



Technische Universität München
Fakultät für Maschinenwesen
Lehrstuhl für Flugsystemdynamik

High Performance Kalman Filter Tuning for Integrated Navigation Systems

Dipl.-Ing. (Univ.) Benjamin Braun

Vollständiger Abdruck der von der Fakultät für Maschinenwesen
der Technischen Universität München
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.) genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. Nikolaus Adams

Prüfer der Dissertation: 1. Univ.-Prof. Dr.-Ing. Florian Holzapfel
2. Univ.-Prof. Dr.-Ing. Alfred Kleusberg,
Universität Stuttgart

Die Dissertation wurde am 23.09.2015 bei der Technischen Universität München eingereicht und
durch die Fakultät für Maschinenwesen am 18.03.2016 angenommen.



Technische Universität München
Institute of Flight System Dynamics

Benjamin Braun

High Performance Kalman Filter Tuning for Integrated Navigation Systems

Copyright © 2016 by Benjamin Braun

Find further information on:

<http://www.fsd.mw.tum.de/research/sensors-data-fusion-and-navigation/>

Preface

I have written the present thesis during my time as research assistant at the Institute of Flight System Dynamics.

First of all, I want to thank Johann Dambeck a lot for the great support in every respect. He took a lot of time for me and my requests. I think for example of the numerous and fruitful discussions about many interesting problems in the field of navigation, often even far into the night, and the “Dorfsteller” afterwards. It turned out that we have a similar mindset and particularly a high demand on clarifying any doubts until we have completely understood the posed problem.

Next, I want to thank Professor Florian Holzapfel to place trust in me and to ask me to support the initial group of research assistants from the very beginning of the newly founded institute. This gave me the chance to assist in establishing the institute, including the acquisition of the first research and industry projects, project management, preparation of new lectures, establishment of the research field navigation at the institute, supervision of students of whom many became later themselves employees at the institute, to name only a few points. This experience, besides the actual scientific research work, will be very helpful for my upcoming work life. Retrospectively, it is very encouraging to see how the institute has evolved in the last years.

Not to forget my nice colleagues at the institute and all the discussions at the meeting table about all relevant topics concerning aerospace. I have for example fond memories of the discussions in preparation of the Corvus Racer identification project. Especially with my office mates Thaddäus and Philip I had a very exciting and diversified time.

Finally, I sincerely thank my family and especially my parents for their great support over all the years.

München, April 2015

Abstract

This thesis discusses possibilities to enhance the performance of modern integrated navigation systems that process inertial sensor measurements, GNSS measurements and observations of other (innovative) aiding methods. The focus is on the statistical consistency of the navigation solution, that is the coincidence of the true and the predicted navigation error statistics, which is an important precondition for the reliable functioning of innovation based fault detection and isolation methods. The tradeoff between consistency on the one hand, coming along with higher computational effort, and efficiency, required for the implementation on embedded systems, is a central point of the discussion. Typical navigation errors like IMU errors, gravity model errors, random vibration, GPS satellite clock and orbit errors, ionosphere and troposphere delays, receiver clock errors and receiver noise are characterized, and physically motivated models are identified. It is shown how these models can be efficiently integrated into the navigation filter, distinguishing between bias-like errors to be estimated and noise-like errors to be merely considered in a statistical sense. A further advantage of the error models is a simplified and less heuristic filter tuning process. Other measures to increase the consistency, which are the correct initialization of the filter on-ground and in-flight and the correct treatment of delayed out-of-sequence measurements are thoroughly discussed. Inertial navigation algorithms with low numerical errors even in unfavorable dynamical situations are prerequisite for statistical consistency, motivating the comparison of different algorithms for integrating IMU regarding robustness against noise and vibration. The thesis is completed with a method for the analytical calculation of the steady-state navigation error covariance for online plausibility tests and a mathematical analysis of filter destabilization mechanisms.

In dieser Arbeit werden Möglichkeiten zur Verbesserung der Leistung moderner integrierter Navigationssysteme, die Trägheitssensormessdaten, Satellitennavigationsmessungen und Beobachtungen anderer (innovativer) Stützverfahren verarbeiten, diskutiert. Das Hauptaugenmerk liegt dabei auf der statistischen Konsistenz der Navigationslösung, sprich der Übereinstimmung von tatsächlicher und vorhergesagter Fehlerstatistik. Diese ist ausschlaggebend für die zuverlässige Funktion von innovationsbasierten Fehlererkennungsmethoden. Dabei steht die Abwägung zwischen Konsistenz einerseits, verbunden mit höherem Rechenaufwand, und Effizienz andererseits, notwendig für die Anwendung auf eingebetteten Systemen, im Mittelpunkt der Diskussion. Typische Navigationsfehler wie Trägheitssensormessfehler, Erdschweremodellierungsfehler, Vibrationen, GPS Satellitenuhren- und Positionsfehler, Ionosphären- und Troposphärenverzögerungen, Empfängeruhrenfehler und Empfängermessrauschen werden charakterisiert und physikalisch motivierte Modelle dafür hergeleitet. Es wird aufgezeigt, wie diese Modelle auf effiziente Weise in den Navigationsfilter integriert werden können. Dabei wird unterschieden zwischen Bias-Fehlern, die geschätzt werden können, und Rausch-Fehlern, die lediglich statistisch berücksichtigt werden. Ein weiterer Vorteil der Modelle ist, dass das Abstimmen des Filters vereinfacht wird und damit weniger heuristisch ist. Andere Maßnahmen zur Erhöhung der Konsistenz wie die korrekte Initialisierung des Filters am Boden und im Flug und die richtige Behandlung von verspäteten und aus der Reihe eintreffenden Messungen werden gründlich besprochen. Voraussetzung für statistische Konsistenz sind Trägheitsnavigationsalgorithmen, die selbst unter ungünstigen Umständen niedrige numerische Fehler aufweisen. Daher werden verschiedene Algorithmen für integrierende Trägheitssensoren bezüglich ihrer Robustheit gegenüber Rauschen und Vibrationen untersucht. Die Arbeit wird mit einer Methode zur analytischen Berechnung der stationären Navigationsfehlerkovarianz z.B. für Plausibilitätstests und einer mathematischen Untersuchung, unter welchen Umständen ein Filter instabil werden kann, abgerundet.

Contents

Nomenclature	xii
1 Introduction	1
1.1 Motivation	1
1.2 Contributions of the Thesis	3
1.3 Look at the Relevant Certification Regulations in Effect	4
1.4 Scope of the Thesis	8
1.5 Outline of the Thesis	8
2 Preliminaries	11
2.1 Inertial Navigation Ordinary Differential Equations	11
2.2 Inertial Navigation Error Ordinary Differential Equations	12
2.3 Discretization	13
2.4 Conventional Kalman Filter	13
2.5 Resampling	13
3 Reality Modeling	15
3.1 Fundamentals of Error Analysis and Identification	15
3.1.1 Motivation.....	15
3.1.2 Autoregressive Moving Average Model	15
3.1.3 Stochastic Error Analysis.....	17
3.1.4 Power-Law Noise	20
3.1.5 General Noise.....	26
3.2 Inertial Measurement Unit Error.....	29
3.2.1 Motivation.....	29
3.2.2 Characteristics	31
3.2.3 Models	34
3.2.4 Parameter Estimation.....	37
3.3 Gravity Model Error	40
3.3.1 Motivation.....	40
3.3.2 Disturbing Gravity Potential Autocorrelation Function.....	41
3.3.3 Gravity Disturbance Model.....	43
3.3.4 Upward Continuation	45
3.3.5 Discrete-Time State-Space Model	45
3.3.6 Parameter Estimation.....	45
3.4 Random Vibration.....	49
3.4.1 Motivation.....	49
3.4.2 Power Spectral Analysis of IMU Measurements.....	51
3.4.3 High-Pass Filtering of IMU Measurements	52
3.4.4 Discrete-Time State-Space Model	54
3.5 GPS Observable Errors	55
3.5.1 Satellite Clock Error	55
3.5.1.1 Motivation.....	55
3.5.1.2 Upload and Dataset Cutovers	55
3.5.1.3 Standard Positioning Service Performance Standard	57
3.5.1.4 Characteristic	60
3.5.1.5 Model.....	66
3.5.1.6 Parameter Estimation.....	71
3.5.1.7 Alternative Sources of Correction Coefficients.....	75

3.5.1.8 Remarks.....	76
3.5.2 Satellite Position Error	77
3.5.2.1 Motivation	77
3.5.2.2 Characteristic.....	77
3.5.2.3 Model.....	81
3.5.2.4 Parameter Estimation.....	86
3.5.3 Ionosphere Error.....	90
3.5.3.1 Motivation	90
3.5.3.2 Ionospheric Delay Models	91
3.5.3.3 Estimation of the Sunspot Number	102
3.5.4 Troposphere Error.....	104
3.5.4.1 Motivation	104
3.5.4.2 Characteristic.....	105
3.5.4.3 Power Spectral Density of the Tropospheric Residual Error.....	106
3.5.4.4 Model.....	108
3.5.4.5 SBAS Tropospheric Error Standard Deviation	111
3.5.5 Receiver Clock Error	113
3.5.5.1 Motivation	113
3.5.5.2 Model.....	116
3.5.5.3 Parameter Estimation.....	117
3.5.5.4 Example: Novatel OEM628	121
3.5.5.5 Remarks on the Integration in the Navigation Filter	124
3.5.6 Receiver Noise	125
3.5.6.1 Motivation	125
3.5.6.2 Satellite Signal Model	125
3.5.6.3 Correlator Model	127
3.5.6.4 Discriminators.....	129
3.5.6.5 Tracking Loop Model	134
4 Efficiently Increasing Navigation Filter Consistency	141
4.1 Inertial Navigation Algorithms	141
4.1.1 Motivation	141
4.1.2 Simple Averaging Algorithm	141
4.1.3 Dual Frequency Algorithms for Orientation	142
4.1.4 Conversion-Integration-Extrapolation Algorithms.....	143
4.1.5 Single Frequency Approach.....	149
4.1.6 Numerical Analysis.....	150
4.1.7 Conclusion.....	154
4.2 Navigation Filter Initialization	155
4.2.1 Motivation	155
4.2.2 Stationary Navigation Filter Initialization.....	156
4.2.3 In-Flight Navigation Filter Initialization.....	165
4.2.4 Conclusion.....	180
4.3 Out-of-Sequence Measurements	181
4.3.1 Motivation	181
4.3.2 Synchronization Concept.....	182
4.3.3 Delayed Measurement Update	183
4.3.4 State Backward Propagation	186
4.3.5 Measurement Prediction.....	187
4.3.6 History State Update.....	187
4.3.7 Comparison of the Three Methods	189
4.3.8 Stability Problem	190

4.3.9 Processing more than one Delayed Measurement.....	190
4.3.10 Example	192
4.3.11 Conclusion.....	198
4.4 Navigation Filter State Augmentation.....	199
4.4.1 Motivation.....	199
4.4.2 Kalman Filter State Augmentation.....	199
4.5 Schmidt-Kalman Filter for Noise-Like Errors.....	204
4.5.1 Motivation.....	204
4.5.2 Schmidt-Kalman Filter.....	204
4.6 Integration of Reality Models into Navigation Filter.....	210
4.6.1 Inertial Measurement Unit Errors.....	210
4.6.2 Gravity Model Error	217
4.6.3 Random Vibration	220
4.6.4 Satellite Clock Error	227
4.6.5 Satellite Position Error.....	231
4.6.6 Ionosphere Error.....	233
4.6.7 Receiver Clock Error.....	236
4.6.8 Receiver Noise	239
5 Navigation Filter Performance Prediction and Stability	245
5.1 Steady-State Kalman Filter.....	245
5.1.1 Motivation.....	245
5.1.2 Preliminaries	245
5.1.3 Alternative Representation of the Riccati Equation	246
5.1.4 Solution of the Matrix Exponential	247
5.1.5 Steady-State Solution of the Riccati Equation.....	247
5.1.6 Application to Navigation Error Filter.....	249
5.1.7 Example	252
5.2 Kalman Filter Stability	254
5.2.1 Motivation.....	254
5.2.2 Discrete-Time Kalman Filter Variants to be Analyzed.....	254
5.2.3 Conversion to Continuous-Time Differential Equations.....	255
5.2.4 Covariance Error Differential Equations.....	257
5.2.5 Stability Analysis	259
5.2.6 Stability Enhancement	265
5.2.7 Out-of-Sequence Measurement Stability Problem	265
5.2.8 Example	266
5.2.9 Conclusion	266
6 Summary	269
6.1 Detailed Summary of the Findings	269
6.2 Significant Achievements of the Thesis beyond State-of-the-Art.....	276
Bibliography	279
A Preliminaries	285
A.1 Quaternions.....	285
A.2 Derivation of the Inertial Navigation Error ODE.....	285
A.3 Block Matrices of the n-frame Inertial Navigation Error ODE	287
B Reality Modeling	289
B.1 Autocorrelation Matrix of the Gravity Disturbance Vector	289
B.2 Satellite Track Frame.....	290

B.3 Ionosphere Error	291
B.3.1 Ionospheric Pierce Point	291
B.3.2 Obliquity Factor.....	292
B.4 Troposphere Error.....	293
B.4.1 Zenith Delay Models.....	293
B.4.2 Temperature, Static Pressure and Partial Pressure of Water Vapor.....	294
B.4.3 Mapping Functions.....	295
B.4.4 Atmospheric Turbulence Structure Function.....	296
C Efficiently Increasing Navigation Filter Consistency	299
C.1 Inertial Navigation Algorithms	299
C.1.1 Solution of the Orientation ODE using Laning-Bortz Parameters	299
C.1.2 Derivation of the Single Frequency Approach	300
C.2 Navigation Filter Initialization	305
C.2.1 Orientation Angle Expected Values	305
C.2.2 IMU Error Cross-Covariances	306
C.2.3 Recursive Averaging.....	307
C.3 Out-of-Sequence Measurements.....	308
C.3.1 State Backward Propagation	308
C.3.2 Measurement Prediction.....	310
C.3.3 History State Update	311
C.4 Filter Augmentation by IMU Biases and Scale Factor Errors.....	314
C.5 Schmidt-Kalman Filter for Noise-Like Errors	315
C.5.1 Correspondence with Kalman Filter for State Cross-Correlated Measurements.....	315
C.5.2 Schmidt-Kalman Filter for Out-of-Sequence Measurements	315
D Navigation Filter Performance Prediction and Stability	319
D.1 Eigenvalues and Eigenvectors of the Hamiltonian Matrix.....	319
D.2 Kalman Filter Stability Enhancement.....	322
E Simulation Scenarios	325
E.1 Trajectory	325
E.1.1 Generic Lissajous.....	325
E.1.2 General Aviation Aircraft.....	328
E.2 Sensors	330
E.2.1 IMU	330
E.2.2 GPS.....	331

Nomenclature

Acronyms

ADC	Analog-digital converter	INS	Inertial navigation system
AHRS	Attitude and heading reference system	IODC	Issue of data, clock
AOD	Age of data	IODE	Issue of data, ephemeris
APC	Antenna phase center	IRI	International Reference Ionosphere
ARMA	Autoregressive moving average	ITU	International Telecommunication Union
CARE	Continuous algebraic Riccati equation	LNAV	Legacy navigation message
CIE	Conversion-integration-extrapolation	LPF	Low-pass filter
CKF	Conventional Kalman filter	MC	Mass center
CNAV	Civil navigation message	MIMO	Multi-input multi-output
CS	Cesium	MOPS	Minimum operational performance standards
DARE	Discrete algebraic Riccati equation	MP	Measurement prediction
DLL	Delay lock loop	NAV	Navigation
DOP	Dilution of precision	NCO	Numerically controlled oscillator
DSP	Digital signal processing	NGA	National Geospatial-Intelligence Agency
EASA	European Aviation Safety Agency	OCXO	Oven controlled crystal oscillator
ECEF	Earth-centered Earth-fixed frame	ODE	Ordinary differential equation
EGM	Earth Gravitational Model	PLL	Phase lock loop
EGNOS	European Geostationary Overlay Navigation System	PRN	Pseudo random number
ETSO	European Technical Standard Order	PSD	Power spectral density
FAA	Federal Aviation Administration	PVT	Position, velocity and time
FLL	Frequency lock loop	RAIM	Receiver autonomous integrity monitoring
FFT	Fast Fourier transformation	RB	Rubidium
FIR	Finite impulse response	RLG	Ring laser gyroscope
FOG	Fiber optical gyroscope	RMS	Root mean square
GGTO	IGS-to-GPS time offset	RTCA	Radio Technical Commission for Aeronautics
GNSS	Global navigation satellite system	SBAS	Satellite based augmentation system
GPS	Global Positioning System	SIS	Signal-in-space
HF	High frequency	SPS	Standard Positioning Service
HPF	High-pass filter	SV	Satellite vehicle
HSU	History state update	SVN	Satellite vehicle number
IBN	Image based navigation	TCXO	Temperature compensated crystal oscillator
ICD	Interface control document	TEC	Total electron count
IEEE	Institute of Electrical and Electronics Engineers	TOC	Time of clock
IGS	International GNSS Service		
IPP	Ionospheric pierce point		
IIR	Infinite impulse response		
IMU	Inertial measurement unit		

TOE	Time of ephemeris	URRE	User range rate error
TOT	Time of transmission	USCG	United States Coast Guard
TRIAD	Three-axis attitude determination	UTC	Coordinated universal time
TSO	Technical Standard Order	VCO	Voltage controlled oscillator
UERE	User equivalent range error	WAAS	Wide Area Augmentation System
URA	User range accuracy	WGS	World Geodetic System
URE	User range error	WMM	World Magnetic Model

Notation

Generally, regular variables x are scalar values, boldface small letter variables \mathbf{x} are vectors and boldface capital letters \mathbf{X} are matrices.

Furthermore:

\mathbf{x}	True value
$\tilde{\mathbf{x}}$	Erroneous value
$\delta\mathbf{x} = \mathbf{x} - \tilde{\mathbf{x}}$	Error vector
$\hat{\mathbf{x}}$	Estimated value
\mathbf{x}_a	Vector given in the a -frame
\mathbf{R}_{ab}	Transformation from b - to a -frame
\mathbf{I}_n	Unity matrix of size $(n \times n)$
$\mathbf{x}(t)$	Continuous-time variable
$\mathbf{x}_k, \mathbf{x}[k]$	Discrete-time variable at time t_k
\mathbf{x}_0	Initial value at time t_0
$E[\cdot]$	Expectation value

$WN(\mathbf{m}, \mathbf{Q})$ White, Gaussian noise with mean value \mathbf{m} and covariance \mathbf{Q}

σ	Standard deviation
λ	Eigenvalue
Λ	Eigenvalue matrix
\mathbf{v}	Eigenvector
\mathbf{V}	Eigenvector matrix
\doteq	Approximated to first order
\otimes	Kronecker product
$\text{diag}(\mathbf{x})$	Diagonal matrix operator
$\text{vec}(\mathbf{X})$	Vector operator
$\text{veck}^{-1}(\mathbf{x})$	Skew-symmetric matrix operator: $\text{veck}^{-1}(\mathbf{x}) \mathbf{y} = \mathbf{x} \times \mathbf{y}$
$\check{\mathbf{q}} = (\mathbf{q}_0, \mathbf{q})$	Quaternion
$h.o.t.$	Higher order terms

Symbols

In the following the most common symbols that are used throughout the thesis are listed. Variables that are not mentioned are used locally within one chapter and are explained at their introduction. Some letters have been assigned several times, but their meaning becomes clear from the context.

Frames

i	Inertial frame
e	Earth-fixed Earth-centered frame
n	North-east-down frame
b	Body-fixed frame
s	Sensor frame
t	Satellite track frame
\mathbf{e}	Unit vector

Inertial sensors

\mathbf{f}_b	Acceleration measurement
$\boldsymbol{\omega}_{ib}$	Angular rate measurement
$\Delta \mathbf{v}_b$	Velocity increment
$\Delta \boldsymbol{\theta}_{ib}$	Angle increment
\mathbf{b}_0	Turn-on bias
\mathbf{b}_T	Temperature sensitive bias
B	Bias instability
K	Random walk
N	White noise
Q	Quantization noise
\mathbf{s}, \mathbf{S}	Scale factor error
\mathbf{m}, \mathbf{M}	Misalignment
$\mathbf{a}_o, \mathbf{A}_o$	Acceleration cross-coupling
T	Temperature
$\boldsymbol{\epsilon}$	Non-modeled errors

Gravity

g	Gravitation vector
Y	Gravity vector
γ	Normal gravity
$\delta_{n/e}$	North/east deflection of the vertical
δy	Gravity disturbance
W	Gravity potential
T	Gravity disturbing potential

Inertial navigation system

ϕ, λ	WGS84 latitude/longitude
h	WGS84 height
x	Cartesian position
v	Kinematic velocity
φ	Roll angle
ϑ	Pitch angle
ψ	Heading angle
\check{q}	Orientation quaternion
ω_{ie}	Earth rate vector
ω_{en}	Transport rate vector
M, N	Meridian/normal curvature radii
R_e	Mean Earth radius

Global navigation satellite system

c	Speed of light
f, λ	Frequency, wave length
τ	Signal travel time
ρ	Pseudorange
Φ	Integrated carrier phase
X	Satellite position
V	Satellite velocity
e	Line-of-sight vector
E	Elevation angle
Δt_s	Satellite clock error
t_{oc}, t_{oe}	Time of clock/ephemeris
$a_{f0/1/2}$	Clock correction parameter
Δt_R	Receiver clock error
I	Ionospheric delay
n	Refraction
F	Obliquity factor
ζ	Ionospheric intersection angle
α_i, β_i	Klobuchar coefficients
R₁₂	Sunspot number

T Tropospheric delay**ε** Non-modeled errors**Modeling & navigation filter**

F	Continuous-time system matrix
Φ	Discrete-time system matrix
G	Continuous-time input matrix
Γ	Discrete-time input matrix
H	Observation matrix
D	Feed-through matrix
K	Kalman gain
P	Covariance matrix
Q	Input noise covariance matrix
R	Measurement noise cov. matrix
s	Innovation
S	Innovation covariance
z	State vector
δz	State error vector
η	White Gaussian input noise
a_i	Autoregressive coefficients
b_i	Moving average coefficients
$H(s)$	Transfer function
$H(z)$	Discrete-time transfer function
$h[n]$	Discrete-time impulse response

Time

t, f	Time, frequency
$\Delta t, f_s$	Sample time, sample rate
T	Covariance propagation time

Signal analysis

G_{yy}	One-sided PSD of the signal y
S_{yy}	Two-sided PSD of the signal y
R_{yy}	Correlation function of the signal y
ρ	Autocorrelation coefficient
τ	Correlation time
r	Spatial correlation length
$\sigma_{avar,y}^2$	Allan variance
$\sigma_{mvar,y}^2$	Modified Allan variance
$\sigma_{hvar,y}^2$	Hadamard variance
n	Number of samples
m	Window size
τ	Averaging time
M	Number of windows

1 Introduction

1.1 Motivation

The thesis on hand deals with aspects related to integrated navigation systems for aerospace applications, that is with the accurate and robust estimation of position, velocity and orientation of an aircraft for flight guidance and control purposes.

The traditional field of navigation is presumed to be fully understood and the comprehensive theory is well established. Conventional navigation systems are technically mature and are widely used in all kinds of aircraft, for example civil transport aircraft, military aircraft and high-end general aviation aircraft like modern business jets. Conventional systems are predominantly freely running navigation grade inertial navigation systems (INS), which are solely aided by barometric height measurements to account for the inherently unstable height channel. Although GPS has evolved to a generally accepted means of navigation for safety critical applications by now, it is not before the two-thousands that first certified global navigation satellite navigation (GNSS) aided inertial navigation systems have been developed and applied in safety critical applications. Standalone GNSS receivers for area navigation, which are augmented by satellite based augmentation systems (SBAS), however, have already found their way into aeronautics and GNSS will replace the traditional terrestrial navigation aids in the medium term.

It is thus legitimate to ask how this thesis can contribute to the elaborated theory of navigation about which everything seems to have already been said. Convincing reasons will be given shortly.

In the last two decades new sensors that are promising for the purpose of navigation came on the market. Especially lower cost microelectromechanical (MEMS) accelerometers and gyroscopes have achieved the required maturity in the meantime to be seriously applied in small attitude and heading reference (AHRS) and navigation systems. The performance of the latest MEMS gyroscopes can indeed keep up with lower grade fiber optical gyroscopes. Moreover, multi-frequency multi-constellation GNSS receivers with more than hundred tracking channels are available, which are rarely larger than two credit cards. Small, powerful embedded computing systems with low power consumption enable more complex algorithms. Together with the new sensors and the modern computing modules, new innovative aiding methods like for example image based navigation or new ranging methods using small laser scanners or lidar came up. This recent development opens up a multitude of new fields of application like unmanned aerial vehicles for civil and military applications or low-cost navigation systems and AHRS for low-end general aviation aircraft.

The central fusion and assessment of the single measurements with respect to their integrity gain in importance and are probably more important than ever. Therefore, it is worth to have once again a closer look on the core algorithms of an integrated navigation system and the corresponding error estimation techniques.

The thesis intends to contribute to the better understanding of the principal sensor and modeling error characteristics, their impact on the navigation solution error and the corresponding covariance

estimation of the navigation filter. The focus is to a smaller extent on the enhancement of the accuracy of the navigation solution in its classical sense but rather on the improvement of its *statistical consistency*. The navigation filter is said to be statistically consistent if the estimated navigation state is equal to the expected value of the true navigation state and the predicted covariance corresponds to the covariance of the true navigation state. The statistically correct prediction of the navigation state error covariance is prerequisite for the correct assessment of the navigation solution, which in turn is crucial for the reliable detection and isolation of measurement failures.

Fault detection and isolation by an integrity monitor is an indispensable part of an integrated navigation system. Fault detection methods often judge the integrity of measurements by any form of innovation testing (for example the simple chi-square test). For this, the innovation covariance containing the state error covariance as well as the measurement error covariance is required as accurate as possible. The better the nominal system inherent errors are understood and represented in the navigation filter, the easier and the earlier even smaller exceptional faults can be detected.

The applications that make use of the outputs of the navigation computer like flight guidance and control rely on the correctness and statistical consistency of the output values and the corresponding uncertainty information. The flight management in unmanned aerial systems, for example, decides based on the navigation state estimate covariance whether the nominal system works properly and is trustworthy or whether a failure exists and it has to be switched to a backup system. Particularly if aiding measurements are absent for longer times or if one or more navigation states cannot be observed by the aiding measurements for a longer period, accurate and coherent prediction of the statistics is essential for continuous operation.

Navigation grade INS feature only very slow error growth because the measurement errors of the incorporated inertial measurement unit (IMU) are comparatively small. The errors are benign and play only a minor role. In contrast, the small new generation lower grade sensors are affected by larger errors. The nature of the encountered errors is most often not purely white, as assumed by the Kalman filter, but colored with different correlation times. In common navigation system designs stochastic measurement errors that are not estimated by the filter are mostly modeled as white noise for the sake of simplicity or against better judgment. Time correlated noise is substituted by white noise and the disregarded correlation is compensated by higher uncertainty. This might be problematic because the navigation solution will be severely distorted if the magnitude of the substitute white noise is assumed too small, resulting in persistent state error estimate offsets. Since the innovation is consequently affected, too, there is the risk that the integrity monitor marks aiding measurements unjustifiably as faulty. Otherwise, if the white noise density is chosen sufficiently large, the navigation solution will not be biased but one gives valuable system performance away because the predicted state error variance is larger than it could be.

Therefore, due to the mentioned reasons, it is desirable that the navigation filter is designed such that all measurements, the inertial as well as the aiding measurements, and modeling errors like the gravity model error are processed in a statistically correct manner.

Moreover, the integration of sensor measurement and modeling error models into the navigation filter has an interesting side effect: it can enormously simplify the filter tuning process. Filter tuning refers to the reasonable adjustment of the process noise and aiding measurement error covariance matrices by the system designer. Filter tuning is in general a difficult task and requires some experience in integrated navigation systems. It is an iterative process, which needs simulations on the one hand and real, recorded measurement data for example from flight experiments on the other hand. The system

designer is relieved from a great part of the filter tuning work since magnitudes and correlations of many errors are automatically represented by the error models.

1.2 Contributions of the Thesis

The basic ways to increase the statistical consistency of the navigation solution are in principle well-known and are addressed in the respective literature. Most textbooks as well as the relevant certification standards are however restricted to naming only the necessity of measures but do not describe how to actually accomplish it since this would go far beyond the scope of the textbooks. The thesis goes a step beyond by elaborating the underlying theory and operationally relevant aspects comprehensively and consistently for typical GPS/INS integrated navigation systems.

The derived models and methods can be used on the one hand for simulation, which is inevitable in the predesign phase of a new navigation system and which is in fact demanded by the certification regulations, and on the other hand for the real-time algorithms on the embedded system.

Furthermore, some interesting aspects concerning the widely-used basic algorithms of an integrated navigation system are discussed, which cannot be found in this form and detail in the standard textbooks, but may help the system designer to achieve a higher level of understanding.

For many presented methods templates are given with the formulas to be implemented in the embedded firmware, which underlines the operational focus of the thesis.

The following questions and solution proposals shall be discussed in detail in this thesis:

How to describe sensor, model and observation errors by appropriate mathematical models, which can be integrated into the navigation filter?

Physically motivated models for the typical sensor measurement and modeling errors in the field of navigation, that is IMU errors, gravity model error, random vibration, GPS pseudorange errors comprising satellite clock and orbit errors, ionospheric and tropospheric errors, receiver clock error and receiver noise, are derived. This involves the identification of the model structure and subsequently the estimation of the model parameters from real data. Common analysis techniques for stochastical errors in time and frequency domain are used to describe their characteristics.

How to efficiently increase the consistency of the navigation solution with reasonable additional computational effort?

The following measures are proposed:

- Accurate and robust inertial navigation algorithms. It is a prerequisite for statistical consistency that the numerical error of the chosen inertial navigation algorithm is many orders smaller than all other involved errors in all imaginable flight situations and even under unfavorable conditions like permanent coning and sculling motion because the numerical error is not taken into account in the navigation filter covariance prediction. Different algorithms for integrating IMU shall be compared with respect to their robustness against noise and vibration.
- Statistically correct navigation filter initialization. The effect of a statistically incorrect initial covariance matrix on the filter consistency in the settling period immediately after initialization shall be clarified. Often, the filter is initialized with a diagonal, over-bounded navigation state error covariance matrix. Cross-covariances for example between the initial orientation errors and the

IMU bias states are mostly neglected. Algorithms for the filter initialization on ground and for the initialization in-flight shall be presented.

- Proper consideration of out-of-sequence measurements. Since modern integrated navigation systems generally process measurements from various sources, the system has to handle different sample rates and different delays. Aiding observations may be only available out-of-sequence. A navigation filter that treats these delays and disorder in a statistically correct manner shall be derived in order to account for the true innovation of each new measurement. It shall be clarified under which circumstances the increased effort is actually necessary.
- Navigation filter state augmentation. The core of an integrated navigation system is an error state space filter (for example conventional Kalman filter) that estimates the navigation state errors. The derived reality models of the bias-like sensor, model and observation errors shall be integrated into the navigation error filter by augmenting the navigation filter states by the reality model states. Time and cross correlations are henceforth correctly accounted for at the expense of higher computational effort.
- Avoiding state augmentation of noise-like errors. A larger portion of the reality model errors are noise-like or are not observable. If these errors are not estimated but only considered by the navigation filter in a statistical sense, the computational effort can be strongly reduced.
- Efficient reality modeling. The number of reality model states has to be decreased in order to increase the efficiency. The order of each reality model has to be adequately chosen such that the main error characteristic is well represented. It has to be carefully analyzed which errors are dominant and which time correlations have to be considered. The more details of the errors are depicted the more additional states, which in turn increases the computational costs.

How to quickly assess steady-state navigation filter performance and how to encounter noticed stability issues?

An efficient tool to quickly calculate the expected stationary covariance of the navigation filter given the process noise covariance matrix, measurement covariance matrix and the measurement frequency is presented. This method is convenient for the specification of integrated navigation systems at the very beginning of the design phase, integrated navigation accuracy predictions and for navigation data fusion testing. Since the navigation filter for delayed measurements tends to become unstable, a thorough mathematical analysis that explains the destabilization mechanisms of the Kalman filter in general is required and appropriate countermeasures have to be found.

Even if the computational power of embedded computers continuously increases all presented measures and methods are traded off against efficiency in order to keep the numerical effort as lean as possible avoiding computations with minor effect on the navigation solution and save spare computational capacity for other tasks.

1.3 Look at the Relevant Certification Regulations in Effect

Since the methods to be presented are part of the safety critical avionics of an aerospace system, they have to meet certain requirements for certification. It is therefore worth to have a brief look at the relevant certification rules in effect first of all and to keep them in mind when discussing innovative methods. The manufacturer has to certify the navigation system according to the applicable Technical

Standard Orders (TSO) of the Federal Aviation Administration (FAA) or the corresponding European Technical Standard Orders (ETSO) of the European Aviation Safety Agency (EASA) before it can be applied in an aircraft. The focus of this thesis is especially on the functional requirements.

GNSS

The relevant TSO for stand-alone GNSS receivers are the FAA TSO-C145d, “Airborne Navigation Sensors Using The Global Positioning System Augmented By The Satellite Based Augmentation System” [1] and the FAA TSO-C146d, “Stand-Alone Airborne Navigation Equipment Using The Global Positioning System Augmented By The Satellite Based Augmentation System” [2]. Both documents refer to the Radio Technical Commission for Aeronautics (RTCA) standard DO-229D, “Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment” [3]. The RTCA standard specifies the minimum required performance and defines in detail how to verify these performance requirements in laboratory tests and flight evaluations. The main focus is on the integrity of the position solution.

AHRS

The decisive document for the certification of attitude and heading reference systems is the FAA TSO-C201, “Attitude and Heading Reference Systems (AHRS)” [4]. The TSO refers to the RTCA standard DO-334, “Minimum Operational Performance Standards (MOPS) for Strapdown Attitude and Heading Reference Systems (AHRS)” [5]. It targets on modern systems incorporating strapdown IMU sensors and also addresses the possibility to aid the system with external sensor measurements like GNSS. In the following some excerpts that are interesting in the context of this thesis are cited:

- (2.1.9) “The equipment shall be designed to operate and output pitch, roll and heading data through all attitudes and headings. [...] The AHRS shall be designed to continuously output data without invalidating the outputs up to at least ± 70 deg/sec angular rates in each axis, at least ± 2 g body axis longitudinal and lateral acceleration, and at least ± 4 g body axis normal acceleration.”
- (2.2.1.5.2) “The update rate of the pitch, roll, and if provided, heading, turn, standard turn bank angle, and slip shall be a minimum of 10 hertz.”
- (2.2.1.5.3) “The latency of the pitch, roll, and if provided, heading, turn, standard turn bank angle, and slip, which is the interval between the motion and the time a measurement is output, shall be a maximum of 200 msec.”
- (2.2.1.5.4) “The filtering of the pitch, roll, heading, and other output parameters, shall be defined by the manufacturer. The noise content and resulting phase delay should be considered.”
- (2.2.1.5.5) “The equipment shall be designed to operate through all attitudes and headings. During and subsequent to rotation of the equipment through pitch angles greater than 89° or less than -89° the equipment shall continue to function correctly in all regards without need for re-initialization, caging or resetting of the equipment.”
- (2.2.6) “The AHRS may use aiding sources, such as GNSS, air data, magnetic sensors, etc., in deriving AHRS outputs. [...] Aiding sensors may be integral to the AHRS equipment design, or they may be separate components.”

- (2.2.6.2) “Following the detected loss of aiding, if the AHRS cannot meet the requirements of its current operational mode, the AHRS shall revert to a different operational mode, a degraded mode, or invalidate its output.”

The standard allows to do parts of the verification by means of simulation instead of flight test evaluation. In the appendix of the standard it is constituted that “A very high level of confidence in the simulation results should be attained to bypass the flight evaluation activity. The importance of achieving accurate results with the simulation cannot be overstated. The simulation should account for all significant sources of error including:

1. Sensor misalignment (gyro, accelerometer)
2. Sensor scale factor (gyro, accelerometer, MSU, barometric altitude)
3. Sensor bias (gyro, accelerometer, MSU, barometric altitude, TAS)
4. Sensor noise (gyro, accelerometer, MSU)
5. Sensor range limit (gyro, accelerometer)
6. Speed-dependent noise (barometric altitude, TAS)
7. Other known sensitivity factors to sensor error if any (e.g. g-sensitivity of gyro)
8. AHRU installation alignment
9. Algorithm estimations
10. Environmental variables including
 - a. Temperature ramps
 - b. Vibration (including acoustically induced vibration)
 - c. Weather induced actions such as horizontal and vertical wind gusts
 - d. Air Data
 - e. GPS Errors“

Moreover, it is stipulated that “The simulated signal errors are accurately modeled. The simulated signal errors can be validated by comparing the statistics of real sensor errors to the statistics of the simulated sensor errors. This comparison has to be done either in the dynamic conditions of the test trajectory, or in any condition ensuring that the modeled sensor errors accurately predict the error characteristics when used in the context of the test trajectory.” This requirement is another justification for the high effort that is made in this thesis to model the reality as accurate as possible.

Integrated Navigation Systems

The FAA TSO-C196b, “Airborne Supplemental Navigation Sensors for Global Positioning System Equipment Using Aircraft-Based Augmentation” [6], is the current basis for the certification of GPS/INS integrated navigation systems. As in the case of AHRS the standard refers to a RTCA document, that is the RTCA standard DO-316, “Minimum Operational Performance Standards for Global Positioning System/Aircraft Based Augmentation System Airborne Equipment” [7], which provides in detail the performance requirements and the corresponding test procedures for an integrated navigation system.

In contrast to the AHRS the focus is on the position solution and not on the orientation solution. As with the PVT solution of stand-alone GNSS receivers, the main issue is the integrity of the position solution. Even if a GNSS/Baro/INS integrated navigation system and a GNSS/Baro aided AHRS are similar systems and mainly use the same algorithms, the standards for the integrated navigation system and for AHRS address different, but complementary aspects. A system that outputs the full

state vector has thus to fulfill the requirements of the RTCA DO-316 standard as well as the RTCA DO-334 standard. The standard corresponds in large parts to the RTCA DO-229D. Specific chapters addressing the aircraft-based augmentation are added in the appendix. These are Appendix G, “Requirements for barometric altimeter aiding”, Appendix K, “Fault detection and exclusion references” and Appendix R, “Requirements and test procedures for tightly integrated GPS/inertial systems”. In the following, some interesting citations from the relevant appendix R are given:

- (R.1) “Tightly integrated systems process and monitor pseudo ranges individually based on inertial information in order to prevent pseudo-range errors from causing system integrity violations.”
- (R.2.2.5) “If pressure altitude aiding is used to enhance integrity, the algorithms that perform calibration shall [GPS 289] be designed to prevent the satellite failure itself from affecting the integrity of the calibration. Conventional Kalman filter integrations using a bias error state with no further enhancements to protect this state in a failure situation will not meet this requirement.”
- (R.2.2.7) “The inertial sensors in tightly integrated GPS/inertial systems are continuously calibrated using GPS measurements. This means that the system can propagate the established position accurately if the GPS signals are lost due to any unexpected event such as interference, scintillation, masking, unexpected satellite failure, etc.“
- (R.2.2.7.1) “The horizontal coasting error distribution as a function of time should be evaluated under the following conditions: Inertial errors should be initialized by either beginning the simulation error-free and flying 60 minutes on pure inertial [...]. Then GPS measurements should be incorporated for a calibration time of 60 minutes. [...] The evaluation should be performed with correct or conservative gyro/accelerometer noise, correct or conservative gyro/accelerometer bias instability and a representative receiver clock model. [...] The coasting error distribution is a statistical measure and covariance propagation techniques may be used to determine the performance. At least 500 Monte Carlo simulations, including calibration and subsequent coasting, using the algorithms implemented in the system should be run to verify the covariance propagation model used to predict the claimed coasting performance.”
- (R.3.1) “[...] The Kalman filter relies on an accurate inertial error model and known statistical inertial sensor error distributions as well as a linearized measurement model for GPS pseudoranges and the associated pseudorange error statistics.”
- (R.3.1.1) “If one or several innovations associated with a measurement far exceed the expected 1-sigma value, the measurement is excluded. This method provides exclusion capability for large steps, ramps and ramp rates. [...] Slow drifts or drift rates are, however, not excluded and detection and exclusion of such failure types can only be provided by other methods.”

Again, as in the case of the AHRS, these requirements substantiate the need of high fidelity models of the sensor measurement and modeling errors for simulation and integration into the navigation filter. The Monte Carlo simulation technique will be extensively used throughout the thesis to test the statistical consistency of the presented methods.

1.4 Scope of the Thesis

The thesis concentrates on the most relevant sensors and core algorithms of an integrated navigation system in order to keep the structure of the thesis clear. The discussion is focused on a system that comprises a simplex IMU, single frequency GPS L1 pseudorange and range rate measurement aiding and the corresponding data fusion algorithms. Other sensors that are usually found in integrated navigation systems like barometer or magnetometer are not explicitly regarded. Moreover, multi-frequency multi-constellation GNSS navigation, which includes novel GNSS systems like Galileo, GLONASS or Beidou and which will play an important role in future designs, go beyond the scope and are hence not part of the discussion. SBAS augmentation plays only a minor role. The selected sensor and modeling errors should be understood as the main examples by which the identification and judgement process of typical navigation errors is demonstrated.

The focus is on the increase of the statistical consistency of the navigation solution, which is required for the fast and reliable fault detection, isolation and recovery (FDIR), but not on the integrity monitoring and FDIR algorithms itself. Furthermore, hardware related aspects like redundancy shall not be discussed.

1.5 Outline of the Thesis

The thesis is organized as follows: in chapter 2, “Preliminaries”, some fundamentals of integrated navigation systems, which are required by the thesis, are briefly recapitulated. In chapter 3, “Reality Modeling”, common signal analysis and model identification methods are presented in general in the beginning and actual navigation errors are analyzed and identified in particular afterwards. In chapter 4, “Efficiently Increasing Navigation Filter Consistency”, several methods to increase the accuracy and statistical consistency of the navigation solution are presented under the premise of efficiency. These are inertial navigation algorithms for integrating IMU, filter initialization on-ground and in-flight, treatment of out-of-sequence measurements, filter state augmentation by error model states, Schmidt-Kalman filter for noise-like process and measurement noise and the analysis of the error models that have been integrated into the navigation filter. In chapter 5, “Navigation Filter Performance Prediction and Stability”, an analytical method for the prediction of the steady-state navigation error covariance matrix is presented and the mechanism how the filter can become unstable is mathematically analyzed. In the final chapter 6 the main aspects of the found results in this thesis are summarized. Additional topics and derivations for further reading have been shifted to the appendix but are referenced in the main document. In Figure 1-1 the outline of the thesis is graphically illustrated on the basis of the components and data flow of a GPS/INS integrated navigation system.

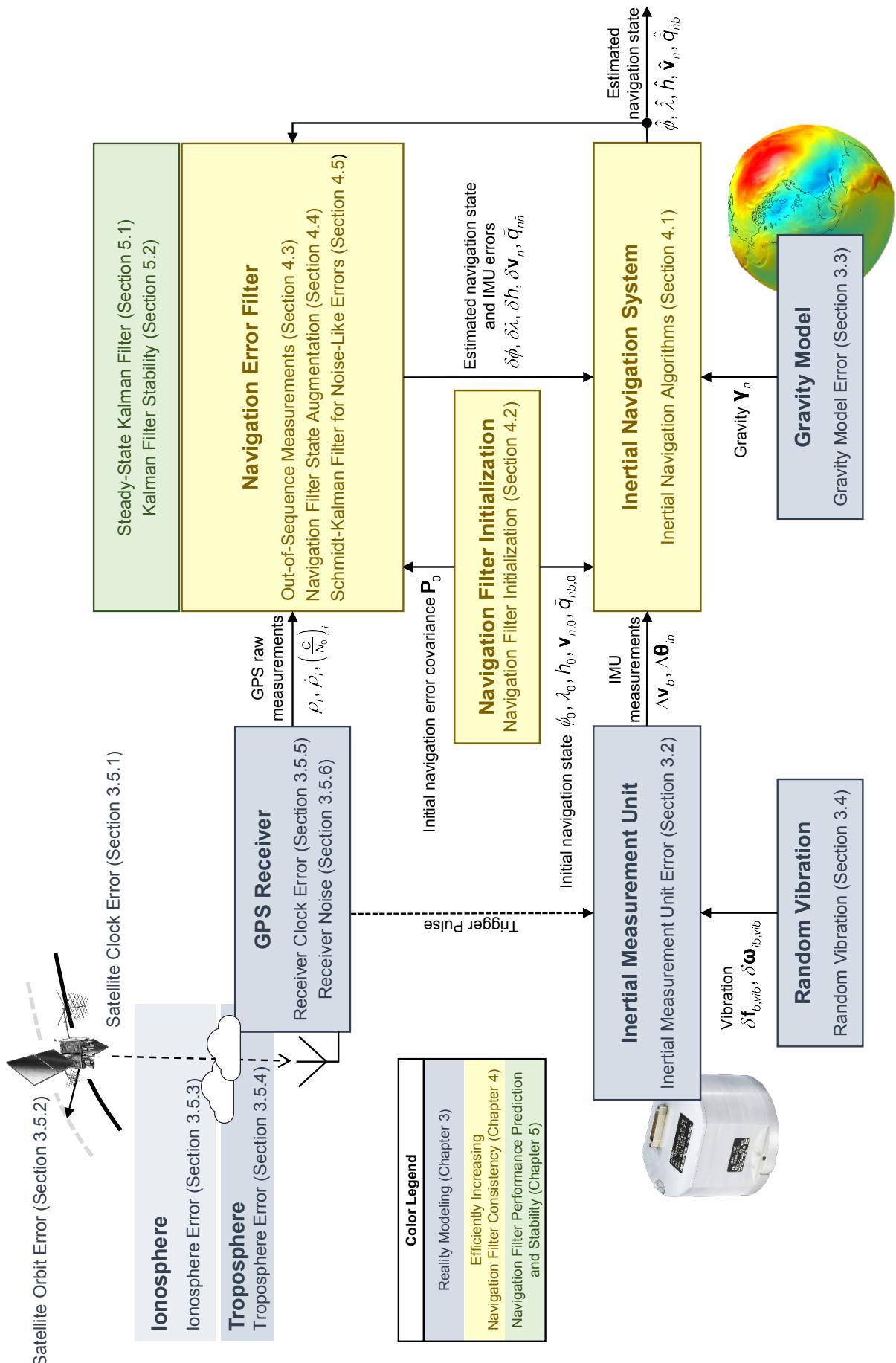


Figure 1-1: Graphical illustration of the outline of the thesis

2 Preliminaries

In this short introductory chapter, the most important basics of integrated navigation are briefly recapitulated for convenience. They are the common basis for all issues throughout the thesis at hand.

2.1 Inertial Navigation Ordinary Differential Equations

The well-known 1st order nonlinear navigation ordinary differential equations (ODE) form the core of an inertial navigation system, here given in the Earth-centered Earth-fixed frame (ECEF, \mathbf{e} -frame)

$$\begin{aligned}\dot{\mathbf{x}}_e(t) &= \mathbf{v}_e(t) \\ \dot{\mathbf{v}}_e(t) &= \mathbf{R}_{eb}(\bar{q}_{eb}) \cdot \mathbf{f}_b(t) + \mathbf{y}_e(\mathbf{x}_e) - 2\boldsymbol{\omega}_{ie} \times \mathbf{v}_e(t) \\ \dot{\bar{q}}_{eb}(t) &= \frac{1}{2}(\bar{q}_{eb}(t)\bar{\omega}_{ib}(t) - \bar{\omega}_{ie}\bar{q}_{eb}(t))\end{aligned}\quad (2.1)$$

(see for example [8], [9], [10]). The navigation states are the Cartesian position vector \mathbf{x}_e , the velocity with respect to Earth \mathbf{v}_e and the orientation of the carrier platform's fixed frame (b -frame) with respect to the e -frame

$$\mathbf{x}_e = \begin{pmatrix} x_e \\ y_e \\ z_e \end{pmatrix}, \quad \mathbf{v}_e = \begin{pmatrix} u_e \\ v_e \\ w_e \end{pmatrix}, \quad \bar{q}_{eb} = (q_{eb_0}, \mathbf{q}_{eb}) \quad (2.2)$$

Therein, the orientation is represented as quaternion \bar{q}_{eb} . $\bar{\omega}_{ib}$ is the angular rate quaternion $(\mathbf{0}, \boldsymbol{\omega}_{ib})$. Quaternions as rotation parameters and particularly the quaternion product are introduced in section A.1 in the appendix. \mathbf{y}_e is the gravity vector, which is the sum of gravitation and centrifugal acceleration and is provided by an appropriate model, $\boldsymbol{\omega}_{ie}$ is the Earth rate vector and $\bar{\omega}_{ie}$ the corresponding quaternion. \mathbf{R}_{eb} is the direction cosine matrix between b - and e -frame. The navigation differential equations expect the acceleration measurement $\mathbf{f}_b(t)$ and the angular rate measurement $\boldsymbol{\omega}_{ib}(t)$ as inputs. The navigation states are obtained by numerically integrating the navigation differential equations, beginning from the initial values $\mathbf{x}_e(t_0)$, $\mathbf{v}_e(t_0)$ and $\bar{q}_{eb}(t_0)$ by means of an adequate integration scheme like 1st order Euler, 2nd order modified Euler or 4th order Runge-Kutta. The navigation differential equations can be alternatively defined in the local North-East-Down frame (n -frame)

$$\begin{aligned}\begin{pmatrix} \dot{\phi}(t) \\ \dot{\lambda}(t) \\ \dot{h}(t) \end{pmatrix} &= \mathbf{D}^{-1} \mathbf{v}_n(t) \\ \dot{\mathbf{v}}_n(t) &= \mathbf{R}_{nb}(\bar{q}_{nb}) \cdot \mathbf{f}_b(t) + \mathbf{y}_n(\phi, \lambda, h) - (2\mathbf{R}_{ne}(\phi, \lambda)\boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}(\phi, \lambda, \mathbf{v}_n)) \times \mathbf{v}_n(t) \\ \dot{\bar{q}}_{nb}(t) &= \frac{1}{2}(\bar{q}_{nb}(t)\bar{\omega}_{ib}(t) - \bar{\omega}_{in}\bar{q}_{nb}(t))\end{aligned}\quad (2.3)$$

with the latitude ϕ , longitude λ , height h , velocity \mathbf{v}_n and orientation \bar{q}_{nb} as navigation states

$$\phi, \lambda, h, \mathbf{v}_n = \begin{pmatrix} u_n \\ v_n \\ w_n \end{pmatrix}, \check{\mathbf{q}}_{nb} = (\mathbf{q}_{nb_0}, \mathbf{q}_{nb}) \quad (2.4)$$

\mathbf{D} maps the latitude, longitude and height errors onto the Cartesian north, east and down directions

$$\mathbf{D} = \begin{pmatrix} M(\phi) + h & 0 & 0 \\ 0 & (N(\phi) + h) \cdot \cos \phi & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (2.5)$$

M and N are the Earth curvature radii in meridian and normal directions, respectively. $\boldsymbol{\omega}_{en}$ is the transport rate vector.

2.2 Inertial Navigation Error Ordinary Differential Equations

The linearized inertial navigation error ODE in e -frame with position error $\delta\mathbf{x}_e$, velocity error $\delta\mathbf{v}_e$ and orientation error $\boldsymbol{\Psi}_{ee} = (\varphi_{ee}, \vartheta_{ee}, \psi_{ee})^\top$, which can be found for example in [10], [11] or [12] and are also derived in the appendix in section A.2, is given with

$$\begin{pmatrix} \dot{\delta\mathbf{x}}_e \\ \dot{\delta\mathbf{v}}_e \\ \dot{\boldsymbol{\Psi}}_{ee} \end{pmatrix} \doteq \underbrace{\begin{pmatrix} 0 & \mathbf{I}_3 & 0 \\ \boldsymbol{\Gamma}_e & -2\boldsymbol{\Omega}_{ie} & -\text{veck}^{-1}(\tilde{\mathbf{f}}_e) \\ 0 & 0 & -\boldsymbol{\Omega}_{ie} \end{pmatrix}}_{\mathbf{F}} \cdot \underbrace{\begin{pmatrix} \delta\mathbf{x}_e \\ \delta\mathbf{v}_e \\ \boldsymbol{\Psi}_{ee} \end{pmatrix}}_{\mathbf{Z}} + \underbrace{\begin{pmatrix} 0 & 0 \\ \mathbf{R}_{eb} & 0 \\ 0 & \mathbf{R}_{eb} \end{pmatrix}}_{\mathbf{G}} \cdot \begin{pmatrix} \delta\mathbf{f}_b \\ \delta\boldsymbol{\omega}_{ib} \end{pmatrix} \quad (2.6)$$

$\boldsymbol{\Gamma}_e = \partial\tilde{\boldsymbol{\Psi}}_e(\mathbf{x}_e)/\partial\mathbf{x}_e^\top$ is the gravity gradient and $\boldsymbol{\Omega}_{ie} = \text{veck}^{-1}(\boldsymbol{\omega}_{ie})$ the skew symmetric matrix of the Earth rate vector. The veck^{-1} operator creates the skew symmetric matrix of the argument vector. The orientation error $\boldsymbol{\Psi}_{ee}$ is defined as small angle transformation matrix $\mathbf{R}_{ee} = \mathbf{R}_3(\psi_{ee})\mathbf{R}_2(\vartheta_{ee})\mathbf{R}_1(\varphi_{ee})$ that is multiplied to the erroneous direction cosine matrix \mathbf{R}_{eb} from left to obtain the true error-free matrix \mathbf{R}_{eb} , that is $\mathbf{R}_{eb} = \mathbf{R}_{ee}\mathbf{R}_{eb}$. This representation is advantageous because platform dynamics do not influence the orientation error. The corresponding linearized inertial navigation error ODE in n -frame with latitude error $\delta\phi$, longitude error $\delta\lambda$, height error δh , velocity error $\delta\mathbf{v}_n$ and orientation error $\boldsymbol{\Psi}_{nn} = (\varphi_{nn}, \vartheta_{nn}, \psi_{nn})^\top$ is

$$\begin{pmatrix} \dot{\delta\phi} \\ \dot{\delta\lambda} \\ \dot{\delta h} \\ \dot{\delta\mathbf{v}}_n \\ \dot{\boldsymbol{\Psi}}_{nn} \end{pmatrix} \doteq \underbrace{\begin{pmatrix} \mathbf{F}_{11} & \mathbf{D}^{-1} & 0 \\ \mathbf{F}_{21} & \mathbf{F}_{22} & -\text{veck}^{-1}(\tilde{\mathbf{f}}_n) \\ \mathbf{F}_{31} & \mathbf{F}_{32} & \mathbf{F}_{33} \end{pmatrix}}_{\mathbf{F}} \cdot \underbrace{\begin{pmatrix} \delta\phi \\ \delta\lambda \\ \delta h \\ \delta\mathbf{v}_n \\ \boldsymbol{\Psi}_{nn} \end{pmatrix}}_{\mathbf{Z}} + \underbrace{\begin{pmatrix} 0 & 0 \\ \mathbf{R}_{nb} & 0 \\ 0 & \mathbf{R}_{nb} \end{pmatrix}}_{\mathbf{G}} \cdot \begin{pmatrix} \delta\mathbf{f}_b \\ \delta\boldsymbol{\omega}_{ib} \end{pmatrix} \quad (2.7)$$

The matrices \mathbf{F}_{11} , \mathbf{F}_{21} , \mathbf{F}_{22} , \mathbf{F}_{31} , \mathbf{F}_{32} and \mathbf{F}_{33} depict the effect of the Earth rate $\boldsymbol{\omega}_{ie}$, transport rate $\boldsymbol{\omega}_{en}$ and gravity gradient $\boldsymbol{\Gamma}_n$ on the navigation error and are given in section A.3 in the appendix.

2.3 Discretization

The continuous-time state-space models (2.6) and (2.7) have first to be converted to discrete-time before they can be integrated into the navigation filter. The discrete-time system matrix Φ and input matrix Γ are obtained from the corresponding continuous-time matrices F and G by discretization with the covariance propagation time T

$$\begin{aligned}\Phi &= e^{FT} = \sum_{i=0}^{\infty} F^i \frac{T^i}{i!} = I_n + FT + F^2 \frac{T^2}{2} + O(T^3) \\ \Gamma &= (e^{FT} - I_n)G = \sum_{i=0}^{\infty} \left(F^i \frac{T^{i+1}}{(i+1)!} \right) G = GT + FG \frac{T^2}{2} + O(T^3)\end{aligned}\quad (2.8)$$

The covariance of the discrete-time input noise \mathbf{Q}_k is related to the covariance of the continuous-time input noise \mathbf{Q} by

$$\mathbf{Q}_k = \frac{\mathbf{Q}}{T} \quad (2.9)$$

2.4 Conventional Kalman Filter

The error of the navigation state vector is estimated by a conventional Kalman filter (CKF). In general, a time-variant discrete-time system model consisting of the propagation and observation equation forms the basis of the discrete-time CKF

$$\begin{aligned}\mathbf{z}_k &= \Phi_{k-1} \mathbf{z}_{k-1} + \Gamma_{k-1} \boldsymbol{\omega}_{k-1} & \boldsymbol{\omega}_k &\sim WN(0, \mathbf{Q}_k) \\ \tilde{\mathbf{y}}_k &= \mathbf{H}_k \mathbf{z}_k + \mathbf{v}_k & \mathbf{v}_k &\sim WN(0, \mathbf{R}_k)\end{aligned}\quad (2.10)$$

Therein, \mathbf{z} is the state vector, Φ is the system matrix, Γ is the input matrix and \mathbf{H} is the output matrix belonging to the measurement $\tilde{\mathbf{y}}$. $\boldsymbol{\omega}$ is the input noise. $\tilde{\mathbf{y}}$ is representative for any arbitrary aiding observation that is corrupted by the measurement noise \mathbf{v} . The CKF expects the process noise $\boldsymbol{\omega}$ as well as the measurement noise \mathbf{v} to be white and normally distributed (WN) with covariances \mathbf{Q}_k and \mathbf{R}_k , respectively.

The state and covariance propagation and update equations are [13]

$\hat{\mathbf{z}}_k^- = \Phi_{k-1} \hat{\mathbf{z}}_{k-1}^+$	State propagation
$\mathbf{P}_k^- = \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^\top + \Gamma_{k-1} \mathbf{Q}_k \Gamma_{k-1}^\top$	Covariance propagation
$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^\top (\mathbf{R}_k + \mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^\top)^{-1}$	Kalman gain
$\hat{\mathbf{z}}_k^+ = \hat{\mathbf{z}}_k^- + \mathbf{K}_k (\tilde{\mathbf{y}}_k - \mathbf{H}_k \hat{\mathbf{z}}_k^-)$	State update
$\mathbf{P}_k^+ = (I_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-$	Covariance update
$= (I_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- (I_n - \mathbf{K}_k \mathbf{H}_k)^\top + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^\top$	(short and Joseph forms)

2.5 Resampling

Often, discrete-time models are derived for a certain, physically motivated time step size Δt , for example the sample time of the inertial sensors. Since navigation error dynamics are normally small, it is sufficient to propagate the covariance in the navigation error filter between subsequent updates

with comparably low rates (for example 10 Hz). The discrete-time model has thus to be resampled to the filter propagation time T before it can be applied in the framework of the navigation error filter. Assuming that T is an integer multiple of the original time step size, $T = m \Delta t$, the down-sampling of the system matrices can be done by

$$\begin{aligned}\bar{\Phi} &= \prod_{i=1}^m \Phi = \Phi^m \\ \bar{\Gamma} &= \sum_{i=1}^m \left(\prod_{j=i+1}^m \Phi \Gamma \right) = \sum_{i=1}^m \Phi^{m-i} \Gamma\end{aligned}\quad (2.12)$$

Alternatively to (2.12), the system and input matrices, whose sizes are assumed to be $n \times n$ and $n \times 1$, can be combined in a matrix

$$\begin{pmatrix} \Phi & \Gamma \\ \mathbf{0}_{1 \times n} & 1 \end{pmatrix} \quad (2.13)$$

and raised to the power of m

$$\begin{pmatrix} \Phi & \Gamma \\ \mathbf{0}_{1 \times n} & 1 \end{pmatrix}^m = \begin{pmatrix} \Phi^m & \sum_{i=1}^m \Phi^{m-i} \Gamma \\ \mathbf{0}_{1 \times n} & 1 \end{pmatrix} \quad (2.14)$$

Then, the overall system and input matrices $\bar{\Phi}$ and $\bar{\Gamma}$ can be found in the upper left and right places of the matrix. The matrix power can be efficiently computed if the matrix is decomposed in its eigenvector matrix \mathbf{V} and diagonal eigenvalue matrix \mathbf{D}

$$\begin{pmatrix} \Phi & \Gamma \\ \mathbf{0}_{1 \times n} & 1 \end{pmatrix}^m = \mathbf{V} \mathbf{D}^m \mathbf{V}^{-1} \quad (2.15)$$

The generalized Schur decomposition can be used to compute the eigenvalues and eigenvectors numerically. The diagonal eigenvalue matrix \mathbf{D} can be easily raised to the power of m . Besides the system matrices Φ and Γ the input noise covariance has to be adequately scaled by

$$\bar{\mathbf{Q}} = \mathbf{Q} \frac{\Delta t}{T} \quad (2.16)$$

It has been observed that it is advantageous to transform the resampled state-space model to modal form in order to increase numerical stability. The transformed system, input and output matrices are

$$\bar{\Phi}^* = \bar{\mathbf{D}}, \quad \bar{\Gamma}^* = \bar{\mathbf{V}}^{-1} \bar{\Gamma}, \quad \mathbf{H}^* = \mathbf{H} \bar{\mathbf{V}} \quad (2.17)$$

with

$$\bar{\Phi} = \bar{\mathbf{V}} \bar{\mathbf{D}} \bar{\mathbf{V}}^{-1} \quad (2.18)$$

Note that the transformation to modal form is only possible if the geometric multiplicity of all eigenvalues is one. Then, all eigenvectors are linearly independent and the eigenvector matrix $\bar{\mathbf{V}}$ is not singular and hence invertible. The invertibility can be simply tested by computing the determinant of the eigenvector matrix $\bar{\mathbf{V}}$. If it is zero, the modal transformation has to be skipped.

3 Reality Modeling

3.1 Fundamentals of Error Analysis and Identification

3.1.1 Motivation

From experience, navigation sensor measurement and modeling errors are seldom pure white Gaussian noise. In fact, sensor and model values are affected by a mixture of time correlated noises with different correlation lengths, that is colored noise. The correlation lengths range from some sample time steps up to periods of some hours. The highly frequent constituents are actually those that are widely understood as noise whereas the slowly varying parts can be characterized as bias-like. In this chapter, fundamental methods and tools for the analysis, identification and modeling of stochastic errors are recapitulated. These will be later applied to the measurement and modeling errors that are encountered in an integrated navigation system. The derived models are used on the one hand in the simulation for the synthesis of more realistic measurement errors and on the other hand in the navigation filter in order to account for the effects of time correlated measurement noise and thus to improve the statistical consistency of the navigation solution.

In the first section autoregressive moving average (ARMA) models are formally introduced. ARMA models and the corresponding discrete-time state-space models form the basis for the mathematical representation of all stochastic errors in the context of integrated navigation systems. Then, Allan variance and periodogram are recapped as suitable stochastic error analysis tools in time and frequency domain. Subsequently, the special group of stochastic errors with power-law power spectral densities is considered more closely. The errors of this group are for example characteristic of IMU measurement errors as well as of the frequency errors of oscillators. Special attention will be paid to the modeling of flicker noise. Finally, a modeling method for general stochastic errors with arbitrary, non-power-law power spectral densities is presented. It will be for example used for the analysis and modeling of random vibration and for the identification of the satellite clock error residual noise.

3.1.2 Autoregressive Moving Average Model

In order to benefit from the analysis of the errors later on, they have to be mathematically described in any adequate way. It turned out that ARMA models offer the most versatile way to represent stochastic errors in time domain. ARMA models are already discrete-time models and can thus be easily embedded in the digital signal processing of navigation systems. An ARMA model is a universal infinite impulse response (IIR) filter, which shapes white Gaussian noise at the input to colored noise with the desired time correlation at the output. By appropriately choosing the order and coefficients of the ARMA model, more or less correlated errors can be described. Even white noise and constant biases are covered as special cases by ARMA models.

The higher the order of the ARMA model, the more details of the time correlated noise are represented. The number of states of the corresponding state-space model is equal to the model order

and hence, the computational effort is proportional to the model order. The order of ARMA models can probably be higher for simulations than for operational real-time applications on embedded systems with limited computational resources. The system designer has to identify the required model order such that the decisive characteristics are mostly represented but the number of additional states is as low as possible. As soon as the proper model order is found, the parameters of the model have to be estimated, for example, from recorded measurements. In order to incorporate the measurement error models later in the integrated navigation filter, the ARMA models have to be converted to state-space form first.

ARMA models consist of autoregressive and moving average portions. A purely moving average model of order q , MA(q), is given with

$$y[n] = \sum_{i=0}^q b_i \eta[n-i] \quad (3.1)$$

with coefficients b_0, b_1, \dots, b_q and input noise $\eta[n]$. The moving average model is a finite impulse (FIR) filter. A purely autoregressive model of order p , AR(p), is

$$y[n] = \sum_{i=1}^p a_i y[n-i] + \eta[n] \quad (3.2)$$

with coefficients a_1, a_2, \dots, a_p and input noise $\eta[n]$. The autoregressive model is an all-pole IIR filter. The combination of (3.1) and (3.2) yields the ARMA(p, q) model

$$y[n] = \sum_{i=1}^p a_i y[n-i] + \sum_{i=0}^q b_i \eta[n-i] \quad (3.3)$$

The ARMA(p, q) model written as state-space model in controllable canonical form is for $p \geq q$

$$\begin{aligned} \mathbf{z}[n+1] &= \begin{pmatrix} 0 & 1 & 0 & \cdots \\ 0 & \ddots & \ddots & 0 \\ & 0 & 1 & \\ a_p & \cdots & a_2 & a_1 \end{pmatrix} \mathbf{z}[n] + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \eta[n] \\ y[n] &= (0 \ \cdots \ 0 \ b_q + b_0 a_q \ \cdots \ b_2 + b_0 a_2 \ b_1 + b_0 a_1) \mathbf{z}[n] + b_0 \eta[n] \end{aligned} \quad (3.4)$$

The state-space model has p states. The output matrix has only q non-zero entries. The first $p-q$ entries of the output matrix are zero. The same considerations can be repeated for the case $p < q$. The number of states is then q . In this case all q elements of the output matrix are non-zero but only the last p column elements in the last row of the system matrix are non-zero and the first $q-p$ column elements are zero. Note that here the input $\eta[n]$, the output $y[n]$ and the feed-through b_0 are scalar. $\eta[n]$ is white, Gaussian distributed noise. The state-space model can alternatively be represented as observable canonical form. In the following, the notation

$$\begin{aligned} \mathbf{z}_{\omega,k} &= \Phi_{\omega} \mathbf{z}_{\omega,k-1} + \Gamma_{\omega} \eta_{\omega,k-1} \\ y_{\omega,k} &= \mathbf{H}_{\omega} \mathbf{z}_{\omega,k} + D_{\omega} \eta_{\omega,k} \end{aligned} \quad (3.5)$$

with system matrix Φ_{ω} , input matrix Γ_{ω} , output matrix \mathbf{H}_{ω} and feed-through D_{ω} is generally used for measurement error models. The results are summarized in Template 3-1.

Template 3-1: State-space representation of an ARMA model**ARMA (p, q) model**

$$y_{\omega}[n] = \sum_{i=1}^p a_i y_{\omega}[n-i] + \sum_{i=0}^q b_i \eta_{\omega}[n-i] \quad \eta_{\omega} \sim WN(0,1)$$

Controllable canonical form

$$\mathbf{z}_{\omega,k} = \begin{pmatrix} 0 & 1 & 0 & \dots \\ 0 & \ddots & \ddots & 0 \\ & & 0 & 1 \\ a_r & \dots & a_2 & a_1 \end{pmatrix} \mathbf{z}_{\omega,k-1} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \eta_{\omega,k-1}$$

$$r = \max(p, q)$$

$$a_r = 0 \text{ for } r > p, b_r = 0 \text{ for } r > q$$

$$y_{\omega,k} = (b_r + b_0 a_r, \dots, b_2 + b_0 a_2, b_1 + b_0 a_1) \mathbf{z}_{\omega,k} + b_0 \eta_{\omega,k}$$

Observable canonical form

$$\mathbf{z}_{\omega,k} = \begin{pmatrix} 0 & 0 & \dots & a_r \\ 1 & \ddots & \ddots & \vdots \\ 0 & \ddots & 0 & a_2 \\ \dots & 0 & 1 & a_1 \end{pmatrix} \mathbf{z}_{\omega,k-1} + \begin{pmatrix} b_r + b_0 a_r \\ \vdots \\ b_2 + b_0 a_2 \\ b_1 + b_0 a_1 \end{pmatrix} \eta_{\omega,k-1}$$

$$y_{\omega,k} = (0 \dots 0 \ 1) \mathbf{z}_{\omega,k} + b_0 \eta_{\omega,k}$$

3.1.3 Stochastic Error Analysis

3.1.3.1 Allan Variance

For the characterization of the stability of oscillators and clocks the Allan variance and its variants are the methods of choice [14]. They are recommended in the IEEE standard 1139 [15] as preferred tool for the analysis of deviations from the nominal frequency of an oscillator over time. Even though the tools were explicitly derived for the time domain analysis of oscillator frequency noise like that of GNSS receivers or satellite clocks, it can be applied to all signals that are affected by colored noise, thus also to the measurement signals of navigation sensors like accelerometers or gyroscopes.

The Allan variance is the sample variance of the difference of two subsequent, over the time τ averaged values of the signal y

$$\sigma_{avar,y}^2 = \frac{1}{2(M-1)} \sum_{i=1}^{M-1} (\bar{y}_{i+1} - \bar{y}_i)^2 \quad (3.6)$$

Therein, M is the number of averaging windows and \bar{y}_i is the mean value of the i^{th} window

$$\bar{y}_i = \frac{1}{m} \sum_{k=1}^m y_{(i-1)m+k} \quad (3.7)$$

with the number of samples in each window

$$m = \frac{\tau}{\Delta t} \quad (3.8)$$

Δt is the base sample time of the signal. Before the Allan variance can be computed, mean values or other trends have to be removed from the signal.

If the frequency signal y is the time derivative of the phase signal x , it can be approximated by the difference quotient

$$y_i = \frac{dx_i}{dt} \approx \frac{x_{i+1} - x_i}{\Delta t} \quad (3.9)$$

With the phase signal x , that is the integral of the frequency signal y , the mean value (3.7) can be expressed as

$$\bar{y}_i = \frac{1}{m} \frac{x_{im} - x_{(i-1)m}}{\Delta t} \quad (3.10)$$

Inserted into the Allan variance (3.6) yields

$$\sigma_{avar,y}^2(\tau) = \frac{1}{2(M-2)\tau^2} \sum_{i=1}^{M-2} (x_{(i+2)m} - 2x_{(i+1)m} + x_{im})^2 \quad (3.11)$$

Usually, the Allan variance is computed with overlapping windows, which increases the number of used samples and hence provides smoother results

$$\sigma_{avar,y}^2(\tau) = \frac{1}{2(n-2m)\tau^2} \sum_{i=1}^{n-2} (x_{i+2m} - 2x_{i+m} + x_i)^2 \quad (3.12)$$

n is the number of signal samples. The computation of the Allan variance from the phase signal x is numerically much faster than the computation of the Allan variance from the frequency signal y . If only the frequency signal y is available, it can be numerically integrated for example either with the forward Euler or the trapezoidal integration scheme (modified Euler) to obtain the corresponding phase signal x . In the case of IMU sensors, the acceleration measurements \mathbf{f}_b and the angular rate measurements $\boldsymbol{\omega}_{ib}$ of non-integrating sensors correspond to the frequency signal y and the velocity increment measurements $\Delta\mathbf{v}_b$ and angle increment measurements $\Delta\boldsymbol{\theta}_{ib}$ of integrating sensors correspond to the phase signal x .

Besides the Allan variance, the Modified Allan variance $\sigma_{mvar,y}^2$ and the Hadamard variance $\sigma_{hvar,y}^2$ can be used for the analysis of noisy time series. In contrast to the standard Allan variance, the Modified Allan variance can distinguish between white phase and flicker phase noise. This will later play a role in the analysis of GNSS receiver oscillators and satellite clock errors, but not in the analysis of IMU measurements because the integrated IMU signals do usually not exhibit flicker noise. The Hadamard variance can cope with frequency signals with drifts and is thus particularly suitable for oscillators that are for example influenced by temperature variations during the measurement recording. If the Hadamard variance is applied, the mean value and drift do not have to be removed before as has to be done with the normal Allan variance.

Template 3-2 summarizes the calculation formulae of the overlapping versions of the Allan, Modified Allan and Hadamard variances of the signal y from the phase signal x .

Template 3-2: Overview over overlapping Allan variance variants [14]

Variance	Phase
Allan	$\sigma_{avar,y}^2 = \frac{1}{2(n-2m)\tau^2} \sum_{i=1}^{n-2m} (x_{i+2m} - 2x_{i+m} + x_i)^2$
Modified Allan	$\sigma_{mvar,y}^2 = \frac{1}{2m^2(n-3m+1)\tau^2} \sum_{j=1}^{n-3m+1} \left[\sum_{i=j}^{j+m-1} (x_{i+2m} - 2x_{i+m} + x_i) \right]^2$
Hadamard	$\sigma_{hvar,y}^2 = \frac{1}{6(n-3m)\tau^2} \sum_{i=1}^{n-3m} (x_{i+3m} - 3x_{i+2m} + 3x_{i+m} - x_i)^2$

3.1.3.2 Periodogram

Besides the Allan variance in time domain, the power spectral analysis of stochastic signals offers valuable clues to the distribution of signal power over frequency and time. The power spectral density (PSD) $S_{yy}(f)$ or $S_{yy}(\omega)$ describes the power of the signal $y(t)$ in dependence of the frequency f or angular frequency ω . A periodogram is an instantaneous estimate of the PSD of a signal. A spectrogram additionally illustrates the change of the PSD with time. Various non-parametric and parametric methods can be found in the literature (for example [16], [17], [18]) to estimate the PSD of a signal.

The two-sided PSD is defined by [16]

$$S_{yy}(f) = \lim_{T \rightarrow \infty} \left(\frac{1}{T} E \left[|Y(f, T)|^2 \right] \right) \quad (3.13)$$

with the finite-range Fourier transform of the signal $y(t)$

$$Y(f, T) = \int_0^T y(t) e^{-j2\pi f t} dt \quad (3.14)$$

The continuous-time signal y is sampled at discrete time steps

$$y = y(t) \Leftrightarrow y_k = y(k \Delta t), \quad k = 0, \dots, n-1 \quad (3.15)$$

where n is the number of samples and Δt the sampling time of the signal. The basic way to estimate the periodogram is to calculate the discrete Fourier transform

$$Y_k = \sum_{i=0}^{n-1} y_i e^{-j2\pi \frac{ik}{n}} \quad (3.16)$$

for example by Fast Fourier Transformation (FFT) and to compute the two-sided PSD subsequently by

$$S_{yy}(f_k) = \frac{\Delta t}{n} |Y_k|^2 \quad (3.17)$$

for the frequencies

$$f_k = \frac{k}{n} f_s \quad (3.18)$$

where $f_s = 1/\Delta t$ is the sample rate of the signal. The variance of this basic PSD estimate can be reduced by means of Bartlett's method or Welch's method. Bartlett's method splits the available data samples in smaller non-overlapping windows, computes the periodogram for each window and finally averages the obtained periodograms. Due to the smaller windows the frequency range of the periodogram is limited at low frequencies. In contrast to Bartlett's method, Welch allows an overlap of the data windows and thus provides even more smoothed periodograms. Bartlett's and Welch's methods are especially interesting if the signal exhibits ergodic character. Note that the two-sided PSD S_{yy} is used throughout this thesis in lieu of the one-sided PSD G_{yy} if the type is not especially specified. If it is referred to the one-sided PSD $G_{yy} = 2S_{yy}$, this is explicitly indicated. Welch's method is recapitulated in Template 3-3.

Template 3-3: Power spectral density estimation with Welch's method

Discrete Fourier transform of signal y

$$Y_{k,i} = \sum_{l=1}^m y_{(i-1)m+l} e^{-j2\pi \frac{k(l-1)}{m}} \quad k = 0, 1, \dots, m-1$$

Power spectral density estimate

$$S_{yy}(f_k) = \frac{1}{M} \frac{\Delta t}{m} \sum_{i=1}^M |Y_{k,i}|^2$$

3.1.4 Power-Law Noise

3.1.4.1 Overview

Power-law noise exhibits a PSD S_{yy} that is proportional to the frequency f raised to the power of α

$$S_{yy}(f) \sim f^\alpha \quad (3.19)$$

Power-law noise is typical for the frequency variations of oscillators but can also be found in IMU signals. The frequency noise of an oscillator and the noise on non-integrating accelerometer and gyroscope measurements is usually composed of power-law noises with exponents α ranging from -2 to +2, that is in sequence the random walk frequency noise, flicker frequency noise, white frequency noise, flicker phase noise and white phase noise with PSD

$$S_{yy}(f) = \begin{cases} \frac{1}{2} \sum_{\alpha=-2}^2 h_\alpha f^\alpha, & 0 \leq f \leq f_s \\ 0, & f > f_s \end{cases} \quad (3.20)$$

h_α is the gain of the noise component with f^α proportional characteristic. The nomenclature is chosen according to the IEEE standard 1139 [15]. The factor $1/2$ is due to the fact that in the definition of [15] the one-sided power spectral density $G_{yy}(f) = 2S_{yy}(f)$ is used instead of the two-sided one.

Time integration of white frequency noise ($\alpha = 0$) results in random walk frequency noise ($\alpha = -2$). Time differentiation of white frequency noise yields white phase noise ($\alpha = 2$). This means that integration decreases and differentiation increases the power α by two. In Laplace domain, integration corresponds to the transfer function $1/s$ and differentiation to the transfer function s . Consequently, in order to obtain flicker frequency noise ($\alpha = -1$) from white frequency noise, thus decreasing the

power α by one, white input noise would have to be filtered by a $1/\sqrt{s}$ transfer function. Likewise, for flicker phase noise ($\alpha=1$) the white input noise would have to be filtered by a \sqrt{s} transfer function. Since these transfer functions are not rational, flicker frequency noise and flicker phase noise cannot be simply created but have to be approximated as described in the next section.

Power-law noise can also be classified by its color. Noise with $\alpha=-2$ is called brown, noise with $\alpha=-1$ pink, noise with $\alpha=0$ white, noise with $\alpha=1$ blue and noise with $\alpha=2$ violet.

Table 3-1 lists the power spectral densities and the corresponding Allan and Modified Allan variances of the five power-law noises. The coefficients $A - E$ given in the table are valid for averaging times $\tau \gg \Delta t/(2\pi)$. In Table 3-2 the power spectral densities, exemplary realizations and the autocorrelation functions are given for illustration purposes.

Table 3-1: Power spectral densities and Allan variances of power-law noise components [15]

Noise Component	PSD $S_{yy}(f)$	PSD $S_{xx}(f)$	Allan $\sigma_{avar,y}^2(\tau)$	Mod. Allan $\sigma_{mvar,y}^2(\tau)$
Random walk frequency ω_{RWF}	$\frac{1}{2} h_{-2} f^{-2}$	$(2\pi)^2 \frac{1}{2} h_{-2} f^{-4}$	$A h_{-2} \tau^1$	$A h_{-2} \tau^1$
Flicker frequency ω_{FF}	$\frac{1}{2} h_{-1} f^{-1}$	$(2\pi)^2 \frac{1}{2} h_{-1} f^{-3}$	$B h_{-1} \tau^0$	$B h_{-1} \tau^0$
White frequency ω_{WF}	$\frac{1}{2} h_0 f^0$	$(2\pi)^2 \frac{1}{2} h_0 f^{-2}$	$C h_0 \tau^{-1}$	$C h_0 \tau^{-1}$
Flicker phase ω_{FP}	$\frac{1}{2} h_1 f^1$	$(2\pi)^2 \frac{1}{2} h_1 f^{-1}$	$D h_1 \tau^{-2}$	$D h_1 \tau^{-2}$
White phase ω_{WP}	$\frac{1}{2} h_2 f^2$	$(2\pi)^2 \frac{1}{2} h_2 f^0$	$E h_2 \tau^{-2}$	$E h_2 \tau^{-3}$
$A = \frac{2\pi^2}{3}$	$B = 2\ln 2$	$C = \frac{1}{2}$	$D = \frac{1.038 + 3\ln(2\pi f_s \tau)}{4\pi^2}$	$E = \frac{3f_s}{4\pi^2}$

3.1.4.2 Flicker Noise Modeling

As has been addressed in the previous section pink flicker noise cannot be simply generated by integration of white noise. The bias instability of accelerometers and gyroscopes is flicker noise. Therefore, a shaping filter that approximates the $1/\sqrt{s}$ transfer function is searched. The filter to be designed is fed by white noise and outputs a random signal with a PSD proportional to $1/f$. The following sketch is based on [19].

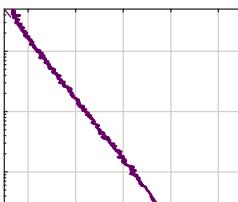
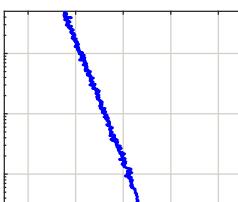
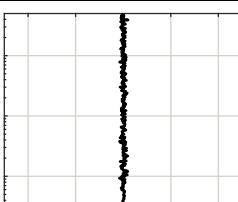
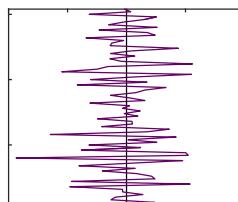
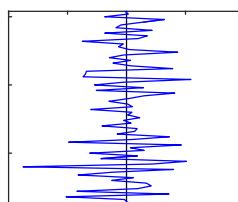
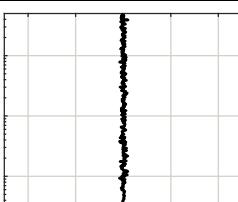
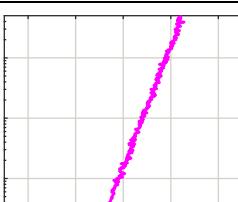
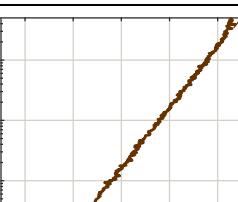
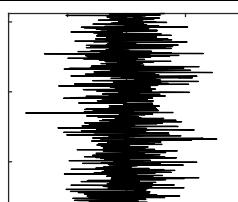
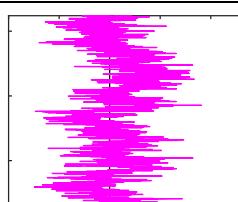
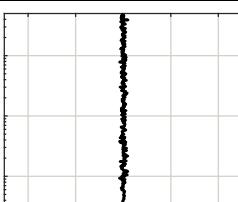
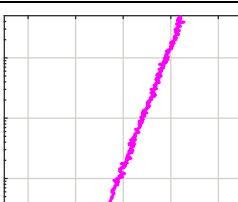
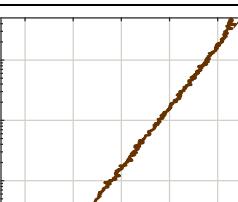
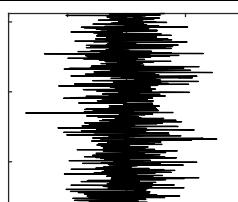
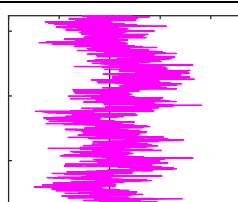
The power spectral density of the white, Gaussian input noise $\eta \sim WN(0, \sigma_\eta^2)$ is converted by the shaping filter according to

$$S_{yy}(f) = |H(f)|^2 \sigma_\eta^2 = \frac{1}{f^\beta} \sigma_\eta^2 \quad (3.21)$$

where $H(f)$ is the frequency response function of the shaping filter.

In discrete-time a transfer function $H(z)$ – represented by an adequate ARMA model – is searched that approximates the $f^\alpha = f^{-\beta}$ behavior over a wide frequency range. In [19] it is shown that transfer functions of numerical integrators (for example backward Euler, trapezoidal) raised to the power of $\beta/2$ have power spectral densities that are proportional to $\sin(2\pi f \Delta t)^{-\beta}$ and thus approximate the $f^{-\beta}$ characteristics for low frequencies adequately.

Table 3-2: Illustration of colored noise components

Type	Random walk frequency	Flicker frequency	White frequency	Flicker phase	White phase
PSD					
Example					
Correlation					
Model	$\omega(t) = \int \eta(t) dt$		$\omega(t) = \eta(t)$	n/a	$\omega(t) = \dot{\eta}(t)$
Transfer function	$\frac{1}{s}$	$\frac{1}{\sqrt{s}}$	1	\sqrt{s}	s
Stationary	no	no	yes	yes	yes
Color	brown	pink	white	blue	violet

A general second order numerical integration scheme evaluates the integrand function f at two subsequent time steps separated by the step size Δt and multiplies the sum of the weighted function values $k_1 f(t - \Delta t)$ and $k_2 f(t)$ by Δt

$$\int_{t-\Delta t}^t f(\tau) d\tau \approx \Delta t (k_1 f(t - \Delta t) + k_2 f(t)) \quad (3.22)$$

with arbitrary weighting factors k_1 and k_2 . The transfer function of the general 2-point numerical integrator (3.22) is given with

$$H(z) = \Delta t \frac{k_2 z + k_1}{z - 1} \quad (3.23)$$

and raised to the power of $\beta/2$ is further

$$H^{\frac{\beta}{2}}(z) = \Delta t^{\frac{\beta}{2}} \left(\frac{k_2 z + k_1}{z - 1} \right)^{\frac{\beta}{2}} \quad (3.24)$$

Now, a substitute ARMA(p, q) model with transfer function

$$\tilde{H}(z) = \frac{\sum_{j=0}^q b_j z^{-j}}{1 - \sum_{i=1}^p a_i z^{-i}} \quad (3.25)$$

is searched whose impulse response $\tilde{h}[n]$ approximates the impulse response $h[n]$ of (3.24). The coefficients a_i and b_j are determined by minimizing the error

$$e[n] = h[n] - \tilde{h}[n] \quad (3.26)$$

The coefficients can be estimated with Prony's method [20] or the Steiglitz McBride method [21]. The latter turned out to yield slightly better results and is recommended. In [19] it has been found that the Al-Alaoui integrator [22] with $k_1 = 1/8$ and $k_2 = 7/8$ yields best results. In Table 3-3 and Table 3-4 the estimated denominator and numerator coefficients of flicker noise ARMA models ($\beta = 1$) with different model orders are listed. p and q are equal. $N = 25000$ samples have been used to represent the impulse response $h[n]$. In Table 3-5 the corresponding poles of the ARMA models are given. All poles are real and approximate +1. The derived models are thus stable and have a steady-state solution. The power spectral densities of generated noise with $\beta = 1$ with orders 1 to 4 are shown in Figure 3-1. The models with orders 1 and 2 feature deviations from the ideal $1/f$ characteristic, but the models with orders 3 and 4 already yield acceptable results. The model of order 4 approximates the $1/f$ behavior in the low frequency range slightly better than the third order model.

The state-space model in controllable canonical form for flicker noise is

$$\begin{aligned} \mathbf{z}_{FF,k} &= \underbrace{\begin{pmatrix} 0 & 1 & 0 & \cdots \\ 0 & \ddots & \ddots & 0 \\ & & 0 & 1 \\ a_{FF,p} & \cdots & a_{FF,2} & a_{FF,1} \end{pmatrix}}_{\Phi_{FF}} \mathbf{z}_{FF,k-1} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}}_{\Gamma_{FF}} \eta_{FF,k-1} \\ \omega_{FF,k} &= \underbrace{\Delta t^{\frac{1}{2}} (b_{FF,p} + b_{FF,0} a_{FF,p}, \dots, b_{FF,1} + b_{FF,0} a_{FF,1})}_{\mathbf{H}_{FF}} \mathbf{z}_{FF,k} + \underbrace{\Delta t^{\frac{1}{2}} b_{FF,0}}_{D_{FF}} \eta_{FF,k} \end{aligned} \quad (3.27)$$

Due to numerical stability it is strongly recommended to transform the state-space model from controllable canonical form to modal form.

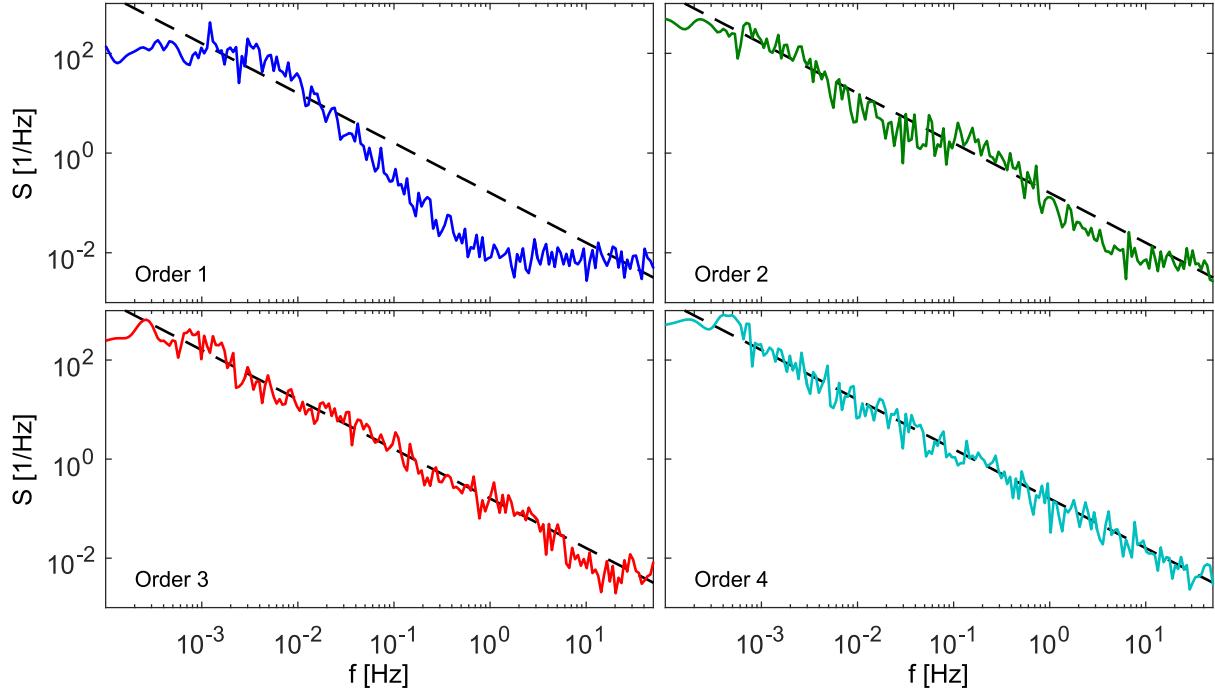


Figure 3-1: Power spectral densities of $1/f$ noise generated with models of orders 1 to 4

Table 3-3: Estimated AR coefficients ($\beta/2 = 0.5$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 25000$)

$p = q$	a_1	a_2	a_3	a_4
1	0.9997082580			
2	1.9842171974	-0.9842186842		
3	2.8964260001	-2.7931352266	0.8967092093	
4	3.7263703996	-5.1839351397	3.1887538680	-0.7311891282

Table 3-4: Estimated MA coefficients ($\beta/2 = 0.5$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 25000$)

$p = q$	b_0	b_1	b_2	b_3	b_4
1	0.9354143467	-0.8995190356			
2	0.9354143489	-1.6422190110	0.7070845809		
3	0.9354142380	-2.2995075441	1.8011773013	-0.4370802553	
4	0.9354057741	-2.9850466277	3.3940470875	-1.5743519333	0.2299457555

Table 3-5: Poles of identified ARMA models ($\beta/2 = 0.5$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 25000$)

$p = q$	λ_1	λ_2	λ_3	λ_4
1	0.9997			
2	0.9999	0.9843		
3	0.9999	0.9973	0.8992	
4	0.9999	0.9980	0.9742	0.7025

3.1.4.3 Power-Law Noise Models

Each of the power-law noises of Table 3-1 can be described by a general discrete-time state-space model

$$\begin{aligned}\mathbf{z}_{X,k} &= \Phi_X \mathbf{z}_{X,k-1} + \Gamma_X \eta_{X,k-1} & X = \{RWF, FF, WF, FP, WP\} \\ \omega_{X,k} &= \mathbf{H}_X \mathbf{z}_{X,k} + D_X \eta_{X,k}\end{aligned}\quad (3.28)$$

where $\eta_{X,k}$ is white, Gaussian noise, $\eta_{X,k} \sim WN(0, \sigma_{\eta,X}^2)$. Table 3-6 lists the system matrices and input noise variances for the power-law noise components on the rate signal y . Table 3-7 lists the corresponding system matrices and input noise variances for the power-law noise components on the integrated rate signal x . The forward Euler integration scheme is used for integrations. The time derivatives are replaced by simple difference quotients. Flicker noise is modeled according to (3.27).

Table 3-6: System matrices and input noise variances for power-law noise on rate signal y

Noise Component	X	Φ_X	Γ_X	\mathbf{H}_X	D_X	$\sigma_{\eta,X}^2$
Random walk frequency	ω_{RWF}	1	Δt	1	0	$(2\pi)^2 \frac{\frac{1}{2}h_{-2}}{\Delta t}$
Flicker frequency	ω_{FF}	Φ_{FF}	Γ_{FF}	\mathbf{H}_{FF}	D_{FF}	$2\pi \frac{\frac{1}{2}h_{-1}}{\Delta t}$
White frequency	ω_{WF}	0	0	0	1	$\frac{\frac{1}{2}h_0}{\Delta t}$
Flicker phase	ω_{FP}	$\begin{pmatrix} 0 & \mathbf{H}_{FF} \\ 0 & \Phi_{FF} \end{pmatrix}$	$\begin{pmatrix} D_{FF} \\ \Gamma_{FF} \end{pmatrix}$	$\begin{pmatrix} -1 & \mathbf{H}_{FF} \\ \Delta t & \Delta t \end{pmatrix}$	$\frac{D_{FF}}{\Delta t}$	$\frac{1}{2\pi} \frac{\frac{1}{2}h_1}{\Delta t}$
White phase	ω_{WP}	0	1	$-\frac{1}{\Delta t}$	$\frac{1}{\Delta t}$	$\frac{1}{(2\pi)^2} \frac{\frac{1}{2}h_2}{\Delta t}$

Table 3-7: System matrices and input noise variances for power-law noise on integrated signal x

Noise Component	X	Φ_X	Γ_X	\mathbf{H}_X	D_X	$\sigma_{\eta,X}^2$
Random walk frequency	ω_{RWF}	$\begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 \\ \Delta t \end{pmatrix}$	$(1 \ 0)$	0	$(2\pi)^2 \frac{\frac{1}{2}h_{-2}}{\Delta t}$
Flicker frequency	ω_{FF}	$\begin{pmatrix} 1 & \Delta t \mathbf{H}_{FF} \\ 0 & \Phi_{FF} \end{pmatrix}$	$\begin{pmatrix} \Delta t D_{FF} \\ \Gamma_{FF} \end{pmatrix}$	$(1 \ 0)$	0	$2\pi \frac{\frac{1}{2}h_{-1}}{\Delta t}$
White frequency	ω_{WF}	1	Δt	1	0	$\frac{\frac{1}{2}h_0}{\Delta t}$
Flicker phase	ω_{FP}	Φ_{FF}	Γ_{FF}	\mathbf{H}_{FF}	D_{FF}	$\frac{1}{2\pi} \frac{\frac{1}{2}h_1}{\Delta t}$
White phase	ω_{WP}	0	0	0	1	$\frac{1}{(2\pi)^2} \frac{\frac{1}{2}h_2}{\Delta t}$

3.1.5 General Noise

3.1.5.1 Overview

In the previous section special noise with power-law power spectral densities was thoroughly discussed. Now, the modeling of noise with more general power spectral densities is considered more closely. Vibrational noise on accelerometer measurements, for example, may not exhibit a power spectral density that follows power-law but normally has an arbitrary power spectral density with power at specific frequencies or smeared over wider frequency bands.

In this section it is shown how to represent general noise by an autoregressive model. For that, the Yule-Walker equations, which will be presented in the beginning, have to be solved. One method to estimate the AR model coefficients amongst others is Burg's method.

3.1.5.2 Yule-Walker Equations

It is assumed that the sampled signal $y[n]$ can be described by an AR(p)-model (3.2). An adequate model order has to be chosen. The higher the order p , the more details of the power spectral density will be covered by the model. We limit ourselves to AR models since the estimation of ARMA models, especially online, is more involved.

The autocorrelation r_{yy} of the data sequence $y[n]$ is defined as

$$r_{yy}(n, m) := E[y[n]y[n-m]] \quad (3.29)$$

It is assumed that the stochastic process within the current window is ergodic. Then, the autocorrelation r_{yy} is independent of the absolute time n but depends only on the relative time difference m

$$r_{yy}(m) = E[y[n+m]y[n]] = E[y[n]y[n-m]] \quad (3.30)$$

or alternatively expressed

$$r_{yy}(m) = r_{yy}(-m) \quad (3.31)$$

With (3.2) the autocorrelation sequence of a stationary AR process is consequently

$$\begin{aligned} E[y[n]y[n-m]] &= \sum_{i=1}^b a_{p,i} E[y[n-i]y[n-m]] + E[\eta[n]\eta[n-m]] \\ &= \sum_{i=1}^b a_{p,i} E[y[n-i]y[n-i-(m-i)]] + \delta_{m0} E[\eta[n]\eta[n]] \end{aligned} \quad (3.32)$$

or simplified with $E[\eta[n]\eta[n]] = Q_\eta$

$$r_{yy}(m) = \sum_{i=1}^b a_{p,i} r_{yy}(m-i) + \delta_{m0} Q_\eta \quad (3.33)$$

The autocorrelation function can be written for time offsets $m = 0 \dots p$ and combined in a linear equation system

$$\begin{pmatrix} r_{yy}(0) & r_{yy}(-1) & \cdots & r_{yy}(-p) \\ r_{yy}(1) & r_{yy}(0) & \cdots & r_{yy}(-p+1) \\ \vdots & \vdots & \ddots & \vdots \\ r_{yy}(p) & \cdots & \cdots & r_{yy}(0) \end{pmatrix} \begin{pmatrix} -1 \\ a_{p,1} \\ \vdots \\ a_{p,p} \end{pmatrix} = \begin{pmatrix} -Q_\eta \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (3.34)$$

These equations are known as the Yule-Walker normal equations. $a_{p,i}$, $i = 1 \dots p$, are the coefficients of an AR(p) model.

It can be shown that the AR coefficients of the AR($p-1$) and AR(p) models are related by the reflection coefficients k_p [17]

$$a_{p,i} = \begin{cases} -1 & i = 0 \\ a_{p-1,i} + k_p a_{p-1,p-i} & i = 1, \dots, p \\ 0 & i = p+1 \end{cases} \quad (3.35)$$

3.1.5.3 Solution Methods for the Yule-Walker Equations

Two different classes of solution approaches can be found in the literature (for example [17] [23]), the correlation function estimation methods and the reflection coefficient estimation methods. The former directly solve the Yule-Walker equations (3.34) for the coefficients $a_{p,i}$ and the latter estimate the reflection coefficients k_p as defined in (3.35). The relation with the reflection coefficients is convenient because it enables the iterative estimation of the model coefficients, beginning with a simple model of order one and iterating estimation until the desired order p . In each iteration step the reflection coefficients k_p have to be determined again. An additional benefit is that the model order can be flexibly adapted because all AR models up to the order p are available due to the recursion.

Table 3-8 lists some of the common solution methods. Burg's method is the favorable method and will be outlined in the following section.

Table 3-8: Solution approaches of the Yule-Walker equations

Class	Methods
Correlation Function Estimation	Autocorrelation method (Yule-Walker method)
Reflection Coefficient Estimation	Forward covariance method Backward covariance method Modified covariance method Burg's method

3.1.5.4 Burg's method

Burg's method estimates the AR coefficients such that the mean value of the sample variances of the forward and backward prediction errors is minimized. Template 3-4 summarizes the equations of Burg's method [17].

Template 3-4: Burg's method

Initialization	
$e_{f,0}[n] = y[n]$	Initial forward prediction error
$e_{b,0}[n] = y[n]$	Initial backward prediction error
$\hat{a}_{0,0} = 1$	Initial AR model parameter
$Q_{\eta,0} = \frac{1}{N} \sum_{i=1}^N y^2[n]$	Initial input noise variance
Recursion $i = 1 \dots p$	
$k_i = \frac{-2 \sum_{n=i+1}^N e_{f,i-1}[n]e_{b,i-1}[n-1]}{\sum_{n=i+1}^N (e_{f,i-1}^2[n] + e_{b,i-1}^2[n-1])}$	Reflection coefficient
$e_{f,i}[n] = e_{f,i-1}[n] + k_i e_{b,i-1}[n-1]$	Update of the forward prediction error
$e_{b,i}[n] = e_{b,i-1}[n-1] + k_i e_{f,i-1}[n]$	Update of the backward prediction error
$\begin{pmatrix} \hat{a}_{i,0} \\ \vdots \\ \hat{a}_{i,i-1} \\ \hat{a}_{i,i} \end{pmatrix} = \begin{pmatrix} \hat{a}_{i-1,0} \\ \vdots \\ \hat{a}_{i-1,i-1} \\ 0 \end{pmatrix} + k_i \begin{pmatrix} 0 \\ \hat{a}_{i-1,i-1} \\ \vdots \\ \hat{a}_{i-1,0} \end{pmatrix}$	Update of the AR parameter
$Q_{\eta,i} = \left(1 - \sum_{j=1}^i k_j^2\right) Q_{\eta,i-1}$	Update of the input noise variance

3.1.5.5 Online Estimation

Burg's method is a batch algorithm, meaning that the AR coefficients are estimated only after all samples of the current estimation window are collected. Alternatively, sequential algorithms could be used that directly update the AR coefficient estimates immediately when a new measurement sample comes in. In [17] a recursive variant of Burg's method is available.

3.2 Inertial Measurement Unit Error

3.2.1 Motivation

An IMU consists of at least an accelerometer triad and a gyroscope triad and forms the core of the integrated navigation system. The measured accelerations and angular rates are integrated by means of the inertial navigation algorithms to obtain the inertial navigation solution. Measurement errors are irrevocably accumulated and deteriorate the inertial navigation solution more and more. Therefore, it is important to consider the most influential errors to predict the uncertainty growth of the position, velocity and orientation solution properly. This helps to improve the statistical consistency of the integrated navigation solution and is particularly crucial if the navigation system runs unaided for a longer time. In order to incorporate the knowledge about the measurement errors into the integrated navigation filter, appropriate models will be derived in this section. Detailed performance and testing requirements are provided by IEEE, for example in [24] for single-axis laser gyroscopes. A standardized terminology is introduced in [25]. Thorough descriptions of the sensing technologies and prominent errors can for example be found in [8] [11] [26] [27]. An assessment of the most relevant errors of low-cost IMU is given in [28]. A good motivation for the need of stochastic IMU error models for the improvement of the integrated navigation solution is presented in [29] [30] [31] [32].

Non-integrating and integrating IMU

It is distinguished between non-integrating and integrating sensors. Non-integrating accelerometers directly measure accelerations \mathbf{f}_b whereas integrating ones output velocity increments $\Delta\mathbf{v}_b$. Non-integrating gyroscopes directly measure angular rates $\boldsymbol{\omega}_{ib}$ whereas integrating ones output angle increments $\Delta\boldsymbol{\theta}_{ib}$.

The servos that are used in a mechanical sensor to keep the proof mass in its zero position can be analog or digital. In the former case the servo is driven by a continuous current whose instantaneous power is proportional to the input acceleration in accelerometers and angular rate in gyroscopes. Sensors with analog servos are thus generally non-integrating sensors. In the latter case short positive or negative pulses with constant currency and voltage and fix time length are applied to drive the servo. One pulse corresponds to a certain positive or negative increment of electric energy, which is proportional to a velocity or angle increment. The measurement is the sum of positive and negative pulses per sampling period. Sensors with digital servos are thus generally integrating sensors [26].

Accelerometers actually do not measure the acceleration w.r.t. the inertial frame \mathbf{v}_i , but the specific net force of all external forces that act on the sensor's proof mass. Non-integrating IMU measurements directly serve as input for the navigation differential equations in e - or n -frame (2.1) or (2.3), which are integrated for example with forward Euler or Runge-Kutta integration schemes. Integrating measurements can either be first converted to non-integrating ones and then used with the equations for non-integrating IMU or can be directly processed by algorithms that have been derived especially for integrating IMU, as will be explained later in section 4.1.

Many sensors sample the analog measurement signal internally with a higher rate than the rate with which they output the sensed value (oversampling). If the sensor is non-integrating, it calculates the mean value of the gathered samples between two subsequent outputs

$$\bar{\mathbf{f}}_b(t_k) = \frac{1}{n} \sum_{i=1}^n \mathbf{f}_b(t_{k-1} + i \Delta\tau), \quad \bar{\boldsymbol{\omega}}_{ib}(t_k) = \frac{1}{n} \sum_{i=1}^n \boldsymbol{\omega}_{ib}(t_{k-1} + i \Delta\tau), \quad n = \frac{\Delta t}{\Delta\tau} \quad (3.36)$$

where $\Delta\tau$ is the internal sample time and Δt is the output sample time. If the sensor is integrating, it has to be distinguished between sensors that actually measure increments due to their physical measurement principle

$$\Delta\mathbf{v}_b(t_k) = \int_{t_{k-1}-\Delta t}^{t_k} \mathbf{f}_b(\tau) d\tau, \quad \Delta\boldsymbol{\theta}_{ib}(t_k) = \int_{t_{k-1}-\Delta t}^{t_k} \boldsymbol{\omega}_{ib}(\tau) d\tau \quad (3.37)$$

and those that internally sample non-integrated values with a high rate and then integrate these gathered measurements numerically within one output time step

$$\Delta\mathbf{v}_b(t_k) = \sum_{i=1}^n \mathbf{f}_b(t_{k-1} + i \Delta\tau) \Delta\tau, \quad \Delta\boldsymbol{\theta}_{ib}(t_k) = \sum_{i=1}^n \boldsymbol{\omega}_{ib}(t_{k-1} + i \Delta\tau) \Delta\tau, \quad n = \frac{\Delta t}{\Delta\tau} \quad (3.38)$$

if for example the backward Euler integration scheme is used. Ring laser gyroscopes (RLG) are true rate-integrating sensors and thus belong to the first group whereas fiber optical gyroscopes (FOG) are rate sensors. However, these sensors often output angle increments, too, and are hence members of the second group. The internal sample rate is mostly between one and two Kilohertz. According to whatever principle the sensor works, the output accelerations (or velocity increments) are affected by the changing orientation of the sensor reference frame during the oversampling or integration period due to the angular rate of the carrier platform. This means that the measurement value cannot be assigned to a unique orientation of the sensor reference frame. The inertial navigation algorithms have to account for this matter, which is especially eminent in applications with high rotational dynamics and coning and sculling motion.

The true, error-free non-integrating accelerometer measurement \mathbf{f}_b , the erroneous accelerometer measurement $\tilde{\mathbf{f}}_b$ and the accelerometer measurement error $\delta\mathbf{f}_b$ are related by

$$\mathbf{f}_b(t) = \tilde{\mathbf{f}}_b(t) + \delta\mathbf{f}_b(t) \quad (3.39)$$

The definition is according to the sign convention of perturbations used throughout this thesis. The error definition of the non-integrating gyroscope measurement $\boldsymbol{\omega}_{ib}$ is analogous

$$\boldsymbol{\omega}_{ib}(t) = \tilde{\boldsymbol{\omega}}_{ib}(t) + \delta\boldsymbol{\omega}_{ib}(t) \quad (3.40)$$

Even if integrating IMU measurements are processed, the well-known navigation error differential equations for non-integrating IMU (2.6) or (2.7) are used in the navigation error filter. The non-integrating IMU errors like biases and scale factor errors that have been estimated by the integrated navigation filter have to be multiplied by the sample time Δt before they can be used for in-flight calibration

$$\Delta\mathbf{v}_b(t) = \Delta\tilde{\mathbf{v}}_b(t) + \delta\mathbf{f}_b(t)\Delta t, \quad \Delta\boldsymbol{\theta}_{ib}(t) = \Delta\tilde{\boldsymbol{\theta}}_{ib}(t) + \delta\boldsymbol{\omega}_{ib}(t)\Delta t \quad (3.41)$$

Consideration of other disturbing influences

Other disturbing effects besides the sensor measurement errors have to be taken into account to obtain a highly accurate result. Higher grade IMU usually accommodate separate sensors for each axis. Due to their extent the sensors cannot be mounted at the same location in the IMU housing. This does not play a role for the angular rates but has to be accounted for when processing the

accelerometer measurements. The accelerations have to be corrected by the internal lever arms because of the different centrifugal accelerations at the different sensing locations.

Non-nominal errors

In this section models for nominal IMU measurement errors will be derived. However, non-nominal faults might occur and have to be detected and isolated with high probability if the specified system performance has to be guaranteed. A simplex IMU features three orthogonal acceleration and angular rate sensing axes. Due to the missing redundancy, faults cannot be autonomously detected by the IMU. For safety critical applications, however, fault detection (and isolation) is a relevant issue. Therefore, these applications make use of skew-redundant IMU (SRIMU) with more than three, not compulsorily orthogonal measurement axes. Accuracy and integrity are increased since the effective noise variance is reduced and faults can be detected because of the redundancy in each axis. In principle, the same fault detection and isolation mechanisms, amongst others the parity space method, are used as in the receiver autonomous integrity monitoring (RAIM) of satellite navigation. The dodecahedron (as five sided frustum with face angles of 58.28°) offers for example the optimal geometry for six axes configurations. If the measurements of the single axes of a SRIMU are converted to three orthogonal values, the same inertial navigation algorithms as with the simplex IMU can be used.

3.2.2 Characteristics

3.2.2.1 Accelerometer

Based on [33] the non-integrating accelerometer measurement model is given with

$$\mathbf{f}_b(t) = (\mathbf{I}_3 + \mathbf{S}_f + \mathbf{M}_f) \tilde{\mathbf{f}}_b(t) + \mathbf{b}_f(t) + \boldsymbol{\varepsilon}_f(t) \quad (3.42)$$

where

\mathbf{S}_f	Scale factor error
\mathbf{M}_f	Misalignment
\mathbf{b}_f	Bias
$\boldsymbol{\varepsilon}_f$	Non-modeled errors, higher order terms

$$\begin{aligned} \mathbf{S}_f &= \mathbf{S}_{0,f} + \mathbf{S}_{T,f}(T - T_{ref}) \\ \mathbf{S}_{0,f} &\quad \text{Scale factor error at reference temperature } T_{ref} \\ \mathbf{S}_{T,f} &\quad \text{Scale factor error temperature sensitivity} \end{aligned} \quad (3.43)$$

$$\begin{aligned} \mathbf{b}_f(t) &= \mathbf{b}_{0,f} + \mathbf{b}_{T,f}(T - T_{ref}) + \mathbf{b}_{Q,f}(t) + \mathbf{b}_{N,f}(t) + \mathbf{b}_{B,f}(t) + \mathbf{b}_{K,f}(t) \\ \mathbf{b}_{0,f} &\quad \text{Turn-on bias at reference temperature } T_{ref} \\ \mathbf{b}_{T,f} &\quad \text{Bias temperature sensitivity} \\ \mathbf{b}_{Q,f} &\quad \text{Output quantization (velocity white noise if integrating sensor)} \\ &\quad \text{and acceleration white noise if non-integrating sensor)} \\ \mathbf{b}_{N,f} &\quad \text{Velocity random walk (acceleration white noise)} \\ \mathbf{b}_{B,f} &\quad \text{Bias instability (acceleration pink noise)} \\ \mathbf{b}_{K,f} &\quad \text{Acceleration random walk (acceleration brown noise)} \end{aligned}$$

Other error terms like ramps $\mathbf{b}_{R,f}$ or additional Markov noise $\mathbf{b}_{M,f}$ are omitted in the model above because of their minor influence on the navigation solution but can of course be easily added to \mathbf{b}_f if

required for example in highly accurate applications. Non-linearities for example of the scale factor and vibration rectification are contained in $\boldsymbol{\epsilon}_f$. A thorough discussion of the physical cause of the single errors is given in [26] for many sensor types.

The scale factor error matrix \mathbf{S}_f is defined as

$$\mathbf{S}_f = \text{diag}(\mathbf{s}_f) = \begin{pmatrix} s_{f,x} & 0 & 0 \\ 0 & s_{f,y} & 0 \\ 0 & 0 & s_{f,z} \end{pmatrix}, \quad \mathbf{s}_f = \begin{pmatrix} s_{f,x} \\ s_{f,y} \\ s_{f,z} \end{pmatrix} \quad (3.44)$$

Misalignment errors stem from non-orthogonal measurement axes due to imperfections of the assembly of the sensor triad. As shown in Figure 3-2, the misalignments can be represented by six independent tilt angles $\delta\varphi_y$, $\delta\varphi_z$, $\delta\vartheta_x$, $\delta\vartheta_z$, $\delta\psi_x$ and $\delta\psi_y$. The nomenclature is as follows: the subindex x , y , z specifies the concerned axis whereas φ , ϑ , ψ defines about which axis the concerned axis is tilted. The misalignment matrix is defined by

$$\mathbf{M}_f \tilde{\mathbf{f}}_b = \begin{pmatrix} 0 & -\delta\psi_{f,y} & \delta\vartheta_{f,z} \\ \delta\varphi_{f,x} & 0 & -\delta\varphi_{f,z} \\ -\delta\vartheta_{f,x} & \delta\varphi_{f,y} & 0 \end{pmatrix} \tilde{\mathbf{f}}_b = \underbrace{\begin{pmatrix} \tilde{f}_{b,z} & -\tilde{f}_{b,y} & 0 & 0 & 0 & 0 \\ 0 & 0 & \tilde{f}_{b,x} & -\tilde{f}_{b,z} & 0 & 0 \\ 0 & 0 & 0 & 0 & \tilde{f}_{b,y} & -\tilde{f}_{b,x} \end{pmatrix}}_{\mathbf{F}_{M_f}} \underbrace{\begin{pmatrix} \delta\vartheta_{f,z} \\ \delta\psi_{f,y} \\ \delta\varphi_{f,x} \\ \delta\varphi_{f,z} \\ \delta\varphi_{f,y} \\ \delta\vartheta_{f,x} \end{pmatrix}}_{\mathbf{m}_f} \quad (3.45)$$

(3.42) solved for the acceleration error $\delta\mathbf{f}_b$ becomes with (3.43), (3.44) and (3.45)

$$\begin{aligned} \delta\mathbf{f}_b(t) &\doteq \mathbf{b}_f(t) + (\mathbf{S}_f + \mathbf{M}_f)\tilde{\mathbf{f}}_b(t) \\ &= \mathbf{b}_{0,f} + \mathbf{b}_{T,f}(T - T_{ref}) + \sum_{i=\{Q,N,B,K\}} \mathbf{b}_{i,f}(t) + \text{diag}(\tilde{\mathbf{f}}_b(t))(\mathbf{s}_{0,f} + \mathbf{s}_{T,f}(T - T_{ref})) + \mathbf{F}_{M_f} \mathbf{m}_f \end{aligned} \quad (3.46)$$

Therein, the non-modeled error $\boldsymbol{\epsilon}_f$ is neglected.

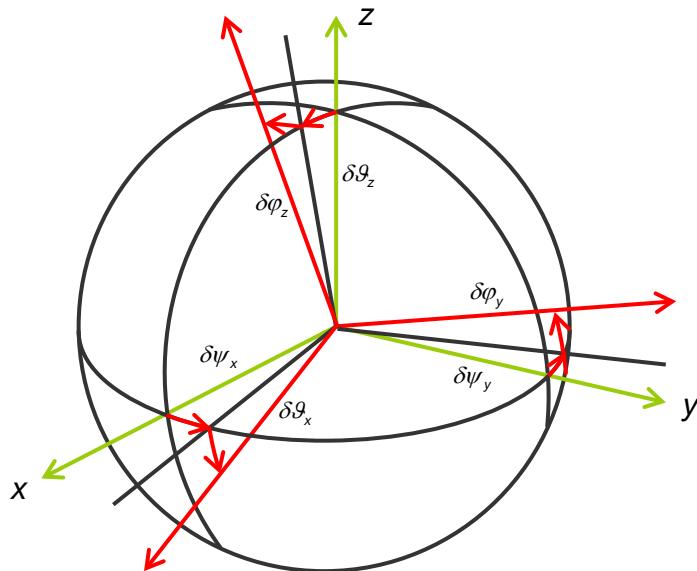


Figure 3-2: Definition of the misalignment angles

3.2.2.2 Gyroscope

Based on [24] the non-integrating gyroscope measurement model is analogous to (3.42)

$$\tilde{\boldsymbol{\omega}}_{ib}(t) = (\mathbf{I}_3 + \mathbf{S}_{\omega} + \mathbf{M}_{\omega}) \tilde{\boldsymbol{\omega}}_{ib}(t) + \mathbf{A}_{\omega} \tilde{\mathbf{f}}_b(t) + \Psi_{bb} \tilde{\boldsymbol{\omega}}_{ib}(t) + \mathbf{b}_{\omega}(t) + \boldsymbol{\varepsilon}_{\omega} \quad (3.47)$$

where

\mathbf{S}_{ω}	Scale factor error
\mathbf{M}_{ω}	Misalignment
\mathbf{A}_{ω}	Acceleration sensitivity
Ψ_{bb}	Mounting orientation error
\mathbf{b}_{ω}	Bias
$\boldsymbol{\varepsilon}_{\omega}$	Non-modeled errors, higher order terms

$$\begin{aligned} \mathbf{S}_{\omega} &= \mathbf{S}_{0,\omega} + \mathbf{S}_{T,\omega}(T - T_{ref}) \\ \mathbf{S}_{0,\omega} &\text{ Scale factor error at reference temperature } T_{ref} \\ \mathbf{S}_{T,\omega} &\text{ Scale factor error temperature sensitivity} \end{aligned} \quad (3.48)$$

$$\begin{aligned} \mathbf{b}_{\omega}(t) &= \mathbf{b}_{0,\omega} + \mathbf{b}_{T,\omega}(T - T_{ref}) + \mathbf{b}_{Q,\omega}(t) + \mathbf{b}_{N,\omega}(t) + \mathbf{b}_{B,\omega}(t) + \mathbf{b}_{K,\omega}(t) \\ \mathbf{b}_{0,\omega} &\text{ Turn-on bias at reference temperature } T_{ref} \\ \mathbf{b}_{T,\omega} &\text{ Bias temperature sensitivity} \\ \mathbf{b}_{Q,\omega} &\text{ Output quantization (rate violet noise if integrating sensor} \\ &\text{ and rate white noise if non-integrating sensor)} \\ \mathbf{b}_{N,\omega} &\text{ Angle random walk (rate white noise)} \\ \mathbf{b}_{B,\omega} &\text{ Bias instability (rate pink noise)} \\ \mathbf{b}_{K,\omega} &\text{ Rate random walk (rate brown noise)} \end{aligned}$$

In contrast to the accelerometer measurement model, the linear acceleration cross-coupling term $\mathbf{A}_{\omega} \tilde{\mathbf{f}}_b(t)$ is added to the gyroscope measurement model

$$\mathbf{A}_{\omega} \tilde{\mathbf{f}}_b = \begin{pmatrix} \mathbf{a}_{\omega,11} & \mathbf{a}_{\omega,12} & \mathbf{a}_{\omega,13} \\ \mathbf{a}_{\omega,21} & \mathbf{a}_{\omega,22} & \mathbf{a}_{\omega,23} \\ \mathbf{a}_{\omega,31} & \mathbf{a}_{\omega,32} & \mathbf{a}_{\omega,33} \end{pmatrix} \tilde{\mathbf{f}}_b = \left(\tilde{\mathbf{f}}_b^T \otimes \mathbf{I}_3 \right) \underbrace{\begin{pmatrix} \mathbf{a}_{\omega,11} \\ \mathbf{a}_{\omega,21} \\ \mathbf{a}_{\omega,31} \\ \mathbf{a}_{\omega,21} \\ \mathbf{a}_{\omega,22} \\ \mathbf{a}_{\omega,23} \\ \mathbf{a}_{\omega,31} \\ \mathbf{a}_{\omega,32} \\ \mathbf{a}_{\omega,33} \end{pmatrix}}_{\mathbf{a}_{\omega}} \quad (3.49)$$

The mounting orientation error is represented as rotation of the orthogonal triad by φ_{bb} , ϑ_{bb} and ψ_{bb}

$$\Psi_{bb} \tilde{\boldsymbol{\omega}}_{ib} = \begin{pmatrix} 0 & -\psi_{bb} & \vartheta_{bb} \\ \psi_{bb} & 0 & -\varphi_{bb} \\ -\vartheta_{bb} & \varphi_{bb} & 0 \end{pmatrix} \tilde{\boldsymbol{\omega}}_{ib} = - \underbrace{\begin{pmatrix} 0 & -\tilde{\omega}_{ib,z} & \tilde{\omega}_{ib,y} \\ \tilde{\omega}_{ib,z} & 0 & -\tilde{\omega}_{ib,x} \\ -\tilde{\omega}_{ib,y} & \tilde{\omega}_{ib,x} & 0 \end{pmatrix}}_{\text{veck}^{-1}(\tilde{\boldsymbol{\omega}}_{ib})} \underbrace{\begin{pmatrix} \varphi_{bb} \\ \vartheta_{bb} \\ \psi_{bb} \end{pmatrix}}_{\Psi_{bb}} \quad (3.50)$$

Note that the mounting orientation error only affects the rotational but not the translational degree of freedom and has thus not to be considered in the accelerometer error. In accordance with (3.46), the angular rate error $\delta\boldsymbol{\omega}_{ib}$ is

$$\begin{aligned}\delta\boldsymbol{\omega}_{ib}(t) &\doteq \mathbf{b}_\omega(t) + (\mathbf{S}_\omega + \mathbf{M}_\omega)\tilde{\boldsymbol{\omega}}_{ib}(t) + \mathbf{A}_\omega \tilde{\mathbf{f}}_b(t) + \boldsymbol{\Psi}_{bb} \tilde{\boldsymbol{\omega}}_{ib}(t) \\ &= \mathbf{b}_{0,\omega} + \mathbf{b}_{T,\omega}(T - T_{ref}) + \sum_{i=\{Q,N,B,K\}} \mathbf{b}_{i,\omega}(t) + \text{diag}(\tilde{\boldsymbol{\omega}}_{ib}(t))(\mathbf{s}_{0,\omega} + \mathbf{s}_{T,\omega}(T - T_{ref})) \\ &\quad + \mathbf{F}_{M_\omega} \mathbf{m}_\omega + (\tilde{\mathbf{f}}_b^\top(t) \otimes \mathbf{I}_3) \mathbf{a}_\omega - \text{veck}^{-1}(\tilde{\boldsymbol{\omega}}_{ib}(t)) \boldsymbol{\Psi}_{bb}\end{aligned}\quad (3.51)$$

3.2.3 Models

The turn-on biases \mathbf{b}_0 , bias temperature sensitivities \mathbf{b}_T , nominal scale factor errors \mathbf{s}_0 , scale factor error temperature sensitivities \mathbf{s}_T as well as misalignment errors \mathbf{m} of the accelerometer and gyroscope and the acceleration sensitivity \mathbf{a}_ω and mounting orientation error $\boldsymbol{\Psi}_{bb}$ of the gyroscope are modeled as constant values, described by the discrete-time model

$$\begin{aligned}\mathbf{z}_{b_0,k} &= \boldsymbol{\Phi}_{b_0} \mathbf{z}_{b_0,k-1} + \boldsymbol{\Gamma}_{b_0} \mathbf{n}_{b_0,k-1}, & \mathbf{z}_{b_0,0} &= \mathbf{b}_0 \\ \mathbf{b}_{0,k} &= \mathbf{H}_{b_0} \mathbf{z}_{b_0,k} + \mathbf{D}_{b_0} \mathbf{n}_{b_0,k} \\ \text{with } \boldsymbol{\Phi}_{b_0} &= \mathbf{I}_3, \quad \boldsymbol{\Gamma}_{b_0} = 0, \quad \mathbf{H}_{b_0} = \mathbf{I}_3, \quad \mathbf{D}_{b_0} = 0\end{aligned}\quad (3.52)$$

exemplarily for the turn-on bias. Although these errors are already small by definition, they are formally split into expected value $\tilde{\mathbf{z}}$ and error $\delta\mathbf{z}$ in order to fit into the context of the navigation state *error* filter if they shall be estimated and not only considered

$$\mathbf{z} = \tilde{\mathbf{z}} + \delta\mathbf{z} \quad (3.53)$$

Then, models for the *errors* of the estimated IMU error components are applied in the navigation filter

$$\begin{aligned}\delta\mathbf{z}_{b_0,k} &= \boldsymbol{\Phi}_{b_0} \delta\mathbf{z}_{b_0,k-1} + \boldsymbol{\Gamma}_{b_0} \mathbf{n}_{b_0,k-1}, & \delta\mathbf{z}_{b_0,0} &= 0 \\ \delta\mathbf{b}_{0,k} &= \mathbf{H}_{b_0} \delta\mathbf{z}_{b_0,k} + \mathbf{D}_{b_0} \mathbf{n}_{b_0,k}\end{aligned}\quad (3.54)$$

instead of the error models (3.52), which are actually the same. In common navigation system designs, the turn-on biases and the nominal scale factor errors are usually estimated and used for in-flight calibration of the raw IMU measurements whereas the remaining errors are treated as considered states because of their low observability.

The velocity/angle random walk \mathbf{b}_N , bias instability \mathbf{b}_B , acceleration/rate random walk \mathbf{b}_K as well as quantization noise \mathbf{b}_Q are noise-like errors and are described by stochastic models.

\mathbf{b}_N is simply white noise on the acceleration/rate signals $\mathbf{f}/\boldsymbol{\omega}$ and a random walk on the velocity/angle increments $\Delta\mathbf{v}/\Delta\boldsymbol{\theta}$ in case of integrating sensors.

The bias instability \mathbf{b}_B is pink colored noise on the acceleration/rate signals $\mathbf{f}/\boldsymbol{\omega}$. It is characterized by the PSD being inversely proportional to the frequency f . According to the IEEE standard [24] the bias instability noise of inertial sensors is best described by pink noise with the high frequency part with $f > 1/T_c$ cut away

$$S_{yy}(f) = \begin{cases} \frac{B^2}{2\pi f} \frac{1}{f} & f \leq 1/T_c \\ 0 & f > 1/T_c \end{cases} \quad (3.55)$$

where B is the bias instability coefficient and T_c the cut-off time constant. A first order low-pass filter with the continuous-time transfer function

$$H_{LPF}(s) = \frac{1}{1 + \frac{T_c}{2\pi}s} \quad (3.56)$$

is adequate to obtain the desired PSD. With the bilinear transform

$$s = \frac{2}{\Delta t} \frac{1-z^{-1}}{1+z^{-1}} \quad (3.57)$$

where Δt is the sample time, the corresponding discrete-time transfer function becomes

$$\begin{aligned} H_{LPF}(z) &= \frac{b_{0,LPF} + b_{1,LPF} z^{-1}}{1 - a_{1,LPF} z^{-1}} \\ a_{1,LPF} &= -\frac{1 - \frac{T_c}{2}}{\frac{2\pi}{T_c} \frac{2}{\Delta t}}, \quad b_{0,LPF} = \frac{1}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}}, \quad b_{1,LPF} = \frac{1}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}} \end{aligned} \quad (3.58)$$

The ARMA coefficients a and b of the overall transfer function $H(z)$ of the low-pass filter transfer function $H_{LPF}(z)$ and the bias instability transfer function $H_B(z)$ connected in series

$$H(z) = H_{LPF}(z) \cdot H_B(z) = \frac{b_{0,LPF} + b_{1,LPF} z^{-1}}{1 - a_{1,LPF} z^{-1}} \frac{\sum_{i=0}^q b_{B,i} z^{-i}}{1 - \sum_{i=1}^p a_{B,i} z^{-i}} \quad (3.59)$$

are obtained by convoluting the denominator and numerator polynomials

$$\begin{aligned} a_1 &= a_{B,1} + a_{1,LPF}, & a_i &= a_{B,i} - a_{1,LPF} a_{B,i-1}, & i &= 2 \dots p, & a_{p+1} &= a_{1,LPF} a_{B,p} \\ b_0 &= b_{0,LPF} b_{B,0}, & b_i &= b_{0,LPF} b_{B,i} + b_{1,LPF} b_{B,i-1}, & i &= 1 \dots q, & b_{q+1} &= b_{1,LPF} b_{B,q} \end{aligned} \quad (3.60)$$

where the bias instability model coefficients $a_{B,i}$ and $b_{B,i}$ can be taken from Table 3-3 and Table 3-4. The PSD of the low-pass filtered bias instability noise are shown in Figure 3-3 for bias instability models of order 1 to 4 and the exemplary cut-off time constant $T_c = 1\text{s}$.

Quantization of a continuous signal generally leads to additional white noise \mathbf{b}_Q . Quantization is owing to the resolution of the analog-digital converters (ADC) sampling the servo circuit current or voltage of analog servos or to the discrete energy amount per pulse of digital servos. If non-integrated acceleration/angular rate signals \mathbf{f}/ω are output, the quantization noise \mathbf{b}_Q superposes with the velocity/angle random walk noise \mathbf{b}_N . Since the sensor manufacturer tests and calibrates the complete sensor unit just as later applied in the navigation system, the quantization effect is already included in the specified white noise and has not to be separately considered in the accelerometer model. If the sensor outputs velocity/angle increments $\Delta\mathbf{v}/\Delta\theta$, the white quantization noise is on the integrated accelerations/angular rates and thus corresponds to acceleration/angular rate violet noise. Looking at the power spectrum of the acceleration/angular rate \mathbf{f}/ω , the PSD of quantization noise is proportional to the frequency f if the sensor is integrating [24]

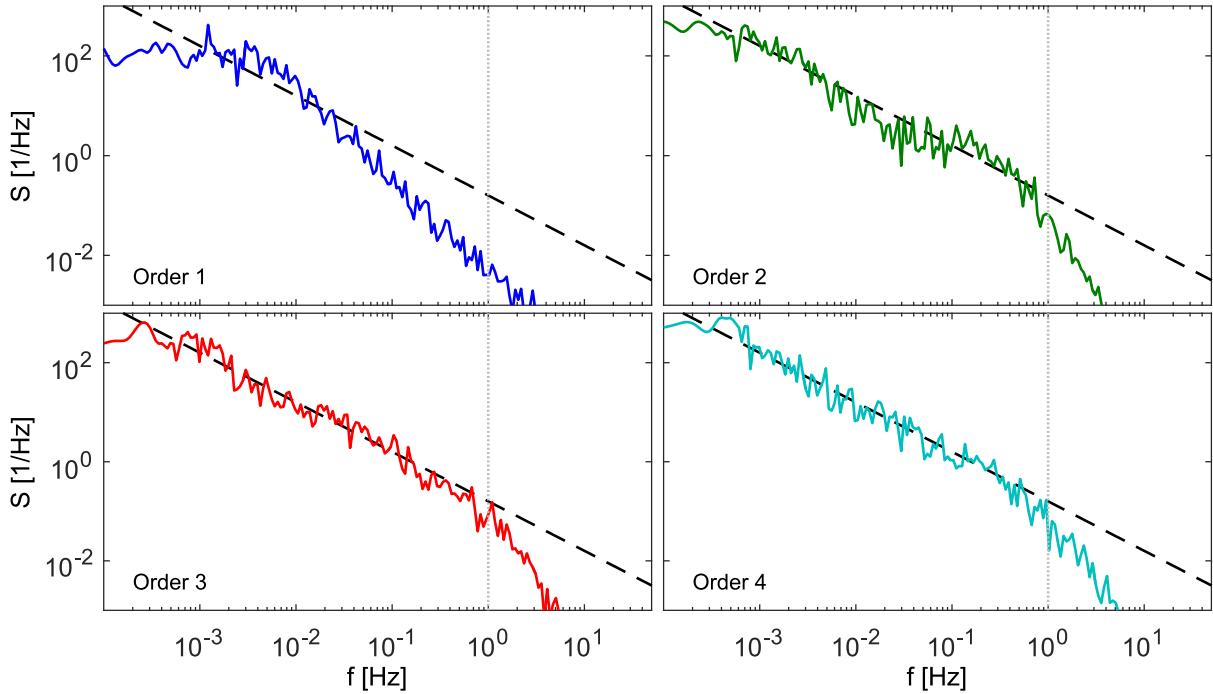
$$S_{f/\omega, b_Q} = 4Q^2 f_s \sin^2(\pi f/f_s) \quad (3.61)$$

and constant with frequency f if the sensor is non-integrating

$$S_{f/\omega, b_Q} = \frac{Q^2}{f_s} \frac{\sin^2(\pi f/f_s)}{(\pi f/f_s)^2} \quad (3.62)$$

For $f < \frac{1}{2}f_s$, the approximations

$$S_{f/\omega, b_Q} \approx 4 \frac{Q^2}{f_s} \pi^2 f^2 \quad (3.63)$$

Figure 3-3: PSD of low-pass filtered $1/f$ noise ($T_c = 1\text{s}$)

and

$$S_{f/\omega, b_Q} \approx \frac{Q^2}{f_s} \quad (3.64)$$

hold. The acceleration/angular rate random walk \mathbf{b}_K is usually very low frequent with a correlation length in the range of hours. The white noise \mathbf{b}_N and the bias instability \mathbf{b}_B have the largest share of the stochastic error.

The corresponding discrete-time state space models have been given in Table 3-6 in section 3.1.4.3. In the models, each of the triad axes is assumed to be independent. The input noise is white and Gaussian distributed and characterized by its PSD specified in $(\text{m}/\text{s}^2)/\sqrt{\text{Hz}}$ ² for accelerometers and $(\text{rad}/\text{s})/\sqrt{\text{Hz}}$ ² for gyroscopes. Table 3-9 lists the corresponding coefficients of the different noises of the acceleration/rate signals $\mathbf{f}/\boldsymbol{\omega}$. Notice that, if integrating sensors are analyzed, the velocity/angle increments have to be numerically differentiated at first.

Table 3-9: PSDs of stochastic errors [24]

Stochastic Error	Type	Exponent α	Coefficient h_α	Unit
\mathbf{b}_K	Random walk frequency	-2	$\frac{K^2}{2\pi^2}$	$(\frac{\text{m}}{\text{s}^2})^2 \text{Hz}$, $(\frac{\text{rad}}{\text{s}})^2 \text{Hz}$
\mathbf{b}_B	Flicker frequency	-1	$\frac{B^2}{\pi}$	$(\frac{\text{m}}{\text{s}^2})^2$, $(\frac{\text{rad}}{\text{s}})^2$
\mathbf{b}_N	White frequency	0	$2N^2$	$(\frac{\text{m}}{\text{s}^2})^2 \frac{1}{\text{Hz}}$, $(\frac{\text{rad}}{\text{s}})^2 \frac{1}{\text{Hz}}$
\mathbf{b}_Q	Non-integrating	0	$2\Delta t Q^2$	$(\frac{\text{m}}{\text{s}^2})^2 \frac{1}{\text{Hz}}$, $(\frac{\text{rad}}{\text{s}})^2 \frac{1}{\text{Hz}}$
	Integrating	+2	$8\pi^2 \Delta t Q^2$	$(\frac{\text{m}}{\text{s}^2})^2 \frac{1}{\text{Hz}^3}$, $(\frac{\text{rad}}{\text{s}})^2 \frac{1}{\text{Hz}^3}$

3.2.4 Parameter Estimation

The parameters K , B , N and Q are specified by the manufacturer. Often, the manufacturer gives rather conservative values if the IMU performance has to be guaranteed. The noise amplitudes of the device at hand are normally up to one order of magnitude lower. If particularly high accuracy is demanded, it is recommended to estimate the parameters again from the recorded measurements of the actual unit to be applied in the integrated navigation system. For that, some hours of data have to be recorded. During the recording time the IMU itself as well as the environmental conditions, first of all the temperature, have to be kept stationary. The overlapping Allan variance $\sigma_{\text{avar},y}^2(\tau_i)$ as given in Template 3-2 is computed for each sensor channel. The parameter vector \mathbf{a} containing

$$\mathbf{a}^\top = (K^2 \quad B^2 \quad N^2 \quad Q^2) \quad (3.65)$$

can be easily estimated by means of the least squares method with the vector \mathbf{s} of the M available Allan variances $\sigma_{\text{avar},y}^2(\tau_i)$, $i=1\dots M$,

$$\mathbf{s} = \begin{pmatrix} \sigma_{\text{avar},y}^2(\tau_1) \\ \vdots \\ \sigma_{\text{avar},y}^2(\tau_M) \end{pmatrix} \quad (3.66)$$

and the coefficient matrix \mathbf{A} , using Table 3-1 and Table 3-9,

$$\mathbf{A} = \begin{pmatrix} \frac{1}{3}\tau_1 & \frac{2\ln 2}{\pi} & \tau_1^{-1} & 3\tau_1^{-2} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{1}{3}\tau_M & \frac{2\ln 2}{\pi} & \tau_M^{-1} & 3\tau_M^{-2} \end{pmatrix} \quad (3.67)$$

It has to be noticed that an estimator with lower boundary constraint has to be applied, since the parameters to be estimated are strictly positive. Furthermore, it is advisable not to minimize the residuals in the classical sense of least-squares

$$\arg \left\{ \min_{\hat{\mathbf{a}}} \left((\mathbf{s} - \mathbf{A} \hat{\mathbf{a}})^\top (\mathbf{s} - \mathbf{A} \hat{\mathbf{a}}) \right) \right\} \quad (3.68)$$

but to minimize the sum of the squares of the differences of the common logarithms

$$\arg \left\{ \min_{\hat{\mathbf{a}}} \left((\log(\mathbf{s}) - \log(\mathbf{A} \hat{\mathbf{a}}))^\top (\log(\mathbf{s}) - \log(\mathbf{A} \hat{\mathbf{a}})) \right) \right\} \quad (3.69)$$

Since the noise is strongly influenced by the temperature, the analysis should be repeated for different environmental temperatures. A temperature chamber would be ideal but alternatively, it is possible to adjust at least a few temperatures with the room heating and air conditioning.

As example, the noise parameters of the gyroscopes of a Northrop Grumman LITEF μIMU-IC device [34] are estimated for a single temperature. The sample rate is 400 Hz. About 18 hours of measurement data have been recorded. The IMU was placed on a leveled plane table. The initial time period, during which the device temperature was transient, has been cut away. By means of a ventilator the device temperature was kept constant at 42 °C as illustrated in Figure 3-4.

The black lines in Figure 3-5 show the actual overlapped Allan deviations of the angular rate measurements. The colored dotted lines illustrate the estimated Allan deviations of the single noise components and the black dotted line is the estimated overall noise Allan deviation. The x -axis gyroscope features only angular random walk and quantization noise. The y - and z -gyroscopes

additionally exhibit bias instability noise. It can be seen that the actual and the estimated Allan deviations coincide well. The estimated parameters of the single gyroscope triad axes are listed in Table 3-10. The estimated angular random walk parameters N of the three axes are almost equal. By comparing them with the specified parameters (Table A-2), it can be seen that the actual performance of the device under consideration is almost one order of magnitude better than specified.

The IMU error models will be integrated into the navigation filter in section 4.6.1.

Table 3-10: Estimated parameters of Northrop Grumman LITEF μIMU-IC gyroscopes

Parameter	x	y	z
B [$^{\circ}/\text{h}$]	n/a	0.157	0.230
N [$^{\circ}/\sqrt{\text{h}}$]	0.063	0.066	0.062
Q [$^{\circ} \sqrt{\text{h}}$]	0.196e-5	0.251e-5	0.148e-5

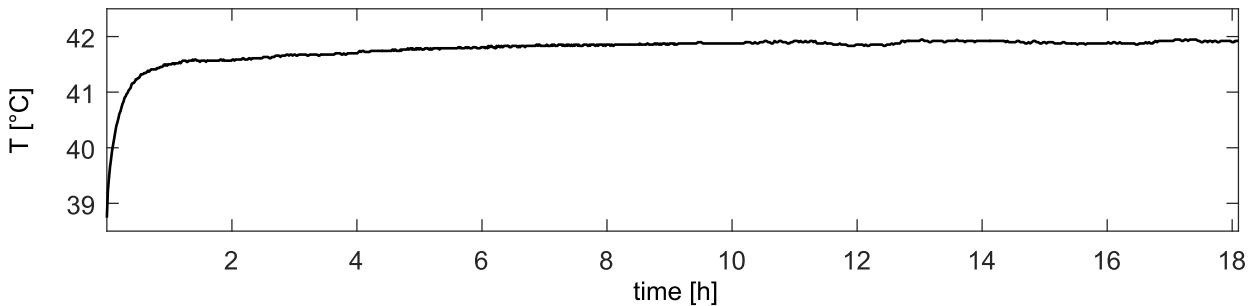


Figure 3-4: Temperature of Northrop Grumman LITEF μIMU-IC during measurement recording

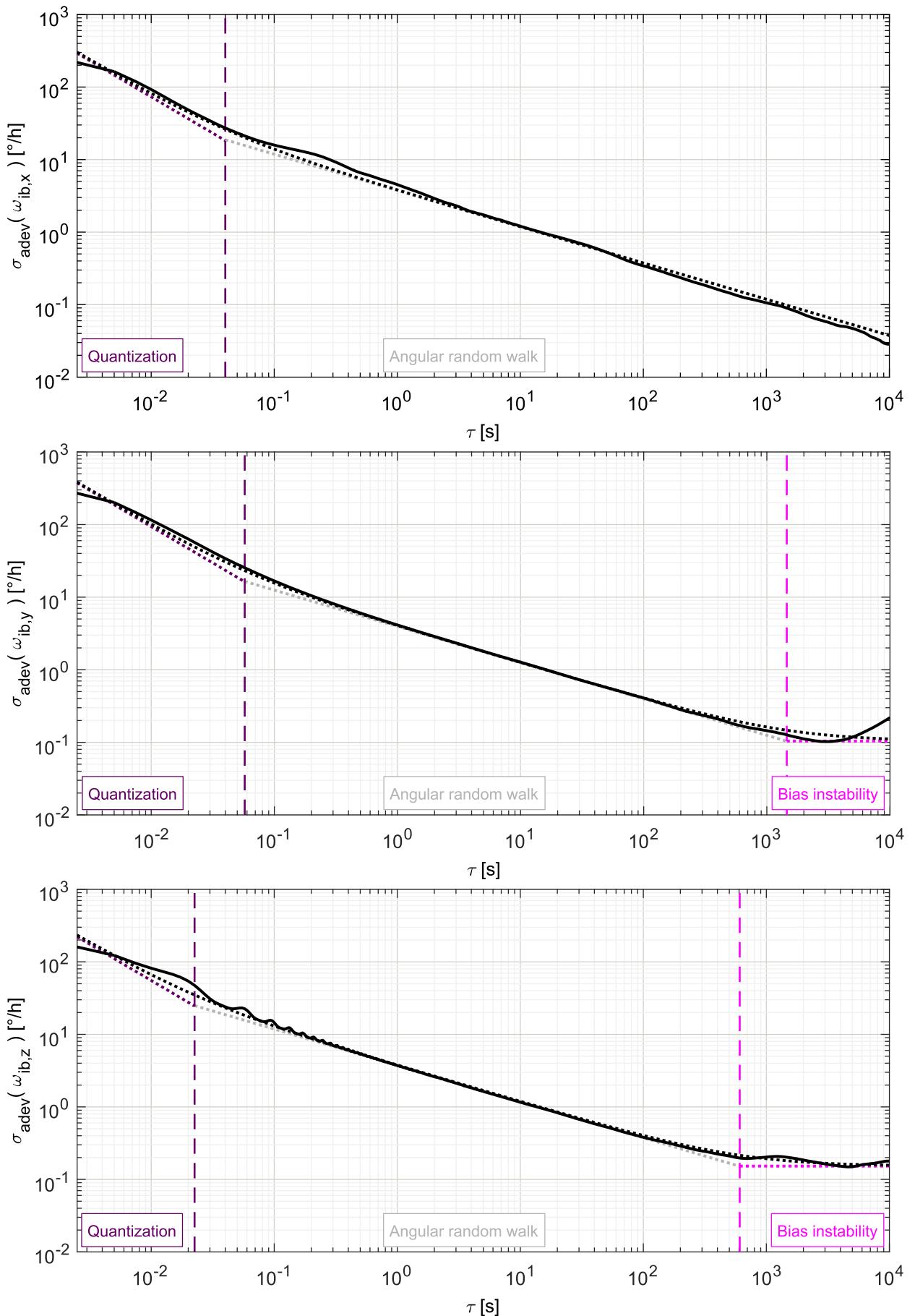


Figure 3-5: Allan deviations of the Northrop Grumman LITEF μIMU-IC gyroscope triad

3.3 Gravity Model Error

3.3.1 Motivation

One of the main problems in inertial navigation is the compensation of the gravity \mathbf{g} . Accelerometers measure the specific force \mathbf{f}_b that acts on the proof mass of the sensor. In order to calculate the acceleration $\dot{\mathbf{v}}_e$ of the platform with respect to Earth with (2.1), the accelerometer measurement has to be transformed from b -frame to e -frame and compensated by the gravity \mathbf{g}_e and the Coriolis acceleration.

Several gravity models exist. Somigliana's gravity model with Taylor series expansion as presented in [35] is valid for heights up to 50 km. Highly accurate applications can use the spherical harmonics gravity model with EGM96 or EGM2008 model coefficients. The gravity potential is described by superposed zonal, sectorial and tesseral surface harmonics. The accuracy of the gravity potential and the gravity vector depends on the degree and order of the harmonics. However, the computational effort increases likewise with degree and order. The gravity vector, which is the partial derivative of the potential, can be calculated with the formulas given in [36].

Especially in flight phases without aiding measurements, for example due to missing GNSS reception, the self-assessment of the accuracy of the pure inertial navigation solution is important. The measurement errors of the inertial sensors and the gravity compensation error are continuously integrated and lead to a gradually growing uncertainty of the navigation solution. This uncertainty has to be properly predicted if the navigation solution is used in safety critical functions like for example flight cancellation of unmanned aircraft running the risk to leave the controlled airspace. In order to calculate the covariance of the inertial navigation solution correctly, gravity model errors have to be identified and adequately modeled besides the already discussed IMU errors.

In section R.2.2.8 of the RTCA standard DO-316, "Minimum Operational Performance Standards for Global Positioning System/Aircraft Based Augmentation System Airborne Equipment" [7], the need of an adequate gravity compensation error model is explained as follows:

"Gravity compensation error is a significant source of position drift for two nautical miles per hour inertial navigation systems. Errors in the gravity models can cause errors in the integrated inertial GPS systems that become problematic particularly during coasting conditions. To reduce these errors, some form of gravity compensation algorithms must be applied. The gravity compensation error is typically expressed in terms of gravity deflection and gravity anomaly. [...] Investigations of the distribution of gravity deflection and anomaly data demonstrate that the data does not form a normal distribution. While most of the world can be modeled assuming one-sigma deflection of 5 arc-sec, there are specific isolated regions where the deflections are much larger -- up to 50 arc-sec. These gravity errors do not produce "random" errors, as the same errors will occur on every flight over a local area. [...] The tightly integrated system shall [GPS294] properly account for the local gravity anomalies and deflections such that the HPL continues to bound the system errors while operating in areas of increased gravity anomaly/deflections, even when coasting. Suitable mechanisms include an appropriate subset of the following:

- a) Over-bounding using a standard model with an elevated sigma level.
- b) Compensation using a gravity map.
- c) Adjustment of the filter parameters (e.g. increase the process noise)."

In section R.5.8 details about the statistical model and its validation are specified: “The effect of gravity disturbances can be simulated by a statistical model, one example is the Gauss-Markov Gravity Model (GMGM), which is generated from a series of medium order (2nd to 4th) Gauss-Markov processes. The statistical model includes processes of varying correlation distance. [...] The SC [Statistical Compensation] technique is implemented by adjusting the parameters of the tightly-coupled filter to overbound the degree of variation of the residual [...] gravity disturbances. The SC technique may incorporate real time adjustment of the filter parameters based on a priori knowledge.”

In [37], it is proposed to model the residual gravity error of the precise orbit determination of satellites as second order Gauss-Markov process. In [11], an adequate third order stochastic gravity error model is presented.

If a simple ellipsoidal normal gravity model like Somigliana is used for gravity compensation in the inertial navigation algorithm, the remaining uncompensated error may be described as stochastic process. [38] has derived a stochastic model for this gravity error. It bases upon the statistical description of the local spatial disturbing gravity potential T . In particular, T is represented by a spatial ergodic and isotropic stochastic Markov process of n^{th} order with exponential autocorrelation function. The gravity vector error is deduced from the disturbing gravity potential model. The spatial process is converted into a temporal process by means of the velocity of the carrier platform.

In this section, the state-space model of [38] for the stochastic gravity compensation error is recapitulated and adapted to the context of integrated navigation filter. The model represents the amplitude and time correlation of the error. It is two-dimensional (latitude, longitude) and valid on the surface of the Earth. The attenuation with height is considered by upward continuation. The Earth is assumed to be flat, which is adequate for correlation lengths shorter than 3000 km. The model can be later integrated into the navigation error filter to correctly consider the gravity error of the integrated navigation solution.

3.3.2 Disturbing Gravity Potential Autocorrelation Function

The actual gravity potential W at the position $\mathbf{x}(t)$ is composed of the normal gravity potential U and the disturbing gravity potential T

$$W(\mathbf{x}(t)) = U(\mathbf{x}(t)) + T(\mathbf{x}(t)) \quad (3.70)$$

The gravity disturbance vector $\delta\mathbf{Y}_n$ is defined as difference between true gravity \mathbf{Y}_n and the normal gravity model value $\tilde{\mathbf{Y}}_n$

$$\delta\mathbf{Y}_n = \mathbf{Y}_n - \tilde{\mathbf{Y}}_n = \begin{pmatrix} 1 & 0 & +\delta_n \\ 0 & 1 & +\delta_e \\ -\delta_n & -\delta_e & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ \gamma \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ \tilde{\gamma} \end{pmatrix} = \begin{pmatrix} \delta_n \gamma \\ \delta_e \gamma \\ \gamma - \tilde{\gamma} \end{pmatrix} = \begin{pmatrix} \delta_n \gamma \\ \delta_e \gamma \\ \delta\gamma \end{pmatrix} \doteq \begin{pmatrix} \delta_n \tilde{\gamma} \\ \delta_e \tilde{\gamma} \\ \delta\gamma \end{pmatrix} \quad (3.71)$$

where γ is the normal gravity, δ_n and δ_e are the deflections of the vertical in north and east direction and $\delta\gamma$ is the gravity disturbance. The gravity model vector $\tilde{\mathbf{Y}}_n$ is assumed to be normal on the Earth ellipsoid. The true gravity vector \mathbf{Y}_n , however, is normal to the local geoid, which is as illustrated in Figure 3-6 an equipotential surface, $W = \text{const}$, with undulation height N above the Earth ellipsoid. The deflections of the vertical in north and east direction δ_n and δ_e are shown in Figure 3-7. Note that the angles are defined with respect to the normal of the Earth ellipsoid. If the deflection δ_e is seen as a rotation about the north axis, the angle is defined in negative sense.

The gravity disturbance vector $\delta \mathbf{V}_n$ is the gradient of the disturbing gravity potential T with respect to latitude ϕ , longitude λ and height h or with respect to north, east and down directions x_n , x_e and x_d if the Earth is assumed to be locally flat

$$\delta \mathbf{V}_n = \begin{pmatrix} \frac{1}{M} \cdot \frac{\partial T}{\partial \phi} \\ \frac{1}{N \cos \phi} \cdot \frac{\partial T}{\partial \lambda} \\ -\frac{\partial T}{\partial h} \end{pmatrix} \stackrel{\text{Flat Earth assumption}}{=} \begin{pmatrix} \frac{\partial T}{\partial x_n} \\ \frac{\partial T}{\partial x_e} \\ \frac{\partial T}{\partial x_d} \end{pmatrix} \quad (3.72)$$

Therein, M is the meridian curvature radius and N is the normal curvature radius of the Earth ellipsoid. The gravity disturbance vector $\delta \mathbf{V}_n$ can thus be consistently derived once a model for the disturbing gravity potential T has been found. The disturbing gravity potential T is modeled as spatial isotropic n^{th} order Gauss-Markov process

$$\frac{d^n T(r)}{dr^n} + \sum_{k=0}^{n-1} (k+1) \beta^{n-k} \frac{d^k T(r)}{dr^k} = \eta(r) \quad (3.73)$$

where $\eta(r)$ is white input noise. The corresponding autocorrelation function is [39]

$$R_{TT}(r) = \sigma_T^2 e^{-\beta|r|} \sum_{k=0}^{n-1} \frac{\Gamma(n) (2\beta|r|)^{n-k-1}}{(2n-2)! k! \Gamma(n-k)} \quad (3.74)$$

Therein, $r = \sqrt{x_n^2 + x_e^2}$ is the horizontal distance. $\Gamma(n)$ is the gamma function. σ_T is the standard deviation of the disturbing gravity potential T . If, for example, a second order model is chosen [38], $n=2$, then the spatial autocorrelation function becomes

$$R_{TT}(r) = \sigma_T^2 (1 + \beta|r|) \cdot e^{-\beta|r|} \quad (3.75)$$

Figure 3-8 shows the normalized autocorrelation function. The correlation length at which the autocorrelation is about $1/e \approx 0.37$ is given with

$$\frac{2.146}{\beta} \quad (3.76)$$

In section B.1 in the appendix the autocorrelation function matrix $\mathbf{R}_{\delta \mathbf{V}_n \delta \mathbf{V}_n}$ of the gravity disturbance vector $\delta \mathbf{V}_n$ is derived from the autocorrelation function of the disturbing gravity potential (3.75).

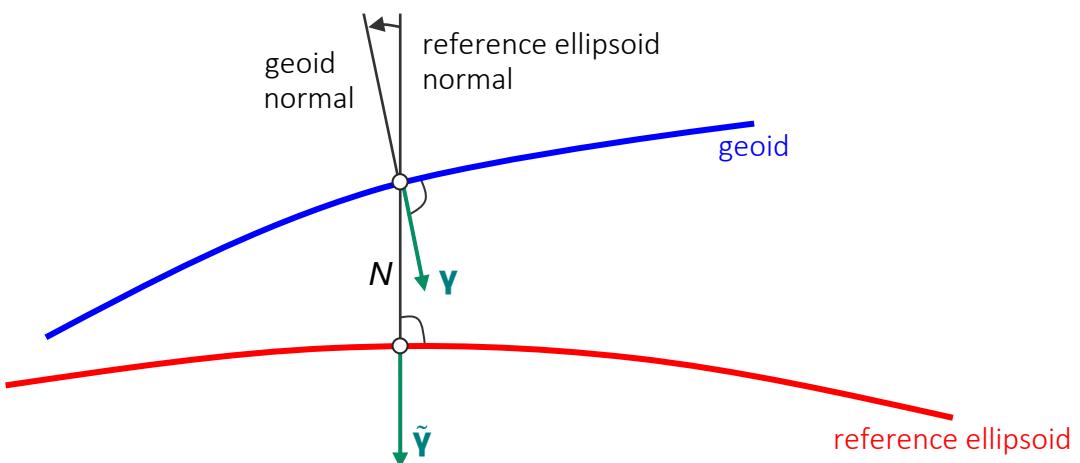


Figure 3-6: Gravity vectors, geoid and reference ellipsoid

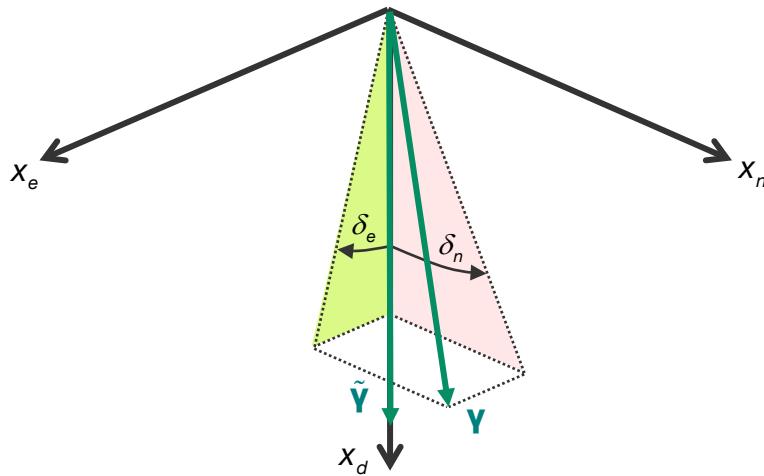


Figure 3-7: Deflections of the vertical δ_n and δ_e

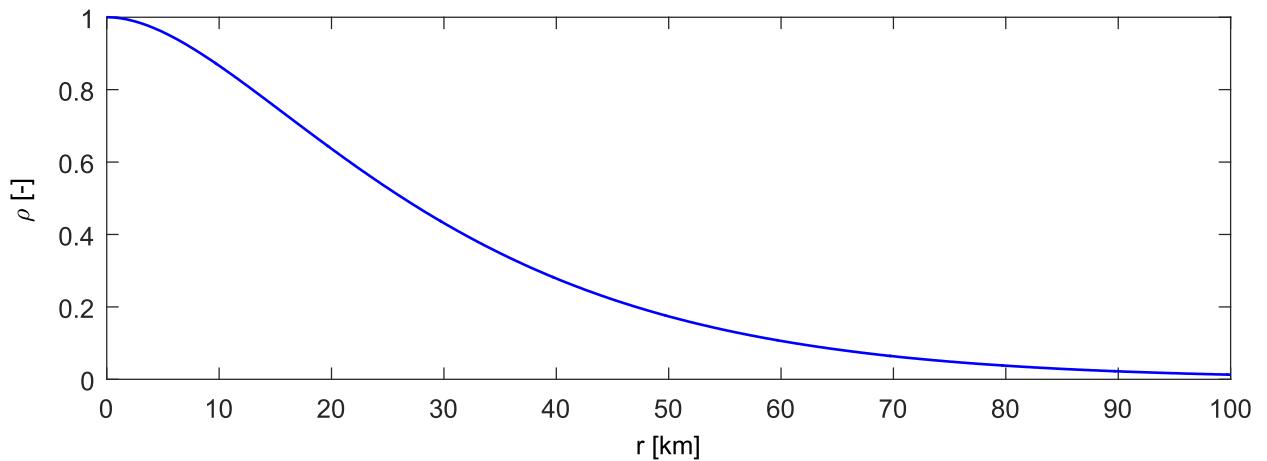


Figure 3-8: Normalized isotropic autocorrelation function of the disturbing gravity potential, which is modeled as 2nd order Gauss-Markov process, $\beta = 6.1 \cdot 10^{-5}$

3.3.3 Gravity Disturbance Model

The continuous-time state-space model for the stochastic gravity disturbance as presented in detail in [38] is given by

$$\dot{\mathbf{z}}(t) = \underbrace{\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\beta^2 v_h^2 & -2\beta v_h & 0 & 0 & 0 \\ 0 & 0 & -\beta v_h & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\beta^2 v_h^2 & -2\beta v_h \end{pmatrix}}_{\mathbf{F}} \cdot \mathbf{z}(t) + \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{G}} \cdot \mathbf{n}(t) \quad (3.77)$$

$$\delta \mathbf{y}_n = \underbrace{\begin{pmatrix} \delta y_n \\ \delta y_e \\ \delta y_d \end{pmatrix}}_{\mathbf{H}} = \sigma_T \beta \sqrt{\beta v_h} \cdot \mathbf{R}_3(\chi) \cdot \underbrace{\begin{pmatrix} 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2} & 0 & 0 \\ 0 & 0 & 0 & \sqrt{2} \beta v_h & \sqrt{6} \end{pmatrix}}_{\mathbf{R}} \cdot \mathbf{z}(t)$$

The model is composed of 2nd order Gauss-Markov processes in along-track and down direction and a 1st order Gauss-Markov process in cross-track direction. The Gauss-Markov processes correspond

to the gravity disturbance vector autocorrelation function (B.8). v_h is the horizontal velocity of the moving carrier platform

$$v_h = \sqrt{v_n^2 + v_e^2} \quad (3.78)$$

The flight track angle χ is calculated by

$$\chi = \text{atan2}(v_e, v_n) \quad (3.79)$$

In (3.77) the correlation distance r has been replaced by a product of the platform velocity v_h and the correlation time τ . The input noise is normalized Gaussian white noise, $\eta \sim WN(0, I_3)$.

The state-space model is valid directly on the Earth ellipsoid with height $h=0$ or close to the topography. The dampening effect of the gravity disturbance with increasing height, that is with increasing distance to the topography, is not included and has to be regarded by the upward continuation of the model, which will be described in the next section. β as well as the standard deviation σ_τ depend on the local roughness of the terrain. In mountainous areas the correlation length $2.146/\beta$ is shorter and the standard deviation σ_τ higher than in level land.

The state-space model (3.77) with five states can be split into three smaller models in along-track, cross-track and downward directions. This is beneficial when the model is implemented in the navigation error filter because the computational effort is reduced. The along-track model is

$$\dot{\mathbf{z}}_a(t) = \underbrace{\begin{pmatrix} 0 & 1 \\ -\beta^2 v_h^2 & -2\beta v_h \end{pmatrix}}_{\mathbf{F}_a} \mathbf{z}_a(t) + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{G}_a} \eta_a(t) \quad (3.80)$$

the cross-track model is

$$\dot{z}_c(t) = \underbrace{-\beta v_h}_{\mathbf{F}_c} z_c(t) + 1 \cdot \eta_c(t) \quad (3.81)$$

and finally the downward model is

$$\dot{\mathbf{z}}_d(t) = \underbrace{\begin{pmatrix} 0 & 1 \\ -\beta^2 v_h^2 & -2\beta v_h \end{pmatrix}}_{\mathbf{F}_d} \mathbf{z}_d(t) + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{G}_d} \eta_d(t) \quad (3.82)$$

The gravity disturbance vector $\delta \mathbf{y}_n$ is accordingly composed of the outputs of the three models

$$\delta \mathbf{y}_n = \begin{pmatrix} \delta \gamma_n \\ \delta \gamma_e \\ \delta \gamma_d \end{pmatrix} = \mathbf{H}_a \mathbf{z}_a(t) + \mathbf{H}_c z_c(t) + \mathbf{H}_d \mathbf{z}_d(t) \quad (3.83)$$

with the single output matrices given as

$$\begin{aligned} \mathbf{H}_a &= \sigma_\tau \beta \sqrt{\beta v_h} \cdot \mathbf{R}_3(\chi) \cdot \begin{pmatrix} 0 & 2 \\ \hline \hline 0 & 0 \\ \hline \hline 0 & 0 \end{pmatrix}, & \mathbf{H}_c &= \sigma_\tau \beta \sqrt{\beta v_h} \cdot \mathbf{R}_3(\chi) \cdot \begin{pmatrix} 0 \\ \hline \hline \sqrt{2} \\ \hline \hline 0 \end{pmatrix} \\ \mathbf{H}_d &= \sigma_\tau \beta \sqrt{\beta v_h} \cdot \mathbf{R}_3(\chi) \cdot \begin{pmatrix} 0 & 0 \\ \hline \hline 0 & 0 \\ \hline \hline \sqrt{2} \beta v_h & \sqrt{6} \end{pmatrix} \end{aligned} \quad (3.84)$$

3.3.4 Upward Continuation

In order to account for the decreasing gravity disturbance with distance to the terrain, the system matrices in (3.80), (3.81) and (3.82) are written as functions of the height h [40]. The adapted state-space models are

$$\begin{aligned}\dot{\mathbf{z}}_i(t) &= \mathbf{F}_i(h) \mathbf{z}_i(t) + \mathbf{G}_i(h) \eta_i(t), & i = \{a, c, d\} \\ \delta \mathbf{y}_n(t) &= \sum_{j=\{a, c, d\}} \mathbf{H}_j(h) \mathbf{z}_j(t)\end{aligned}\quad (3.85)$$

The system matrices depend on h as follows

$$\mathbf{F}_i(h) = \frac{1}{(1+2\bar{h})} \mathbf{F}_i(0), \quad \mathbf{G}_i(h) = \frac{1}{\sqrt{1+2\bar{h}}} \mathbf{G}_i(0), \quad \mathbf{H}_i(h) = \frac{1}{(1+2\bar{h})^{\frac{3m}{2}}} \mathbf{H}_i(0) \quad (3.86)$$

Therein, \bar{h} is the normalized height and is calculated as

$$\bar{h} = \frac{h\beta}{2.146} \quad (3.87)$$

m is a free parameter and has to be chosen adequately.

3.3.5 Discrete-Time State-Space Model

Next, the continuous-time model (3.85) is converted to discrete-time as described in section 2.3 so that it can be implemented into the integrated navigation filter

$$\begin{aligned}\mathbf{z}_{i,k} &= \Phi_i(h_{k-1}) \mathbf{z}_{i,k-1} + \Gamma_i(h_{k-1}) \eta_{i,k-1}, & i = \{a, c, d\} \\ \delta \mathbf{y}_{n,k} &= \sum_{j=\{a, c, d\}} \mathbf{H}_j(h_k) \mathbf{z}_{j,k}\end{aligned}\quad (3.88)$$

The linear terms are sufficient for the system matrices of the gravity disturbance model

$$\begin{aligned}\Phi_{a/d,k} &= \frac{1}{(1+2\bar{h}_k)} \begin{pmatrix} 1 & \Delta t \\ -\beta^2 v_{h,k}^2 \Delta t & 1 - 2\beta v_{h,k} \Delta t \end{pmatrix}, & \Gamma_{a/d,k} &= \frac{1}{\sqrt{1+2\bar{h}_k}} \begin{pmatrix} 0 \\ \Delta t \end{pmatrix} \\ \Phi_{c,k} &= \frac{1}{(1+2\bar{h}_k)} (1 - \beta v_{h,k} \Delta t), & \Gamma_{c,k} &= \frac{1}{\sqrt{1+2\bar{h}_k}} \Delta t\end{aligned}\quad (3.89)$$

The discrete-time input noise is Gaussian distributed white noise

$$\eta_{i,k} \sim WN\left(0, \frac{1}{\Delta t}\right) \quad (3.90)$$

3.3.6 Parameter Estimation

The stochastic gravity error model parameters σ_T and β shall be estimated from real gravity data. In the absence of the knowledge of the true gravity the accurate EGM2008 gravity model serves as reference for the truth. The gravity error is then defined as difference between the EGM2008 gravity and the WGS84 ellipsoidal gravity, which is commonly used in inertial navigation systems for gravity compensation.

The National Geospatial-Intelligence Agency (NGA) already provides EGM2008 gravity anomalies and deflections of the vertical as global $2.5' \times 2.5'$ grids [41]. The maximum degree and order of 2159

have been used for the computation of these grid values. The deflections of the vertical δ_n and δ_e are multiplied by the normal gravity \tilde{g} to obtain the north and east gravity errors $\delta_n \tilde{g}$ and $\delta_e \tilde{g}$. Figure 3-9 shows the north, east and down gravity errors $\delta\gamma_n = \delta_n \tilde{g}$, $\delta\gamma_e = \delta_e \tilde{g}$ and $\delta\gamma_d = \delta\gamma$ for parts of the Midwestern and Southern territories of the U.S. The magnitude of the error components is below 0.08 mg.

The discrete spatial autocorrelation functions with the discrete north offsets $x_{n,i}$ and discrete east offsets $x_{e,i}$, $i=1\dots N$, are computed from the grid values according to

$$R_{\delta\gamma_n \delta\gamma_n}(x_{n,i}, x_{e,j}) = \frac{1}{N} \sum_{j=1}^N \delta\gamma_{n,j}(0,0) \cdot \delta\gamma_{n,j}(x_{n,i}, x_{e,j}) \quad (3.91)$$

where N is the number of grid points and $\delta\gamma_{n,j}(0,0)$ are the grid values.

A non-linear least squares solver allowing for boundary conditions is applied to estimate the standard deviation of the disturbing gravity potential σ_T and the correlation length parameter β from the gridded data. The cost function J to be minimized by the solver is

$$\begin{aligned} J = \sum_{i=1}^N & \left\{ \left[R_{\delta\gamma_n \delta\gamma_n}(x_{n,i}, x_{e,i}) - \sigma_T^2 \beta^2 e^{-\beta|r_i|} (1 - \beta|r_i| \cos^2 \chi_i) \right]^2 \right. \\ & + \left[R_{\delta\gamma_e \delta\gamma_e}(x_{n,i}, x_{e,i}) - \sigma_T^2 \beta^2 e^{-\beta|r_i|} (1 - \beta|r_i| \sin^2 \chi_i) \right]^2 \\ & + \left[R_{\delta\gamma_n \delta\gamma_e}(x_{n,i}, x_{e,i}) + \sigma_T^2 \beta^2 e^{-\beta|r_i|} \beta|r_i| \sin \chi_i \cos \chi_i \right]^2 \\ & \left. + \left[R_{\delta\gamma_d \delta\gamma_d}(x_{n,i}, x_{e,i}) - \sigma_T^2 \beta^2 e^{-\beta|r_i|} (2 - \beta|r_i|) \right]^2 \right\} \end{aligned} \quad (3.92)$$

The estimated parameters for the Midwestern and Southern territories of the U.S. are

$$\begin{aligned} \sigma_T &= 6.0740 \text{ m}^2/\text{s}^2 \\ \beta &= 2.39 \cdot 10^{-5} \text{ m}^{-1} \end{aligned} \quad (3.93)$$

The corresponding correlation length is $\frac{2.146}{\beta} = 89.8 \text{ km}$. Figure 3-10 shows the spatial autocorrelation function of the north gravity error as computed by (3.91) and the autocorrelation function with the estimated parameters (3.93). Figure 3-11 shows both autocorrelation functions for the east gravity error and Figure 3-12 the autocorrelation functions for the downward pointing gravity error. The true and the estimated autocorrelation functions fit well in all three directions. Note that the stochastic gravity error model parameters are not global but vary regionally. The standard deviation of the disturbing gravity potential and the correlation length depend, for example, on the roughness of the terrain surface. For an operational system it is recommended to repeat the parameter estimation offline for the different regions in which the system is intended to be operated and to file the computed values in a table that is accessible online by the navigation algorithm.

The model will be integrated into the navigation filter and analyzed in section 4.6.2.

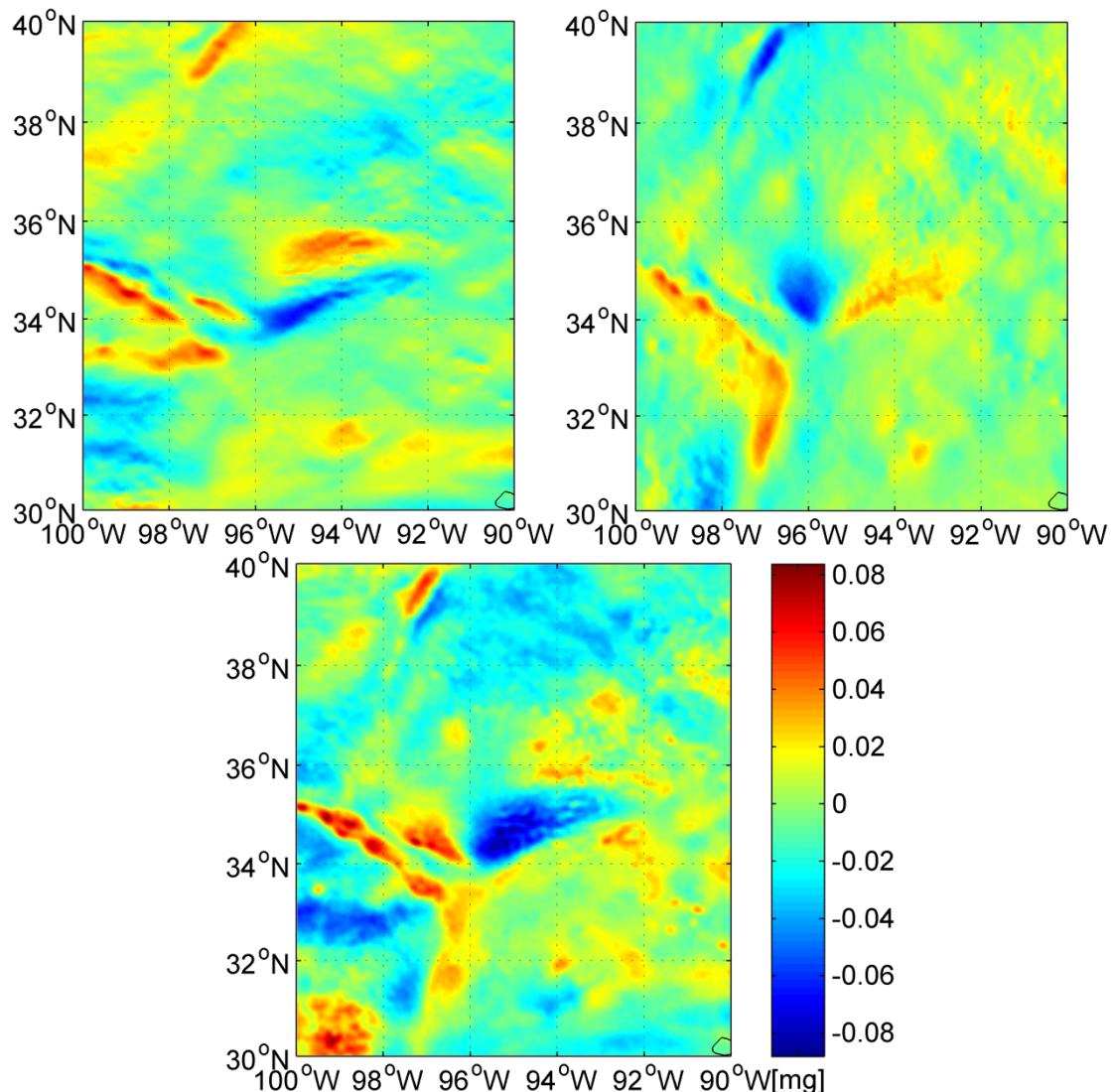


Figure 3-9: Difference between the EGM2008 gravity and Somigliana's normal gravity in north, east and down direction in parts of the Midwestern and Southern U.S. territories

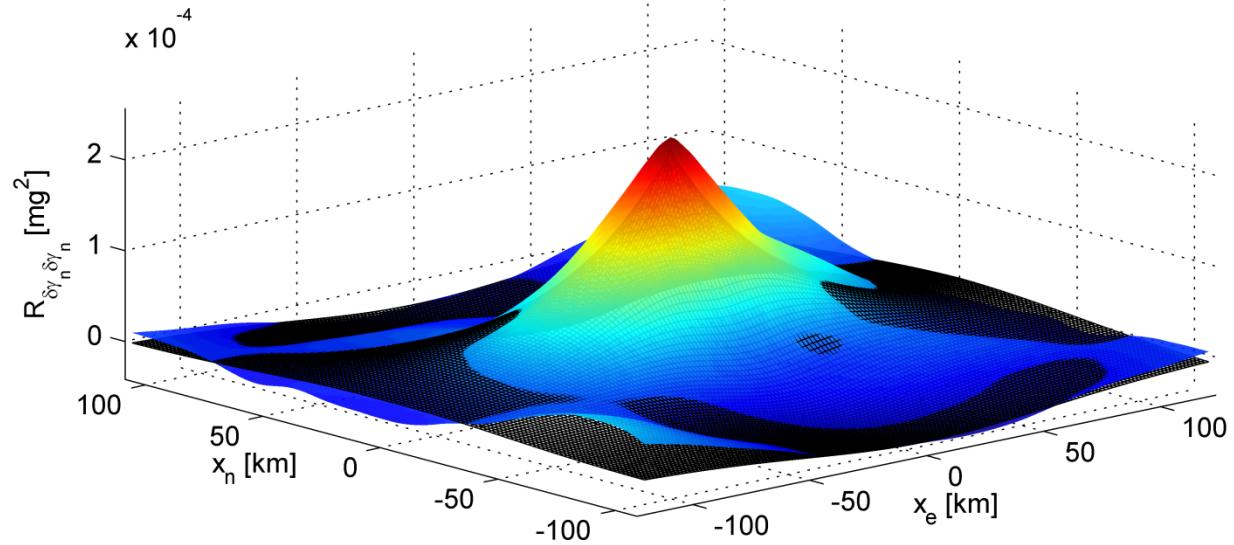


Figure 3-10: True (colored surface) and estimated (black mesh) autocorrelation function $R_{\delta\gamma_n \delta\gamma_n}$

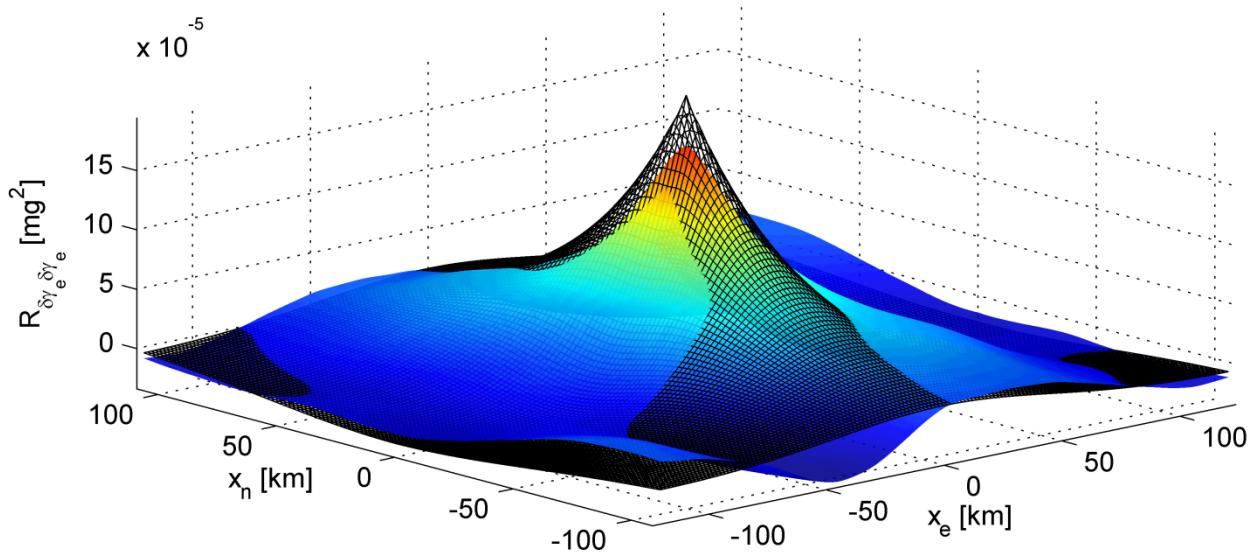


Figure 3-11: True (colored surface) and estimated (black mesh) autocorrelation function $R_{\delta\gamma_e \delta\gamma_e}$

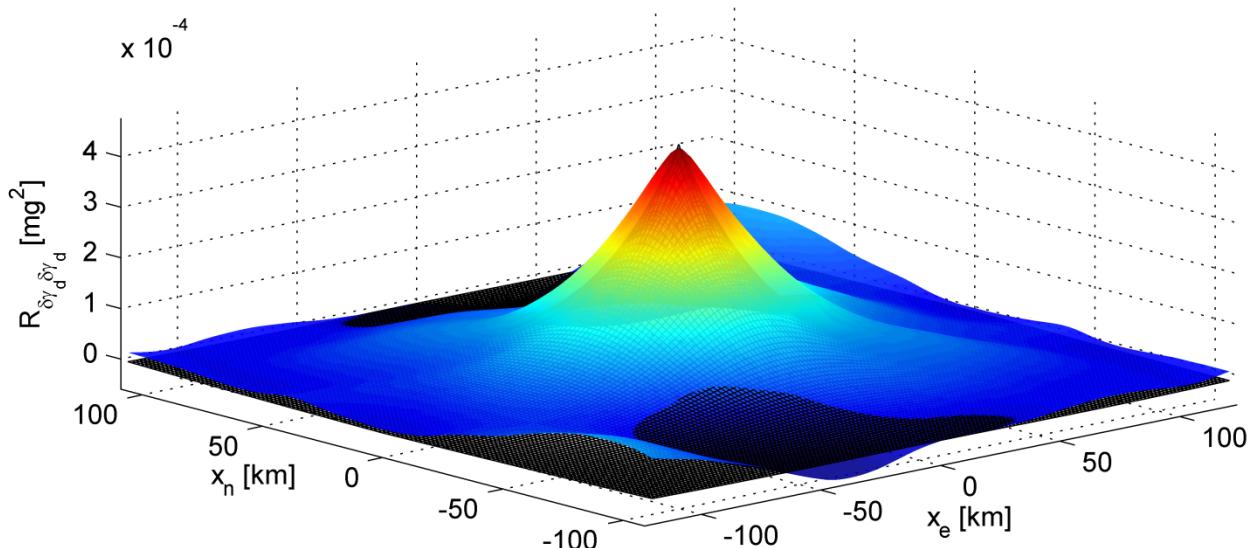


Figure 3-12: True (colored surface) and estimated (black mesh) autocorrelation function $R_{\delta\gamma_d \delta\gamma_d}$

3.4 Random Vibration

3.4.1 Motivation

Under unfavorable conditions, the influence of vibrational noise on the inertial navigation error can exceed the influence of the inertial sensor measurement errors. In these cases, the power spectral density of the highly frequent vibrations might be larger than that of the IMU measurement noise and bias instabilities. Only if the IMU is effectively isolated from these vibrations, the IMU errors are the main contributing factor to the inertial navigation solution error. In this section, a method is presented to estimate the PSD of the accelerometer and gyroscope signals in real-time and to adapt the process noise model in the integrated navigation filter accordingly.

Vibrations are superposed oscillating motions of the structure with different frequencies and amplitudes. They occur on the translational as well as rotational degrees of freedom. The vibration frequencies are usually higher than of the vehicle motion. Aerodynamic or propulsion forces and moments that act on the vehicle may excite the eigenmodes of the platform structure, resulting in small amplitude oscillations at the installation location of the IMU with frequencies close to the natural ones of the structure. Furthermore, imbalances of rotating shafts in engines, electric motors or other mechanical devices that are mounted in the vicinity of the IMU may also cause vibrations. Another source may be aeroacoustically generated noise. It is distinguished between vibration with distinct frequencies and narrow line spectra and vibrational noise with broadband spectra.

In principal, vibrations are part of the motion of the vehicle or at least of the IMU and one cannot necessarily distinguish between “true” vehicle motion and vibration from the accelerometer and gyroscope measurements. However, there are two aspects why the system designer may nevertheless be interested in isolating the vibrational part from the remainder of the motion. First, if the overall motion of the carrier platform shall be determined, vibrations that are locally induced by a mechanical device in the proximity of the IMU are not characteristic for the overall motion and will hence falsify the inertial navigation solution. And second, more important, only the low frequent parts of the motion with frequencies smaller than about a fifth of the Nyquist frequency are integrated by the numerical integration method without loss, meaning that the accumulated numerical error actually corresponds to the expected truncation error of the chosen scheme. Motion parts with higher frequencies are sampled with too few points. If oscillations are sampled with less than five points per period, they may not be correctly represented after the discretization. The integration error will be larger than expected from the truncation order.

Therefore, the location as well as the mounting of the IMU have to be carefully chosen. The device should be installed at a place where the effect of externally and internally induced structural vibrations is as small as possible. From simulations, for example, with the finite element method or from previously recorded flight data the frequencies and amplitudes of structural vibrations can be determined at the intended installation location. Periodograms and spectrograms of the simulated or recorded accelerations and angular rates help to gain insight into the distribution of signal power over the frequency range during the flight. Well-suited locations feature low signal power in the higher frequency range. Once a feasible place has been found for the IMU, the effect of random vibration is further reduced by mounting the IMU with suitable damping elements whose resonant frequencies are tuned to the predominant frequencies in order to damp most of the interfering motion energy.

The remaining disturbing influence can be further countered by adequate signal processing. Notch filters or narrow band-pass filters are one possibility to wipe out motion parts with distinct frequencies or narrow frequency bands with especially high interfering signal energy content.

In an integrated navigation filter the effect of the inertial navigation errors due to insufficiently resolved vibrational motion in the high frequency range has to be accounted for by the process noise covariance matrix \mathbf{Q} . The question arises how to tune the process noise covariance matrix in this case. The simplest way is to look at the power spectral densities of the accelerations \mathbf{f}_b and angular rates $\boldsymbol{\omega}_{ib}$ that have been previously recorded at a representative flight. The PSDs, which usually vary over frequency, are substituted by white noise with constant density over frequency. The task is to find the amplitude of the substitute white noise such that its PSD best fits or slightly over-bounds the actual characteristic in the high frequency range. One possibility is to calculate the signal power by integrating the PSD over the considered frequency range and subsequently dividing it by the same frequency interval to obtain the substitute white noise with constant power spectral density but same power content as the original. The procedure has to be separately done for all six channels of the IMU. The computed white noise PSDs can be directly inserted into the process noise covariance matrix of the integrated navigation filter without further modifications.

In higher sophisticated systems, the system designer can identify a model that considers the varying power over the frequency range. It is thus possible to concentrate signal power at certain distinct frequencies or frequency bands. Autoregressive models are appropriate for that purpose. The usage of such models has also the advantage that time correlations of the errors are described in contrast to the substitute Gaussian white noise assumption. The higher the chosen model order, the more details of the PSD can be represented. The chosen order is merely a question of computational load. In quasi-stationary conditions, constant models are adequate. This is, for example, the case if electric motors or ventilators with constant revolution speeds cause the vibrations. When different regimes like high dynamic motion, flight in turbulence or slow flight in calm conditions alternate, an adaption of the model to each situation is proposed.

Even if the vibration model is not integrated into the navigation filter, the RTCA standard DO-334 [5] requires that vibration (including acoustically induced vibration) has to be accounted for in the simulation model if parts of the certification process of an AHRS are done by simulation instead of flight evaluation. Then, knowledge about the prevailing vibrations at the intended installation location and deduced from that a high fidelity model are necessary for the synthesis of the acceleration and angular rate measurements in the simulation.

Finally, it has to be noted that random vibration can additionally provoke bias-like measurement errors besides the vibrational noise, which primarily result from the non-linearity of the scale factor and are characterized by the vibration rectification coefficients. These errors are however not part of this discussion here and will have to be analyzed in future in the ongoing research. Background information about the origin of vibration rectification can be found in [42] [43] [44] [45]. A brief overview about the effect of vibration on the inertial solution can also be read in [8] and [27].

First, some remarks are made about the power spectral analysis of the signals to get an insight. The power spectral analysis is actually not required in the algorithm of the integrated navigation system. Then, a real-time capable power spectral estimation algorithm and the derivation of a suitable autoregressive model are presented.

3.4.2 Power Spectral Analysis of IMU Measurements

Periodograms of the separate IMU signals at different instants of time along the trajectory reveal the actual motion of the platform, mainly in the low frequency range, and superposed vibrations and sensor errors, mainly in the high frequency range. Due to the aliasing effect the PSD is symmetrical to multiples of the Nyquist frequency $f_s/2$ as well as the sampling rate f_s as illustrated in Figure 3-13. The PSD cannot distinguish whether the power at frequency f stems from signal frequencies f , $f_s - f$ or integer multiples of these.

In order to separate the motion part that can be integrated without loss by the inertial navigation algorithm from the perturbing vibrational part of the signal, the low frequency range containing the motion has to be removed by means of a high-pass filter. It has to be noted that then, owing to the aliasing effect, not only the desired frequency range below the chosen cut-off frequency $\omega_c/(2\pi)$ is filtered but also the frequency band about the sampling frequency f_s in the range between $f_s - \omega_c/(2\pi)$ and $f_s + \omega_c/(2\pi)$. The same happens at $2f_s$, however, the power content at this high frequency is assumed to be small and its influence should be small.

The military standard of the Department of Defense MIL-STD-810G “Environmental Engineering Considerations and Laboratory Tests” [46] gives in section 514.6 examples for vibrational noise PSDs that are usually encountered in aircraft (jet/propeller aircraft, helicopter, ...) at different locations (instrument panel, stores, ...). The standard defines, amongst others, how devices like IMUs or integrated navigation systems to be installed in a military aerial vehicle have to be tested with respect to the expected vibrations for example in the laboratory in order to obtain their certification. On the one hand, the equipment has to resist the mechanical load induced by vibrations and shall not get broken under the expected vibrations. On the other hand, the navigation system has to fulfill the specified performance, even under these expected vibrations. Especially categories 12, 13 and 14 apply for installed materiel on jet, propeller and helicopter aircraft. Categories 16, 17 and 18 hold for materiel that is installed in the stores of jet, propeller and helicopter aircraft. Category 19 specifies the vibrational regime of freely flying missiles with running engine. Figure 3-14 shows the typical PSD that can be expected on jet aircraft (MIL-STD-810G 514.6 Category 12). Figure 3-15 illustrates the typical vibrational noise PSD of propeller aircraft (Category 13). Note that the high frequency vibrations are damped by appropriately tuned shock absorbers and thus do not interfere the IMU/integrated navigation system.

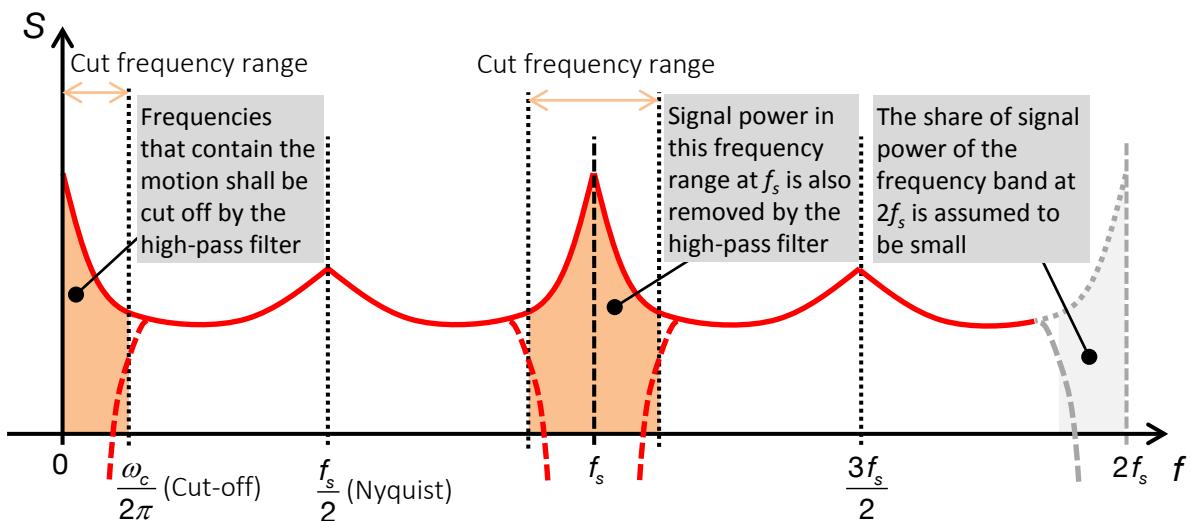
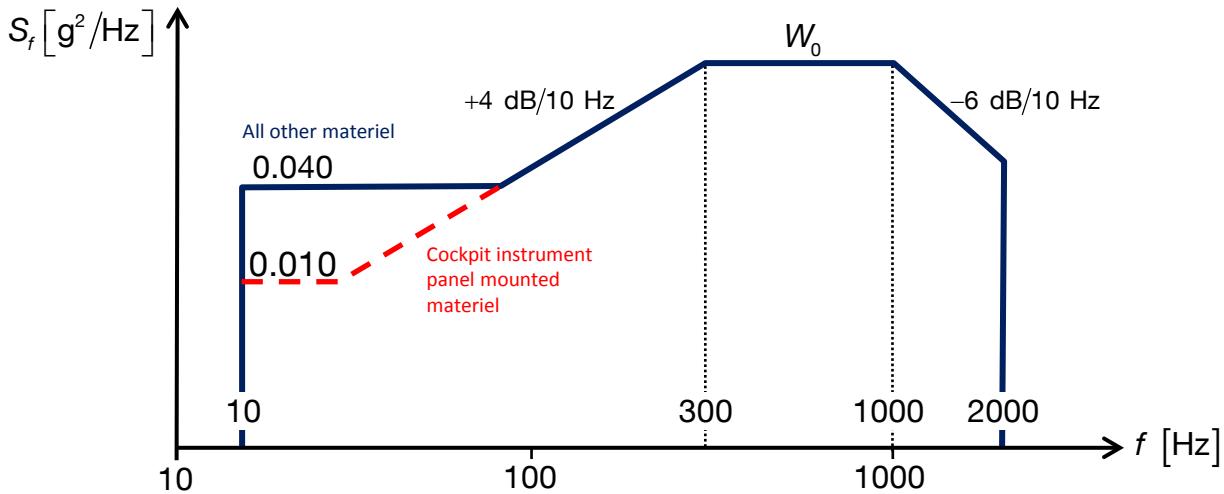
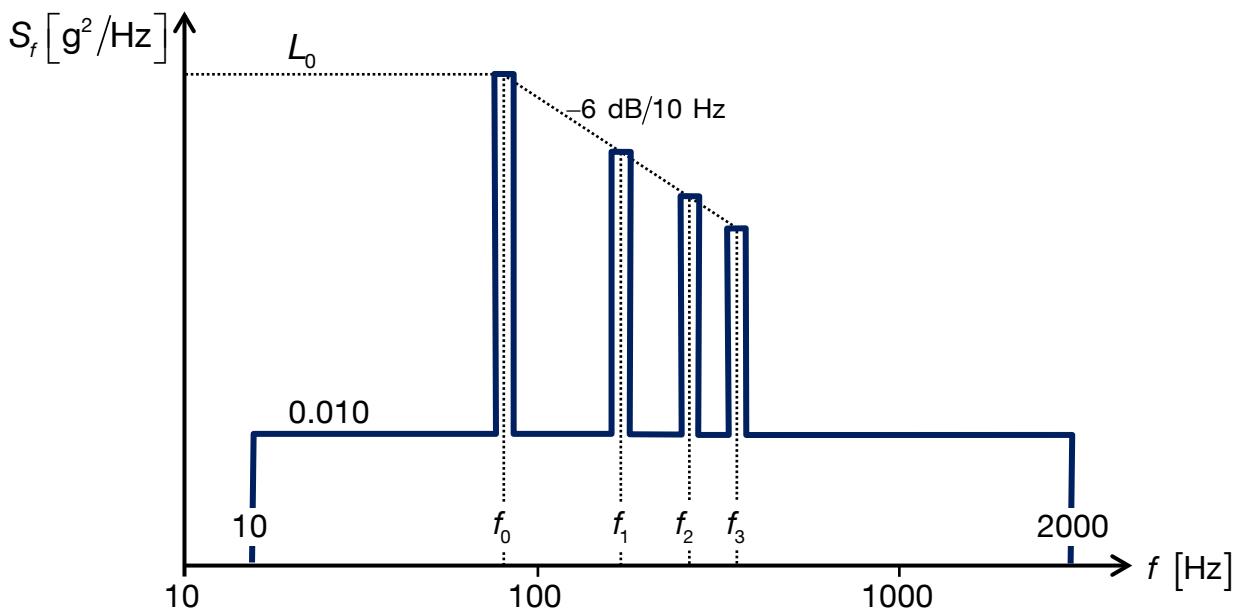


Figure 3-13: Exemplary power spectral density and cut-off ranges



**Figure 3-14: Typical vibrational noise PSD for material installed on jet aircraft
(MIL-STD-810G 514.6 Category 12, [46])**



**Figure 3-15: Typical vibrational noise PSD for material installed on propeller aircraft
(MIL-STD-810G 514.6 Category 13, [46])**

3.4.3 High-Pass Filtering of IMU Measurements

The low frequency accelerations and angular rates are integrated by the inertial navigation algorithm without loss because the motion is resolved with a sufficiently high number of samples. The focus lies on the frequencies higher than a fifth of Nyquist where numerical errors due to under-sampling are expected.

It is therefore recommended to cut off the low frequent part of the motion by means of high-pass filtering before the coefficients of the vibration model are estimated. The high-pass filter is characterized by its cut-off frequency and its rate of frequency roll-off. The higher the order of the filter is chosen, the higher the roll-off rate but also the phase lag. However, since the filter does not work in closed-loop and the phase lag does not play a role, the filter order can be chosen arbitrarily. The cut-off can be set very sharply, for example with a filter order of 10. In principal, only the numerical effort limits the order of the chosen filter. It is proposed to use a Butterworth filter.

The transfer function of the resulting high-pass filter takes the general form

$$H_{HPF}(s) = \frac{s^n}{s^n + a'_{n-1} \omega_c s^{n-1} + \dots + a'_1 \omega_c^{n-1} s + a'_0 \omega_c^n} \quad (3.94)$$

In Table 3-11 the denominator coefficients a'_i of the high-pass filter transfer function H_{HPF} are calculated for filters up to order $n = 4$.

Table 3-11: High-pass filter coefficients a'_i

n	a'_0	a'_1	a'_2	a'_3
1	1			
2	1	$\sqrt{2}$		
3	1	2	2	
4	1	$\sqrt{2 + \sqrt{2}} + \sqrt{2 - \sqrt{2}}$	$2 + \sqrt{2 + \sqrt{2}} \sqrt{2 - \sqrt{2}}$	$\sqrt{2 + \sqrt{2}} + \sqrt{2 - \sqrt{2}}$

Finally, the continuous-time high-pass filter transfer function (3.94) has to be converted to discrete-time. This is accomplished by substituting the continuous-time integrator $1/s$ by a discrete-time forward Euler integrator according to

$$s = \frac{1}{\Delta t} \frac{1 - z^{-1}}{z} \quad (3.95)$$

Therein, Δt is the sample time. Alternatively, a bilinear transformation, which is actually a trapezoidal integrator of second order accuracy, could be used. The discrete-time high-pass filter transfer function (3.94) will generally be

$$H_{HPF}(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_{n-1} z^{-(n-1)} + b_n z^{-n}}{1 - a_1 z^{-1} - \dots - a_{n-1} z^{-(n-1)} - a_n z^{-n}} \quad (3.96)$$

with the numerator coefficients b_i , $i = 0 \dots n$ and the denominator coefficients a_i , $i = 1 \dots n$. Table 3-12 gives the coefficients for high-pass filters up to order $n = 4$, where $T = \omega_c \Delta t$.

Table 3-12: Discrete-time high-pass filter coefficients a_i and b_i ($T = \omega_c \Delta t$)

n	0	1	2	3	4
1	b	1	-1		
	a		$-(a'_0 T - 1)$		
2	b	1	-2	1	
	a		$-(a'_1 T - 2)$	$-(a'_0 T^2 - a'_1 T + 1)$	
3	b	1	-3	3	-1
	a		$-(a'_2 T - 3)$	$-(a'_1 T^2 - 2 a'_2 T + 3)$	$-(a'_0 T^3 - a'_1 T^2 + a'_2 T - 1)$
4	b	1	-4	6	-4
	a		$-(a'_3 T - 4)$	$-(a'_2 T^2 - 3 a'_3 T + 6)$	$-(a'_1 T^3 - 2 a'_2 T^2 + \dots - 3 a'_3 T - 4)$

3.4.4 Discrete-Time State-Space Model

Next, AR models are searched that adequately represent the PSD of the filtered IMU signals. The estimation of the AR model coefficients has to be real-time capable so that it can be applied in the integrated navigation filter. The AR model coefficients are estimated with Burg's method, which has been presented in section 3.1.5.4.

The state-space model for the translational vibrational noise in the three orthogonal directions is

$$\begin{aligned} \mathbf{z}_{f_x,k} &= \underbrace{\begin{pmatrix} 0 & 1 & 0 & \cdots \\ 0 & \ddots & \ddots & 0 \\ & & 0 & 1 \\ \hat{a}_{f_x,p} & \cdots & \hat{a}_{f_x,2} & \hat{a}_{f_x,1} \end{pmatrix}}_{\Phi_{f_x,k}} \mathbf{z}_{f_x,k-1} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}}_{\Gamma_{f_x}} \eta_{f_x,k-1}, & X = \{x, y, z\} \\ \delta f_{b,X,k} &= \underbrace{\left(\hat{a}_{f_x,p}, \dots, \hat{a}_{f_x,2}, \hat{a}_{f_x,1} \right)}_{\mathbf{H}_{f_x,k}} \mathbf{z}_{f_x,k} + \eta_{f_x,k} & \eta_{f_x,k} \sim WN(0, Q_{\eta,f_x,k}) \end{aligned} \quad (3.97)$$

and analogous for the rotational vibrational noise in the three orthogonal directions. The input noise is white and Gaussian distributed with the estimated input noise variance Q_η .

The model coefficients are regularly updated by the online estimation algorithm, for example at every covariance propagation step of the navigation filter. The models are thus adjusted to the currently prevailing vibration regime. It is best to choose a window of the latest m measurement samples as depicted in Figure 3-16. The integrated navigation filter can estimate the AR coefficients with lower priority in parallel to the inertial navigation solution.

Even if the AR model coefficients are estimated only at the covariance propagation steps of the integrated navigation filter, the time step size of the discrete-time vibration state-space models (3.97) is the IMU sample time $\Delta t = t_i - t_{i-1}$. In the propagation step of the navigation filter, however, a discrete-time model at the propagation rate is required. The propagation time is an integer multiple of the IMU sample time, $T = t_k - t_{k-1} = n \Delta t$. The vibration state-space matrices are assumed to be constant between two propagation steps, $\Phi_i = \Phi$, $\Gamma_i = \Gamma$. The conversion from the IMU sample rate to the propagation rate is described in section 2.5.

The model will be integrated into the navigation filter and analyzed in section 4.6.3.

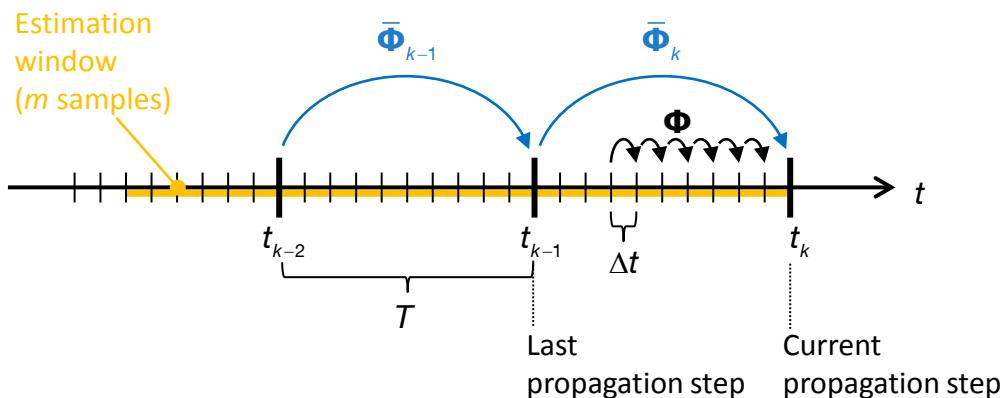


Figure 3-16: State propagation, covariance propagation and estimation window

3.5 GPS Observable Errors

3.5.1 Satellite Clock Error

3.5.1.1 Motivation

Satellite navigation bases on measuring the time of flight of a radio signal between the satellite and user antenna. The signal travel time is determined by subtracting the broadcast time from the time of receipt. In order to obtain accurate range measurements, the broadcast time has to be accurately known. Since the signal travels with speed of light, a 1 ns time error already yields a 0.3 m range error. Although GPS satellites incorporate highly stable atomic cesium (CS) and rubidium (RB) clocks, the clock time gradually deviates from true reference (GPS) time. The control segment monitors the errors of each satellite clock and provides clock correction parameters in the navigation message. However, after applying the clock correction to the measured pseudoranges, uncompensated range errors of up to 4 m remain, depending on the age of the navigation message, the satellite block type and the activated clock in the satellite. The range error is typically bias-like with correlation length in the range of an hour. These bias-like range errors result in permanently biased position solutions. Pseudorange measurement errors are often modeled as white noise with sufficiently large variance to over-bound the effect of the omitted correlations. If the variance of the substitute noise is too small, the navigation error covariance settles too much, which leads to too optimistic statistics. This in turn can cause false fault detections and exclusions of measurements by the innovation based integrity monitor.

In this section a model for the residual satellite clock error shall be identified. At first, the short and long term satellite clock error characteristics are analyzed by comparing the satellite clock error that is calculated with the parameters in the broadcast navigation message with the precise satellite clock error that is provided as product from the International GNSS Service (IGS) network. For that, data of a complete year is evaluated. Afterwards, the model structure is found and the corresponding model coefficients are estimated from the real data. The model will consider upload and dataset cutovers of the navigation message and will account for the increasing satellite clock error with elapsing time since the last ephemeris update. Finally, an outlook on the new navigation messages that are modulated on the new GPS signals will be given. It is expected that the accuracy of the satellite clock error will increase with the additional correction parameters, such that it will probably not be required anymore to augment the navigation error filter by the models presented in this section in future. In [47] [48] [49] [50] [51], GPS clock and orbit ephemeris errors are analyzed. In [47], standard deviations and Gaussian overbounds are derived for the instantaneous GPS clock and orbit errors of the IIA, IIR and IIR-M satellite blocks. None of the contributions presents a statistical model that describes the temporal correlation of the errors.

3.5.1.2 Upload and Dataset Cutovers

The navigation message contains, amongst others, the ephemeris data that is required to determine the satellite position at broadcast time and the satellite clock correction coefficients [52]. The navigation messages are regularly updated to reduce range errors stemming from the space segment. The event when a satellite begins to broadcast a new navigation dataset is called cutover. Two kinds of cutovers are possible, the upload cutover and the dataset cutover. An upload cutover occurs when

a fresh navigation dataset is uploaded to the satellite by the control segment and transmitted by the satellite afterwards. At a dataset cutover the navigation message is autonomously updated by the satellite. In this case the dataset is slightly modified with the results of the prediction computations that are independently performed by the satellite. The changes of the ephemeris parameters are usually larger at an upload cutover than at a dataset cutover. A new navigation message is indicated to the user by a change of the Clock Issue of Data (IODC) value and the Ephemeris Issue of Data (IODE) value, respectively. The IODC value is unique within the last seven days. When dealing with navigation datasets, it is important to distinguish between the following times. First, the time of transmission (TOT) is the time when the satellite starts to broadcast a new ephemeris dataset. Second, the time of ephemeris (TOE) is the reference time for the ephemeris parameters and third, the time of clock (TOC) is the reference time for the satellite clock error correction parameters. Dataset cutovers occur on full hour whereas upload cutovers may also occur during the hour. At normal operation, a dataset is continuously broadcast during a two hour transmission interval. Cutovers usually take place at 0 h, 2 h, 4 h, ... The curve fit interval, which is the validity time of the current dataset, begins right after the start of transmission and lasts four hours. The TOE is two hours after the TOT of the dataset. In order to indicate that an upload cutover has occurred instead of a dataset cutover, the TOE differs from the full hour in the first and second navigation datasets after the upload cutover. In this case, the TOE is usually 16 seconds before the full hour. The highest accuracy of the satellite position, velocity and clock error is obtained if the ephemeris dataset is used during the two hour interval around TOE. This means that the currently broadcast dataset is not applied until one hour before its TOE (unless an irregular upload cutover occurs in the meantime). The selection procedure for dataset cutovers is illustrated in Figure 3-17. The transmission of dataset 1 begins at midnight. It is broadcast for two hours until two o'clock am. The TOE of dataset 1 is two o'clock am, that is exactly in the middle of the four hour curve fit interval between midnight and four o'clock am. It would be optimal regarding navigation accuracy to use this dataset from one o'clock am to three o'clock am. Transmission of dataset 2 starts at two o'clock am and so on. Figure 3-18 shows exemplarily how an upload cutover may be initiated. The situation is the same as in Figure 3-17 until 3:10:32. At this time, the new dataset 3 is uploaded by the control segment. The new navigation data should be immediately applied. The TOE of the uploaded dataset is at 3:59:44 instead of 4 o'clock am to announce the occurred upload cutover. The TOE of the following dataset 4 is at 5:59:44 instead of 6 o'clock.

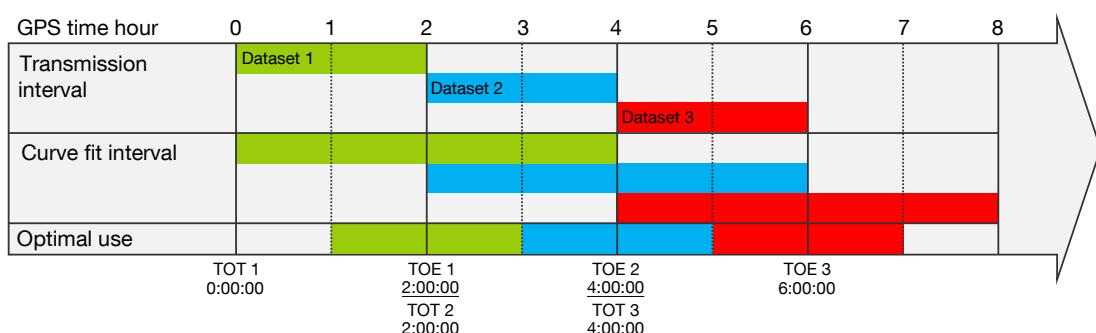
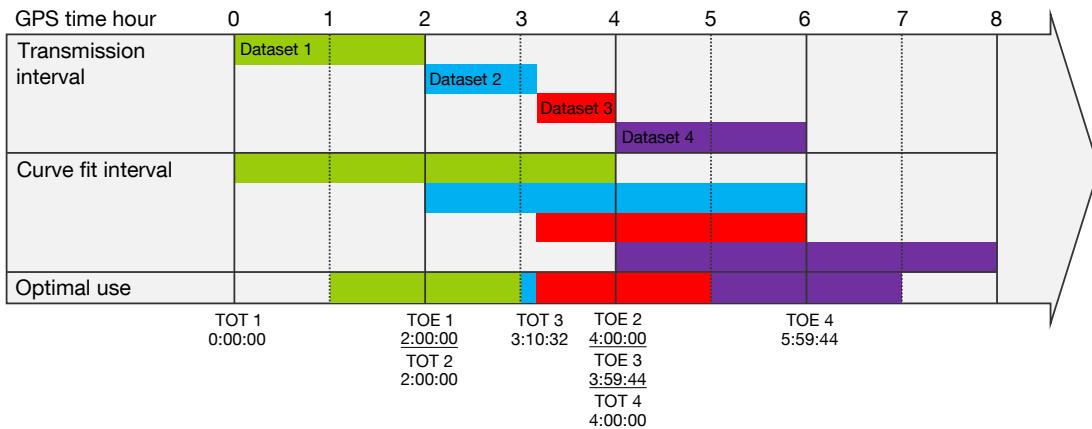


Figure 3-17: Dataset cutovers

**Figure 3-18: Upload cutover**

3.5.1.3 Standard Positioning Service Performance Standard

The range error of the signal in space (SIS) is called User Range Error (URE) [53]. It includes all errors that are caused by the control and the space segment. These are predominantly non-compensated satellite position and satellite clock errors and inter-signal biases. The range rate error of the SIS is accordingly denoted as User Range Rate Error (URRE). The User Equivalent Range Error (UERE) is the actual ranging error that is experienced by the user when processing the pseudorange measurements. In addition to the URE, the UERE contains the errors stemming from the atmosphere and the user segment. These errors are primarily non-compensated ionospheric and tropospheric delays, multipath effects and receiver noise. The User Range Accuracy (URA) index is a conservative measure of the current URE and is transmitted by the satellites in the navigation message. The receiver can use the URA to judge the current SIS accuracy. The conventional URA definition that is included in the legacy navigation (LNAV) message, which is modulated on the L1 and L2 signals, is cited in Table 3-13. In the new civil navigation (CNAV) message, which is available on the L2C and L5 signal, further URA indices are added for better accuracies. A similar URA definition can be found in the SBAS standard RTCA DO-229D [3].

The GPS standard positioning service (SPS) performance standard [53] specifies the guaranteed accuracy of the SIS. The relevant accuracies of the L1 signal with C/A code are cited in Table 3-14 for the URE and in Table 3-15 for the URRE. Looking at the nominal situation in the first row of Table 3-14, the expected pseudorange error is with 6 m at 95% smallest immediately after an upload cutover. With increasing age of data (AOD), the SIS error increases. The overall mean value of the SIS error over space and time is 7.8 m at 95%.

The maximum error is 12.8 m at 95%. It will likely occur towards the end of the period between two subsequent upload cutovers. The URRE is with 0.006 m/s at 95% very small. Therefore, additional models for the range rate error are not required.

Besides the cited performance numbers of the SIS error, the GPS SPS performance standard [53] gives quantitatively the characteristics of the SIS error. Figure 3-19 shows the increase of the SIS error 95% deviation with the AOD. In nominal operation, there is about one upload cutover per day. If the satellite clock drifts faster than usual, the control segment can switch to more frequent upload cutovers, as illustrated by an exemplary realization of the SIS error in Figure 3-20. The minor steps in the SIS error stem from the dataset cutovers that occur regularly every two hours.

Table 3-13: URA index as defined in IS-GPS-200G [52]

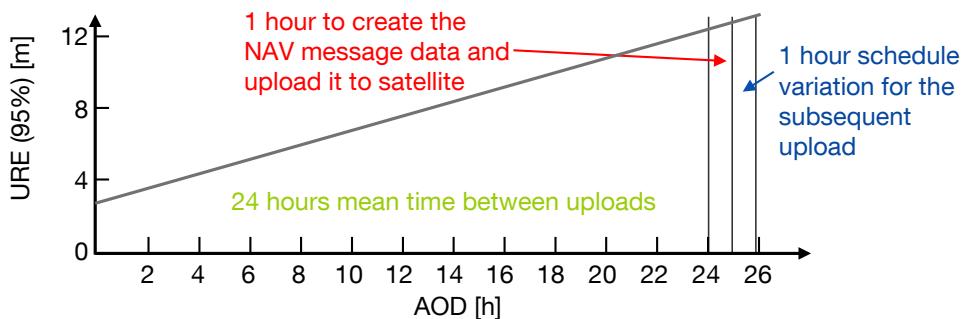
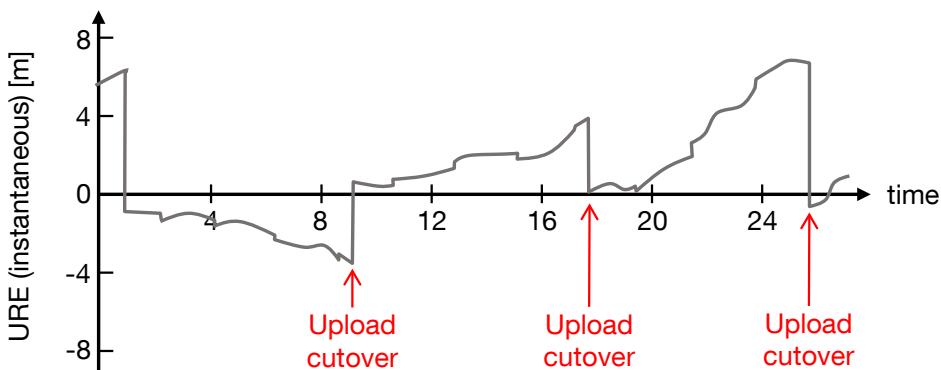
LNAV Message		CNAV Message			
URA Index	URA Value [m]	URA Index	URA Value [m]	URA Index	URA Value [m]
0	2.40	-16	Use at own risk	0	2.40
1	3.40	-15	0.01	1	3.40
2	4.85	-14	0.02	2	4.85
3	6.85	-13	0.03	3	6.85
4	9.65	-12	0.04	4	9.65
5	13.65	-11	0.06	5	13.65
6	24.00	-10	0.08	6	24.00
7	48.00	-9	0.11	7	48.00
8	96.00	-8	0.15	8	96.00
9	192.00	-7	0.21	9	192.00
10	384.00	-6	0.30	10	384.00
11	768.00	-5	0.43	11	768.00
12	1536.00	-4	0.60	12	1536.00
13	3072.00	-3	0.85	13	3072.00
14	6144.00	-2	1.20	14	6144.00
15	Do not use	-1	1.70	15	Do not use

Table 3-14: SPS SIS URE accuracy standards [53], p.22

SIS Accuracy Standard	Conditions and Constraints
Single-Frequency C/A-Code:	<ul style="list-style-type: none"> For any healthy SPS SIS Neglecting single-frequency ionospheric delay model errors Including group delay time correction (TGD) errors at L1 Including inter-signal bias (P(Y)-code to C/A-code) errors at L1
Single-Frequency C/A-Code:	<ul style="list-style-type: none"> For any healthy SPS SIS Neglecting single-frequency ionospheric delay model errors Including group delay time correction (TGD) errors at L1 Including inter-signal bias (P(Y)-code to C/A-code) errors at L1 Standard based on measurement interval of one year; average of daily values within the service volume Standard based on 3 service failures per year, lasting no more than 6 hours each
Single-Frequency C/A-Code:	<ul style="list-style-type: none"> For any healthy SPS SIS

Table 3-15: SPS SIS URRE accuracy standards [53], p.23

SIS Accuracy Standard	Conditions and Constraints
Single-Frequency C/A-Code: <ul style="list-style-type: none"> • 0.006 m/sec 95% Global Average URRE over any 3-second interval during Normal Operations at Any AOD 	<ul style="list-style-type: none"> • For any healthy SPS SIS • Neglecting all perceived pseudorange rate errors attributable to pseudorange step changes caused by NAV message data cutovers • Neglecting single-frequency ionospheric delay model errors

**Figure 3-19: URE variation during normal operation, one upload per day [53]****Figure 3-20: URE variation during normal operation, three uploads per day [53]**

Even if the characteristic of the instantaneous variance of the URE is given at least quantitatively in the GPS SPS performance standard, it does not contain any information about the time correlation of the SIS error. The SBAS standard RTCA DO-229D provides in fact more detailed information about the change of the instantaneous covariance matrix, but does also not make a statement about the time correlation of the URE. For the statistically consistent data fusion within the navigation error filter, the knowledge of the time correlation is essential.

As will be seen later by comparing the separate contributions of the satellite clock error and the orbit error to the URE by means of real data, it can be stated that the satellite clock error is responsible for the larger share of the URE. The pseudorange error owing to the satellite position error is in the range of one meter at high AODs. The increase of the URE variance between subsequent upload cutovers is mainly due to the satellite clock error whereas the orbit error depends less on the AOD of the ephemeris set.

3.5.1.4 Characteristic

3.5.1.4.1 Broadcast Clock Error Correction

The master control station continuously monitors the satellite clock errors, determines correction parameters and provides them in the navigation message. The user has to apply these parameters to correct the measurement. The satellite clock error is defined as

$$\Delta t_s = t_s - t \quad (3.98)$$

[52]. t is the true GPS time and t_s denotes the current erroneous satellite clock time. A clock offset $\Delta t_s > 0$ means that the satellite clock runs faster, a clock offset $\Delta t_s < 0$ means that the satellite clock is slower compared to true GPS time. GPS uses a quadratic polynomial for the satellite clock error $\tilde{\Delta t}_s$

$$\tilde{\Delta t}_s(t) = \tilde{a}_{f0} + \tilde{a}_{f1}(t - t_{oc}) + \tilde{a}_{f2}(t - t_{oc})^2 + \Delta t_r - T_{GD} \quad (3.99)$$

The polynomial coefficients \tilde{a}_{f0} , \tilde{a}_{f1} , \tilde{a}_{f2} as well as the mean group delay differential T_{GD} are contained in the LNAV message. The relativistic correction Δt_r can be computed from parameters in the navigation message. t_{oc} is the TOC and represents the reference time of the currently broadcast clock correction coefficient set. The mean group delay differential T_{GD} has only to be applied by single (L1) frequency users. With the clock correction coefficients in the navigation message, the medium-to long-term bias, drift and drift rate of the satellite clock are compensated. Due to short-term, noise-like clock errors and erroneous correction coefficients, there is a residual error $\delta\Delta t_s$ left over after applying the correction. The true satellite clock error Δt_s can be decomposed into the satellite clock error correction $\tilde{\Delta t}_s$ and the residual error $\delta\Delta t_s$

$$\Delta t_s = \tilde{\Delta t}_s + \delta\Delta t_s \quad (3.100)$$

The satellite clock error drift $\dot{\tilde{\Delta t}}_s$, which is required to correct range rate measurements, is accordingly

$$\dot{\tilde{\Delta t}}_s = \tilde{a}_{f1} + 2\tilde{a}_{f2}(t - t_{oc}) + \dot{\Delta t}_r \quad (3.101)$$

Since the satellite clock error drift is observed to be nearly constant between two subsequent cutovers, the \tilde{a}_{f2} coefficient is usually set to zero in the navigation message.

3.5.1.4.2 Precise Clock Error

Whereas real-time users have to rely on the currently broadcast correction parameters in the navigation message, users with post-processing applications alternatively have access to precise clock products. IGS is a federation of several world-wide distributed institutions that operate a network of stationary monitoring receivers [54]. With the centrally collected measurements, the actual satellite orbit and clock errors as well as ionospheric and tropospheric delays are estimated. The IGS makes the processed data available free of charge. Depending on the processing time, different products with different accuracies are distributed (Table 3-16). There are ongoing plans to supply the user also with precise correction information in real-time.

It is important to note that the IGS clock products refer to the own IGS timescale and not to GPS time. This reference time is formed from a weighted combination of a few highly accurate atomic clocks within the IGS network [55]. The IGS timescale is loosely coupled to GPS time [56]. It is freely running but is roughly adapted once per day to reduce the deviation from GPS time. The current

difference between both timescales is noted as GNSS time offset (GGTO) and is provided with the clock products by the IGS [57]. In order to compare the broadcast satellite clock error with the precise clock error, the GGTO has to be applied to the IGS precise clock error. Furthermore, the relativistic correction Δt_r as well as the group delay T_{GD} are not contained in the IGS precise clock error and have thus to be added before the comparison.

Table 3-16: Overview about IGS clock products [58]

Product	Latency	Root Mean Square (RMS)	Standard Deviation	Sample Interval
Ultra-Rapid	3 – 9 hr	150 ps	50 ps	15 min
Rapid	17 – 41 hr	75 ps	20 ps	15 min
Final	12 – 18 days	75 ps	20 ps	30 s

3.5.1.4.3 Comparison of Broadcast and Precise Clock Errors

Table 3-17 lists the GPS satellite constellation in the year 2012 [59]. The current and former GPS constellation status (OPS advisories) can be obtained from the United States Coast Guard (USCG) Navigation Center. Apart from PRN 24 and PRN 27, all PRNs were continuously occupied by only one satellite. At PRN 27, the block IIA satellite SVN 27 was decommissioned at 10/17/2012 and replaced by the block IIR-M satellite SVN 49. At PRN 24, several interim satellites were parked for short periods until finally the third block IIF satellite SVN 65 was activated on this PRN. Most of the satellites used the RB clock. Only PRNs 3, 9, 8, 10 and 27 until decommission applied the CS clock. It is obvious that the CS clocks were activated only in the older block type IIA satellites. In the following, the broadcast and the precise satellite clock error are compared, exemplarily for the year 2012. For the comparison, the clock difference $\delta\Delta t_s$ between broadcast clock error $\Delta\tilde{t}_s$ and precise clock error $\Delta t_{s,prec}$ are calculated as $\delta\Delta t_s = \Delta\tilde{t}_s - \Delta t_{s,prec}$. Since the precise clock error $\Delta t_{s,prec}$ is more accurate than the broadcast clock error $\Delta\tilde{t}_s$ by some orders of magnitude, the precise clock error $\Delta t_{s,prec}$ is henceforth assumed to be the true satellite clock error $\Delta t_s = \Delta t_{s,prec}$. The final IGS clock product with 30 s sample interval is used for the precise clock error. Apart from block type IIA, one satellite of each block is analyzed. From block IIA, one satellite with activated CS clock and one with activated RB clock are selected. The analyzed satellites are highlighted in Table 3-17.

In Figure 3-21, Figure 3-23, Figure 3-25, Figure 3-27, Figure 3-29 and Figure 3-31 the clock error $\delta\Delta t_s$ is shown for seven subsequent days, that is the first week of 2012. The day changes at midnight are illustrated by the grey solid lines, the dataset cutovers every two hours by the grey dotted lines and the upload cutovers by the blue dotted lines. It can be seen that upload cutovers occur roughly every 24 hours, that is once per day. Only in the case of PRN 10, the CS clock drifts faster and the master control station provokes interim upload cutovers. It is obvious that the clock error corrections at upload cutovers are more considerable than at dataset cutovers. In Figure 3-22, Figure 3-24, Figure 3-26, Figure 3-28, Figure 3-30 and Figure 3-32 all periods of 2012 between two succeeding upload cutovers are considered in parallel, i.e. in upload cutover windows. The x-axis shows the AOD, which is the time since the last upload crossover. At time 0 the last upload crossover occurred. The grey dotted lines every two hours indicate the dataset cutovers. Six sample curves were randomly chosen to illustrate the time correlation of the error. The black dotted lines are the 3σ borders of all upload cutover windows. It can be observed that broadcast satellite clock errors are smallest after upload crossover and increase afterwards. Again, it can be seen that dataset crossover corrections are small compared to upload crossover corrections for satellites of type IIA, IIR-A, IIR-B and IIR-M, but are,

however, in the range of the upload cutover corrections for satellites of type IIF. The noise-like short period clock error of the block type IIF satellite seems to be smaller than those of the other block types. The clock errors of the oldest type IIA satellites can become largest. Many interim upload cutovers are necessary to correct the satellite clock error of PRN 10 with the CS clock activated. The steps in the 3σ line hint at frequent upload cutovers ahead of the planned time.

Table 3-17: GPS constellation in 2012, sorted by start of operation

Block	PRN	SV	Start of Operation	Clock	Remark
IIA	32	23	11/26/1990	RB	continuous
IIA	27	27	02/02/1992	CS	Decommission at 10/17/2012, IIR-M G49 afterwards
IIA	26	26	07/07/1992	RB	continuous
IIA	30	35	08/30/1992	RB	since 08/05/2011
IIA	9	39	06/26/1993	CS	continuous
IIA	4	34	10/26/1993	RB	continuous
IIA	6	36	03/10/1994	RB	continuous
IIA	3	33	03/28/1996	CS	continuous
IIA	10	40	07/16/1996	CS	continuous
IIR-A	13	43	07/23/1997	RB	continuous
IIA	8	38	11/06/1997	CS	continuous
IIR-A	11	46	10/07/1999	RB	continuous
IIR-A	20	51	05/11/2000	RB	continuous
IIR-A	28	44	07/16/2000	RB	continuous
IIR-A	14	41	11/10/2000	RB	continuous
IIR-A	18	54	01/30/2001	RB	continuous
IIR-A	16	56	01/29/2003	RB	continuous
IIR-A	21	45	03/31/2003	RB	continuous
IIR-B	22	47	12/21/2003	RB	continuous
IIR-B	19	59	03/20/2004	RB	continuous
IIR-B	23	60	06/23/2004	RB	continuous
IIR-B	2	61	11/06/2004	RB	continuous
IIR-M	17	53	09/26/2005	RB	continuous
IIR-M	31	52	09/25/2006	RB	continuous
IIR-M	12	58	11/17/2006	RB	continuous
IIR-M	15	55	10/17/2007	RB	continuous
IIR-M	29	57	12/20/2007	RB	continuous
IIR-M	7	28	03/15/2008	RB	continuous
IIR-M	5	50	08/17/2009	RB	continuous
IIF	25	62	05/28/2010	RB	continuous
IIF	1	63	07/16/2011	RB	continuous
	24			RB	IIR-M G49, IIA G32, IIA G37, IIF G65

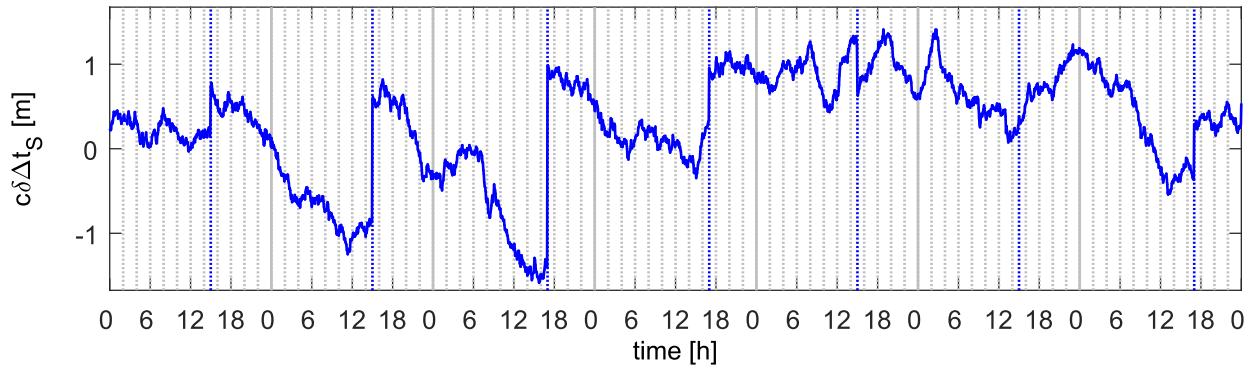


Figure 3-21: Satellite clock error of PRN 4 (IIA, RB clock), 01/01/2012 – 01/08/2012

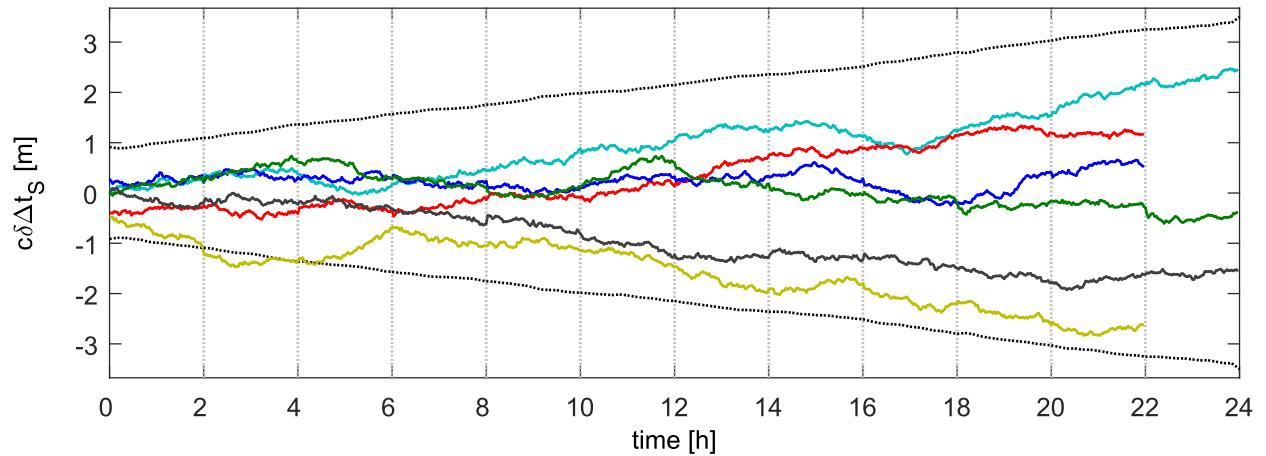


Figure 3-22: Satellite clock error between upload cutovers of PRN 4 (IIA, RB clock), 2012

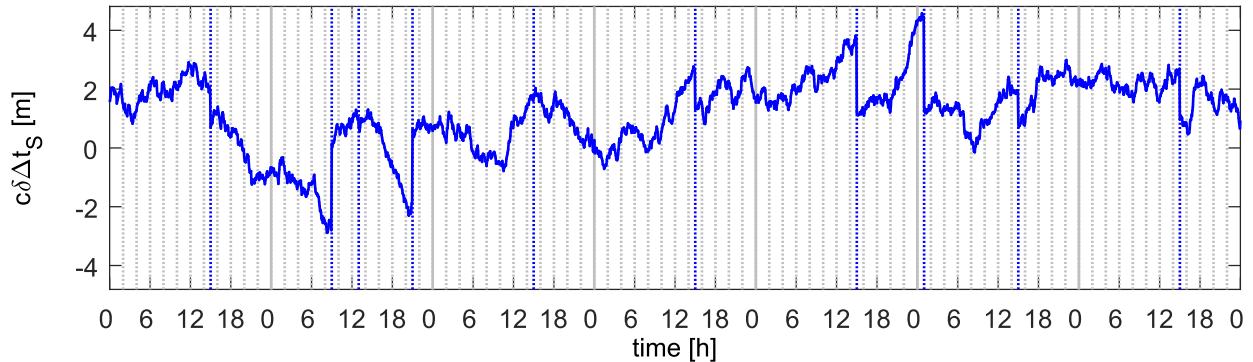


Figure 3-23: Satellite clock error of PRN 10 (IIA, CS clock), 01/01/2012 – 01/08/2012

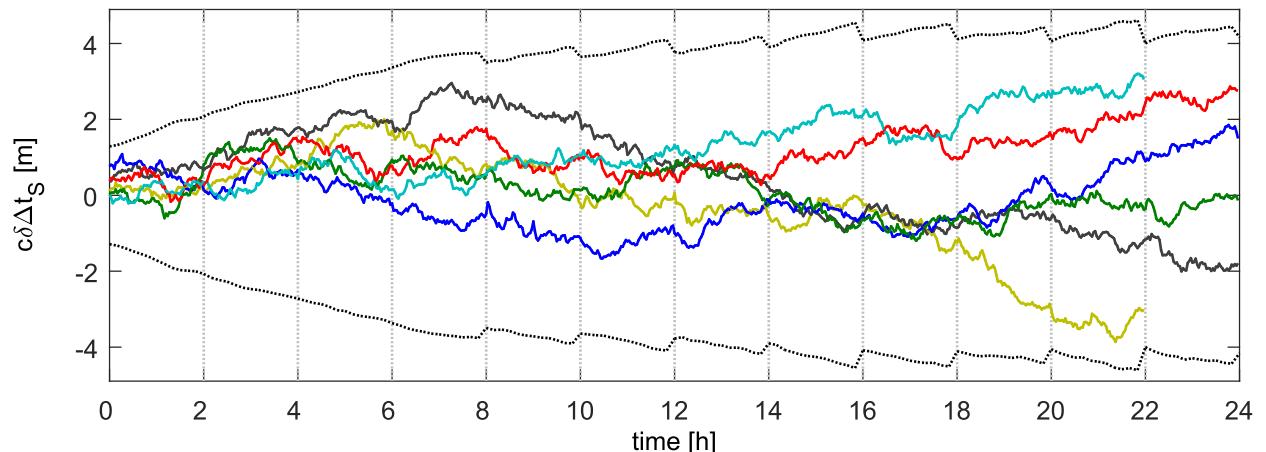


Figure 3-24: Satellite clock error between upload cutovers of PRN 10 (IIA, CS clock), 2012

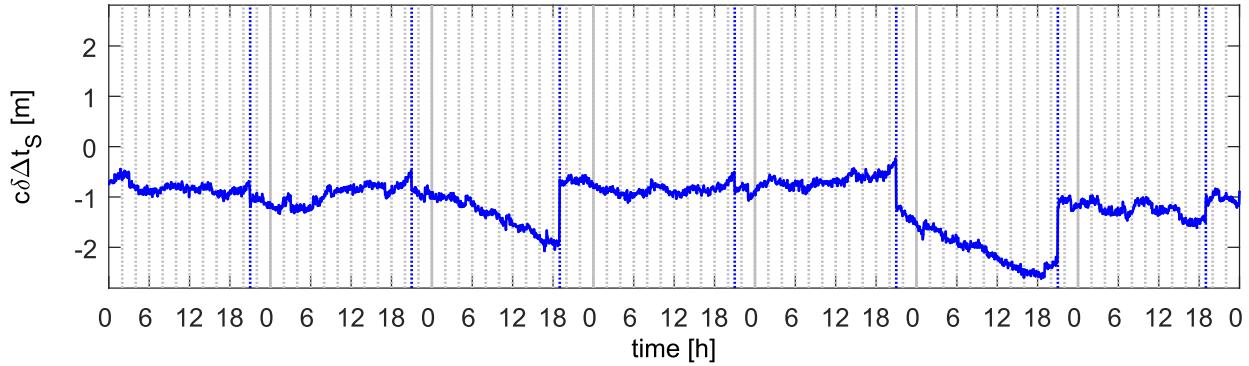


Figure 3-25: Satellite clock error of PRN 13 (IIR-A, RB clock), 01/01/2012 – 01/08/2012

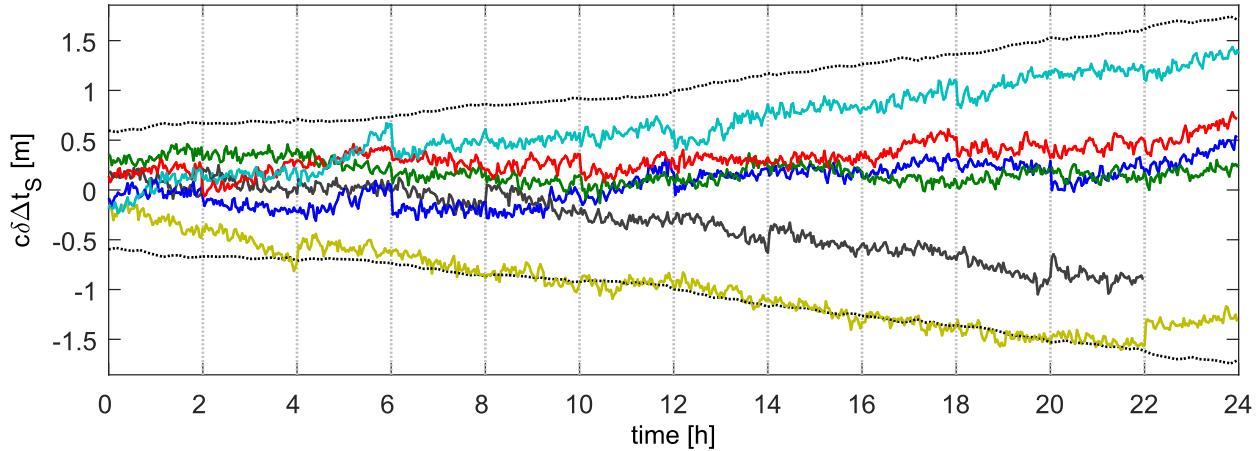


Figure 3-26: Satellite clock error between upload cutovers of PRN 13 (IIR-A, RB clock), 2012

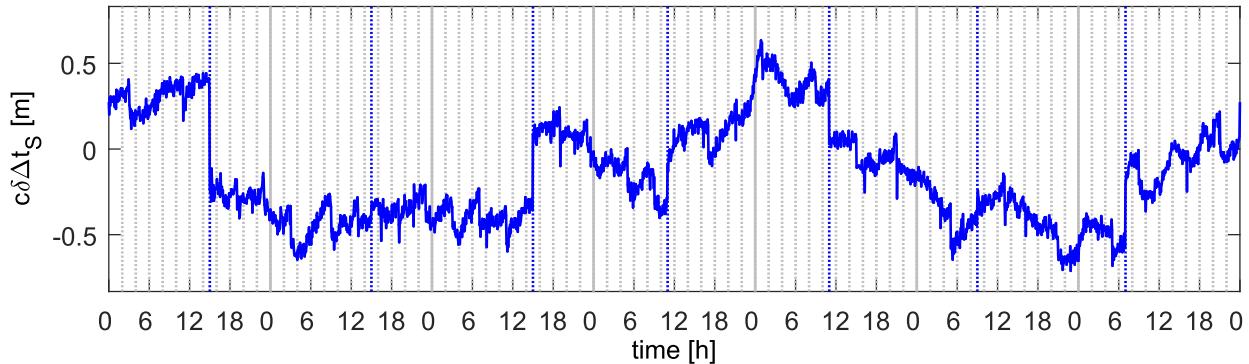


Figure 3-27: Satellite clock error of PRN 19 (IIR-B, RB clock), 01/01/2012 – 01/08/2012

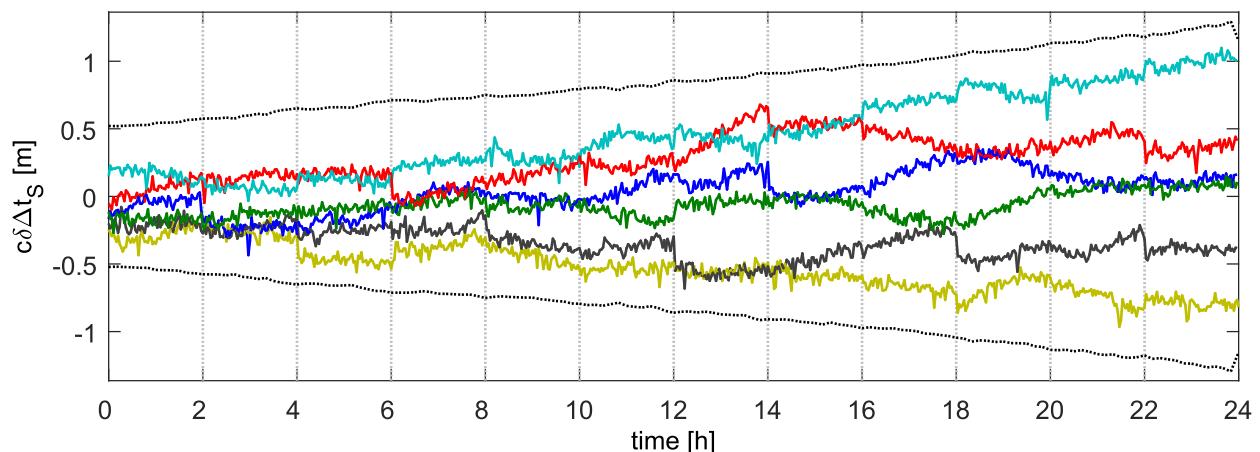


Figure 3-28: Satellite clock error between upload cutovers of PRN 19 (IIR-B, RB clock), 2012

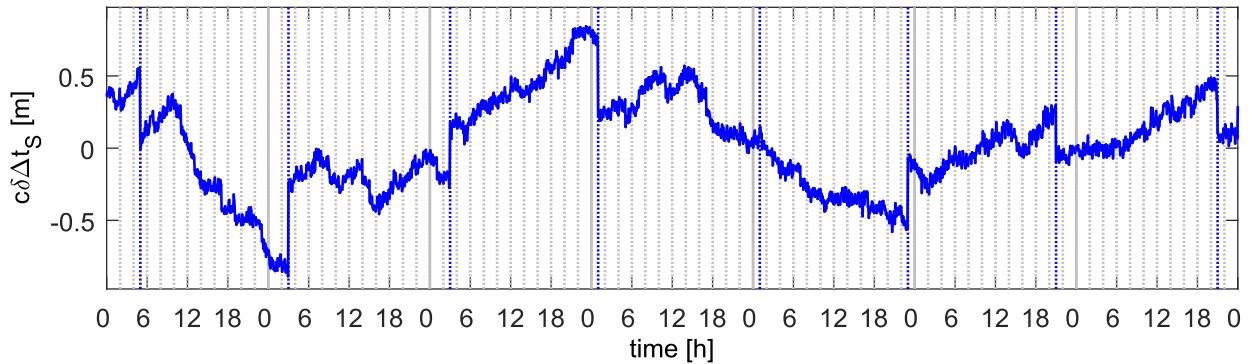


Figure 3-29: Satellite clock error of PRN 7 (IIR-M, RB clock), 01/01/2012 – 01/08/2012

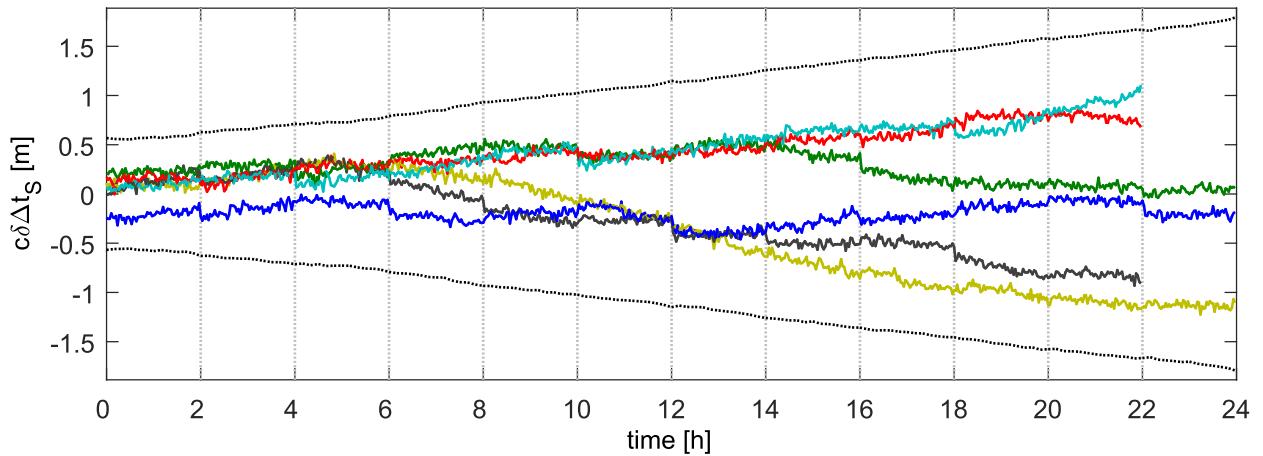


Figure 3-30: Satellite clock error between upload cutovers of PRN 7 (IIR-M, RB clock), 2012

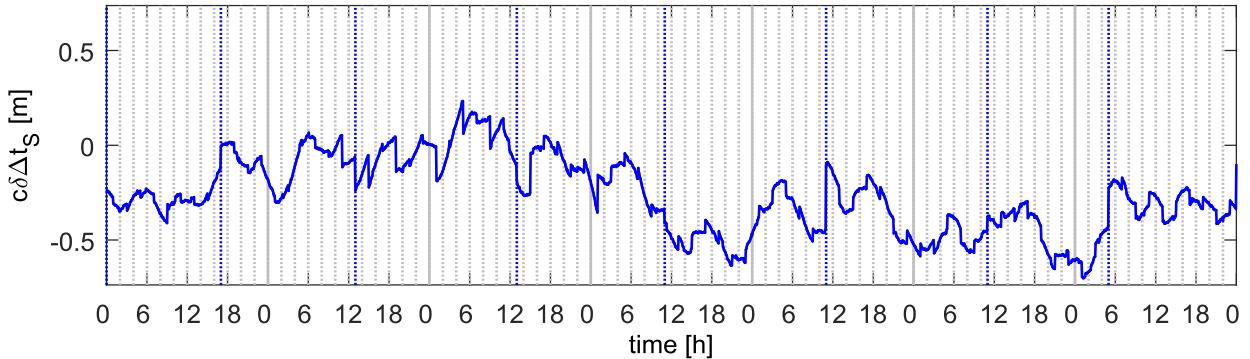


Figure 3-31: Satellite clock error of PRN 25 (IIF, RB clock), 01/01/2012 – 01/08/2012

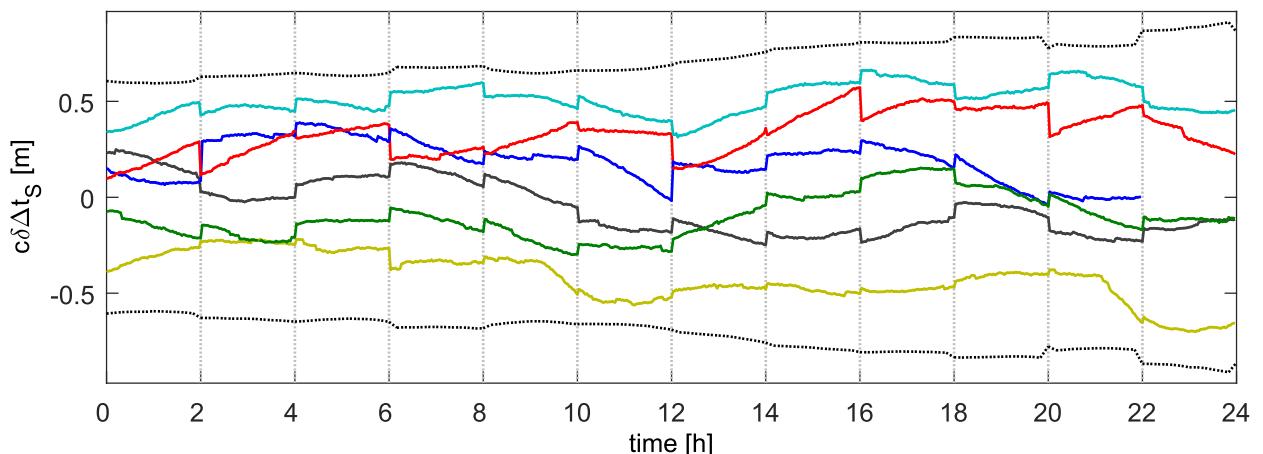


Figure 3-32: Satellite clock error between upload cutovers of PRN 25 (IIF, RB clock), 2012

3.5.1.5 Model

3.5.1.5.1 Structure

The satellite clock error Δt_s is adequately described by the discrete-time three-state model

$$\underbrace{\begin{pmatrix} \mathbf{z}_{\Delta t_s, k+1} \\ \mathbf{z}_{\Delta \ddot{t}_s, k+1} \\ \mathbf{z}_{\Delta \ddot{\ddot{t}}_s, k+1} \end{pmatrix}}_{\mathbf{z}_{\Delta t_s, k+1}} = \underbrace{\begin{pmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix}}_{\Phi_{\Delta t_s}} \underbrace{\begin{pmatrix} \mathbf{z}_{\Delta t_s, k} \\ \mathbf{z}_{\Delta \ddot{t}_s, k} \\ \mathbf{z}_{\Delta \ddot{\ddot{t}}_s, k} \end{pmatrix}}_{\mathbf{z}_{\Delta t_s, k}} + \underbrace{\begin{pmatrix} T & \frac{1}{2}T^2 \\ 0 & T \\ 0 & 0 \end{pmatrix}}_{\Gamma_{\Delta t_s}} \underbrace{\begin{pmatrix} \eta_{WF, k} + \omega_{FF, k} \\ \eta_{RWF, k} \\ \boldsymbol{\omega}_{\Delta t_s, k} \end{pmatrix}}_{\boldsymbol{\omega}_{\Delta t_s, k}} \quad (3.102)$$

$$\Delta t_{s,k} = \underbrace{(1 \ 0 \ 0)}_{\mathbf{H}_{\Delta t_s}} \mathbf{z}_{\Delta t_s, k} + \underbrace{\eta_{WP, k} + \omega_{FP, k}}_{\mathbf{v}_{\Delta t_s, k}} \quad (3.103)$$

with the clock bias state $\mathbf{z}_{\Delta t_s}$, clock drift state $\mathbf{z}_{\Delta \ddot{t}_s}$, clock drift rate state $\mathbf{z}_{\Delta \ddot{\ddot{t}}_s}$ and the clock sample time T . η_{WP} is the white phase noise, η_{WF} is the white frequency noise, η_{RWF} is the random walk frequency noise, ω_{FP} is the flicker phase noise and ω_{FF} is the flicker frequency noise of the satellite clock as depicted in section 3.1.4.3.

The initial state $\mathbf{z}_{\Delta t_s, 0}$ is composed of the clock bias a_{f0} , drift a_{f1} and drift rate a_{f2} , valid at the clock reference time t_{oc}

$$\mathbf{z}_{\Delta t_s, 0} = \begin{pmatrix} a_{f0} + a_{f1}(t_0 - t_{oc}) + a_{f2}(t_0 - t_{oc})^2 \\ a_{f1} + 2a_{f2}(t_0 - t_{oc}) \\ 2a_{f2} \end{pmatrix} \quad (3.104)$$

In order to fit into the framework of the navigation *error* filter, the error model of the satellite clock error is required since the pseudoranges have already been corrected by the broadcast satellite clock error (3.99). Only the difference between the true (unknown) and broadcast satellite clock error $\delta \Delta t_s = \Delta t_s - \tilde{\Delta t}_s$ has to be considered in the navigation error filter. For that, the satellite clock error model (3.102) and (3.103) is formally perturbed with $\mathbf{z}_{\Delta t_s} = \tilde{\mathbf{z}}_{\Delta t_s} + \delta \mathbf{z}_{\Delta t_s}$ and $\Delta t_s = \tilde{\Delta t}_s + \delta \Delta t_s$. The error model is given by

$$\begin{aligned} \delta \mathbf{z}_{\Delta t_s, k+1} &= \Phi_{\Delta t_s} \delta \mathbf{z}_{\Delta t_s, k} + \Gamma_{\Delta t_s} \boldsymbol{\omega}_{\Delta t_s, k} \\ \delta \Delta t_{s,k} &= \mathbf{H}_{\Delta t_s} \delta \mathbf{z}_{\Delta t_s, k} + \mathbf{v}_{\Delta t_s, k} \end{aligned} \quad (3.105)$$

with the initial state error

$$\delta \mathbf{z}_{\Delta t_s, 0} = \begin{pmatrix} \delta a_{f0} + \delta a_{f1}(t_0 - t_{oc}) + \delta a_{f2}(t_0 - t_{oc})^2 \\ \delta a_{f1} + 2\delta a_{f2}(t_0 - t_{oc}) \\ 2\delta a_{f2} \end{pmatrix} \quad (3.106)$$

The satellite clock error is thus affected on the one hand by the clock correction coefficient errors δa_{f0} , δa_{f1} and δa_{f2} and on the other hand by the noises $\boldsymbol{\omega}_{\Delta t_s}$ and $\mathbf{v}_{\Delta t_s}$.

3.5.1.5.2 Curve Fitting

Neglecting the noise and considering only the correction coefficient errors in (3.105), the satellite clock error $\delta\Delta t_s$ is described by the quadratic polynomial

$$\delta\Delta t_s = \delta a_{f0} + \delta a_{f1}(t - t_{oc}) + \delta a_{f2}(t - t_{oc})^2 = \begin{pmatrix} 1 & t - t_{oc} & (t - t_{oc})^2 \end{pmatrix} \begin{pmatrix} \delta a_{f0} \\ \delta a_{f1} \\ \delta a_{f2} \end{pmatrix} \quad (3.107)$$

In the following, this quadratic polynomial is fit to each of the dataset cutover windows of the available data. The clock correction coefficient errors δa_{f0} , δa_{f1} and δa_{f2} are determined by least-squares estimation for each window. For that, the quadratic equations of all m samples within one window are stacked

$$\underbrace{\begin{pmatrix} \delta\Delta t_{s,1} \\ \vdots \\ \delta\Delta t_{s,m} \end{pmatrix}}_{\delta\Delta t_s} = \underbrace{\begin{pmatrix} 1 & t_1 - t_{oc,i} & (t_1 - t_{oc,i})^2 \\ \vdots & \vdots & \vdots \\ 1 & t_m - t_{oc,i} & (t_m - t_{oc,i})^2 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} \delta a_{f0} \\ \delta a_{f1} \\ \delta a_{f2} \end{pmatrix}}_{\delta\hat{\mathbf{a}}_f} \quad (3.108)$$

The equation system is solved for the polynomial coefficients

$$\delta\hat{\mathbf{a}}_f = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \delta\Delta t_s \quad (3.109)$$

In Figure 3-33 and Figure 3-36 the estimated clock correction coefficient errors $\delta\hat{a}_{f0}$, $\delta\hat{a}_{f1}$ and $\delta\hat{a}_{f2}$ are shown for the upload cutover windows of PRN 4 and PRN 25 in the year 2012. The satellite clock errors that are calculated according to (3.107) from the determined coefficient errors are illustrated in Figure 3-34 and Figure 3-37. In the upper plots, the clock bias at the time of the upload cutover is shown for the year 2012.

By looking at Figure 3-34, it can be stated that the clock drift correction coefficient error $\delta\hat{a}_{f1}$ is correlated with the clock bias correction coefficient error $\delta\hat{a}_{f0}$. In fact, the dataset cutover corrections for IIA, IIR-A, IIR-B and IIR-M satellites are normally small. This means that the clock drift is more or less equal to the difference of two subsequent clock biases $\delta\hat{a}_{f0}$ divided by the dataset cutover timespan. For IIF satellites, the dataset cutover corrections are in the range of the upload cutover corrections. This means that the clock drift correction coefficient error δa_{f1} is less correlated with δa_{f0} , which is confirmed by Figure 3-37. The quadratic clock drift correction coefficient error δa_{f2} is small and can be neglected.

In order to check the suitability of the assumed model structure and the accuracy of the estimated coefficients, the residuals of the clock errors are calculated by

$$r_{\delta\Delta t_s} = \delta\Delta t_s - \delta\hat{t}_s \quad (3.110)$$

and are plotted in Figure 3-35 and Figure 3-38 for the first week of 2012. It can be observed that the 3σ bounds of the residuals lie below 0.15 m.

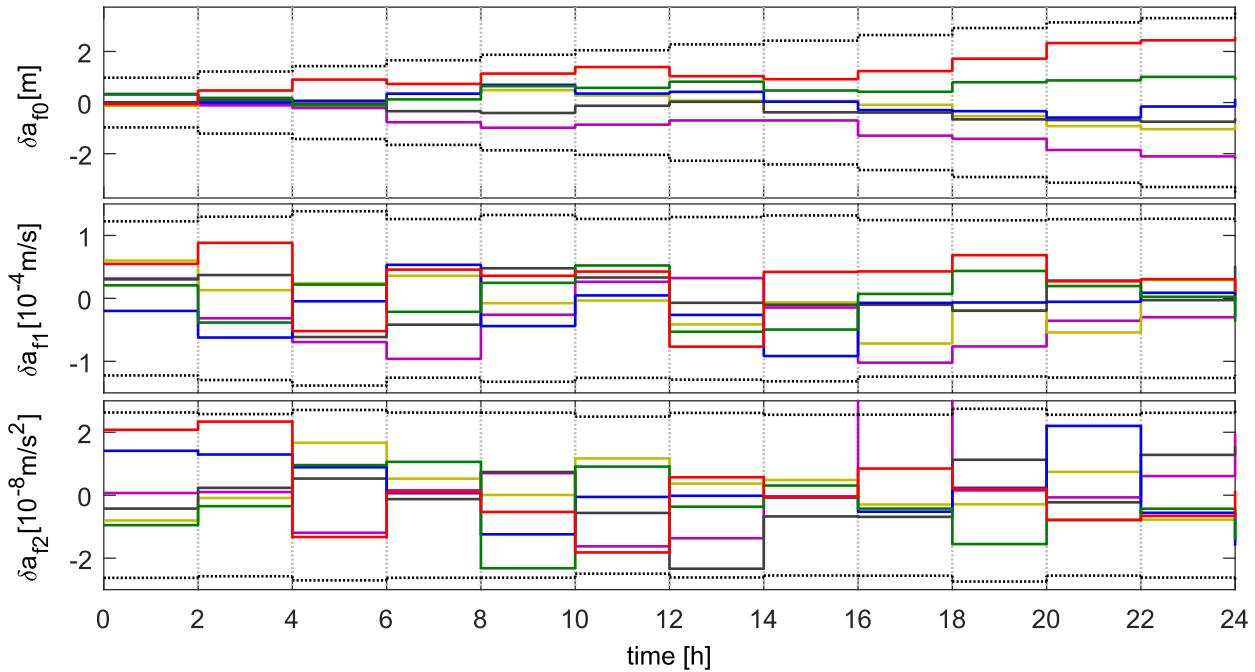


Figure 3-33: Fitted satellite clock error coefficients of PRN 4, 01/01/2012 – 12/31/2012

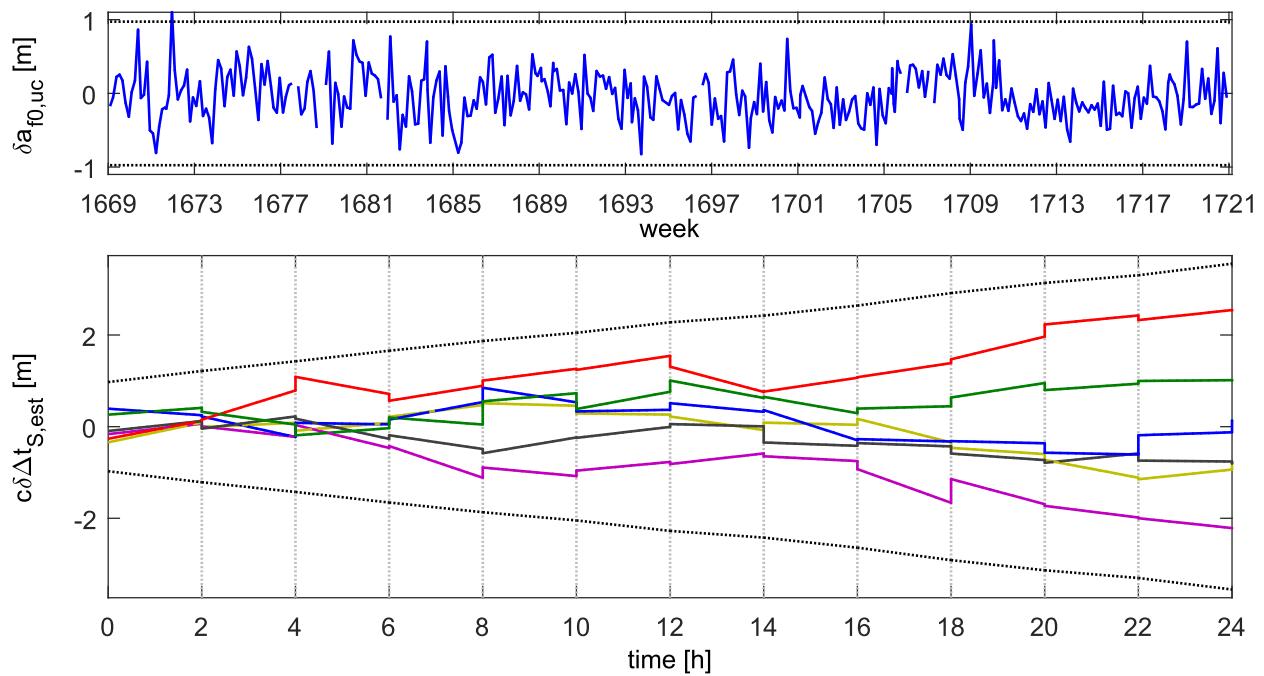


Figure 3-34: Fitted satellite clock error of PRN 4, 01/01/2012 – 12/31/2012

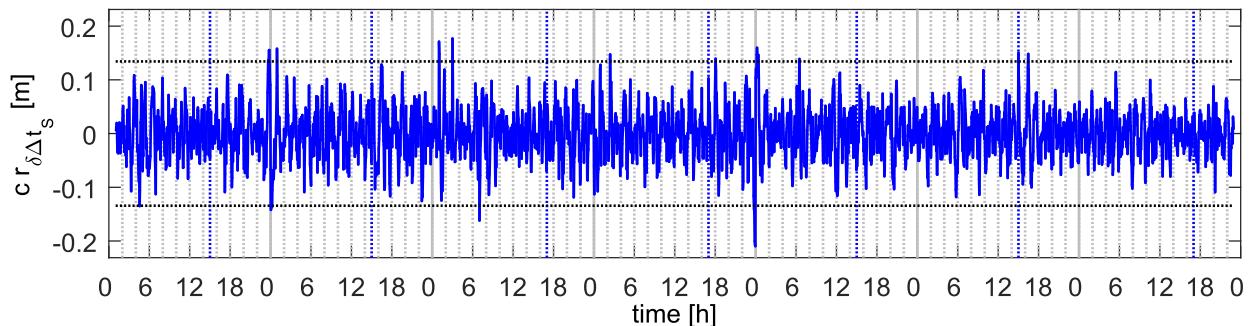


Figure 3-35: Residual satellite clock error of PRN 4, 01/01/2012 – 01/08/2012

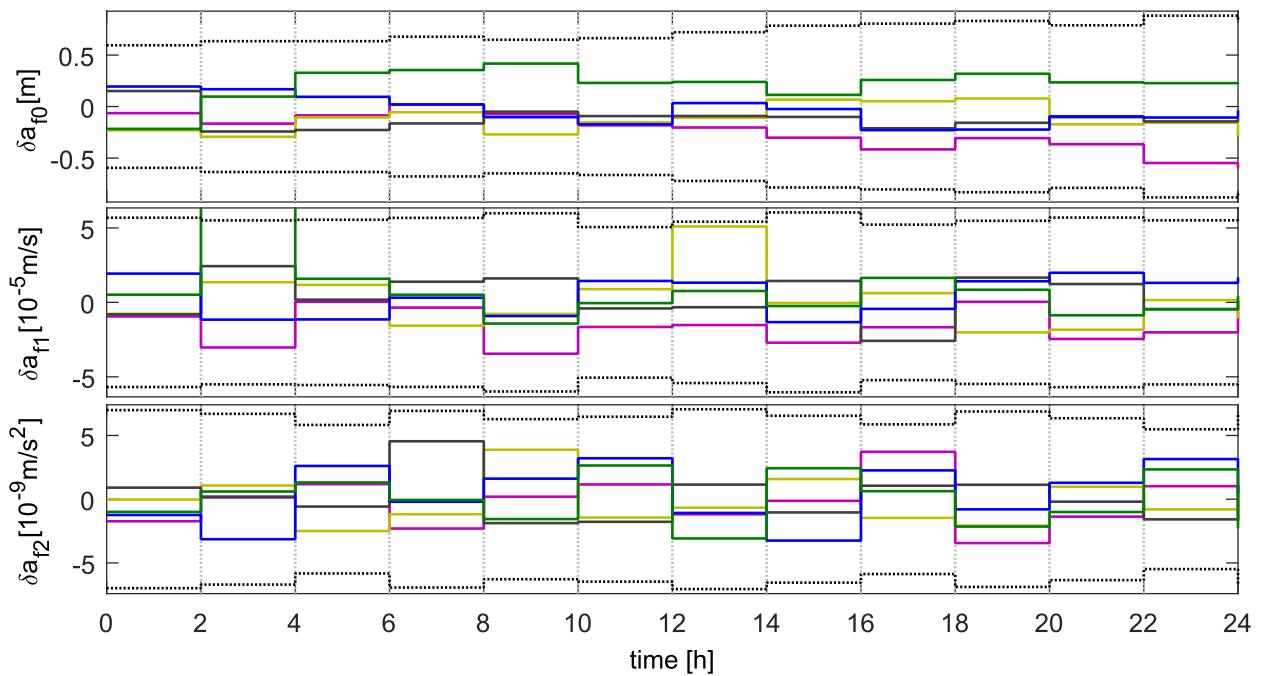


Figure 3-36: Fitted satellite clock error coefficients of PRN 25, 01/01/2012 – 12/31/2012

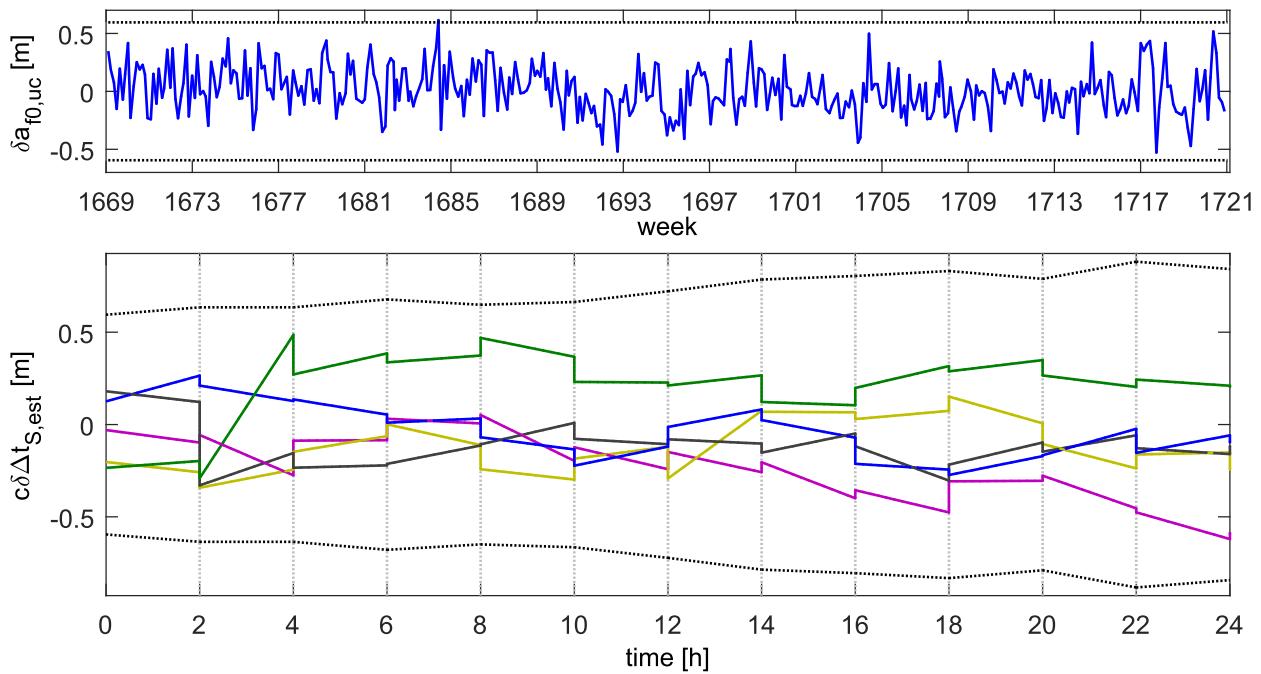


Figure 3-37: Curve fitted satellite clock error of PRN 25, 01/01/2012 – 12/31/2012

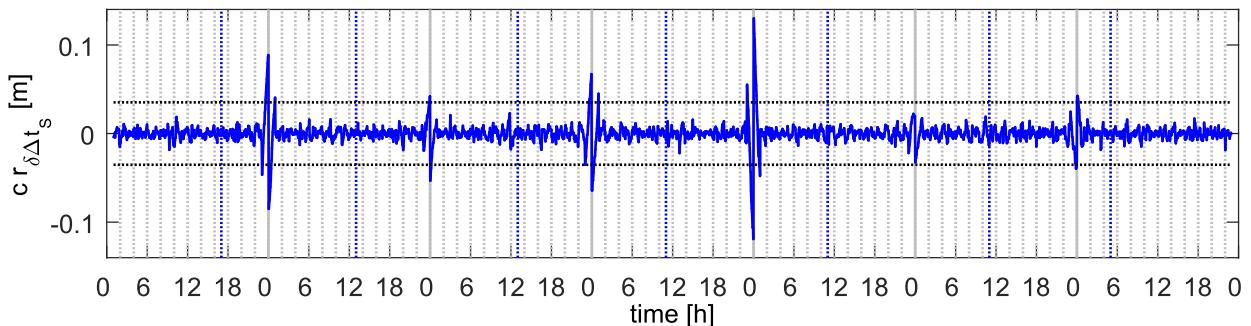


Figure 3-38: Residual satellite clock error of PRN 25, 01/01/2012 – 01/08/2012

3.5.1.5.3 Correction Coefficient Error

The non-stationary behavior of the correction coefficient errors δa_{f_0} and δa_{f_1} between two subsequent upload cutovers is best described by an unsteady or even slightly unstable multi-input multi-output (MIMO) model that is able to represent the correlation between both parameters. The general model structure for a two dimensional MIMO model is given with

$$\begin{aligned} \mathbf{z}_{dc,i+1} &= \Phi_{dc} \mathbf{z}_{dc,i} + \Gamma_{dc} \mathbf{n}_{dc,i} \\ \begin{pmatrix} \delta a_{f_0,i} \\ \delta a_{f_1,i} \end{pmatrix} &= \mathbf{H}_{dc} \mathbf{z}_{dc,i} + \mathbf{D}_{dc} \mathbf{n}_{dc,i} \end{aligned} \quad (3.111)$$

The input noise is two dimensional, white and Gaussian distributed, $\mathbf{n}_{dc} \sim WN(\mathbf{0}, \mathbf{Q}_{dc})$. δa_{f_0} and δa_{f_1} are both coupled with the common state vector \mathbf{z}_{dc} via the fully occupied matrix \mathbf{H}_{dc} . The model is newly initialized after each upload cutover. The upload cutover time window is generally about 24 hours. For most applications with no or only one upload cutover during runtime, it is sufficient to assume that the clock correction coefficient errors are not correlated between these single upload cutovers. In these cases, the correlation between subsequent upload cutovers does not have to be considered and only the variance of the initial clock bias is of interest. Between two succeeding dataset cutovers, the correction coefficient errors δa_{f_0} and δa_{f_1} are constant. Propagation takes only place at the instants of dataset cutovers

$$\mathbf{z}_{dc,k} = \begin{cases} \Phi_{dc} \mathbf{z}_{dc,i-1} + \Gamma_{dc} \mathbf{n}_{dc,i-1} & i^{\text{th}} \text{ dataset cutover at } t_k = t_{oc,i} \\ \mathbf{z}_{dc,k-1} & \text{else} \end{cases} \quad (3.112)$$

3.5.1.5.4 Noise

The modified overlapping Allan deviation is computed from the residuals (3.110) with (3.12) in section 3.1.3.1 in order to characterize the white phase noise, flicker phase noise, white frequency noise, flicker frequency noise and random walk frequency noise. Figure 3-39 shows the modified Allan deviation of the residuals of the year 2012. The blue lines belong to block IIA satellites with RB clock, the red lines to block IIA satellites with CS clock, the green lines to IIR-A and IIR-B satellites with RB clock, the cyan lines to IIR-M satellites with RB clock and the magenta lines to the new IIF satellites with RB clock activated. The grey dotted lines with slopes $-3/2$, -1 and $-1/2$ serve as reference for the white phase noise, flicker phase noise and white frequency noise. The modified Allan deviation is advantageous compared to the standard Allan deviation because it is capable of resolving white phase noise and flicker phase noise. It can be observed that satellites of the same block and clock types have similar characteristics. The IIA satellites with CS and RB clocks both exhibit white frequency noise for averaging times up to 1000 seconds. However, as already stated in [56], the IIA satellites with RB clock are more stable than the IIA satellites with CS clock in the short term and additionally feature white phase and flicker phase noise for time intervals up to 100 seconds. The Allan deviations of the IIR-A, IIR-B and IIR-M satellite residuals are more or less identical. They exhibit white frequency noise for short averaging times and fade to flicker phase noise for larger averaging times. The clocks of the IIF satellites are most stable which is also proven by the Allan deviation. Their characteristics are similar to those of the older IIA satellites with RB clock. Flicker frequency and random walk frequency noise do not occur.

The satellite clock residuals consist of white phase, flicker phase and white frequency noise components. An obvious way to capture the noise characteristic quantitatively is to fit the sum of

power law curves with exponents $-3/2$, -1 and $-1/2$ to the modified Allan deviation curves as has been done for example at the identification of the IMU noise and to estimate the magnitudes of the three noise components. In this case, however, it turned out that the method does not yield satisfying results. A better approach is to substitute the white phase, flicker noise and white frequency noise by an overall colored noise that directly acts on the satellite clock error (3.105) according to

$$\delta\Delta t_{S,k} = \mathbf{H}_{\Delta t_S} \delta\mathbf{z}_{\Delta t_S,k} + \omega_{\Delta t_S,k} \quad (3.113)$$

The substitute noise $\omega_{\Delta t_S}$ in turn can be represented by an ARMA model

$$\begin{aligned} \mathbf{z}_{\omega_{\Delta t_S},k+1} &= \Phi_{\omega_{\Delta t_S}} \mathbf{z}_{\omega_{\Delta t_S},k} + \Gamma_{\omega_{\Delta t_S}} \eta_{\omega_{\Delta t_S},k} \\ \omega_{\Delta t_S,k} &= \mathbf{H}_{\omega_{\Delta t_S}} \mathbf{z}_{\omega_{\Delta t_S},k} + D_{\omega_{\Delta t_S}} \eta_{\omega_{\Delta t_S},k} \end{aligned} \quad (3.114)$$

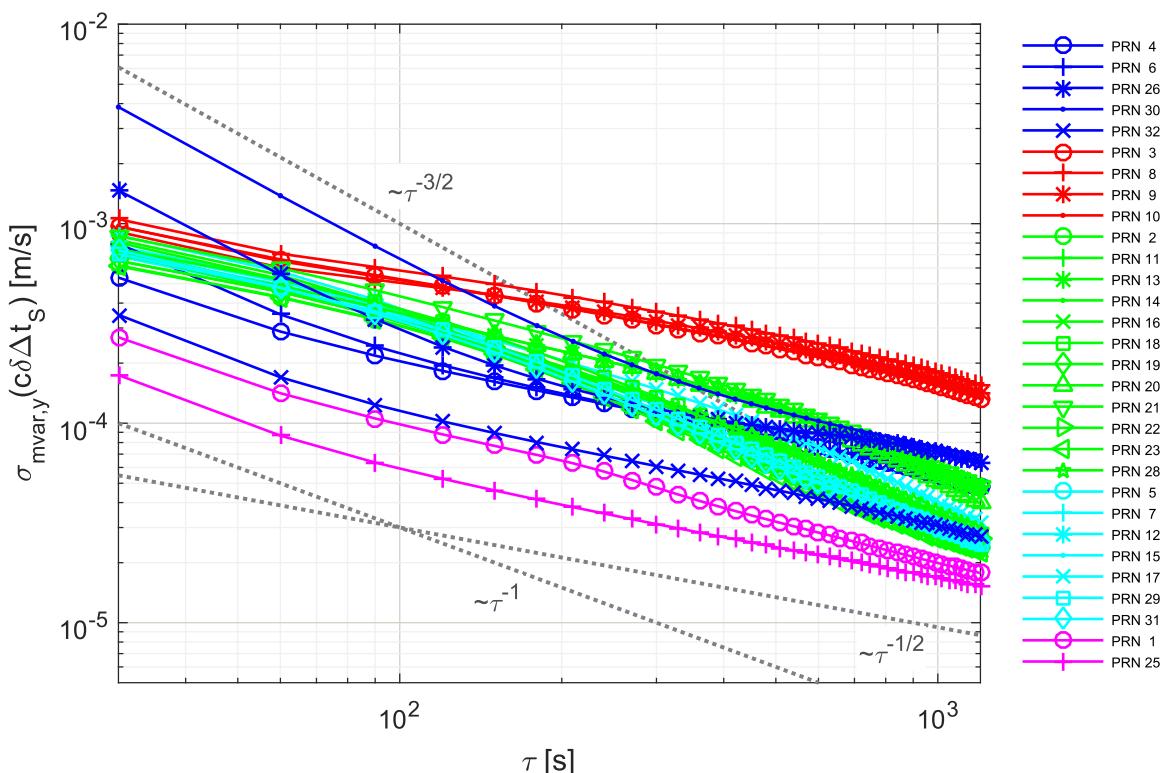


Figure 3-39: Modified Allan deviation of the residual clock error, 2012

3.5.1.6 Parameter Estimation

3.5.1.6.1 Correction Coefficient Error

The parameters of the matrices of the MIMO model (3.111) are estimated with the prediction error method. MATLAB®'s pem function can be used to identify multi-input multi-output models. The prediction error method applies a numerical optimization algorithm that adapts the free model parameters such that the output error is minimized. The algorithm of the pem function is capable of handling purely stochastic processes. The function provides different search methods for the minimum: the subspace Gauss-Newton method, an adaptive version of the Gauss-Newton approach, the Levenberg-Marquardt method, a non-linear least-squares solver and the steepest decent gradient search method [60]. The function has been setup to select the best search method automatically. The prediction error method requires an initial solution. It is obtained using the subspace method

(n4sid). The pem function outputs the model in observable canonical form. Table 3-18 lists the estimated entries of the system matrices Φ_{dc} and Table 3-19 the entries of the input matrices Γ_{dc} . The elements of the input noise covariance matrices \mathbf{Q}_{dc} as well as the initial covariance matrices $\mathbf{P}_{dc,0}$ are listed in Table 3-20. The output matrix \mathbf{H}_{dc} is a unit matrix. The feed-through matrix \mathbf{D}_{dc} is zero. The eigenvalues of the system matrices are plotted in Figure 3-40. All eigenvalues are real and between 0.59 and 1.04. The second eigenvalues are close to 1, indicating asymptotically stable or slightly unstable models.

Figure 3-41 shows the simulated clock error of PRN 4 between subsequent upload cutovers, Figure 3-42 of PRN 10, Figure 3-43 of PRN 13, Figure 3-44 of PRN 19, Figure 3-45 of PRN 7 and Figure 3-46 of PRN 25. The satellite clock error has been computed by means of the MIMO model (3.111) together with the model matrices and input noise variances from Table 3-18 to Table 3-20. The black dotted lines illustrate the 3σ boundaries that are predicted by the model whereas the grey dotted lines are the actual 3σ standard deviations that have been computed from the data of 2012. The simulated satellite clock errors agree well with the fitted satellite clock errors (compare with Figure 3-34 and Figure 3-37).

Table 3-18: Estimated entries of the system matrix Φ_{dc}

PRN	$\Phi_{dc,11}$	$\Phi_{dc,12}$	$\Phi_{dc,21}$	$\Phi_{dc,22}$
4	9.99624869e-01	1.10325845e+03	-8.20928642e-07	9.06469102e-01
10	7.35035731e-01	-1.34601979e+04	-2.16982982e-06	7.99964106e-01
13	1.00751789e+00	-8.32365448e+02	3.45180782e-06	6.18607406e-01
19	1.00119725e+00	-6.19370534e+02	2.53421074e-07	8.69268104e-01
7	1.02434763e+00	-1.85962490e+03	-1.27960123e-07	1.00881482e+00
25	8.00088448e-01	-2.80335083e+03	2.35903798e-06	1.05003617e+00

Table 3-19: Estimated entries of the input matrix Γ_{dc}

PRN	$\Gamma_{dc,11}$	$\Gamma_{dc,12}$	$\Gamma_{dc,21}$	$\Gamma_{dc,22}$
4	5.23215084e-01	5.28052221e+03	8.29343619e-06	-1.99363523e-02
10	3.37303781e-01	4.90796843e+03	1.99761268e-06	-7.28177104e-02
13	7.98120569e-01	3.25956446e+03	-3.64854388e-05	2.74081669e-01
19	7.53242344e-01	3.04808975e+03	-4.14710245e-05	2.29531441e-01
7	6.07008971e-01	4.37829725e+03	1.71074691e-06	2.86291107e-02
25	4.62756103e-01	4.06690103e+03	-8.13194823e-06	-5.82571478e-02

Table 3-20: Estimated entries of the noise covariance matrix \mathbf{Q}_{dc} and initial state covariance $\mathbf{P}_{dc,0}$

PRN	$Q_{dc,11}$	$Q_{dc,12}$	$Q_{dc,22}$	$P_{dc,0,11}$	$P_{dc,0,12}$	$P_{dc,0,22}$
4	4.52040417e-02	6.96540178e-06	1.92739998e-09	9.92496141e-02	3.19642811e-06	1.62058562e-09
10	2.18254807e-01	3.11568879e-05	1.01772168e-08	2.78289922e-01	1.72878402e-05	7.83586184e-09
13	1.06008692e-02	1.12416848e-06	4.43953450e-10	3.89503074e-02	7.61451551e-07	4.39397604e-10
19	7.45240032e-03	7.60324529e-07	3.01156849e-10	3.09670337e-02	5.72349259e-07	2.56833673e-10
7	8.08240916e-03	1.12727448e-06	3.75182852e-10	2.80125803e-02	-7.00323256e-07	0.00000000e+00
25	7.56951025e-03	9.43954534e-07	3.19884591e-10	3.84103732e-02	-3.21865673e-08	3.84624722e-10

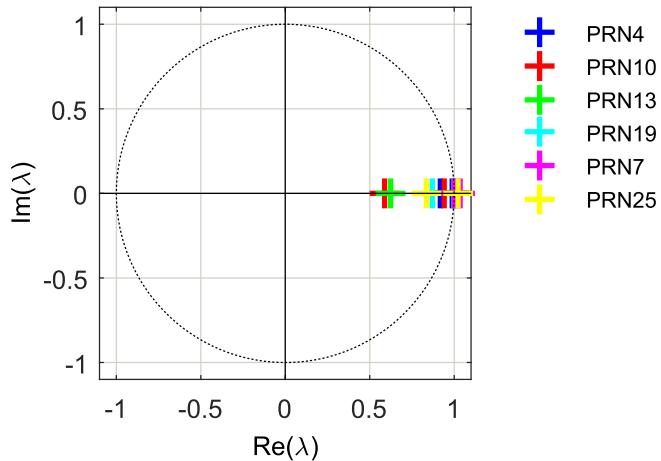
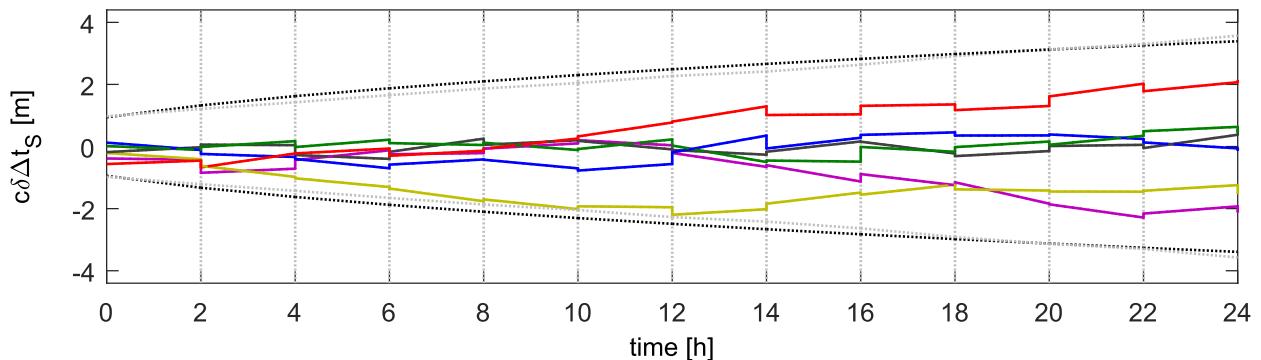
Figure 3-40: Eigenvalues of the system matrix Φ_{dc} 

Figure 3-41: Simulated satellite clock error of PRN 4, 01/01/2012 – 12/31/2012

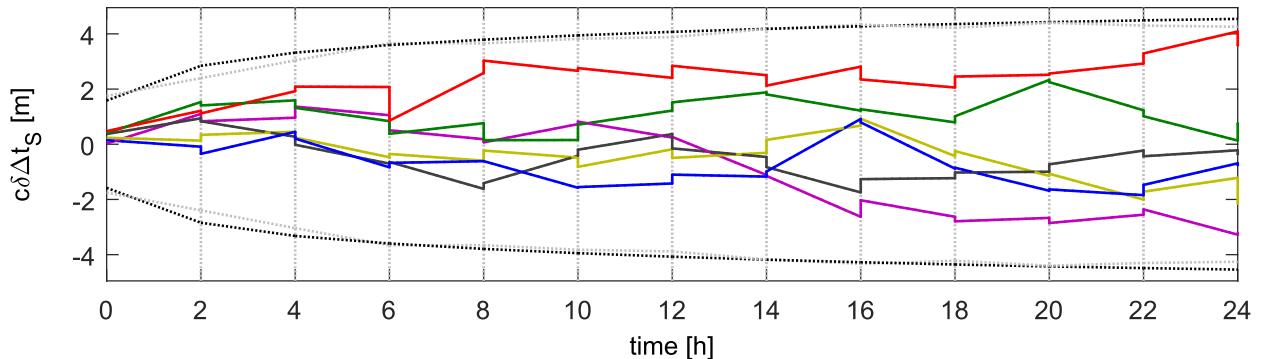


Figure 3-42: Simulated satellite clock error of PRN 10, 01/01/2012 – 12/31/2012

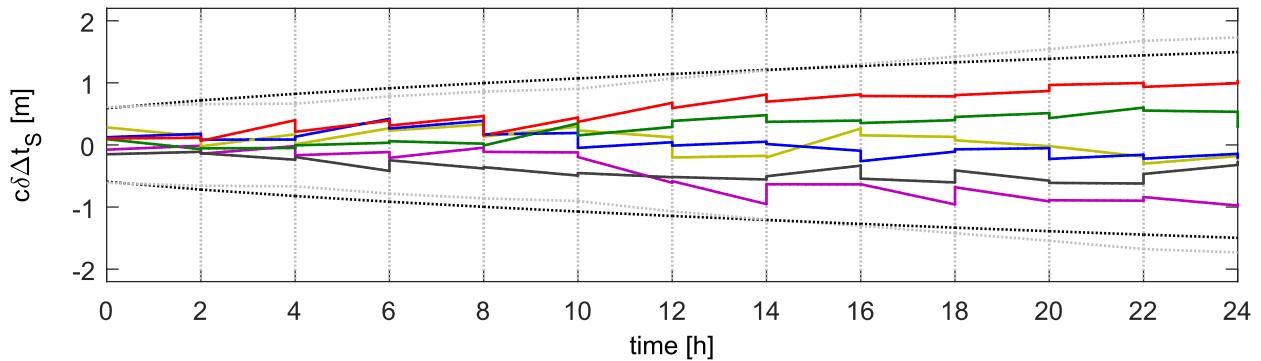


Figure 3-43: Simulated satellite clock error of PRN 13, 01/01/2012 – 12/31/2012

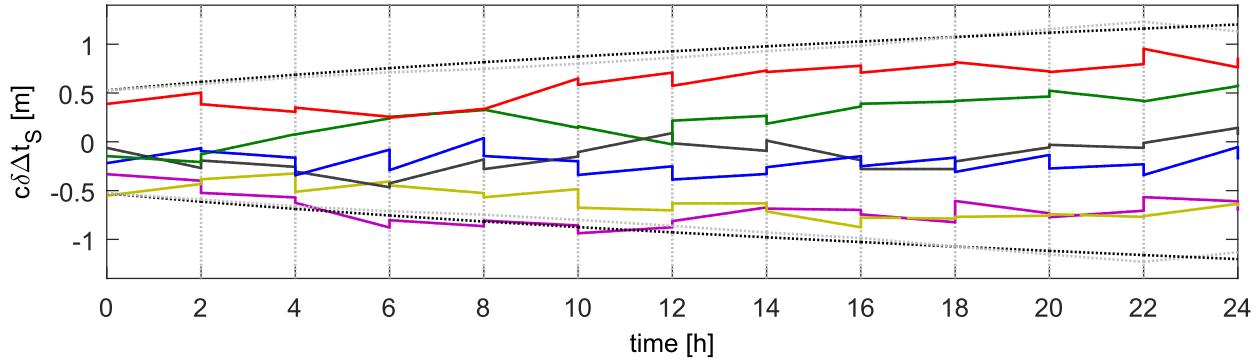


Figure 3-44: Simulated satellite clock error of PRN 19, 01/01/2012 – 12/31/2012

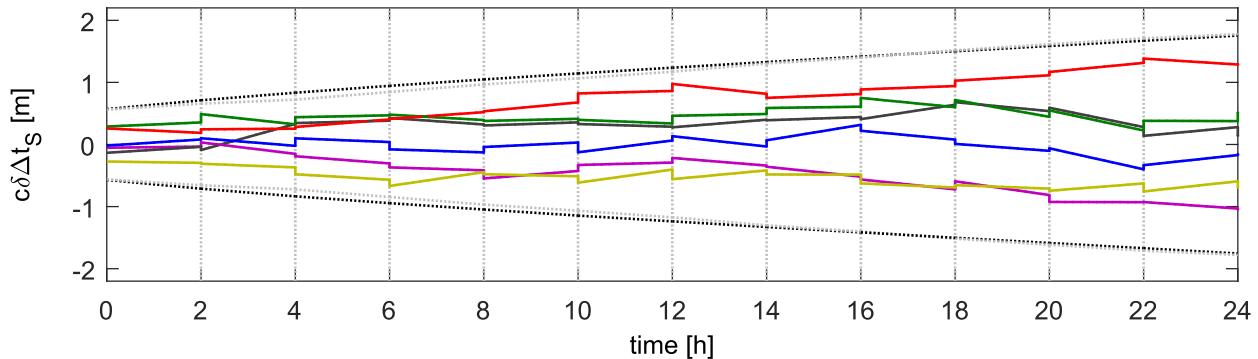


Figure 3-45: Simulated satellite clock error of PRN 7, 01/01/2012 – 12/31/2012

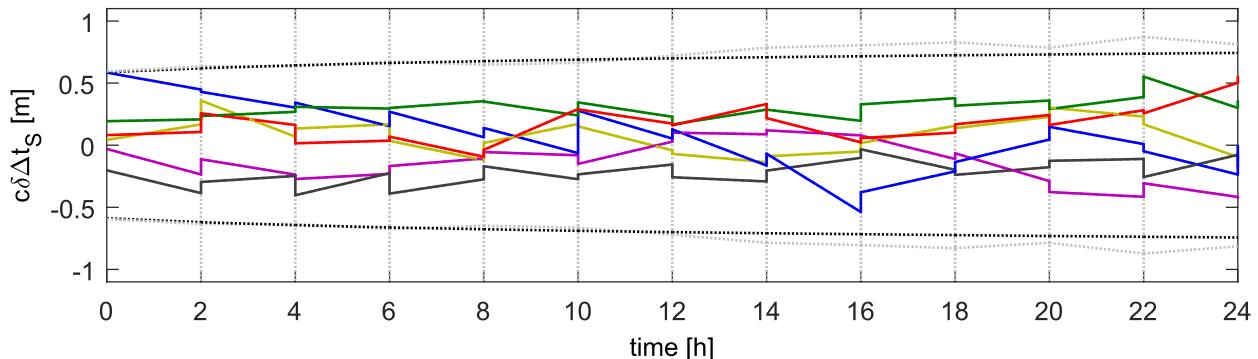


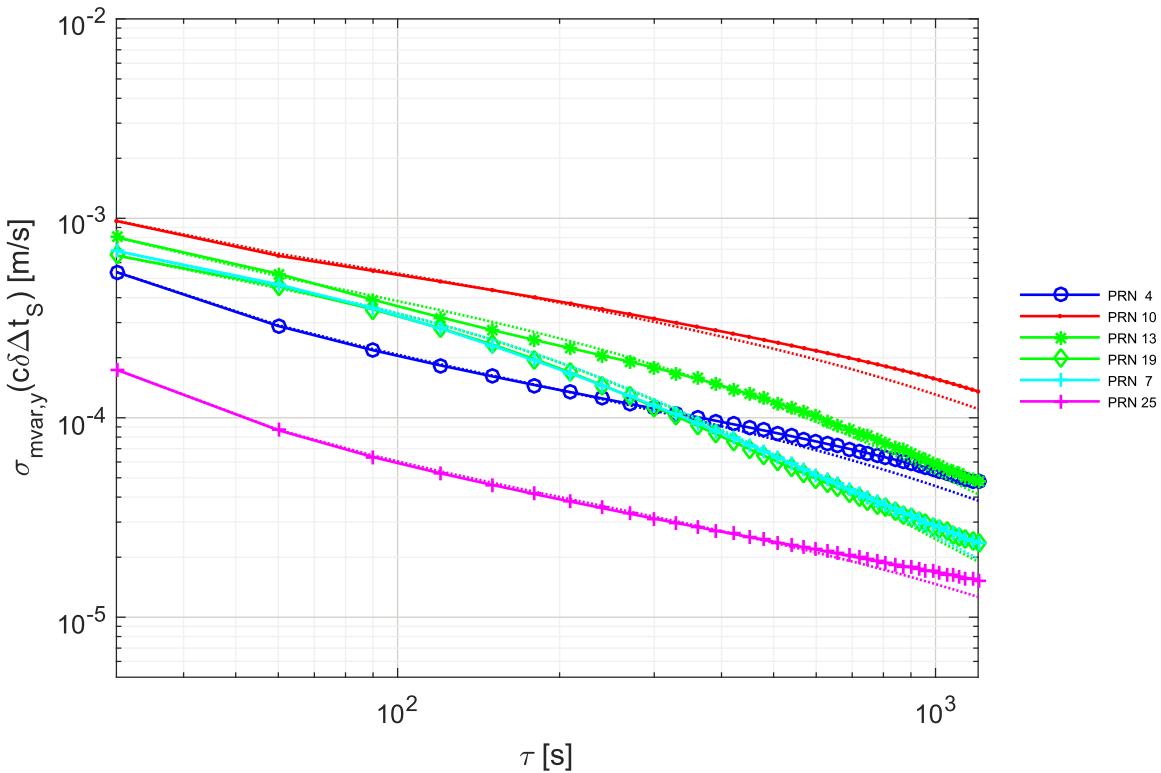
Figure 3-46: Simulated satellite clock error of PRN 25, 01/01/2012 – 12/31/2012

3.5.1.6.2 Noise

The coefficients of the ARMA model (3.114) can be estimated for example with Burg's method (section 3.1.5.4). An ARMA(2,0) model is already sufficient to represent the main character of the satellite clock noise. Table 3-21 lists the estimated ARMA(2,0) model coefficients for the six selected satellites in the year 2012. The sample time of the model is $T = 30\text{ s}$. This means that the propagation of the model in the navigation error filter has only to be done every 30 seconds. If another sample time is required, the model has to be resampled accordingly. Figure 3-47 shows the modified Allan deviations of the clock residuals of the year 2012 and the corresponding modified Allan deviations of the estimated ARMA models (dotted lines). It can be observed that there is a good coincidence between true and estimated deviations.

Table 3-21: Estimated ARMA(2,0) coefficients of the $\omega_{\Delta t_s}$ model

PRN	a_1	a_2	b_0	$Q_{\eta_{\omega_{\Delta t_s}}}$
4	9.11883305e-01	3.24955273e-02	1	2.38435549e-04
10	1.19301690e+00	-2.31739854e-01	1	1.00486936e-03
13	1.03078715e+00	-1.45725295e-01	1	5.92959072e-04
19	1.01087955e+00	-2.31270384e-01	1	3.75586419e-04
7	9.79313042e-01	-2.00524655e-01	1	4.06296325e-04
25	8.41701870e-01	1.15105091e-01	1	2.31494163e-05

**Figure 3-47: Modified Allan deviations of the estimated noise models**

3.5.1.7 Alternative Sources of Correction Coefficients

CNAV message

In order to increase the SPS accuracy, new CNAV messages have been introduced with the new L1C, L2C and L5 signals [52]. The CNAV navigation messages contain more and particularly higher accurate navigation data than the LNAV message on L1 and L2. The CNAV data is contemporarily organized in message types. The clock correction parameters \tilde{a}_{f0} , \tilde{a}_{f1} and \tilde{a}_{f2} that are also contained in the conventional LNAV navigation message can be found in message type 30. Additionally, differential correction parameters δa_{f0} and δa_{f1} for the satellite clock bias and drift are broadcast in messages type 13 and 14, respectively. The expanded satellite clock error model (3.99) is then

$$\Delta \tilde{t}_s = (\tilde{a}_{f0} + \delta a_{f0}) + (\tilde{a}_{f1} + \delta a_{f1})(t - t_{oc}) + \tilde{a}_{f2}(t - t_{oc})^2 + \Delta t_r - T_{GD} \quad (3.115)$$

In this case, the residual satellite clock error $\delta \Delta t_s$ is expected to be small, so that the satellite clock error model is probably not required.

SBAS navigation messages

The SBAS navigation messages types 2 – 5 and type 24 “Fast Corrections” contain additional correction parameters with which the short-term URE can be reduced [3]. The pseudorange correction $\delta\rho$ to be applied is computed from the transmitted pseudorange offset PRC and the pseudorange rate RRC by

$$\delta\rho_{fc} = PRC(t_{of}) + RRC(t_{of})(t - t_{of}) \quad (3.116)$$

where t_{of} is the time of applicability of the most recent fast correction. The range rate correction RRC is not contained in the navigation message but calculated by forming the difference quotient between the current and the previous pseudorange correction parameters

$$RRC(t_{of,k}) = \frac{PRC(t_{of,k}) - PRC(t_{of,k-1})}{t_{of,k} - t_{of,k-1}} \quad (3.117)$$

Furthermore, message type 25 “Long Term Satellite Error Corrections” contains satellite clock correction coefficients δa_{f0} and δa_{f1} . Analogous to (3.115), they can be used to reduce the satellite clock correction errors

$$\begin{aligned} \Delta\tilde{t}_s &= \tilde{a}_{f0} + \tilde{a}_{f1}(t - t_{oc}) + \tilde{a}_{f2}(t - t_{oc})^2 + \Delta t_r - T_{GD} \\ &\quad + \delta a_{f0} + \delta a_{f1}(t - t_0) \end{aligned} \quad (3.118)$$

where t_0 is the time of applicability of the long-term corrections, which is also contained in message type 25 and is not equal to the time of clock t_{oc} of the GPS navigation message. By means of message type 7 “Fast Correction Degradation” even the standard deviation of the fast correction parameters can be computed and thus their degradation with time can be monitored. The covariance of the long-term satellite clock correction coefficients can be calculated with the content of message type 28 “Clock-Ephemeris Covariance Matrix Message”.

3.5.1.8 Remarks

A problem is that the satellite block type is not contained in the navigation message. Therefore, the navigation system needs an own database with the current SVN-to-PRN assignment. This has to be kept up-to-date by the operator in order to account for changes of the satellite constellation. Furthermore, if a new satellite comes into view, it is not known when the last upload cutover has been taken place. The time is required to set up the initial covariance matrices. This problem can be circumvented if almanac datasets of the previous days are loaded into the navigation system to evaluate the last upload cutover time. If the high-performance navigation system under consideration in this thesis is used in special applications with dedicated mission preparation like the singular launch of a sounding rocket, the satellite clock errors of the current satellite constellation can be analyzed in advance for the previous days. The SVN-to-PRN assignment as well as the model coefficients can be updated before the flight. Satellites with suspiciously high errors can be excluded if enough other satellites are available and the dilution of precision (DOP) value does not degrade too much. Furthermore, an optimal launch time can be chosen probably right after an upload of fresh navigation messages. If the navigation system is intended to be used in day-to-day operation, the models have to be generalized and the amount of information that is required in advance has to be kept small due to missing connections with appropriate information sources. The derived satellite clock error models will be integrated into the navigation filter in section 4.6.4.

3.5.2 Satellite Position Error

3.5.2.1 Motivation

Besides the satellite clock error, the uncorrected satellite position error contributes to the URE. As will be seen later, the magnitude of the orbit error is considerably smaller than of the satellite clock error. Splitting the position error in along-track, cross-track and radial components, it can be observed that the radial error in the range of a few decimeters is smallest of the three. This is because the radial error component is better observable by the control segment than the two other components. This is a pleasing matter of fact because it is just the radial error component that affects the pseudorange error most. The position error is mainly characterized by the orbital period and exhibits less degradation with time between subsequent upload cutovers compared to the satellite clock error. The satellite position error thus plays a secondary role in the URE.

Analogous to the satellite clock error model, it is looked for an adequate model structure at first. Then, the coefficients of this model are estimated by means of the real data that has already been used for the identification of the satellite clock error model. Finally, a way is shown how to integrate the found model into the navigation error filter to consider the radial component of the satellite position error.

3.5.2.2 Characteristic

3.5.2.2.1 Broadcast Position

The satellite position at broadcast time is calculated with the ephemeris parameters contained in the navigation message according to table 20-IV in the GPS ICD [52]. In principal, the satellite orbit is described by the six Kepler elements, that is the semi-major axis of the ellipse A , the eccentricity of the ellipse e , the inclination of the orbit i , the longitude of the ascending node Ω , the argument of perigee ω and the time of perigee passage or alternatively the mean anomaly M . The argument of latitude Φ_k , the radius r_k and the inclination i_k are additionally corrected by harmonic functions with periods of approximately 5 hours 58 minutes

$$\begin{aligned}\delta\Phi_k &= c_{us} \sin(2\Phi_k) + c_{uc} \cos(2\Phi_k) \\ \delta r_k &= c_{rs} \sin(2\Phi_k) + c_{rc} \cos(2\Phi_k) \\ \delta i_k &= c_{is} \sin(2\Phi_k) + c_{ic} \cos(2\Phi_k)\end{aligned}\quad (3.119)$$

The correction coefficients c_{us} , c_{uc} , c_{rs} , c_{rc} , c_{is} and c_{ic} are also contained in the ephemeris set of the navigation message. A similar harmonic structure will be used later on for the position error model.

3.5.2.2.2 Precise Position

Besides the clock products, IGS provides precise orbit products. Contrary to the broadcast ephemerides, these orbit products are sampled satellite positions on the orbit at intervals of 15 minutes. The satellite positions are specified in the e -frame of the corresponding time. Lagrange polynomial interpolation can be applied to calculate the satellite positions at arbitrary times in between the sampling points. The interpolation accuracy lies in the range of centimeters if ten nodes are used for the interpolation [61]. The expected accuracies and availabilities of the different orbit products are given in Table 3-22.

The satellite position that is calculated from the broadcast ephemerides actually represents the position of the antenna phase center (APC), which is decisive when processing pseudorange measurements. IGS orbit products however refer to the position of the mass center (MC) of the satellite. When comparing broadcast orbits with precise orbits in order to determine the position error resulting from broadcast ephemeris errors, the lever arm between the satellite MC and APC has to be accounted for. Since the lever arms are neither published by the manufacturers nor by the U.S. department of defense, IGS made great efforts to estimate the lever arms of the different satellite types from observations. The estimated lever arms are provided in the so-called annex file. The lever arm vector is specified in a satellite-fixed coordinate frame that is spanned by the radius vector from the satellite MC to the Earth center, the vector being normal on the radius vector and the vector between satellite MC and Sun and the third vector completing the right hand system [62].

Next to IGS, the National Geospatial-Intelligence Agency (NGA) provides precise orbit products [63]. NGA additionally offers precise ephemerides that already refer to the APC instead of the MC. These APC ephemerides are convenient since the lever arm correction has not to be done and the location vector of the satellite MC w.r.t. the Sun has not to be computed.

Table 3-22: Overview about IGS precise orbit products [58]

Product	Latency	Root Mean Square (RMS)	Sample Interval
Ultra-Rapid	3 – 9 hr	3 cm	15 min
Rapid	17 – 41 hr	2.5 cm	15 min
Final	12 – 18 days	2.5 cm	15 min

3.5.2.2.3 Comparison of Broadcast and Precise Positions

In the following, the precise and broadcast satellite positions $\mathbf{X}_{e,prec}$ and $\mathbf{X}_{e,brdc}$ are compared and the position difference $\delta\mathbf{X}_e$ is calculated as

$$\delta\mathbf{X}_e = \mathbf{X}_{e,prec} - \mathbf{X}_{e,brdc} \quad (3.120)$$

Since the satellite position error of the precise ephemerides is significantly smaller than the satellite position error of the broadcast ephemerides, it is assumed that the precise satellite position represents the true position of the satellite, $\mathbf{X}_e = \mathbf{X}_{e,prec}$, and the position difference solely stems from the erroneous broadcast ephemerides, meaning $\tilde{\mathbf{X}}_e = \mathbf{X}_{e,brdc}$. The position error becomes $\delta\mathbf{X}_e = \mathbf{X}_e - \tilde{\mathbf{X}}_e$. It is advantageous to split the satellite position error $\delta\mathbf{X}_e$ into along-track, cross-track and radial direction because the pseudorange measurement $\tilde{\rho}$ is predominantly affected by the radial position error. It may thus be sufficient to consider only the radial error of the satellite, which has to be analyzed in more detail later on.

For that purpose, the satellite track frame (t -frame) is introduced in the appendix in section B.2. The x -axis points in the along-track direction, the y -axis in the cross-track direction and the z -axis completes the orthogonal right hand system in the radial direction. As in the case of the satellite clock error, the broadcast and precise satellite positions are exemplarily compared for the year 2012. Figure 3-48 shows the position error of PRN 4 for the first seven days of April, that is GPS week 1682, Figure 3-49 the position error of PRN 10, Figure 3-50 the position error of PRN 13, Figure 3-51 the position error of PRN 19, Figure 3-52 the position error of PRN 7 and finally Figure 3-53 the position error of PRN 25. The magnitude of the radial position error lies for all satellites within the sub-meter range and is smaller than the satellite clock error. Since the radial error influences the ranging error in

first order, the control segment predominantly adapts the correction coefficients to reduce the radial error. The along-track error and the cross-track error are worse observable and correctable by the master control station. These errors thus increase up to three meters in the example. On the other hand, they have minor influence on the ranging error in equal measure. As could be expected, the position error is similar for all satellites, independent of the satellite block type and the activated clock. Upload cutovers are indicated by the blue dotted lines and take place roughly once per day. Dataset cutovers indicated by the grey dotted lines occur every two hours. The effect of upload and dataset cutovers is less obvious than with the clock error. It can be seen that the along-track, cross-track and radial position errors are characterized by 12 h period and 2 h period harmonics resulting from the orbital period as well as the upload and dataset cutover periods.

In the following, it is sufficient to analyze only one of the satellites, e.g. PRN 25, to identify an adequate satellite position error model.

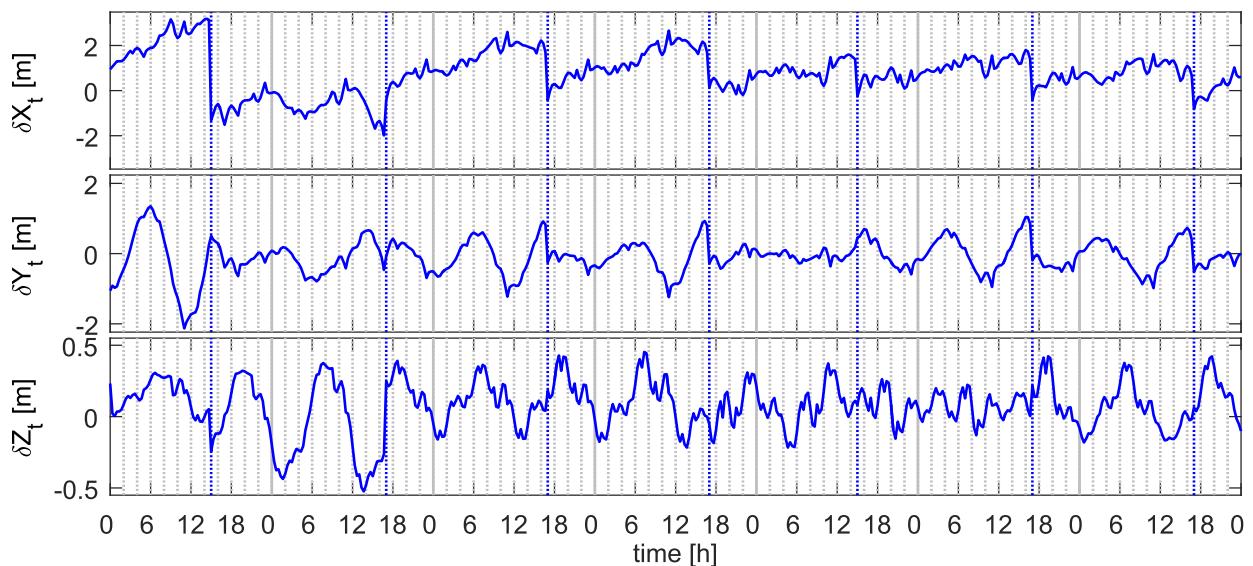


Figure 3-48: Along-track, cross-track and radial position error of PRN 4, GPS week 1682

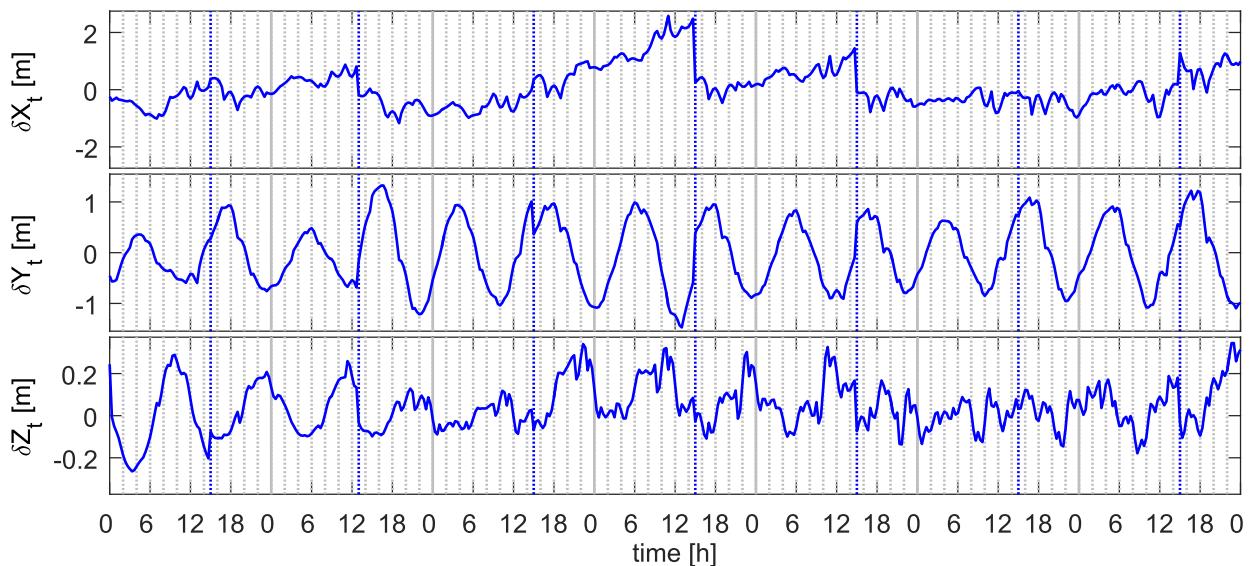


Figure 3-49: Along-track, cross-track and radial position error of PRN 10, GPS week 1682

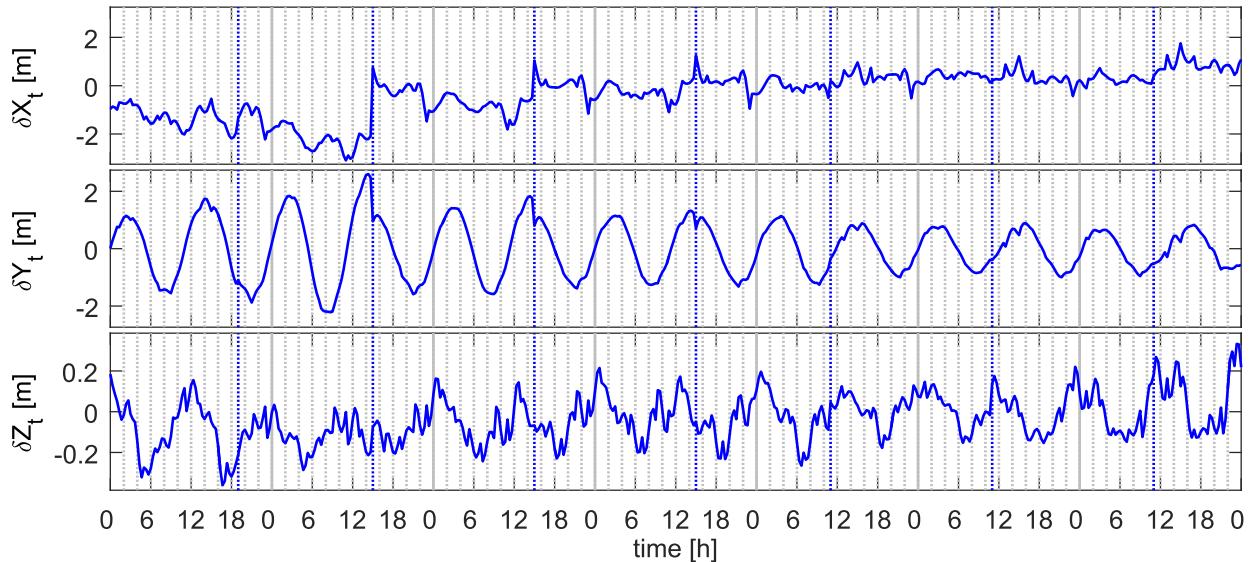


Figure 3-50: Along-track, cross-track and radial position error of PRN 13, GPS week 1682

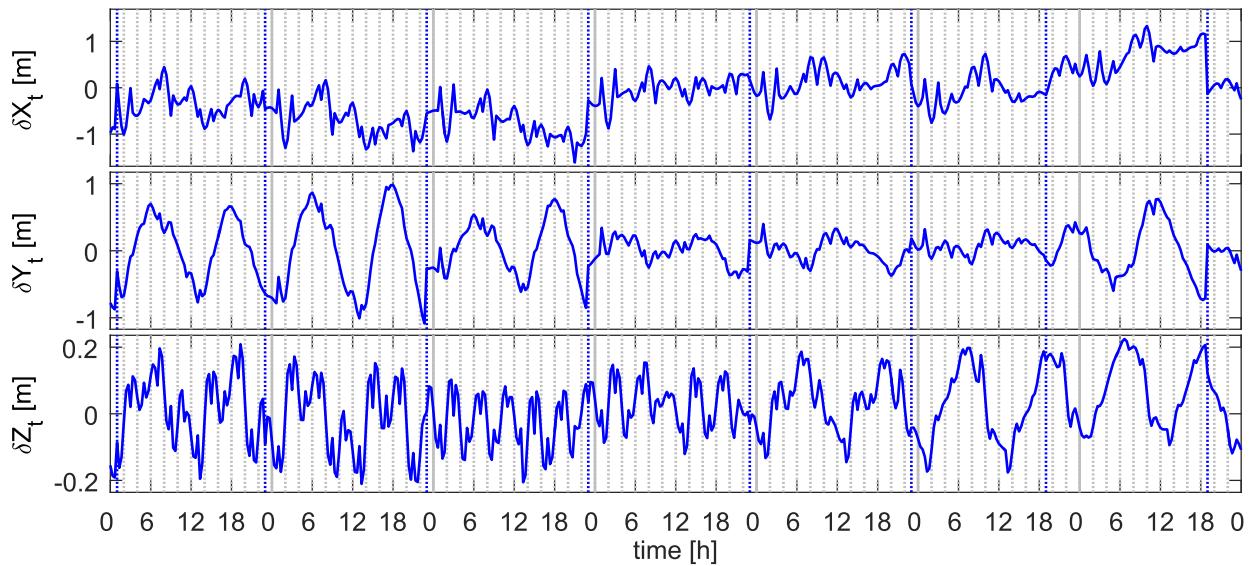


Figure 3-51: Along-track, cross-track and radial position error of PRN 19, GPS week 1682

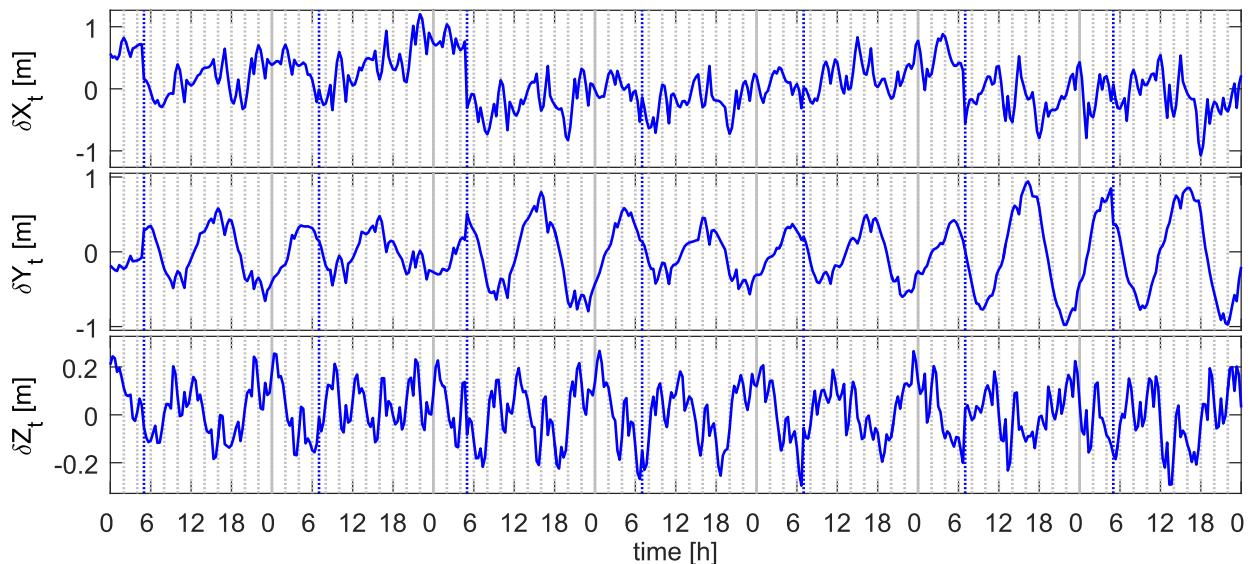


Figure 3-52: Along-track, cross-track and radial position error of PRN 7, GPS week 1682

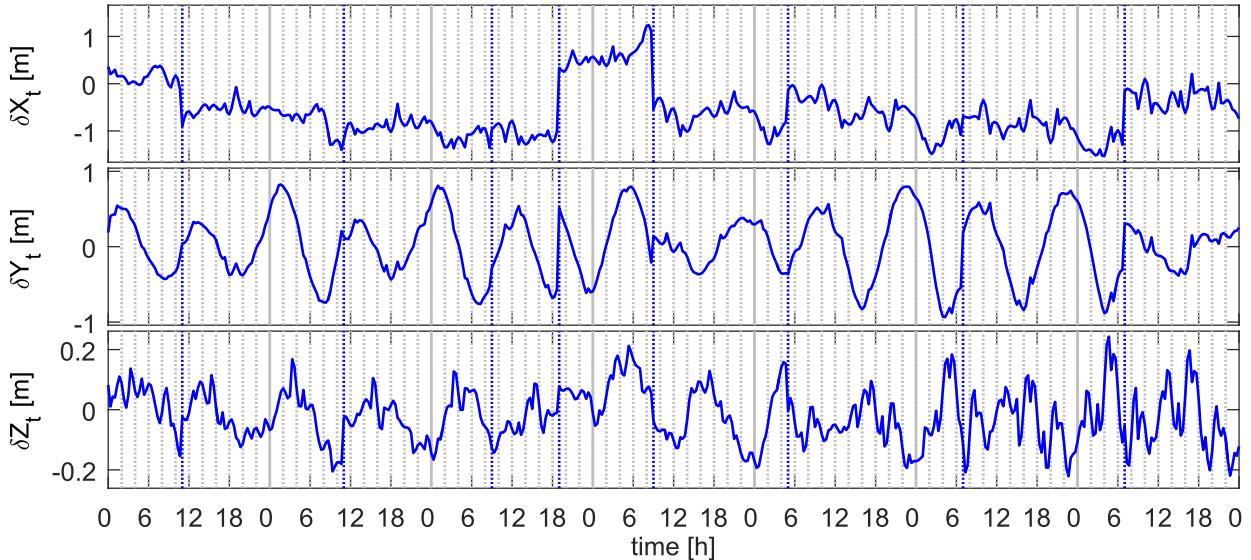


Figure 3-53: Along-track, cross-track and radial position error of PRN 25, GPS week 1682

3.5.2.3 Model

3.5.2.3.1 Structure

The along-track, cross-track and radial position errors are characterized by 12 h period and superposed 2 h period harmonics. The amplitude and phase of the 12 h period harmonics change at upload cutovers. The amplitude and phase of the 2 h period harmonics change at dataset cutovers. In order to correctly represent the harmonic character and thus the corresponding time correlation of the position error, it is favorable to assume a harmonic function model whose coefficients are described by stochastic models instead of a pure stochastic model. Inspired by the harmonic corrections used in the broadcast navigation message (3.119), a superposition of sine and cosine functions with 12 h periods and 2 h periods, each offset by a constant, is proposed for the components of the position error

$$\begin{aligned} \delta X_t(t) = & a_{x,12} \sin \frac{\pi(t - t_{oe,uc})}{6 \cdot 3600 \text{ s}} + b_{x,12} \cos \frac{\pi(t - t_{oe,uc})}{6 \cdot 3600 \text{ s}} + c_{x,12} \\ & + a_{x,2} \sin \frac{\pi(t - t_{oe,dc})}{3600 \text{ s}} + b_{x,2} \cos \frac{\pi(t - t_{oe,dc})}{3600 \text{ s}} + c_{x,2} \end{aligned} \quad (3.121)$$

The same model structure can be applied for the cross-track error $\delta Y_t(t)$ with the coefficients $a_{y,12}$, $b_{y,12}$, $c_{y,12}$, $a_{y,2}$, $b_{y,2}$ and $c_{y,2}$ and the radial error $\delta Z_t(t)$ with the coefficients $a_{z,12}$, $b_{z,12}$, $c_{z,12}$, $a_{z,2}$, $b_{z,2}$ and $c_{z,2}$. $t_{oe,uc}$ is the time of ephemeris of the last upload crossover and $t_{oe,dc}$ is the time of ephemeris of the last dataset crossover. The position error can alternatively be represented with amplitude and phase

$$\delta X_t(t) = A_{x,12} \sin \left(\frac{\pi(t - t_{oe,uc})}{6 \cdot 3600 \text{ s}} + \varphi_{x,12} \right) + c_{x,12} + A_{x,2} \sin \left(\frac{\pi(t - t_{oe,dc})}{3600 \text{ s}} + \varphi_{x,2} \right) + c_{x,2} \quad (3.122)$$

Both representations can be converted into the other form with

$$\begin{aligned} A_{i,12} &= \sqrt{a_{i,12}^2 + b_{i,12}^2}, & \varphi_{i,12} &= \text{atan2}(b_{i,12}, a_{i,12}) \\ A_{i,2} &= \sqrt{a_{i,2}^2 + b_{i,2}^2}, & \varphi_{i,2} &= \text{atan2}(b_{i,2}, a_{i,2}) \quad i = \{x, y, z\} \end{aligned} \quad (3.123)$$

It is assumed that the a and b coefficients, respectively, are not correlated.

3.5.2.3.2 Curve Fitting

First, the coefficients of the 12 h period sine and cosine functions with the amplitudes $a_{x,12}$ and $b_{x,12}$ and the constant offset $c_{x,12}$ are fitted to the sample data

$$\delta X_{t,k} = \begin{pmatrix} \sin \frac{\pi(t_k - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & \cos \frac{\pi(t_k - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & 1 \end{pmatrix} \cdot \begin{pmatrix} a_{x,12} \\ b_{x,12} \\ c_{x,12} \end{pmatrix} \quad (3.124)$$

All available sample position errors $\delta X_{t,1} \dots \delta X_{t,M}$ with the sample times $t_1 \dots t_M$ within one upload cutover window with the corresponding time of ephemeris $t_{oe,uc}$ are collected to form the $\mathbf{A}_{x,12}$ -matrix and $\mathbf{y}_{x,12}$ -vector

$$\mathbf{A}_{x,12} = \begin{pmatrix} \sin \frac{\pi(t_1 - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & \cos \frac{\pi(t_1 - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & 1 \\ \vdots & \vdots & \vdots \\ \sin \frac{\pi(t_M - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & \cos \frac{\pi(t_M - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & 1 \end{pmatrix}, \quad \mathbf{y}_{x,12} = \begin{pmatrix} \delta X_{t,1} \\ \vdots \\ \delta X_{t,M} \end{pmatrix} \quad (3.125)$$

The coefficients are found by least-squares estimation

$$\begin{pmatrix} \hat{a}_{x,12} \\ \hat{b}_{x,12} \\ \hat{c}_{x,12} \end{pmatrix} = (\mathbf{A}_{x,12}^\top \mathbf{A}_{x,12})^{-1} \mathbf{A}_{x,12}^\top \mathbf{y}_{x,12} \quad (3.126)$$

The curve fitting is repeated for all available upload cutover windows and afterwards for the cross-track and radial satellite position errors. In a second step, with the results of the upload cutover fitting, the dataset cutover fitting can be done. For that purpose, the actual position error on the left hand side in (3.121) is corrected by the estimated upload cutover window error

$$\begin{aligned} \delta X_{t,k} - \begin{pmatrix} \sin \frac{\pi(t_k - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & \cos \frac{\pi(t_k - t_{oe,uc})}{6 \cdot 3600 \text{ s}} & 1 \end{pmatrix} \cdot \begin{pmatrix} \hat{a}_{x,12} \\ \hat{b}_{x,12} \\ \hat{c}_{x,12} \end{pmatrix} \\ = \begin{pmatrix} \sin \frac{\pi(t_k - t_{oe,dc})}{3600 \text{ s}} & \cos \frac{\pi(t_k - t_{oe,dc})}{3600 \text{ s}} & 1 \end{pmatrix} \cdot \begin{pmatrix} a_{x,2} \\ b_{x,2} \\ c_{x,2} \end{pmatrix} \end{aligned} \quad (3.127)$$

Again, the N samples within common dataset cutover windows $\delta X_{t,1} \dots \delta X_{t,N}$ are collected in the $\mathbf{A}_{x,2}$ -matrix and the $\mathbf{y}_{x,2}$ -vector

$$\mathbf{A}_{x,2} = \begin{pmatrix} \sin \frac{\pi(t_1 - t_{oe,dc})}{3600 \text{ s}} & \cos \frac{\pi(t_1 - t_{oe,dc})}{3600 \text{ s}} & 1 \\ \vdots & \vdots & \vdots \\ \sin \frac{\pi(t_N - t_{oe,dc})}{3600 \text{ s}} & \cos \frac{\pi(t_N - t_{oe,dc})}{3600 \text{ s}} & 1 \end{pmatrix}, \quad (3.128)$$

$$\mathbf{y}_{x,2} = \begin{pmatrix} \delta X_{t,1} - \hat{a}_{x,12} \sin \frac{\pi(t_1 - t_{oe,uc})}{6 \cdot 3600 \text{ s}} - \hat{b}_{x,12} \cos \frac{\pi(t_1 - t_{oe,uc})}{6 \cdot 3600 \text{ s}} - \hat{c}_{x,12} \\ \vdots \\ \delta X_{t,N} - \hat{a}_{x,12} \sin \frac{\pi(t_N - t_{oe,uc})}{6 \cdot 3600 \text{ s}} - \hat{b}_{x,12} \cos \frac{\pi(t_N - t_{oe,uc})}{6 \cdot 3600 \text{ s}} - \hat{c}_{x,12} \end{pmatrix}$$

Like in the first step, the coefficients of the 2 h period harmonics are obtained by least-squares estimation

$$\begin{pmatrix} \hat{a}_{x,2} \\ \hat{b}_{x,2} \\ \hat{c}_{x,2} \end{pmatrix} = (\mathbf{A}_{x,2}^\top \mathbf{A}_{x,2})^{-1} \mathbf{A}_{x,2}^\top \mathbf{y}_{x,2} \quad (3.129)$$

Figure 3-54 exemplarily illustrates the position error $\delta \mathbf{X}_t$ of PRN 25 (blue) and the fitted position error $\delta \hat{\mathbf{X}}_t$ (red) for April 1 & 2, 2012. In Figure 3-55, the along-track error coefficients $\hat{a}_{x,12}$, $\hat{b}_{x,12}$, $\hat{c}_{x,12}$, in Figure 3-56 the cross-track error coefficients $\hat{a}_{y,12}$, $\hat{b}_{y,12}$, $\hat{c}_{y,12}$, in Figure 3-57 the radial error coefficients $\hat{a}_{z,12}$, $\hat{b}_{z,12}$, $\hat{c}_{z,12}$, in Figure 3-58 the along-track error coefficients $\hat{a}_{x,2}$, $\hat{b}_{x,2}$, $\hat{c}_{x,2}$, in Figure 3-59 the cross-track error coefficients $\hat{a}_{y,2}$, $\hat{b}_{y,2}$, $\hat{c}_{y,2}$ and finally in Figure 3-60 the radial error coefficients $\hat{a}_{z,2}$, $\hat{b}_{z,2}$, $\hat{c}_{z,2}$ are shown for April 2012. In Figure 3-61, the residual position error $\mathbf{r}_t = \delta \mathbf{X}_t - \delta \hat{\mathbf{X}}_t$ is plotted. It can be seen that the residual radial position error $r_{t,z}$ does not exceed some centimeters.

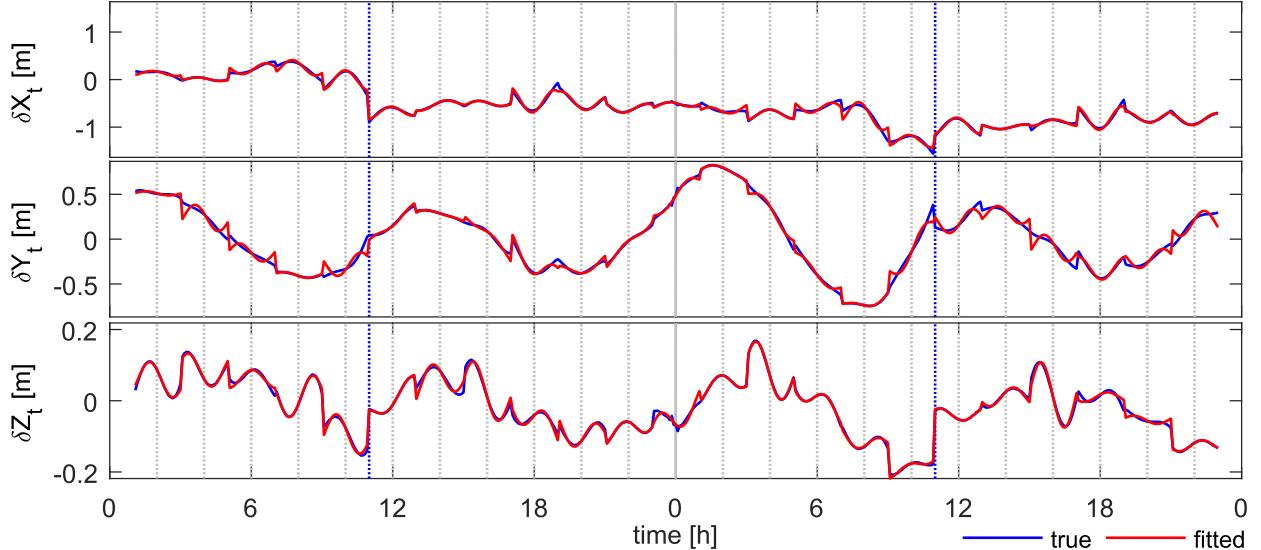


Figure 3-54: True position error and fitted position error of PRN 25, April 1 & 2, 2012

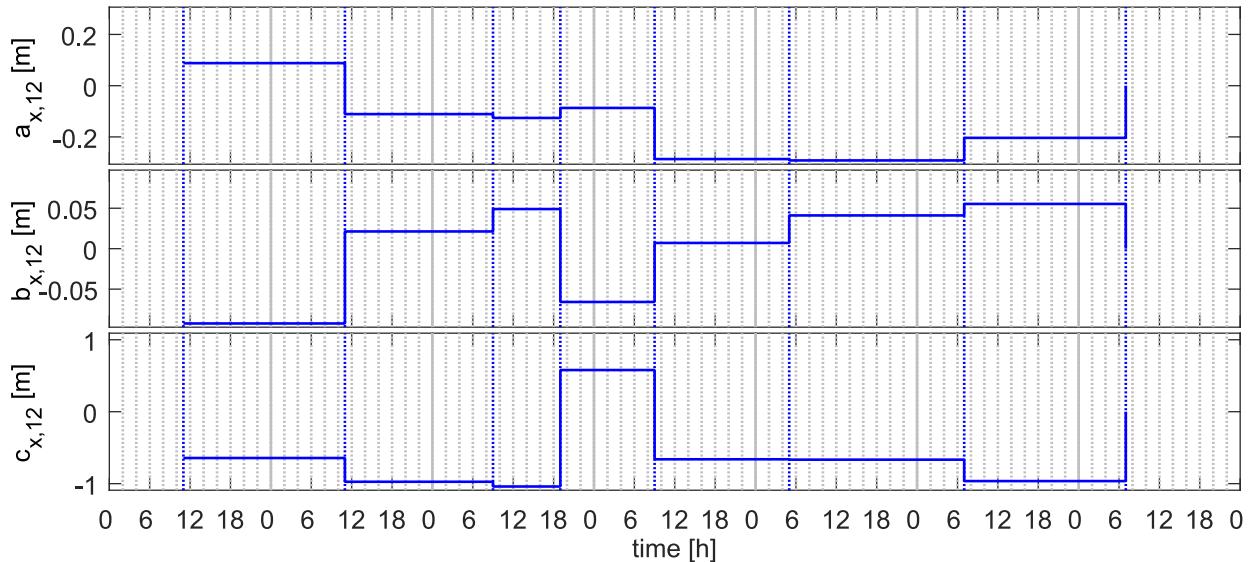


Figure 3-55: Estimated along-track error 12 h coefficients of PRN 25, GPS week 1682

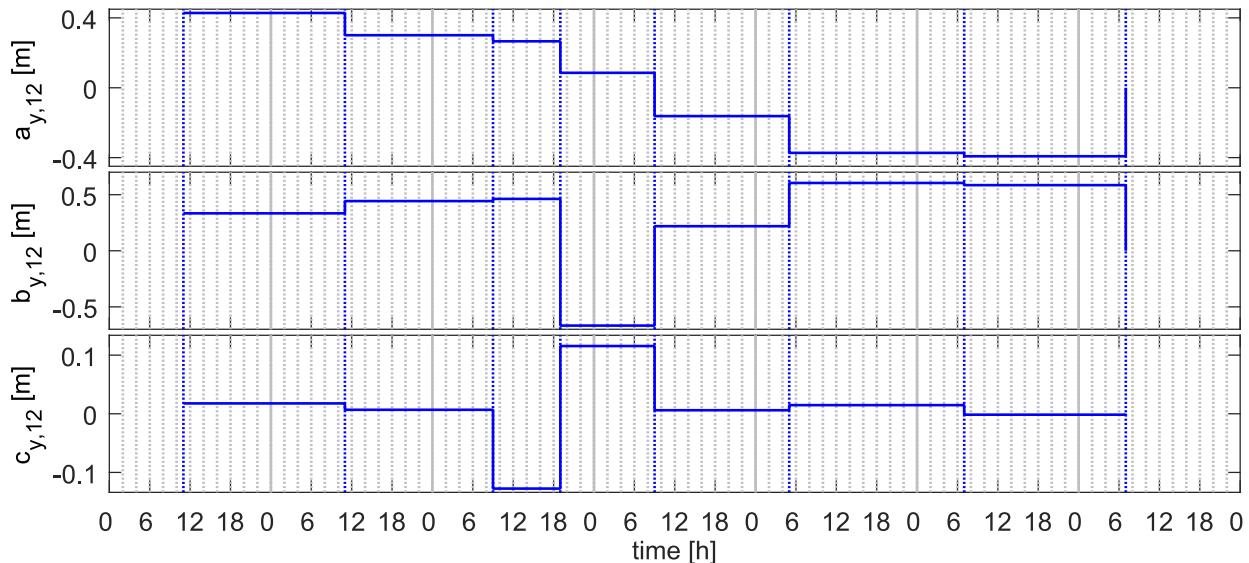


Figure 3-56: Estimated cross-track error 12 h coefficients of PRN 25, GPS week 1682

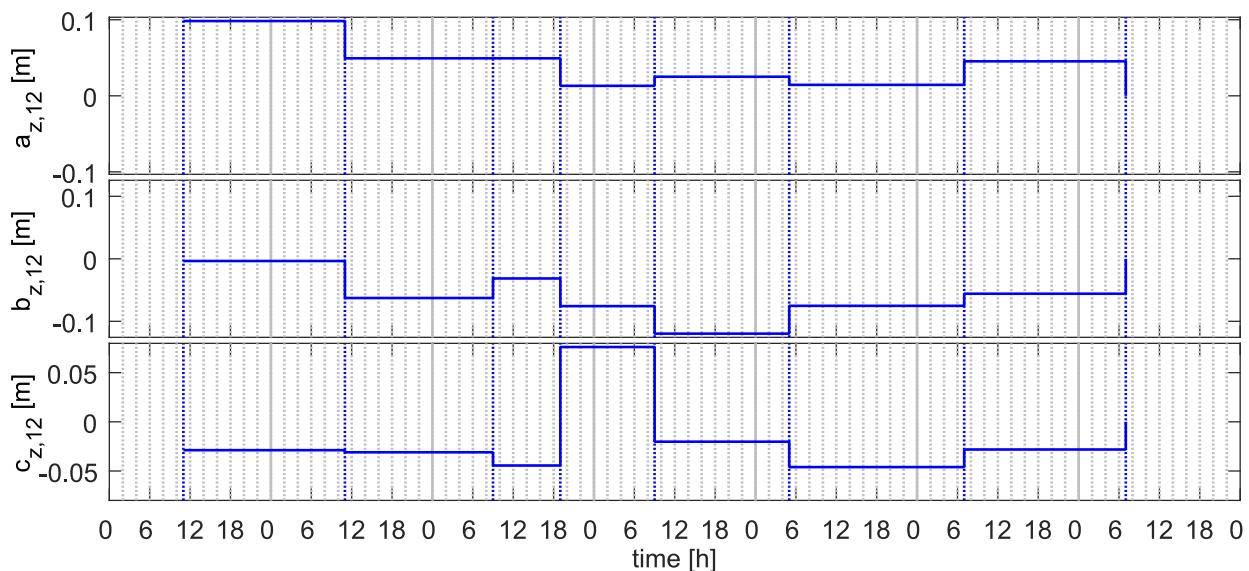


Figure 3-57: Estimated radial error 12 h coefficients of PRN 25, GPS week 1682

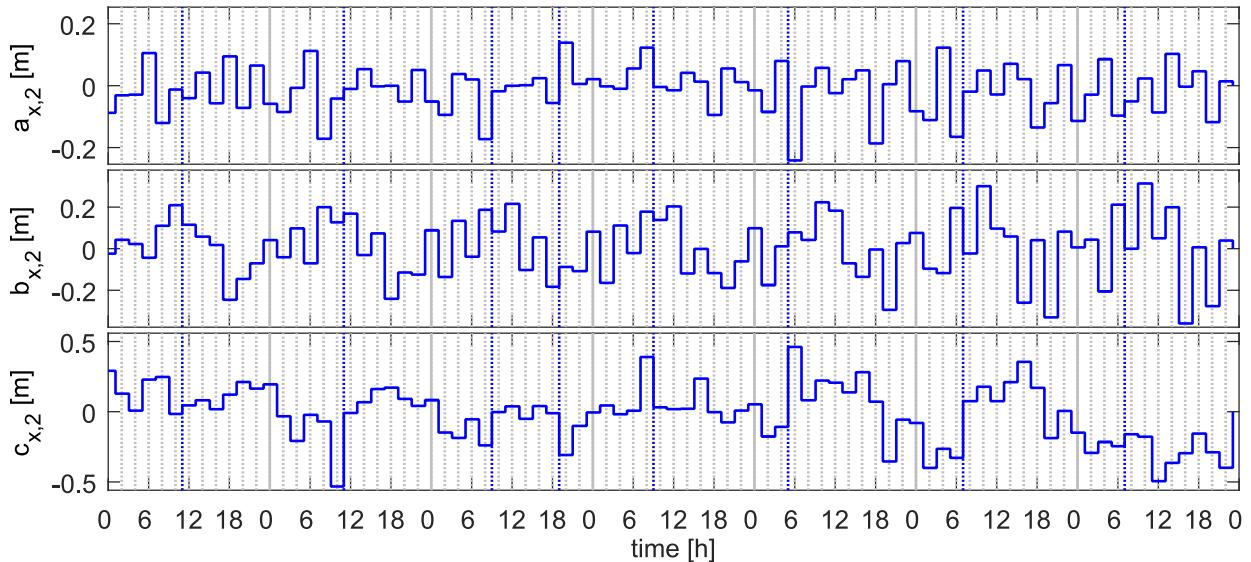


Figure 3-58: Estimated along-track error 2 h coefficients of PRN 25, GPS week 1682

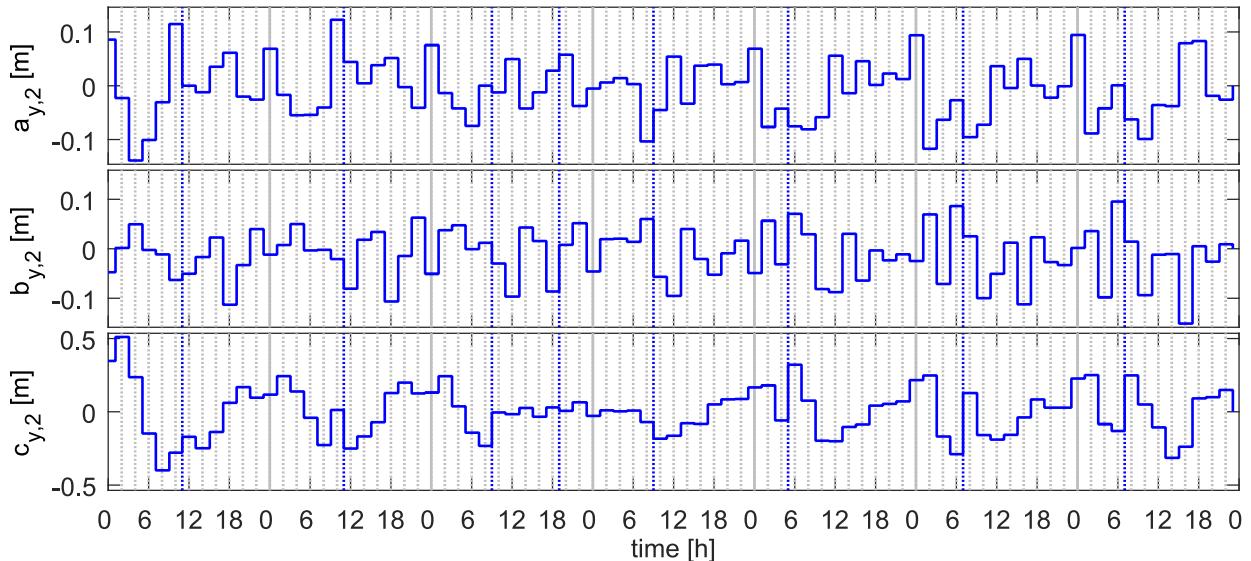


Figure 3-59: Estimated cross-track error 2 h coefficients of PRN 25, GPS week 1682

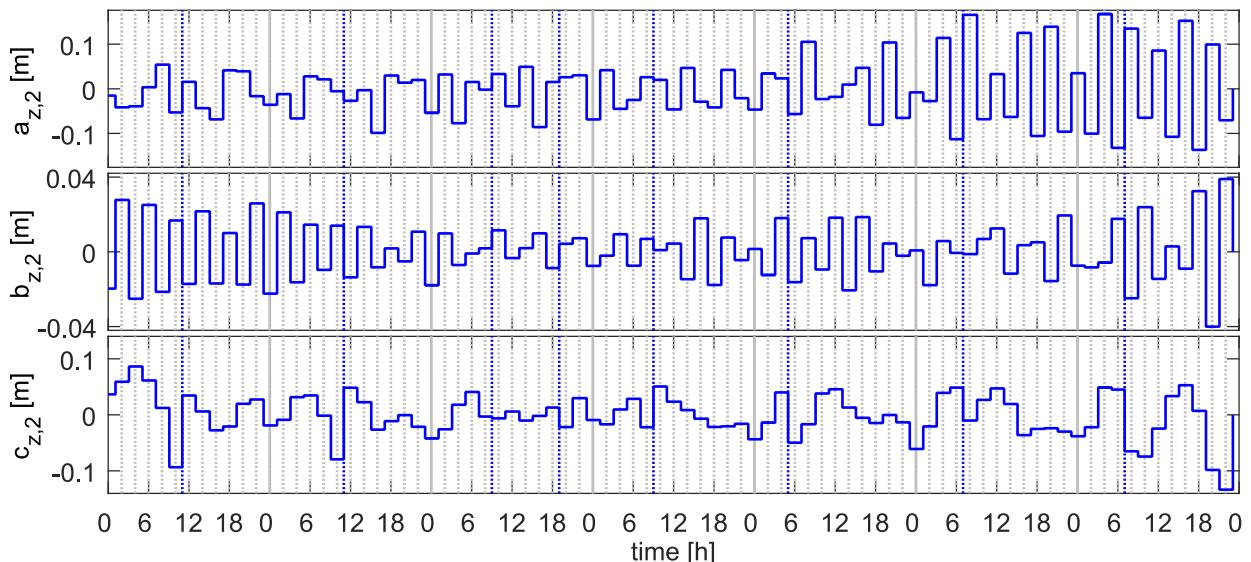


Figure 3-60: Estimated radial error 2 h coefficients of PRN 25, GPS week 1682

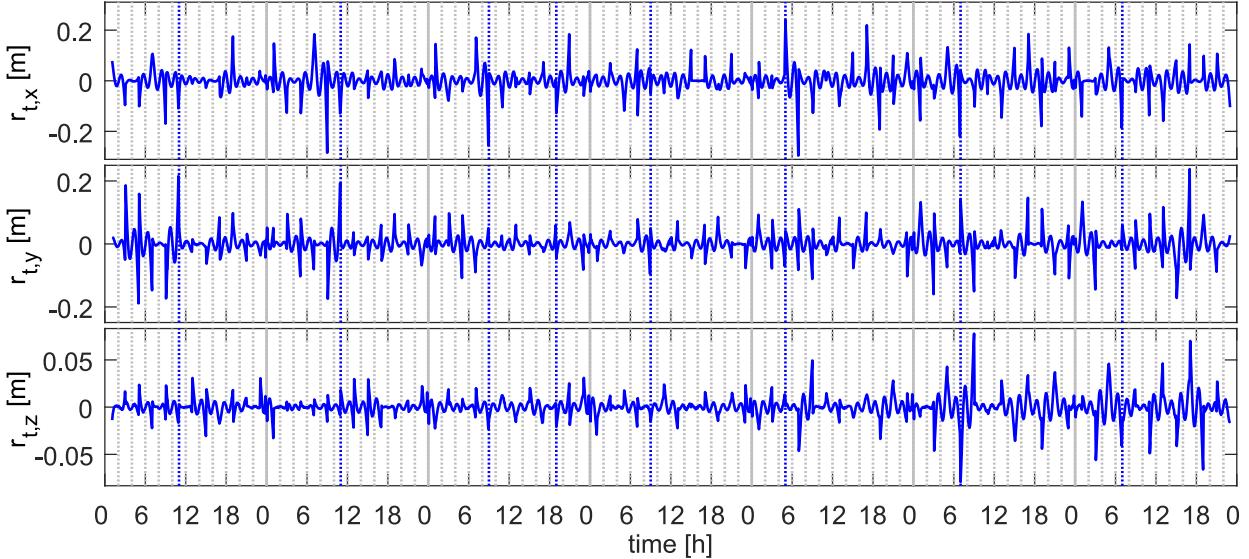


Figure 3-61: Residual position error of PRN 25, GPS week 1682

3.5.2.4 Parameter Estimation

It is proposed to describe the stochastic behavior of each position error coefficient by separate stationary ARMA($p, 0$) models. This makes in total 18 models. The model structure for the a coefficients is given with

$$\begin{aligned} \mathbf{z}_{a_{i,j},k} &= \Phi_{a_{i,j}} \mathbf{z}_{a_{i,j},k-1} + \Gamma_{a_{i,j}} \eta_{a_{i,j},k-1} & i = \{x, y, z\}, \quad j = \{2, 12\} \\ a_{i,j,k} &= \mathbf{H}_{a_{i,j}} \mathbf{z}_{a_{i,j},k} + D_{a_{i,j}} \eta_{a_{i,j},k} \end{aligned} \quad (3.130)$$

and for the b coefficients

$$\begin{aligned} \mathbf{z}_{b_{i,j},k} &= \Phi_{b_{i,j}} \mathbf{z}_{b_{i,j},k-1} + \Gamma_{b_{i,j}} \eta_{b_{i,j},k-1} & i = \{x, y, z\}, \quad j = \{2, 12\} \\ b_{i,j,k} &= \mathbf{H}_{b_{i,j}} \mathbf{z}_{b_{i,j},k} + D_{b_{i,j}} \eta_{b_{i,j},k} \end{aligned} \quad (3.131)$$

and finally for the c coefficients

$$\begin{aligned} \mathbf{z}_{c_{i,j},k} &= \Phi_{c_{i,j}} \mathbf{z}_{c_{i,j},k-1} + \Gamma_{c_{i,j}} \eta_{c_{i,j},k-1} & i = \{x, y, z\}, \quad j = \{2, 12\} \\ c_{i,j,k} &= \mathbf{H}_{c_{i,j}} \mathbf{z}_{c_{i,j},k} + D_{c_{i,j}} \eta_{c_{i,j},k} \end{aligned} \quad (3.132)$$

The AR coefficients are determined by means of MATLAB®'s system identification toolbox with the versatile `armax` function. Data of the complete year 2012 is evaluated. It has been revealed that a model order of $p = 2$ already yields satisfactory results. The estimated coefficients and input noise covariance matrices are listed in Table 3-23 to Table 3-28. A suitable general model for all satellites can be found by computing the mean values of each coefficient and covariance. The relation between the AR coefficients and the system matrices is given in Template 3-1. All feed-through gains D are one.

Table 3-23: Estimated ARMA(2,0) model coefficients for 12 h along-track error

PRN	$a_{x,12}$			$b_{x,12}$			$c_{x,12}$		
	a_1	a_2	Q	a_1	a_2	Q	a_1	a_2	Q
4	4.669e-01	1.993e-01	3.414e-02	5.192e-01	2.714e-01	3.544e-02	9.716e-02	1.426e-01	4.004e-01
10	4.801e-01	1.542e-01	7.459e-02	4.934e-01	2.067e-01	8.909e-02	2.148e-01	2.410e-01	9.105e-01
13	-7.041e-03	2.416e-04	3.102e-01	1.303e-01	7.188e-02	4.282e-02	2.098e-01	1.727e-01	5.975e-01
19	4.950e-01	3.188e-01	1.728e-02	4.330e-01	2.745e-01	1.856e-02	3.247e-01	2.450e-01	4.517e-01
7	5.040e-01	3.335e-01	1.047e-02	5.277e-01	2.492e-01	1.141e-02	2.668e-01	1.413e-01	2.946e-01
25	7.403e-01	1.407e-01	1.291e-02	6.722e-01	2.466e-01	1.265e-02	2.614e-01	1.077e-01	3.039e-01

Table 3-24: Estimated ARMA(2,0) model coefficients for 12 h cross-track error

PRN	$a_{y,12}$			$b_{y,12}$			$c_{y,12}$		
	a_1	a_2	Q	a_1	a_2	Q	a_1	a_2	Q
4	3.911e-01	3.055e-01	9.985e-02	3.034e-01	3.188e-01	1.270e-01	4.604e-01	2.916e-01	3.436e-03
10	2.300e-01	2.854e-01	2.891e-01	2.731e-01	2.899e-01	3.264e-01	2.896e-01	2.124e-01	1.133e-02
13	5.171e-01	3.617e-01	9.037e-02	5.472e-01	3.316e-01	8.094e-02	1.174e-01	9.717e-02	2.464e-03
19	4.080e-01	4.011e-01	1.636e-01	5.554e-01	3.296e-01	9.241e-02	3.349e-01	2.115e-01	1.714e-03
7	5.178e-01	3.122e-01	8.637e-02	4.801e-01	2.812e-01	1.068e-01	5.097e-01	1.870e-01	1.018e-03
25	4.142e-01	1.433e-01	8.543e-02	2.728e-01	6.853e-02	8.229e-02	4.780e-01	2.889e-01	1.461e-03

Table 3-25: Estimated ARMA(2,0) model coefficients for 12 h radial error

PRN	$a_{z,12}$			$b_{z,12}$			$c_{z,12}$		
	a_1	a_2	Q	a_1	a_2	Q	a_1	a_2	Q
4	5.362e-01	2.253e-01	7.677e-03	4.398e-01	1.206e-01	8.147e-03	2.149e-01	1.757e-01	2.799e-03
10	5.950e-01	7.723e-02	1.726e-02	5.482e-01	1.521e-01	1.618e-02	2.837e-01	3.283e-01	8.402e-03
13	2.683e-01	2.283e-01	7.956e-03	2.211e-01	1.682e-01	1.084e-02	6.220e-02	3.480e-02	8.563e-03
19	5.686e-01	1.472e-01	1.932e-03	5.025e-01	2.740e-01	3.017e-03	2.957e-01	1.108e-01	3.058e-03
7	5.935e-01	2.545e-01	2.453e-03	5.072e-01	4.136e-01	1.875e-03	2.754e-01	1.302e-01	1.473e-03
25	6.966e-01	2.586e-01	1.841e-03	6.661e-01	2.720e-01	1.871e-03	4.000e-01	1.643e-01	1.592e-03

Table 3-26: Estimated ARMA(2,0) model coefficients for 2 h along-track error

PRN	$a_{x,2}$			$b_{x,2}$			$c_{x,2}$		
	a_1	a_2	Q	a_1	a_2	Q	a_1	a_2	Q
4	-3.072e-01	-5.469e-02	8.973e-03	4.488e-01	-3.128e-01	6.008e-02	6.027e-01	3.995e-02	6.598e-02
10	-1.069e-01	-8.063e-02	1.497e-02	9.037e-02	1.657e-01	4.768e-02	5.490e-01	8.652e-03	1.444e-01
13	-6.459e-01	-4.165e-01	7.841e-03	5.462e-02	1.543e-01	5.582e-02	3.665e-01	1.042e-01	5.425e-02
19	-4.982e-01	-3.702e-01	9.529e-03	1.516e-01	9.602e-02	4.802e-02	4.948e-01	5.335e-02	7.212e-02
7	-5.855e-01	-3.014e-01	8.674e-03	-1.508e-03	2.349e-01	5.389e-02	4.344e-01	8.347e-02	5.763e-02
25	-2.258e-01	-5.266e-03	7.388e-03	-6.052e-02	4.309e-01	3.065e-02	5.055e-01	3.408e-02	4.698e-02

Table 3-27: Estimated ARMA(2,0) model coefficients for 2 h cross-track error

PRN	$a_{y,2}$			$b_{y,2}$			$c_{y,2}$		
	a_1	a_2	Q	a_1	a_2	Q	a_1	a_2	Q
4	-1.109e-01	-3.666e-01	4.525e-03	-5.059e-03	-6.837e-02	1.667e-02	3.629e-01	-3.147e-01	2.235e-02
10	-2.237e-01	-3.534e-01	8.106e-03	-2.118e-01	-3.193e-01	8.964e-03	2.854e-01	-2.446e-01	3.476e-02
13	-4.362e-02	-2.685e-01	2.931e-03	-1.160e-02	-1.197e-01	7.326e-03	4.782e-01	-3.197e-01	1.325e-02
19	-8.271e-02	-2.875e-01	3.790e-03	-5.697e-01	-6.619e-01	4.677e-03	4.124e-01	-2.524e-01	1.812e-02
7	-1.763e-01	-3.648e-01	3.061e-03	-2.795e-01	-3.721e-01	5.775e-03	4.431e-01	-2.792e-01	1.365e-02
25	-1.500e-01	-3.304e-01	4.056e-03	-2.342e-01	-3.905e-01	4.541e-03	2.947e-01	-2.093e-01	1.787e-02

Table 3-28: Estimated ARMA(2,0) model coefficients for 2 h radial error

PRN	$a_{z,2}$			$b_{z,2}$			$c_{z,2}$		
	a_1	a_2	Q	a_1	a_2	Q	a_1	a_2	Q
4	-4.707e-01	2.931e-01	2.571e-03	-5.013e-01	3.639e-01	1.763e-04	5.634e-01	-1.500e-01	6.532e-04
10	-5.283e-01	2.974e-01	2.840e-03	-5.949e-01	2.931e-01	1.233e-04	6.077e-01	-2.408e-01	7.601e-04
13	-7.429e-01	1.048e-01	2.508e-03	-7.145e-01	1.229e-01	1.896e-04	4.477e-01	-1.472e-01	2.982e-04
19	-4.645e-01	3.388e-01	3.029e-03	-6.548e-01	2.418e-01	9.093e-05	4.072e-01	-9.986e-02	2.782e-04
7	-7.045e-01	1.076e-01	3.398e-03	-8.191e-01	7.400e-02	1.331e-04	5.849e-01	-1.457e-01	3.473e-04
25	-6.373e-01	1.707e-01	2.877e-03	-5.618e-01	2.844e-01	1.873e-04	2.897e-01	-1.608e-01	1.124e-03

3.5.2.4.1 Measurement Equation

The linearized pseudorange measurement error $\delta\rho$ is given with

$$\delta\rho \doteq (-\mathbf{e}_e^\top, 1) \begin{pmatrix} \delta\mathbf{x}_e \\ c \delta\Delta t_R \end{pmatrix} + \mathbf{e}_e^\top \delta\mathbf{X}_e \quad (3.133)$$

The focus is here on the contribution of the satellite position error $\delta\mathbf{X}_e$. Since the effect of the along-track error and the cross-track error on the pseudorange error $\delta\rho$ is smaller than of the radial error, the along-track and cross-track error components are neglected. With (B.10) and (3.121) the position error $\delta\mathbf{X}_e$ becomes

$$\delta\mathbf{X}_e = \frac{\tilde{\mathbf{V}}_e \times (\tilde{\mathbf{V}}_e \times \tilde{\mathbf{X}}_e)}{\|\tilde{\mathbf{V}}_e \times \tilde{\mathbf{X}}_e\|} \cdot \left(a_{z,12} \sin \frac{\pi(t - t_{oe,uc})}{6 \cdot 3600 \text{s}} + b_{z,12} \cos \frac{\pi(t - t_{oe,uc})}{6 \cdot 3600 \text{s}} + c_{z,12} \right. \\ \left. + a_{z,2} \sin \frac{\pi(t - t_{oe,dc})}{3600 \text{s}} + b_{z,2} \cos \frac{\pi(t - t_{oe,dc})}{3600 \text{s}} + c_{z,2} \right) \quad (3.134)$$

The linearized pseudorange error equation (3.133) can then be written as

$$\delta\rho_k \doteq \mathbf{H}_k \begin{pmatrix} \delta\mathbf{x}_{e,k} \\ c \delta\Delta t_{R,k} \end{pmatrix} + H_{a,12,k} a_{z,12,k} + H_{b,12,k} b_{z,12,k} + H_{c,k} c_{z,12,k} \\ + H_{a,2,k} a_{z,2,k} + H_{b,2,k} b_{z,2,k} + H_{c,k} c_{z,2,k} \quad (3.135)$$

with the abbreviations defined as follows

$$\begin{aligned}
\mathbf{H}_k &= (-\mathbf{e}_{e,k}^T, 1) \\
H_{a,12,k} &= H_{c,k} \sin \frac{\pi(t_k - t_{oe,uc})}{6 \cdot 3600 \text{ s}} \\
H_{b,12,k} &= H_{c,k} \cos \frac{\pi(t_k - t_{oe,uc})}{6 \cdot 3600 \text{ s}} \\
H_{c,k} &= \mathbf{e}_{e,k}^T \frac{\tilde{\mathbf{v}}_{e,k} \times (\tilde{\mathbf{v}}_{e,k} \times \tilde{\mathbf{x}}_{e,k})}{\|\tilde{\mathbf{v}}_{e,k} \times \tilde{\mathbf{x}}_{e,k}\|} \\
H_{a,2,k} &= H_{c,k} \sin \frac{\pi(t_k - t_{oe,dc})}{3600 \text{ s}} \\
H_{b,2,k} &= H_{c,k} \cos \frac{\pi(t_k - t_{oe,dc})}{3600 \text{ s}}
\end{aligned} \tag{3.136}$$

Most applications run less than 24 hours. Therefore, at most one upload cutover will occur during the operation. In this case, the correlation of the 12 h period coefficients between subsequent upload cutovers does not have to be considered. These coefficients can be assumed as random constants that realize independently from upload cutover to upload cutover with given variance. Only for the 2 h period harmonics, the identified models will be used to describe the time correlation between dataset cutovers. With the output equations of the models (3.130), (3.131) and (3.132) for the 2 h period harmonics, the pseudorange error becomes

$$\begin{aligned}
\delta\rho_k \doteq \mathbf{H}_k &\left(\begin{matrix} \delta\mathbf{x}_{e,k} \\ c \delta\Delta t_{R,k} \end{matrix} \right) + H_{a,12,k} a_{z,12,k} + H_{b,12,k} b_{z,12,k} + H_{c,k} c_{z,12,k} \\
&+ H_{a,2,k} (\mathbf{H}_{a_{z,2}} \mathbf{z}_{a_{z,2},k} + \eta_{a_{z,2},k}) + H_{b,2,k} (\mathbf{H}_{b_{z,2}} \mathbf{z}_{b_{z,2},k} + \eta_{b_{z,2},k}) + H_{c,k} (\mathbf{H}_{c_{z,2}} \mathbf{z}_{c_{z,2},k} + \eta_{c_{z,2},k})
\end{aligned} \tag{3.137}$$

The derived satellite position error models will be applied to the integrated navigation filter in section 4.6.5.

3.5.3 Ionosphere Error

3.5.3.1 Motivation

The ionosphere is the layer of the atmosphere of the Earth between 70 km and 1000 km above the Earth ellipsoid. It is characterized by a large amount of ions and free electrons. Ultraviolet sun rays partly ionize the air molecules in this layer and produce on the one hand O⁺H⁺ ions in the topside region, O⁺ ions in the F region and O₂⁺NO⁺ ions in the E region and on the other hand free electrons. The density of electrons is highest in the F region at a height of about 300 km to 350 km. A typical free electron density profile is shown in Figure 3-62.

Especially the free electrons influence the propagation speed of electro-magnetic signals. The ionospheric delay of the ranging code along the signal path from the satellite to the receiver is

$$\Delta\rho_{iono} = \frac{40.3}{f_{carrier}^2} \int_0^{\|\mathbf{x}-\mathbf{x}\|} n_e(s) ds \quad (3.138)$$

where \mathbf{X} is the satellite position, \mathbf{x} the user position, n_e the number of free electrons in a square meter about the wave and s the line-of-sight parameter [64]. The integrated carrier phase advance $\Delta\Phi_{iono}$ is accordingly

$$\Delta\Phi_{iono} = -\frac{2\pi}{\lambda_{carrier}} \frac{40.3}{f_{carrier}^2} \int_0^{\|\mathbf{x}-\mathbf{x}\|} n_e(s) ds \quad (3.139)$$

The number of free electrons and hence the ionospheric delay depend on the daily insolation. It is lowest at night and increases during the day. It reaches its maximum roughly at 14:00 local time. In the northern hemisphere it is lower in winter and higher during summer. Furthermore, it is affected by the solar activity. In periods with high activity with many solar flares and eruptions and subsequent solar winds the ionospheric delays are usually higher than in calm periods.

Ionospheric pseudorange delays of 3 m after midnight and 15 m at early afternoon are observed for satellites in zenith [64]. The ionospheric delay of satellites with low elevation between 0 and 10 degree can be as high as 45 m during the day due to the long path of the signal through the ionosphere.

Because of the dispersive character of the refraction n in the ionosphere, dual frequency receivers, which track, for example, simultaneously GPS L1 and L5 spreading codes, can accurately determine the ionospheric delay from the difference of both pseudorange measurements. Even if the code is not accessible on the second signal as in the case of GPS L1 and L2 and only the carrier phase of the second signal can be tracked, the receiver can still determine the ionospheric delay by comparing the integrated carrier phase measurements. Single frequency users, however, have to rely on a model to compensate for the ionospheric delays on the pseudorange measurements to the satellites in view or have to estimate them by means of the integrated navigation filter.

In the following, three popular ionospheric delay models, that are the GPS Klobuchar model, the NeQuick2 model serving as basis for the Galileo broadcast correction model (NeQuickG) and the Total Electron Count (TEC) maps as for example provided by IGS are presented and compared. It is discussed how to integrate the ionospheric pseudorange error into the navigation error filter.

In section B.3 in the appendix, the ionospheric shell substitution and the calculation of the obliquity factor that are used by the Klobuchar model and TEC maps are described.

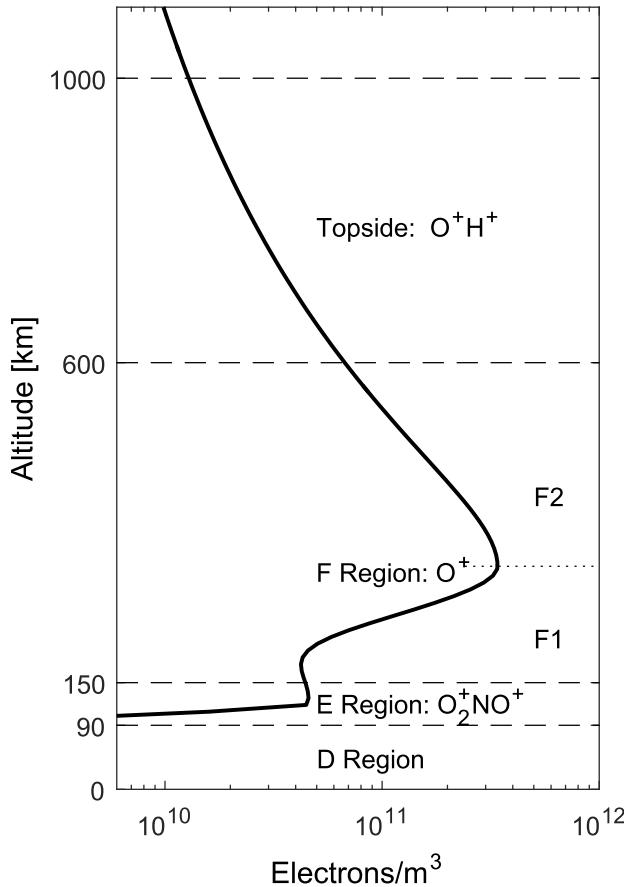


Figure 3-62: Typical free electron density profile of the ionosphere

3.5.3.2 Ionospheric Delay Models

3.5.3.2.1 Klobuchar

The Klobuchar model was originally presented in [65]. It is the officially proposed broadcast correction model for GPS single frequency users [52]. The model first calculates the total electron contents and hence the vertical path delays at the ionospheric pierce points (IPP) and then maps them by means of the obliquity factors onto the slanted line-of-sight directions to obtain the actual delays along the signal paths from the satellites to the user.

The model uses the local time to account for the current elevation of the Sun. The amplitude and the time phase of the vertical delay are each described by third order polynomials of the geomagnetic latitude of the IPP. The seasonal displacement between northern and southern hemisphere is covered by the coefficients of the polynomials. The coefficients $\alpha_i, i = 0 \dots 3$ and $\beta_i, i = 0 \dots 3$ are contained in the broadcast navigation message. With the Klobuchar model about 50% of the ionospheric error can be corrected [64].

3.5.3.2.2 NeQuick2 / NeQuickG

The NeQuick2 model is a more realistic model of the ionosphere than the Klobuchar model because it does without the shell substitution and considers the actual free electron density profiles over height like the International Reference Ionosphere (IRI) [66]. The ionospheric delay along the signal path is obtained by computing the integral (3.138). The original model was presented in [67]. Since then, the model has experienced some modifications up to the present form [68]. It is recommended by the

International Telecommunication Union (ITU) as standard ITU-R P.531 for the analysis of radio wave propagation that is affected by the ionosphere [69]. Moreover, the model is proposed as standard broadcast correction model for single frequency Galileo users in slightly modified form, NeQuickG [70]. A preliminary analysis study comparing the Klobuchar and NeQuickG models is presented in [71] [72].

A detailed description of the model can be found in [73]. Meanwhile, the European Union has published a first version of the official NeQuickG document [74], too. The NeQuick2 model uses continuous functions to describe the electron density profiles. In detail, the profile is separated in five Epstein layers for the E and F regions, that is for the bottom part of E, top part of E, bottom part of F1, top part of F1 and for F2 and a function for the topside region. The model requires the ionosonde parameters f_{0F2} and $M_{(3000)F2}$ as input. These values can be stored as worldwide grids for each month of the year. The values can be taken from the recommendation ITU-R P.1239 [75]. The required values can be obtained by linear interpolation with position and time. Furthermore, the monthly mean sunspot number R_{12} or, alternatively, the $F_{10.7}$ solar radio flux are required. Since there is an empirical relation between both values, they can be interchangeably used

$$F_{10.7} = 63.7 + 0.728 R_{12} + 8.9 \cdot 10^{-4} R_{12}^2 \quad (3.140)$$

and

$$R_{12} = -408.99 + \sqrt{167271.8 + 1123.6 (F_{10.7} - 63.7)} \quad (3.141)$$

respectively. In Figure 3-63, the course of the sunspot number R_{12} is plotted for the years 2001 to 2014. It can be seen that the daily sunspot number varies strongly but the monthly smoothed values follow a long term trend. This trend corresponds to the solar cycle with a period of about 11 years. In Figure 3-64, the sunspot number R_{12} is shown for the year 2012.

Figure 3-65 shows the electron density profiles that have been calculated with the NeQuick2 model for the position N47° E11° and 0 m height at four hours of the day 06/22/2012 with two different sunspot numbers. As expected, it can be observed that the densities are higher during day than at night. The sunspot number influences the magnitude as well as the shape of the electron density profiles.

The Galileo SIS [70] prescribes to replace the monthly mean sunspot number R_{12} by the effective ionization level A_z , which actually constitutes the main difference between the NeQuick2 and the NeQuickG model

$$A_z = a_{i0} + a_{i1} \mu + a_{i2} \mu^2 \quad (3.142)$$

Therein, μ is the modified dip latitude [76]

$$\tan \mu = \frac{I}{\sqrt{\cos \phi}} \quad (3.143)$$

I is the true magnetic inclination angle. This angle can for example be calculated by means of the World Magnetic Model (WMM). It has to be noted that the Earth magnetic field changes slowly with time. Therefore, the stored WMM coefficients have to be updated every five years. The parameters a_{i0} , a_{i1} and a_{i2} are contained in the Galileo broadcast navigation message.

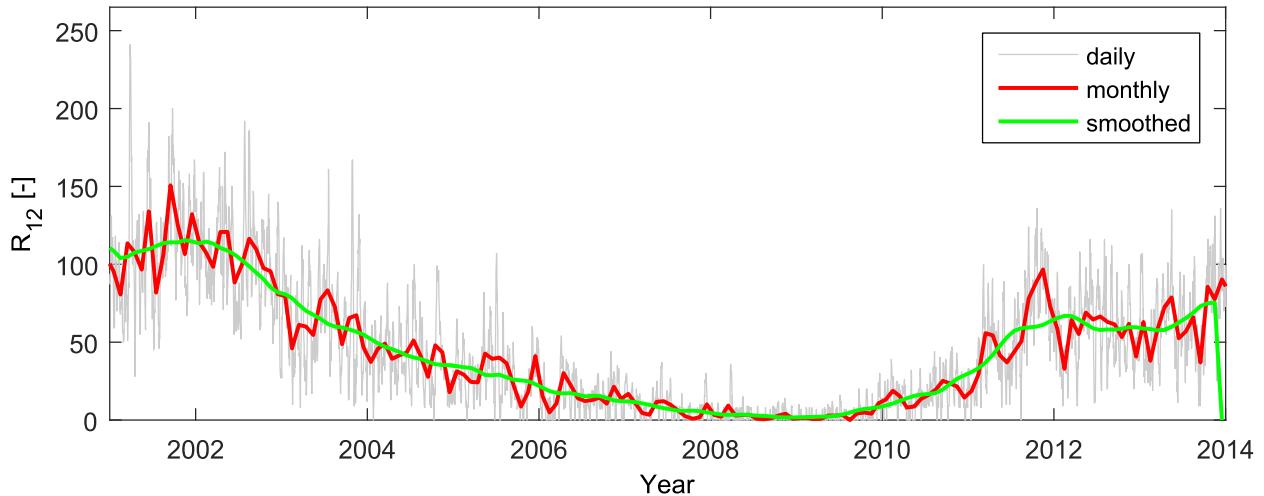


Figure 3-63: Daily, monthly mean and smoothed monthly mean sunspot number R_{12} . Data from [77]

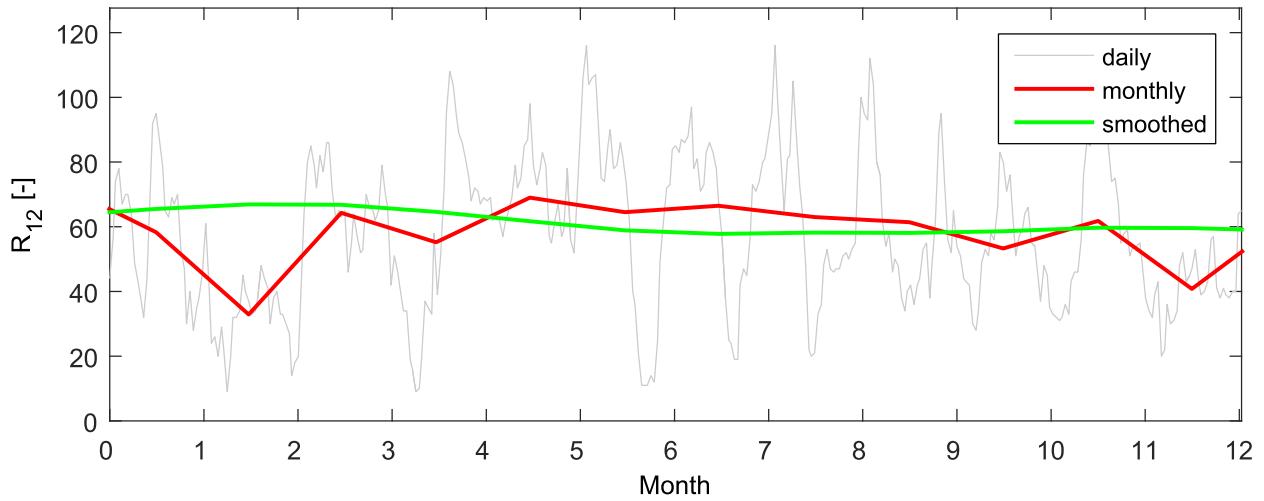


Figure 3-64: Daily, monthly mean and smoothed monthly mean sunspot number R_{12} during 2012 [77]

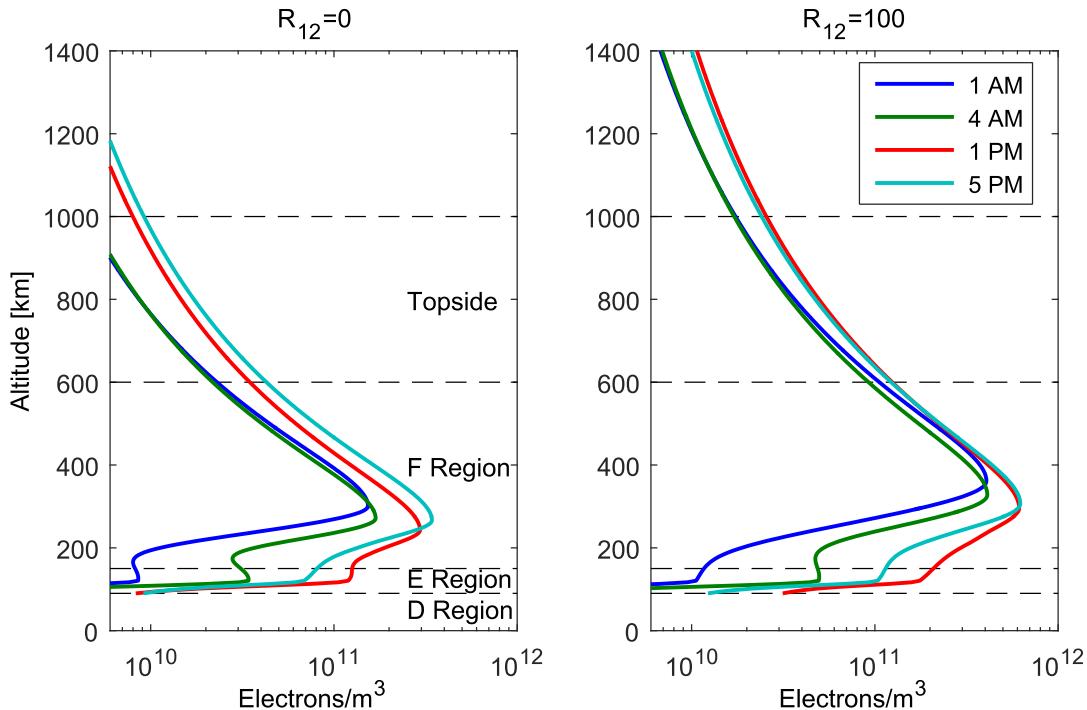


Figure 3-65: NeQuick2 electron density profiles at 06/22/2012, at N47° E11°

3.5.3.2.3 TEC Map

Amongst others IGS provides global TEC maps that are suitable for computing the vertical ionospheric delay $\Delta\rho_{iono,v,i}$ [58]. The TEC value is the integral of the vertical electron density profile over height

$$N = \int_0^{\infty} n_e(h) dh \quad (3.144)$$

According to (3.138) the vertical path delay is related to the TEC value at the IPP by

$$\Delta\rho_{iono,v} = \frac{40.3}{f_{carrier}^2} N \quad (3.145)$$

Again, the slant delay is calculated from the vertical delay with (B.19). The TEC values are supplied in grids with 5 degree longitudinal, 2.5 degree latitudinal and 2 hour time resolution. The TEC values at the IPP are obtained by linear interpolation of the gridded TEC values in position and time. The TEC maps are generated by fusion of the dual frequency measurements from the worldwide IGS network of stationary GPS receivers. IGS provides two types of maps, a rapid and a final ionospheric TEC grid. The former is available after 24 hours, is daily updated and has an accuracy of 2 – 9 TECU (TEC units). The latter is available after eleven days, is weekly updated and has an accuracy of 2 – 8 TECU. One TECU is defined as 10^{16} electrons per square meter [64]. Considering the L1 signal with frequency $f_{L1} = 1575.42 \text{ MHz}$, the corresponding errors are 0.33 – 1.46 m and 0.33 – 1.30 m, respectively. It can be observed that the difference between the rapid and the final TEC maps is comparatively small. The effective height of the substitute spherical shell above the Earth ellipsoid h_i is assumed to be 350 km. Because of the delayed availability of the TEC maps, the model is less suited for real-time navigation systems but rather for accurate post-processing analysis. For systems that are operated daily without user interactivity frequent uploads of new TEC data are out of question, for example, due to a missing connection to adequate databases. However, for special missions and well-planned events like the launch of a sounding rocket the operating point in time is usually exactly known beforehand and it is imaginable to upload the latest available TEC maps from the previous day. Although changes of the ionosphere are more mid- to long-term by their nature, the variation of the ionospheric delay during the previous days should be carefully monitored because incidents like for example sudden eruptions of the Sun and subsequent solar winds can severely distort the ionospheric signal delay.

Alternatively, SBAS like the North American Wide Area Augmentation System (WAAS) or the European Geostationary Overlay Navigation System (EGNOS) offer gridded TEC values in the navigation message. In principal, the RTCA standard DO-229D [3] specifies the SBAS navigation messages containing ionospheric correction data. These are the message type 18 “Ionospheric Grid Point Masks” and the message type 26 “Ionospheric Delay Corrections”. The adaption to EGNOS can be found in [78]. However, SBAS is not available worldwide but only in the dedicated reception areas.

3.5.3.2.4 Comparison of Klobuchar, NeQuick2 and TEC Map

In order to assess the accuracy, the three presented models are exemplarily analyzed at 06/22/2012. A stationary user with a GPS receiver is supposed to be located at N47° E11° and 0 m height. The Klobuchar model uses the coefficients that were broadcast at that day. Since the Galileo broadcast navigation message had not yet been activated at this date, the monthly mean sunspot number R_{12} is used as input for the NeQuick2 model. For the TEC map model the final IGS TEC maps were chosen. According to Figure 3-64 the monthly mean sunspot number was 75. The daily sunspot number was below 20 and thus very low at this day. The situation is evaluated at two different hours: first one hour after noon at 1pm (UTC) and second in the evening at 7pm (UTC). The corresponding GPS sky plots are given in Figure 3-66 and Figure 3-69.

In Figure 3-67 the ionospheric path delays of the satellites in view are shown for a two hour period beginning at 1pm. It is suggested to take the TEC map results as reference for the other models since their error is expected to be lowest of all three models. It can be observed that all three models match quite well. Although the magnitudes of the NeQuick2 model delays seem to be too low, which is probably due to a higher solar activity than assumed, they correspond qualitatively slightly better to the TEC map model than the Klobuchar delay magnitudes do. In Figure 3-68 the vertical path delays are shown. Obviously, the variability of the delay with time is much smaller. This is because the slant effect is removed. Again the magnitudes of the delays of the Klobuchar and NeQuick2 models seem to be too low. Whereas the TEC map and the Klobuchar model readily provide the vertical delay, the vertical delay of the NeQuick2 model has been computed from the slant delay using (B.15).

Figure 3-70 shows the ionospheric delays and Figure 3-71 the vertical ionospheric delays for 7pm. The magnitudes are smaller than at 1pm. The delays of the NeQuick2 model resemble the TEC map delays more than the Klobuchar model delays. The vertical delays of the Klobuchar model are too low and particularly decrease with progressing time towards dusk, which indicates that the Klobuchar model might have a shortcoming at night.

In order to get a better understanding of the global situation, worldwide maps of the vertical delay at 1pm are given in Figure 3-72 for the TEC map model, in Figure 3-73 for the Klobuchar model and in Figure 3-74 for the NeQuick2 model. The corresponding maps at 7pm are shown in Figure 3-75, Figure 3-76 and Figure 3-77. The NeQuick2 model is capable to represent the typical form of the maximum delay area, where the Sun is in zenith, with two separate maximum zones in the north and in the south. However, the maximum delays seem to be exaggerated suggesting that the sunspot number input parameter is actually lower than originally assumed.

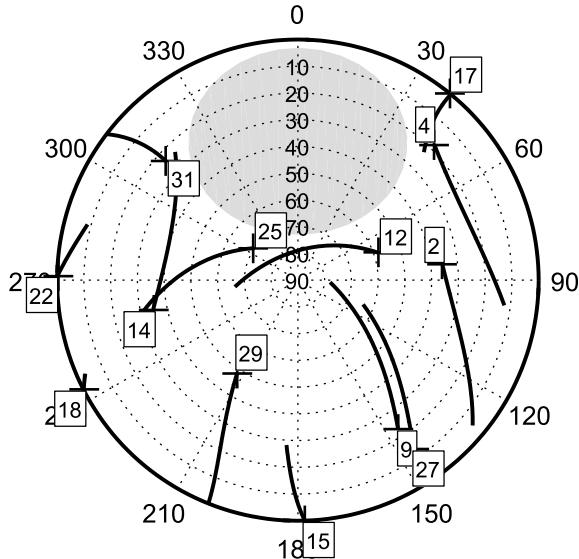
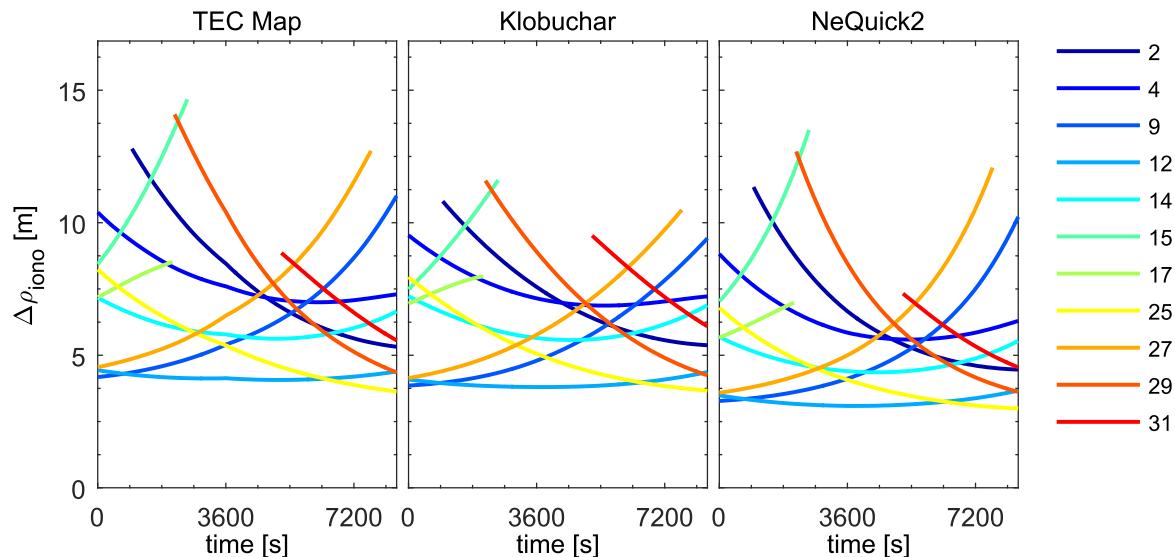
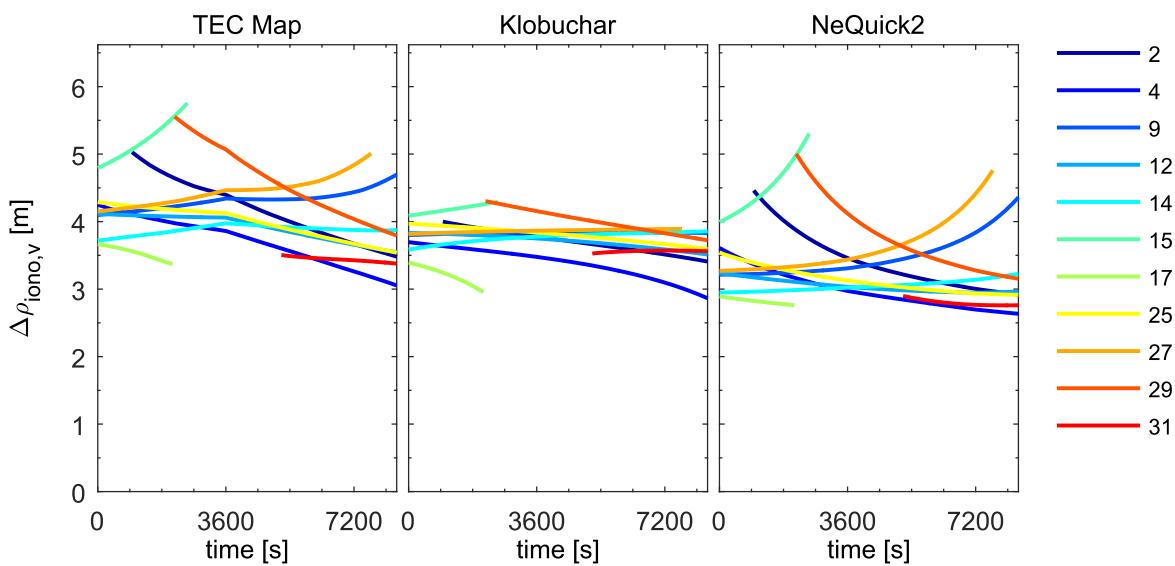


Figure 3-66: GPS sky plot at N47° E11°, 06/22/2012, 1pm (UTC)

Figure 3-67: Comparison of ionospheric delays at N47° E11°, 06/22/2012, 1pm (UTC), $R_{12} = 100$ Figure 3-68: Comparison of vert. ionospheric delays at N47° E11°, 06/22/2012, 1pm (UTC), $R_{12} = 100$

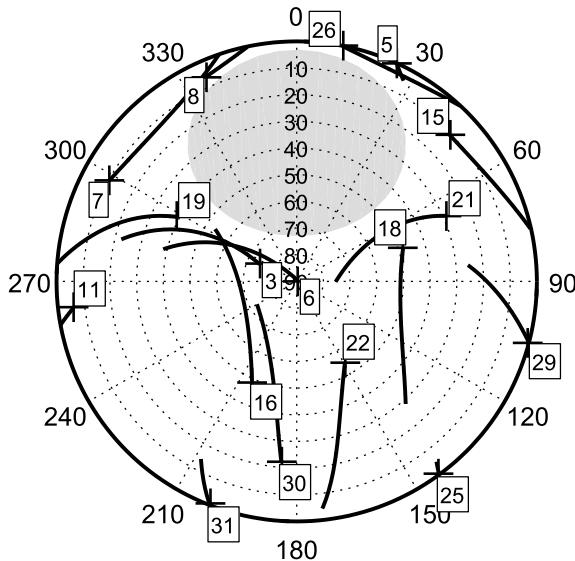


Figure 3-69: GPS sky plot at N47° E11°, 06/22/2012, 7pm (UTC)

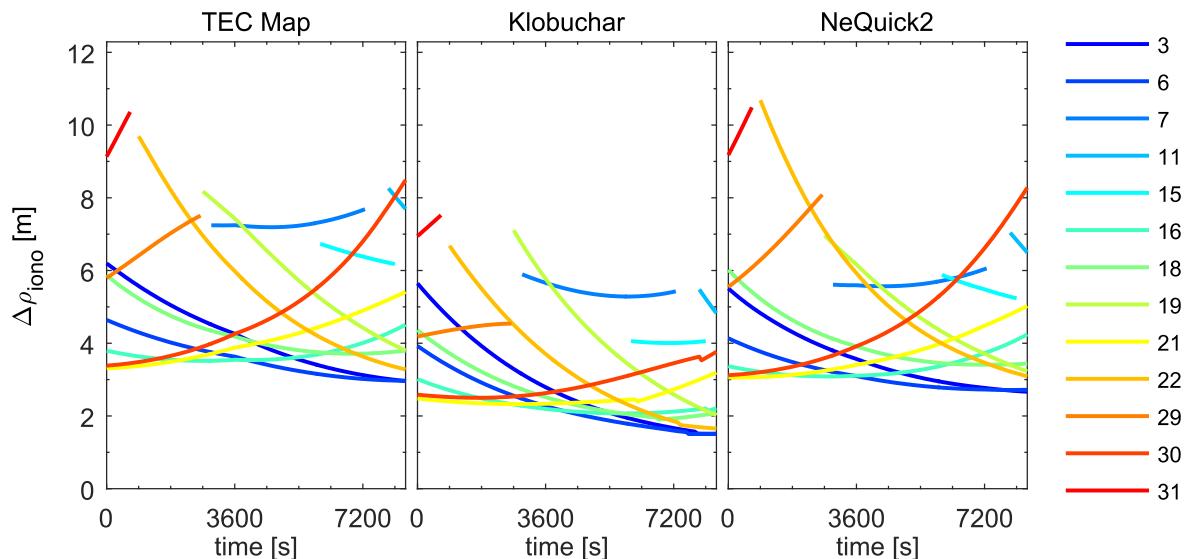


Figure 3-70: Comparison of ionospheric delays at N47° E11°, 06/22/2012, 7pm (UTC), $R_{12} = 100$

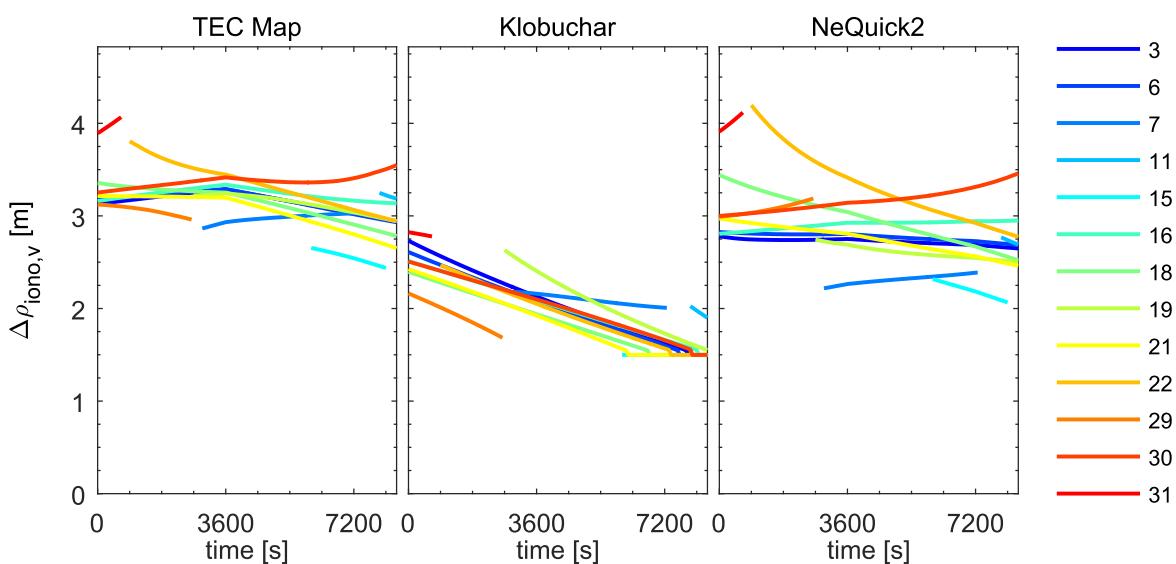


Figure 3-71: Comparison of vert. ionospheric delays at N47° E11°, 06/22/2012, 7pm (UTC), $R_{12} = 100$

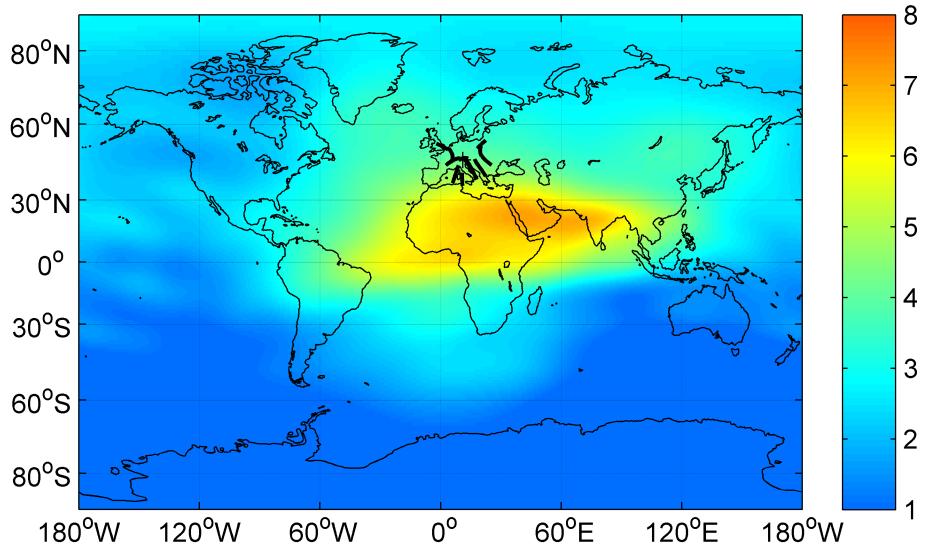


Figure 3-72: Worldwide TEC map vertical ionospheric delay [m] at 06/22/2012, 1pm (UTC)

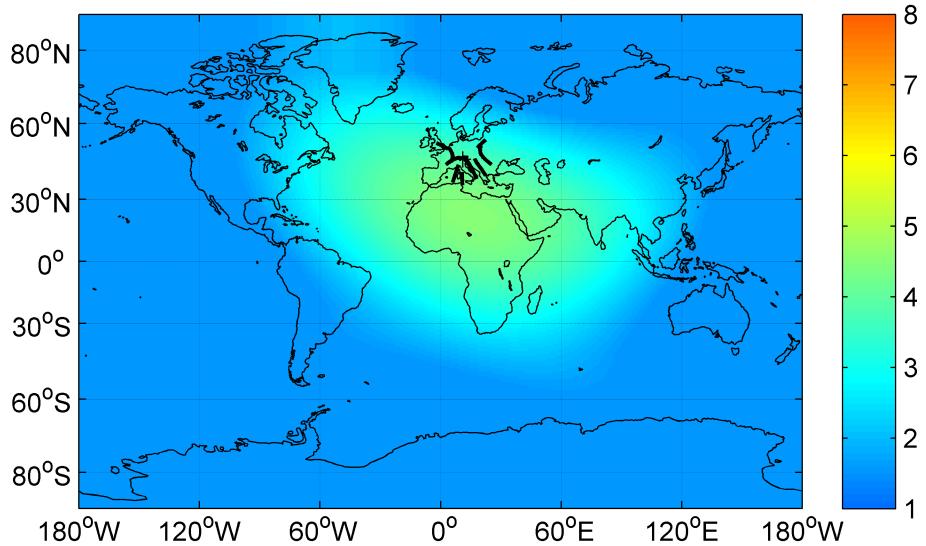


Figure 3-73: Worldwide Klobuchar vertical ionospheric delay [m] at 06/22/2012, 1pm (UTC)

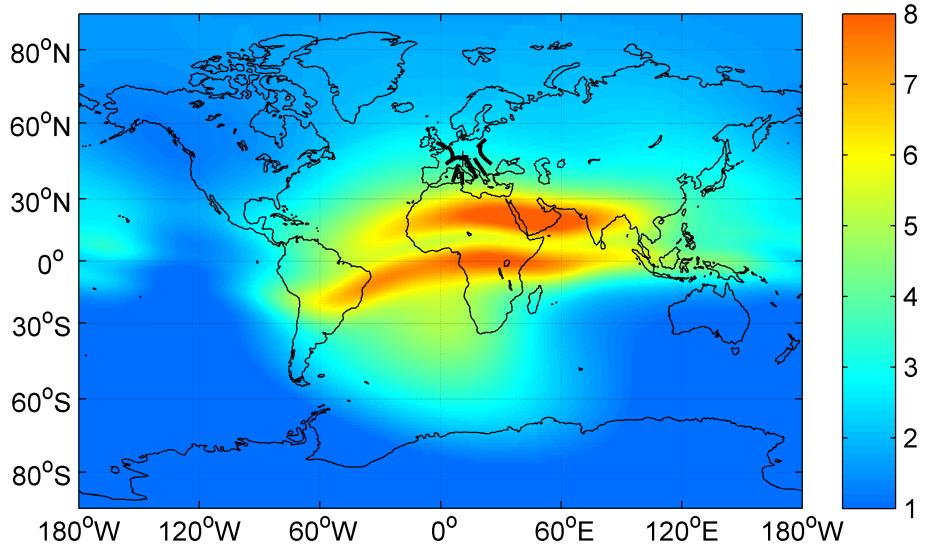


Figure 3-74: Worldwide NeQuick2 vertical ionospheric delay [m] at 06/22/2012, 1pm (UTC), $R_{12} = 100$

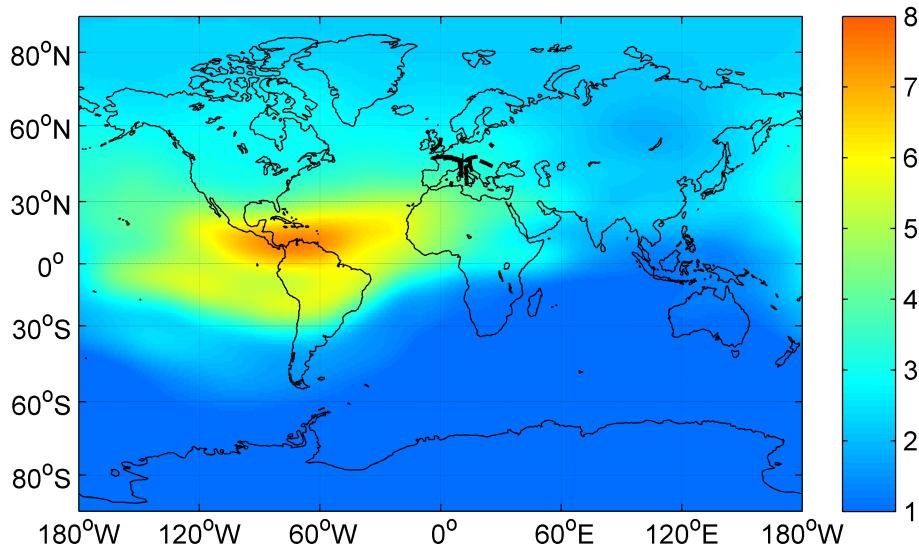


Figure 3-75: Worldwide TEC map vertical ionospheric delay [m] at 06/22/2012, 7pm (UTC)

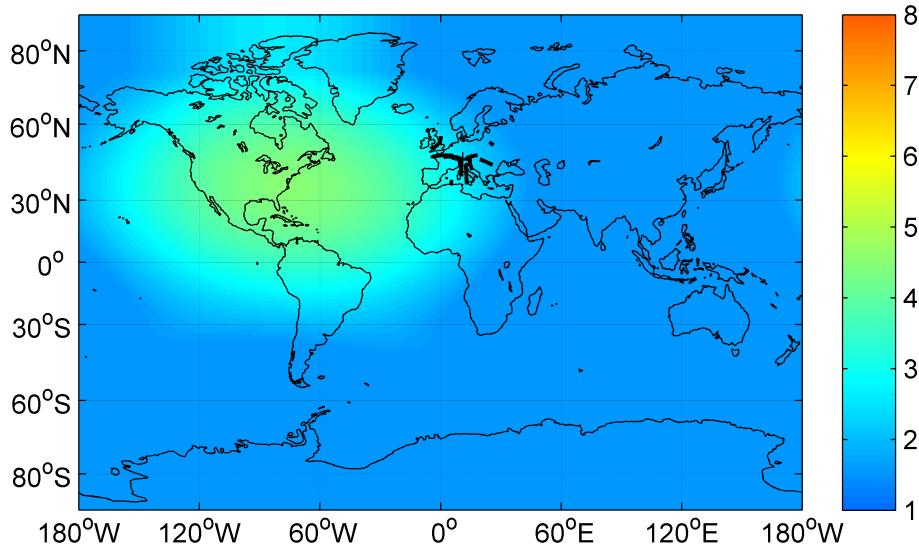


Figure 3-76: Worldwide Klobuchar vertical ionospheric delay [m] at 06/22/2012, 7pm (UTC)

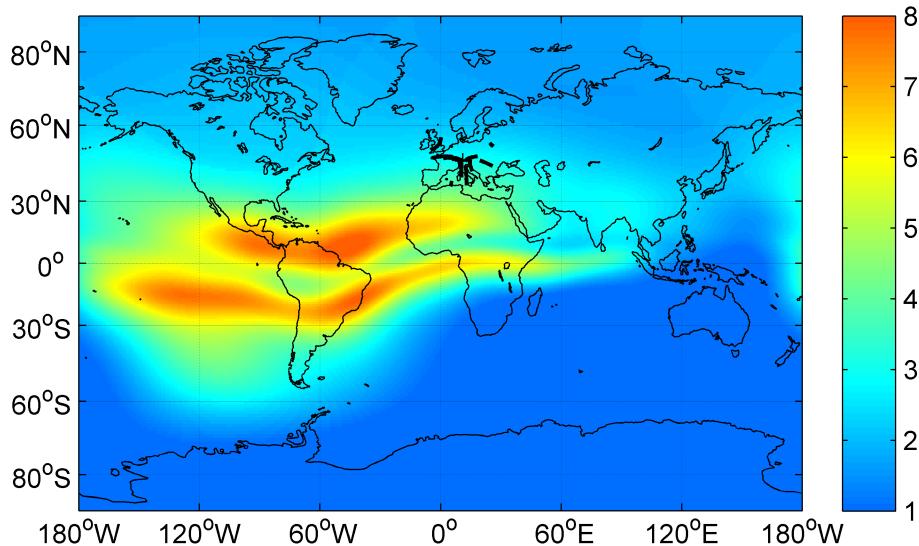


Figure 3-77: Worldwide NeQuick2 vertical ionospheric delay [m] at 06/22/2012, 7pm (UTC), $R_{12} = 100$

In order to get an even better understanding of the magnitudes as well as the spatial and time variability of the ionospheric delay, the flight of the SHEFEX 2 sounding rocket is considered [79]. It took place at June 22, 2012 at 19:18 UTC time (21:18 local time). The launch site was Andenes on the island of Andøya, Norway. Figure 3-78 illustrates the motion of the IPPs during the 480 seconds long flight. The PRN numbers mark the beginning of the pierce point ground tracks at launch. The color depicts the vertical ionospheric delay in meters, which has been calculated from final IGS TEC maps at lift-off time. Since the launch was in the evening and the zenith of the Sun is somewhere in the west, the highest vertical ionospheric delays occur in the lower left corner of the figure.

Figure 3-79 shows the ionospheric delays and Figure 3-80 the corresponding vertical ionospheric delays of the signals from the satellites in view as calculated with the three models. The NeQuick2 model uses again the sunspot number $R_{12} = 100$. As in the previous stationary example the ionospheric delay of the NeQuick2 model seems to be underestimated. Also, as in Figure 3-71, the vertical delay of the Klobuchar model is too low. At lift-off time it has already reached the model's absolute minimum delay of 1.5 m, which corresponds to the vertical delay that is generally assumed by the model for night. An interesting effect can be observed in the delays of the NeQuick2 model that cannot be detected in the delays of the other two models. The vertical delays decrease continuously with time, reach a minimum and increase afterwards again. This is due to the height profile of the flight trajectory. Because of the large flight height of the sounding rocket of up to 170 km, the signal path length through the ionosphere is appreciably shortened and only free electrons in the upper part of the ionosphere affect the satellite signal propagation. This effect can even not be represented by the precise TEC map model. This is an advantage of the NeQuick2 model that works with electron density profiles over height and integrates exactly the electron density along the signal path and a disadvantage of the models that substitute the extent of the ionosphere with height by a thin shell. In order to emphasize the effect of the flight height and changing line-of-sight geometry on the effective ionospheric delays, Figure 3-81 and Figure 3-82, respectively, show the ionospheric delays that are computed for the launch site for the same time. The delays are much less dynamic and all three models yield at least qualitatively similar results. The variability of the vertical delays with time during the 480 seconds long flight is quite low and can be approximated as quasi-constant. Additionally, the differences between the separate vertical delays are small. The vertical delays of the TEC map model lie, for example, between 2.3 m and 2.65 m.

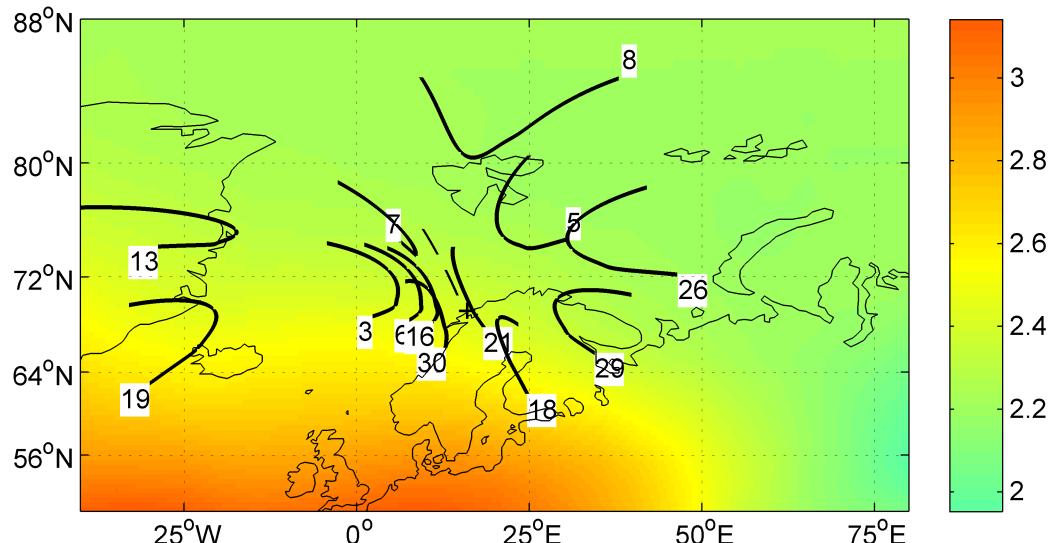


Figure 3-78: Ground tracks of pierce points and TEC map vertical ionospheric delay in meters

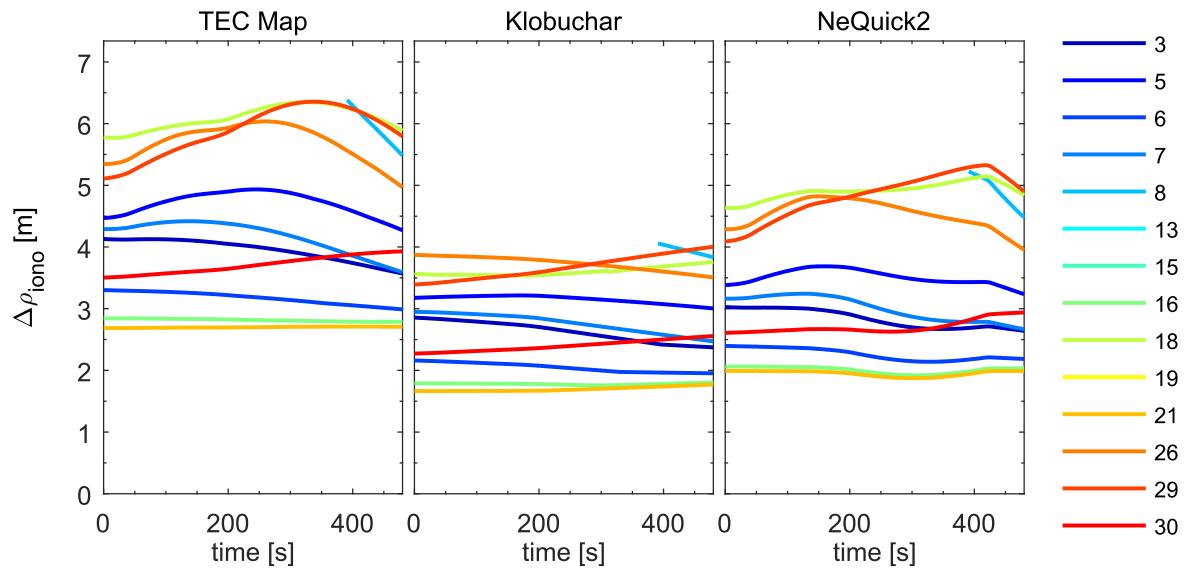


Figure 3-79: Comparison of ionospheric delays along flight trajectory

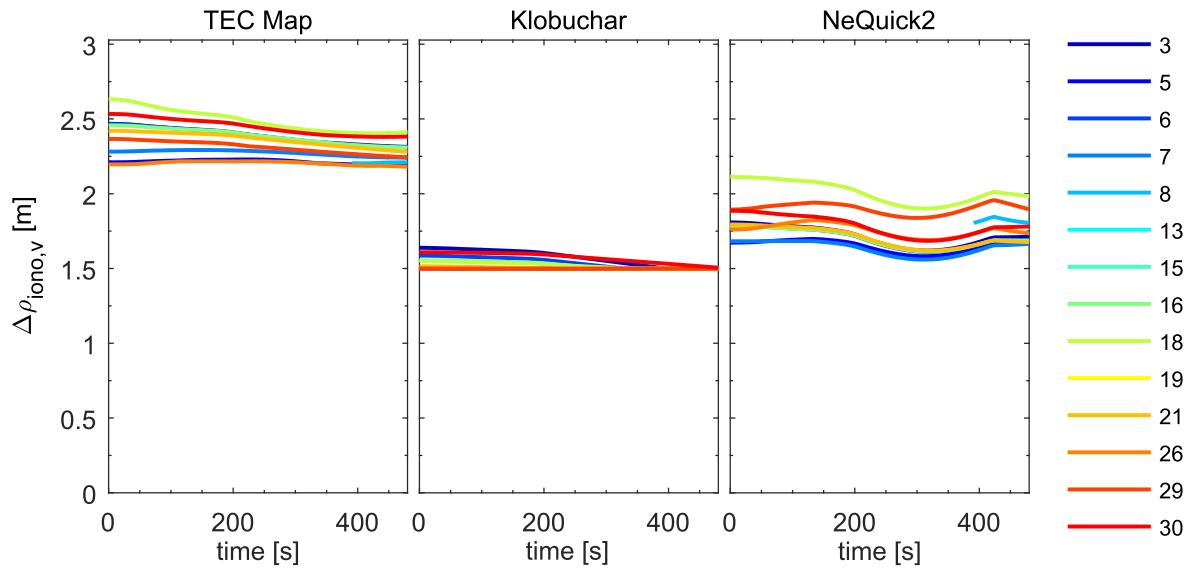


Figure 3-80: Comparison of vertical ionospheric delays along flight trajectory

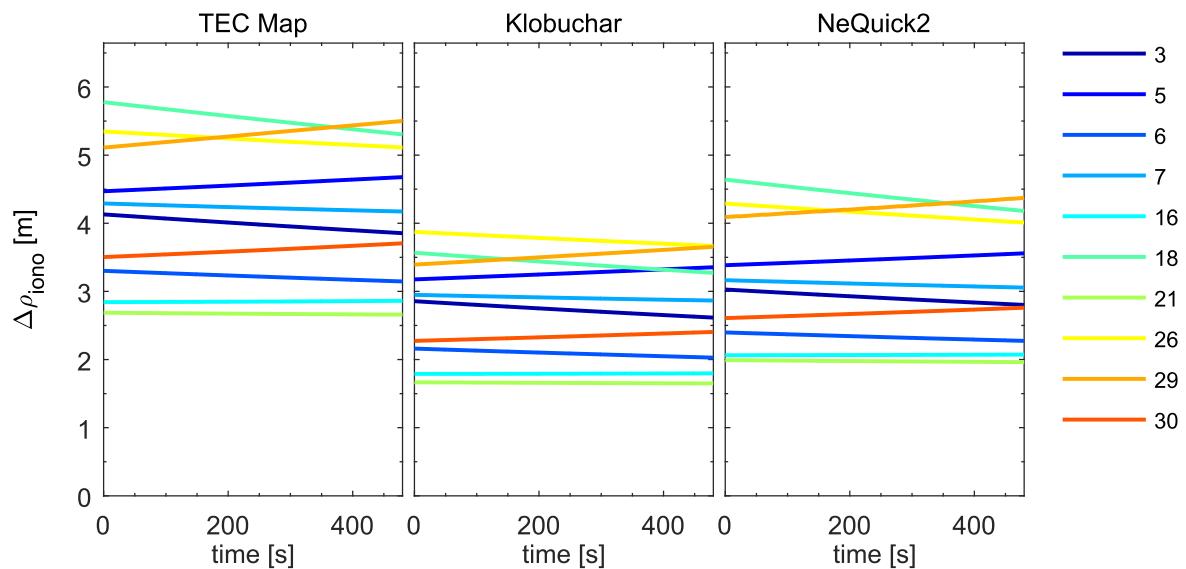


Figure 3-81: Comparison of ionospheric delays at launch site

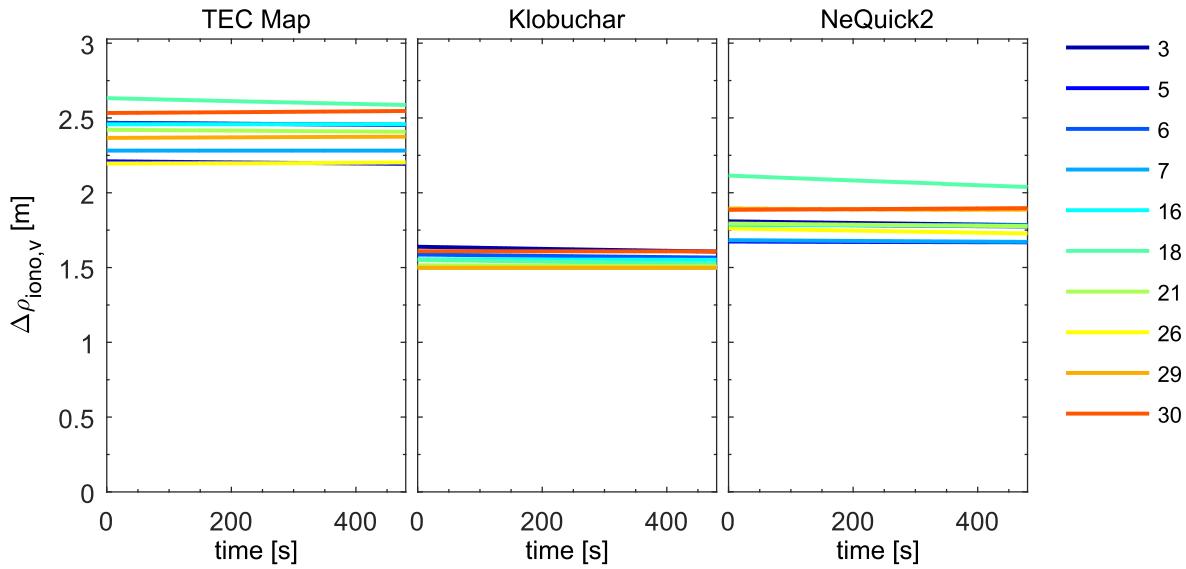


Figure 3-82: Comparison of vertical ionospheric delays at launch site

3.5.3.3 Estimation of the Sunspot Number

As has been revealed in the comparison of the different ionospheric models in section 3.5.3.2.4, there is a good qualitative agreement of the TEC map and the NeQuick2 model. Quantitatively, offsets of some meters between the TEC map and NeQuick2 ionospheric delays have been observed. The magnitude of the NeQuick2 ionospheric delay depends very much on the sunspot number R_{12} input parameter. Therefore, it is proposed to estimate the sunspot number R_{12} by the navigation filter to adapt the NeQuick2 model output to the actual ionospheric situation. This is an obvious approach because the sunspot number R_{12} is also the parameter that is replaced by the effective ionization level A_z in the NeQuickG model and the currently valid value is broadcast in the Galileo navigation message.

As usual, a discrete-time model for the sunspot number error δR_{12} is required for the integration into the navigation error filter. It is assumed that the sunspot number error changes only very slowly with time, such that a simple random walk model is considered adequate

$$\begin{aligned} z_{R_{12},k} &= z_{R_{12},k-1} + \Delta t \eta_{R_{12},k-1} \\ \delta R_{12,k} &= z_{R_{12},k} \end{aligned} \quad (3.146)$$

The power spectral density of the normally distributed, white input noise $\eta_{R_{12}} \sim WN(0, \sigma_{\eta, R_{12}}^2)$ should be selected in the range of one per hour. Furthermore, the Jacobian of the pseudorange error $\delta \rho_i$ with respect to the sunspot number error δR_{12} is necessary for the pseudorange observation error equation

$$\delta \rho_i = (-\mathbf{e}_i^\top \quad 1) \left(\frac{\partial \mathbf{x}}{\partial \delta R_{12}} \right) + \frac{\partial \rho_i}{\partial R_{12}} \Big|_{\delta R_{12}} \delta R_{12} + \varepsilon_\rho \quad (3.147)$$

From the equations of the NeQuick2 model it becomes clear that it is not possible to calculate the Jacobian $\partial \rho_i / \partial R_{12} \Big|_{\delta R_{12}}$ analytically. As workaround, it has to be evaluated numerically. In Figure 3-83, the dependence of the TEC value (and thus of the ionospheric delay on the pseudorange measurement) on the sunspot number is exemplarily illustrated for a stationary receiver at N47° E11° and 0 m height at June 22, 2012 for different times of day. It can be clearly seen that the curves are nearly linear without extreme values and there is nothing to be said against approximating the Jacobian

by a difference quotient. The NeQuick2 model provides a mathematically non-linear function of the sunspot number R_{12} for the ionospheric delay $\Delta\rho_{iono}$

$$\Delta\tilde{\rho}_{iono} = f(\tilde{R}_{12}) \quad (3.148)$$

The wanted difference quotient

$$\frac{\partial\rho}{\partial R_{12}} \approx \frac{f(\tilde{R}_{12} + 10) - f(\tilde{R}_{12} - 10)}{20} \quad (3.149)$$

is hence obtained by evaluating the NeQuick2 model for the sunspot numbers $\tilde{R}_{12} + 10$ and $\tilde{R}_{12} - 10$. The initial value $\tilde{R}_{12}(t_0)$ could be taken from Figure 3-63. The initial covariance of the sunspot number error should be set to a value in the range of one to two decades, for example

$$E[\delta R_{12}^2(t_0)] = 10^2 \quad (3.150)$$

In contrast to the estimation of the vertical ionospheric delays of the single IPPs, only one additional state instead of the number of parallel receiver channels has to be estimated by the navigation filter. Furthermore, the user does not have to worry about the correlation of the separate vertical delays. In terms of numerical efficiency, this approach is hence promising. It will be further analyzed in section 4.6.6.

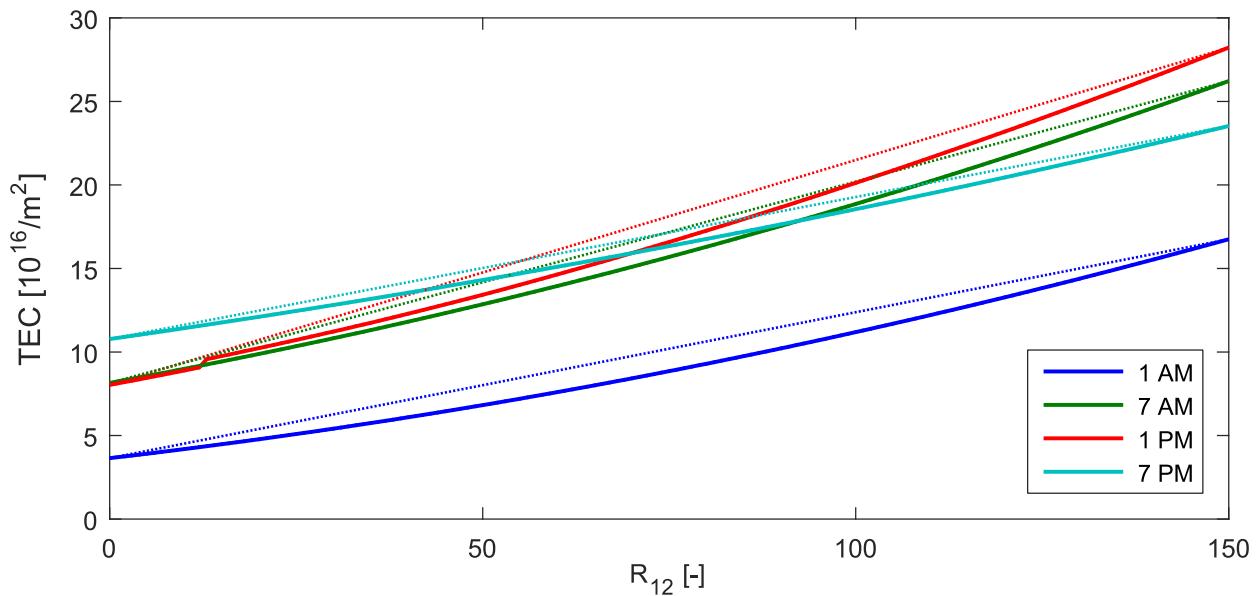


Figure 3-83: Dependence of the TEC value on the sunspot number R_{12} at N47° E11° and 0 m height, 06/22/2012. The dotted curves are linear and serve as reference

3.5.4 Troposphere Error

3.5.4.1 Motivation

The troposphere is the lowest layer of the atmosphere of the Earth. It contains about 90% of the air and almost all water vapor of the atmosphere. At mid latitudes the troposphere ranges from ground to about eleven kilometers height.

Analogous to the free electrons of the ionosphere the refraction and hence the propagation speed of the satellite signals is reduced by the air and water molecules in the troposphere. In contrast to the ionosphere, the effect is non-dispersive in the frequency range of GNSS signals and influences the carrier wave and spreading code propagation likewise. The pseudorange delay can range from about 2.5 m for satellites at zenith up to 25 m for satellites at elevation angles of five degrees [64]. It is distinct between the hydrostatic delay caused by the dry air and the wet delay caused by the water vapor contained in the air for example as clouds.

Usually, there are different models for the hydrostatic and wet delays. Most often, the zenith delays are computed at first and are subsequently projected onto the actual line of sight by means of mapping functions to account for the slant effect, which is the longer travel time through the troposphere of signals from lower elevated satellites. Models and mapping functions consider seasonal effects or expect the actual temperature and pressure as inputs.

The hydrostatic delay can be well modeled and almost completely removed, for example, with the Saastamoinen zenith delay model [80] and the Niell mapping function [81]. The wet delay, however, is more difficult to describe since it mainly depends on turbulent fluctuations of the water vapor with eddies of different scales, which may strongly vary locally and with time. This effect cannot be adequately accommodated by a simple, global model. Figure 3-84 exemplarily shows the height profiles of the temperature, air pressure, wind speed and relative humidity measured by four German radiosondes at Stuttgart ($N48.83^\circ E9.20^\circ$, station number 10739), Kümmersbruck ($N49.43^\circ E11.90^\circ$, station id ETGK, station number 10771), Idar-Oberstein ($N49.70^\circ E7.33^\circ$, station id ETGI, station number 10618) and Meiningen ($N50.56^\circ E10.38^\circ$, station number 10548) at noon of April 23, 2013. It can be observed that the static values, that is the temperature and the air pressure, are very similar at all four locations. Even the wind speed profiles are similar. The relative humidity, however, varies much over height and location.

In section B.4.1 in the appendix, common zenith delay models are presented. The required temperature, static pressure and water vapor values of the standard atmosphere are given in section B.4.2. Common mapping functions to account for the slant effect are summarized in section B.4.3. In contrast to the ionosphere, the pierce points of the line of sights to the satellites in view through the troposphere are in close vicinity of the user antenna. It can be assumed that all pseudorange measurements are affected by the same zenith delay. The single tropospheric delays solely differ because of the different obliquity factors.

The tropospheric residual error after application of the models for the hydrostatic and wet delays on the pseudorange measurements is mainly due to the non-modeled water vapor fluctuations, has stochastic nature and can be described by a spatiotemporal stochastic process. The magnitude of the residual delay is in the range of ten centimeters for satellites in zenith and up to 1.20 m for low satellites.

In this section, a stochastic model is derived that represents the time correlation of the tropospheric residual error and that can be integrated into the navigation error filter. The stochastic model bases upon the spatial structure function of atmospheric turbulence. The spatial stochastic process is converted to a temporal process using the frozen troposphere model assumption. The corresponding power spectral densities are derived and from that, an adequate ARMA model is searched.

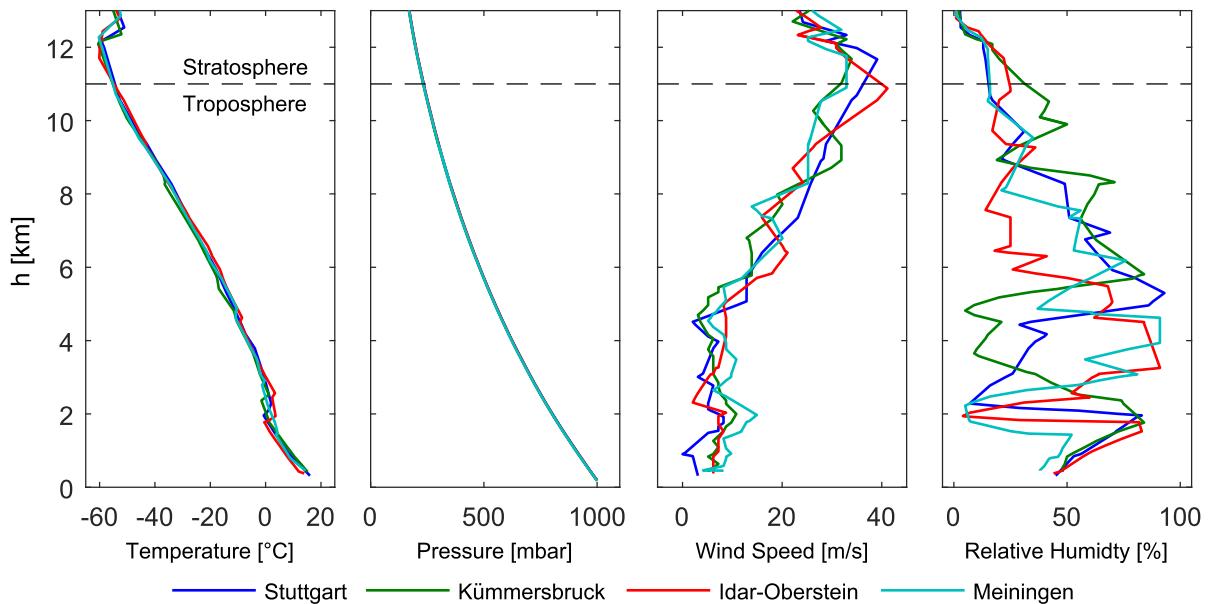


Figure 3-84: Atmospheric data from radiosondes in Germany at 04/23/2013 [82]

3.5.4.2 Characteristic

The tropospheric zenith path delay is exemplarily shown for Warnemünde in the year 2012 in Figure 3-85. Warnemünde is part of the IGS reference receiver network (id: warn). Besides the satellite and ionosphere products, IGS provides estimated tropospheric zenith path delays at the locations of the monitoring receiver stations sampled in five minute intervals. The accuracy is in the range of some millimeters and the IGS zenith path delays are thus close to reality. Next to the IGS zenith path delay, the modeled wet, hydrostatic and combined delays are plotted, using the UNB3 model. The hydrostatic delay is about 2.3 m and does not vary much over the year. The wet delay is smaller than two decimeters and features a stronger seasonal dependence. It is largest in summer. It can be seen that the UNB3 model represents the actual tropospheric zenith path delay quite well. In Figure 3-86, the difference between the IGS and the UNB3 tropospheric zenith path delay is shown for the year 2012. The standard deviation is in the range of some centimeters.

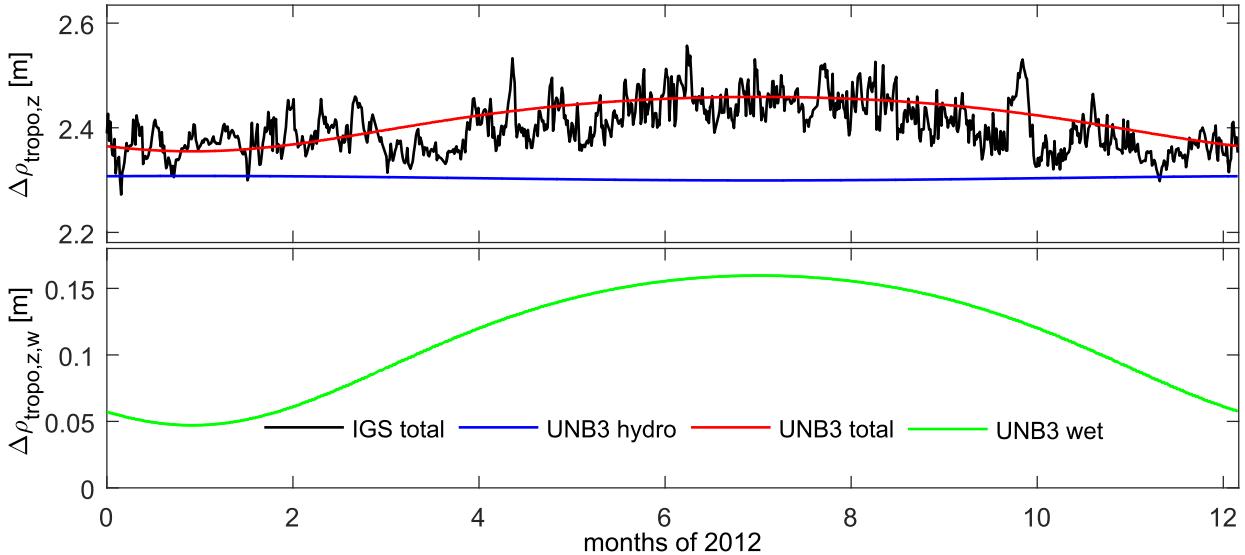


Figure 3-85: IGS and UNB3 tropospheric zenith path delay

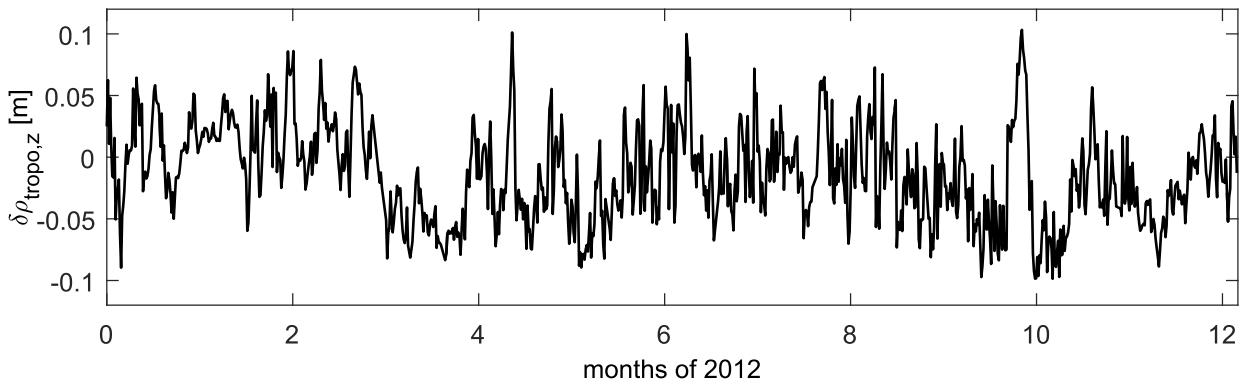


Figure 3-86: Residual tropospheric zenith path delay at Warnemünde at 2012

3.5.4.3 Power Spectral Density of the Tropospheric Residual Error

In section B.4.4 in the appendix the atmospheric turbulence structure function is given. It is converted to an autocorrelation function. The autocorrelation function of the zenith delay, which corresponds to the structure function (B.50), is according to (B.44)

$$R_{\delta\rho_{tropo,z}}(\tau) = R_{\delta\rho_{tropo,z}}(0) - \frac{1}{2}D_{\delta\rho_{tropo,z}}(\tau) \quad (3.151)$$

The (two-sided) PSD function $S_{\delta\rho_{tropo,z}}(\omega)$ is obtained by Fourier transformation of the autocorrelation function. The autocorrelation function is an even function. Using the linearity property and a table of Fourier (cosine) transform pairs, the PSD as function of the angular frequency ω is given by

$$S_{\delta\rho_{tropo,z}}(\omega) = 2\pi R_{\delta\rho_{tropo,z}}(0) \delta(\omega) + C_{\delta\rho_{tropo,z}}^2 \left(\frac{V_{A,h}}{H} \right)^p \Gamma(p+1) \sin\left(\frac{\pi}{2} p\right) |\omega|^{-(p+1)} \quad (3.152)$$

[83]. Γ is the gamma function. The PSD of the small scale eddies $S_{\delta\rho_{tropo,z,s}}(\omega)$ with $p = \frac{5}{3}$ is, omitting the term with the Dirac function that contributes only for zero frequency,

$$S_{\delta\rho_{tropo,z,s}}(\omega) = C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h}}{H} \right)^{\frac{5}{3}} \Gamma\left(\frac{8}{3}\right) \sin\left(\frac{5\pi}{6}\right) |\omega|^{-\frac{8}{3}} \quad (3.153)$$

The PSD function of the large scale eddies $S_{\delta\rho_{tropo,z,l}}(\omega)$ with $p = \frac{2}{3}$ is accordingly described by

$$S_{\delta\rho_{tropo,z,l}}(\omega) = C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h}}{H} \right)^{\frac{2}{3}} \Gamma\left(\frac{5}{3}\right) \sin\left(\frac{\pi}{3}\right) |\omega|^{-\frac{5}{3}} \quad (3.154)$$

Before the PSDs of the small and large scale noises can be summed, the small scale noise has to be high-pass filtered to cut off the low frequencies whereas the large scale noise has to be low-pass filtered to cut off the high frequencies. Simple first order filters should be sufficient. The PSD of the small scale noise is then, multiplied by the squared magnitude of the transfer function of the high-pass filter in frequency domain

$$S_{\delta\rho_{tropo,z,s}}(\omega) = \underbrace{\frac{T_c^2 \omega^2}{4\pi^2 + T_c^2 \omega^2}}_{\text{High-pass filter } |H_{HPF}(\omega)|^2} \cdot C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h}}{H} \right)^{\frac{5}{3}} \Gamma\left(\frac{8}{3}\right) \sin\left(\frac{5\pi}{6}\right) |\omega|^{-\frac{8}{3}} \quad (3.155)$$

and of the large scale noise

$$S_{\delta\rho_{tropo,z,l}}(\omega) = \underbrace{\frac{4\pi^2}{4\pi^2 + T_c^2 \omega^2}}_{\text{Low-pass filter } |H_{LPF}(\omega)|^2} \cdot C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h}}{H} \right)^{\frac{2}{3}} \Gamma\left(\frac{5}{3}\right) \sin\left(\frac{\pi}{3}\right) |\omega|^{-\frac{5}{3}} \quad (3.156)$$

The cut-off time constant of the filters is chosen as

$$T_c = \frac{H}{v_{A,h}} \quad (3.157)$$

The superposed PSD functions contain the small scale as well as large scale effects and are illustrated in Figure 3-87

$$S_{\delta\rho_{tropo,z}}(\omega) = S_{\delta\rho_{tropo,z,s}}(\omega) + S_{\delta\rho_{tropo,z,l}}(\omega) \quad (3.158)$$

The dotted lines represent the PSDs of the unfiltered small and large scale noises.

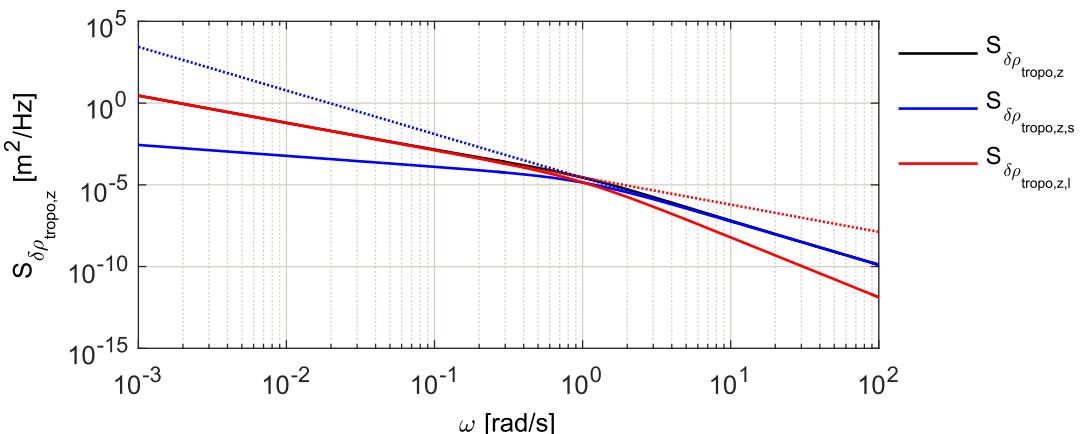


Figure 3-87: Large scale, small scale and overall PSDs of the tropospheric zenith residual error

3.5.4.4 Model

In this section a discrete-time model is searched that shapes white Gaussian input noise in such a way that the PSD of its output approximates the characteristic shown in Figure 3-87 as best as possible. Since the single noise components follow power-law, the method that has been applied for the derivation of the flicker noise model in section 3.1.4 can here be used again.

In the end, there is a state-space model for the small scale eddies with power $\beta = 8/3$

$$\begin{aligned} \mathbf{z}_{s,k+1} &= \Phi_s \mathbf{z}_{s,k} + \Gamma_s \eta_{s,k} \\ \delta\rho_{tropo,z,s,k} &= \mathbf{H}_s \mathbf{z}_{s,k} + D_s \eta_{s,k} \end{aligned} \quad (3.159)$$

of size p with the system matrices and input noise

$$\begin{aligned} \Phi_s &= \begin{pmatrix} 0 & 1 & 0 & \dots \\ 0 & \ddots & \ddots & 0 \\ & & 0 & 1 \\ a_{s,p} & \dots & a_{s,2} & a_{s,1} \end{pmatrix}, \quad \Gamma_s = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \quad D_s = \Delta t^{\frac{4}{3}} b_{s,0} \\ \mathbf{H}_s &= \Delta t^{\frac{4}{3}} (b_{s,p} + b_{s,0} a_{s,p}, \dots, b_{s,1} + b_{s,0} a_{s,1}) \\ \eta_s &\sim WN \left(0, \frac{C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h}}{H} \right)^{\frac{5}{3}} \Gamma(\frac{8}{3}) \sin(\frac{5\pi}{6})}{\Delta t} \right) \end{aligned} \quad (3.160)$$

and a state-space model for the large scale eddies with power $\beta = 5/3$

$$\begin{aligned} \mathbf{z}_{l,k+1} &= \Phi_l \mathbf{z}_{l,k} + \Gamma_l \eta_{l,k} \\ \delta\rho_{tropo,z,l,k} &= \mathbf{H}_l \mathbf{z}_{l,k} + D_l \eta_{l,k} \end{aligned} \quad (3.161)$$

of size p with the system matrices and input noise

$$\begin{aligned} \Phi_l &= \begin{pmatrix} 0 & 1 & 0 & \dots \\ 0 & \ddots & \ddots & 0 \\ & & 0 & 1 \\ a_{l,p} & \dots & a_{l,2} & a_{l,1} \end{pmatrix}, \quad \Gamma_l = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \quad D_l = \Delta t^{\frac{5}{6}} b_{l,0} \\ \mathbf{H}_l &= \Delta t^{\frac{5}{6}} (b_{l,p} + b_{l,0} a_{l,p}, \dots, b_{l,1} + b_{l,0} a_{l,1}) \\ \eta_l &\sim WN \left(0, \frac{C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h}}{H} \right)^{\frac{2}{3}} \Gamma(\frac{5}{3}) \sin(\frac{\pi}{3})}{\Delta t} \right) \end{aligned} \quad (3.162)$$

The ARMA model coefficients for the small scale noise, which have been estimated as described in section 3.1.4.2, are listed in Table 3-29 and Table 3-30 for orders of up to four. The ARMA model coefficients for the large scale noise are given in Table 3-31 and Table 3-32. The PSDs of the noise that has been created with the models (3.159) and (3.161) with the coefficients as given in the tables is shown in Figure 3-88. It can be seen that there is a good accordance between the PSDs even for low orders.

Table 3-29: Estimated small scale noise AR coefficients ($\beta/2 = 4/3$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 5000$)

$p = q$	a_1	a_2	a_3	a_4
1	1.00000000000000			
2	1.896312815819	-0.896312815819		
3	2.514836545091	-2.043199403568	0.528362858477	
4	3.081137891912	-3.400849974586	1.556351190900	-0.236639108227

Table 3-30: Estimated small scale noise MA coefficients ($\beta/2 = 4/3$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 5000$)

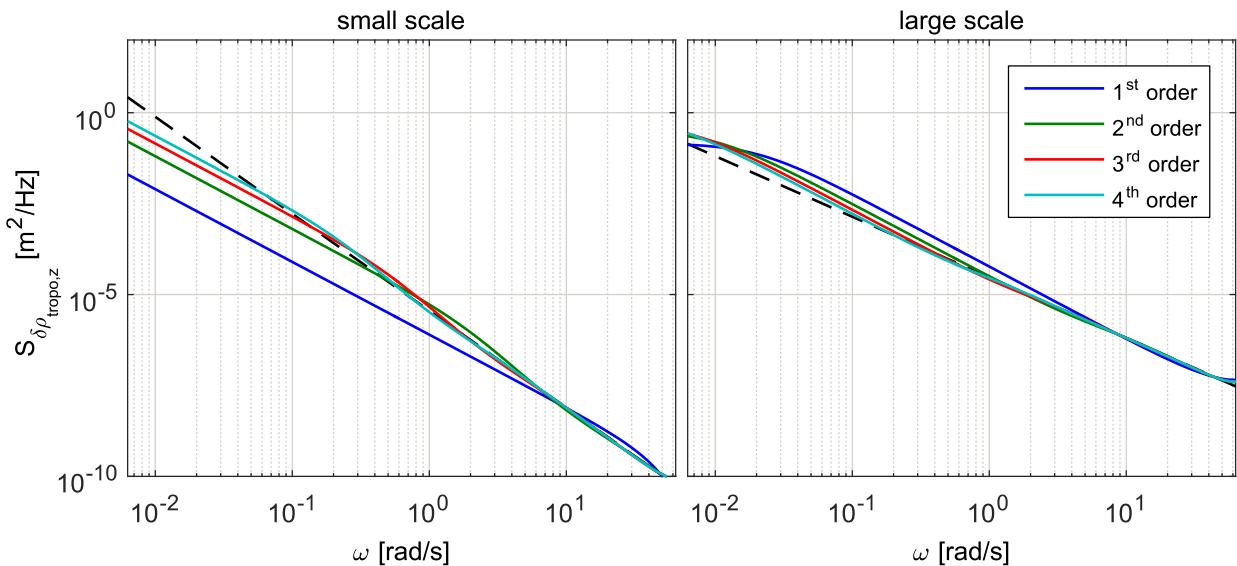
$p = q$	b_0	b_1	b_2	b_3	b_4
1	0.836907392	0.438241290			
2	0.836907392	-0.311884971	-0.150097990		
3	0.836907392	-0.829525253	0.021027395	0.044867567	
4	0.836907392	-1.303459121	0.435125934	0.056181761	-0.011406248

Table 3-31: Estimated large scale AR coefficients ($\beta/2 = 5/6$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 5000$)

$p = q$	a_1	a_2	a_3	a_4
1	0.998991567178			
2	1.652113804552	-0.652285982097		
3	2.208253703943	-1.499586410542	0.291306160245	
4	2.748152964506	-2.610275499932	0.959021480961	-0.096902902531

Table 3-32: Estimated large scale MA coefficients ($\beta/2 = 5/6$, $k_1 = \frac{1}{8}$, $k_2 = \frac{7}{8}$, $N = 5000$)

$p = q$	b_0	b_1	b_2	b_3	b_4
1	0.894691636	-0.041702127			
2	0.894691636	-0.626045130	-0.053214229		
3	0.894691636	-1.123618847	0.230978647	0.040578924	
4	0.894691636	-1.606662200	0.764661598	-0.026644184	-0.018360797

**Figure 3-88: Approximated small and large scale PSDs (order 1 – 4)**

Next, the short distance noise has to be high-pass filtered and the large distance noise has to be low-pass filtered, respectively. The continuous-time transfer function of the first order high-pass filter is

$$H_{HPF}(s) = \frac{T_c s}{2\pi + T_c s} \quad (3.163)$$

The continuous-time transfer function is converted into a discrete-time one by means of the bilinear transform

$$s = \frac{2}{\Delta t} \frac{1-z^{-1}}{1+z^{-1}} \quad (3.164)$$

where Δt is the sample time. The discrete-time transfer function of the high-pass filter is thus

$$\begin{aligned} H_{HPF}(z) &= \frac{b_{0,HPF} + b_{1,HPF} z^{-1}}{1 - a_{1,HPF} z^{-1}} \\ a_{1,HPF} &= -\frac{1 - \frac{T_c}{2\pi} \frac{2}{\Delta t}}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}}, \quad b_{0,HPF} = \frac{\frac{T_c}{2\pi} \frac{2}{\Delta t}}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}}, \quad b_{1,HPF} = -\frac{\frac{T_c}{2\pi} \frac{2}{\Delta t}}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}} \end{aligned} \quad (3.165)$$

and consequently the ARMA model of the high-pass filter

$$\begin{aligned} z_{HPF,k+1} &= a_{1,HPF} z_{HPF,k} + u_k \\ y_k &= (b_{1,HPF} + b_{0,HPF} a_{1,HPF}) z_{HPF,k} + b_{0,HPF} u_k \end{aligned} \quad (3.166)$$

The state-space model (3.160) and the high-pass filter are connected in series, yielding the overall model for the short distance noise

$$\begin{aligned} \begin{pmatrix} \mathbf{z}_{s,k+1} \\ z_{HPF,k+1} \end{pmatrix} &= \begin{pmatrix} \Phi_s & 0 \\ \mathbf{H}_s & a_{1,HPF} \end{pmatrix} \begin{pmatrix} \mathbf{z}_{s,k} \\ z_{HPF,k} \end{pmatrix} + \begin{pmatrix} \Gamma_s \\ D_s \end{pmatrix} \eta_{s,k} \\ \delta p_{tropo,z,s,k} &= (b_{0,HPF} \mathbf{H}_s + b_{1,HPF} + b_{0,HPF} a_{1,HPF}) \begin{pmatrix} \mathbf{z}_{s,k} \\ z_{HPF,k} \end{pmatrix} + b_{0,HPF} D_s \eta_{s,k} \end{aligned} \quad (3.167)$$

The continuous-time transfer function of the low-pass filter is given with

$$H_{LPF}(s) = \frac{2\pi}{2\pi + T_c s} \quad (3.168)$$

With the bilinear transform (3.164), the corresponding discrete-time transfer function becomes

$$\begin{aligned} H_{LPF}(z) &= \frac{b_{0,LPF} + b_{1,LPF} z^{-1}}{1 - a_{1,LPF} z^{-1}} \\ a_{1,LPF} &= -\frac{1 - \frac{T_c}{2\pi} \frac{2}{\Delta t}}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}}, \quad b_{0,LPF} = \frac{1}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}}, \quad b_{1,LPF} = \frac{1}{1 + \frac{T_c}{2\pi} \frac{2}{\Delta t}} \end{aligned} \quad (3.169)$$

and finally the ARMA model of the low-pass filter

$$\begin{aligned} z_{LPF,k+1} &= a_{1,LPF} z_{LPF,k} + u_k \\ y_k &= (b_{1,LPF} - b_{0,LPF} a_{1,LPF}) z_{LPF,k} + b_{0,LPF} u_k \end{aligned} \quad (3.170)$$

The state-space model (3.162) and the low-pass filter are connected in series, yielding the overall model for the large distance noise

$$\begin{aligned} \begin{pmatrix} \mathbf{z}_{I,k+1} \\ \mathbf{z}_{LPF,k+1} \end{pmatrix} &= \begin{pmatrix} \Phi_I & 0 \\ \mathbf{H}_I & a_{1,LPF} \end{pmatrix} \begin{pmatrix} \mathbf{z}_{I,k} \\ \mathbf{z}_{LPF,k} \end{pmatrix} + \begin{pmatrix} \Gamma_I \\ D_I \end{pmatrix} \eta_{I,k} \\ \delta\rho_{tropo,z,I,k} &= (b_{0,LPF} \mathbf{H}_I - b_{1,LPF} a_{1,LPF}) \begin{pmatrix} \mathbf{z}_{I,k} \\ \mathbf{z}_{LPF,k} \end{pmatrix} + b_{0,LPF} D_I \eta_{I,k} \end{aligned} \quad (3.171)$$

The outputs of both models (3.167) and (3.171) are superposed

$$\delta\rho_{tropo,z,k} = \delta\rho_{tropo,z,s,k} + \delta\rho_{tropo,z,I,k} \quad (3.172)$$

in order to obtain the complete tropospheric zenith residual error. Finally, the zenith delay has to be mapped onto the single line-of-sight directions. For that, the same mapping function as for the deterministic wet delay model can be used

$$\delta\rho_{tropo,k} = m_w(\zeta) \delta\rho_{tropo,z,k} \quad (3.173)$$

The derived model can be simplified if the model for the short distance regime with $r < H$ is omitted. It has to be noted that an adequate dampening of the tropospheric wet delay error with height has to be additionally introduced in order to account for the decreasing water vapor in the air with increasing height. The tropospheric wet delay residual error is illustrated by means of an example. Figure 3-89 shows an exemplary realization of the stochastic wet delay for $v = 10 \text{ m/s}$ and Figure 3-90 the PSD. The solid lines belong to the periodograms that have been calculated from the actual realization. The dotted lines are the theoretical PSD curves. Figure 3-91 and Figure 3-92 are the corresponding figures of the stochastic wet delay for $v = 250 \text{ m/s}$. 4th order models have been used for the simulation. Due to the comparatively low cost-benefit ratio of the derived model it will not pursued and not integrated into navigation filter.

3.5.4.5 SBAS Tropospheric Error Standard Deviation

The RTCA standard DO-229D [3] and RTCA standard DO-316 [7] suggest to model the tropospheric zenith residual error as white Gaussian noise

$$\delta\rho_{tropo,z,k} = \eta_{\delta\rho_{tropo,z,k}} \quad (3.174)$$

where $\eta_{\delta\rho_{tropo,z}} \sim WN(0, \sigma_{TVE}^2)$ with the standard deviation

$$\sigma_{TVE} = 0.12 \text{ m} \quad (3.175)$$

In the test procedures for tightly integrated GPS/inertial systems in appendix R in [7] it is recommended to model the tropospheric residual error as first-order Gauss-Markov process with a correlation length of 30 minutes to adequately represent the pass-through of a typical storm system [GPS 297].

The statistical distribution of the residual error of the UNB3 tropospheric model has been thoroughly analyzed in [84]. It is concluded that the error can be assumed to be normally distributed with a standard deviation that depends on the current tropospheric wet delay

$$\sigma_{TVE} = 0.01 \left(3.22 + \frac{\delta\rho_{w,z}}{3.53} \right) [\text{m}] \quad (3.176)$$

However, no statement is made about the time correlation of the residual error.

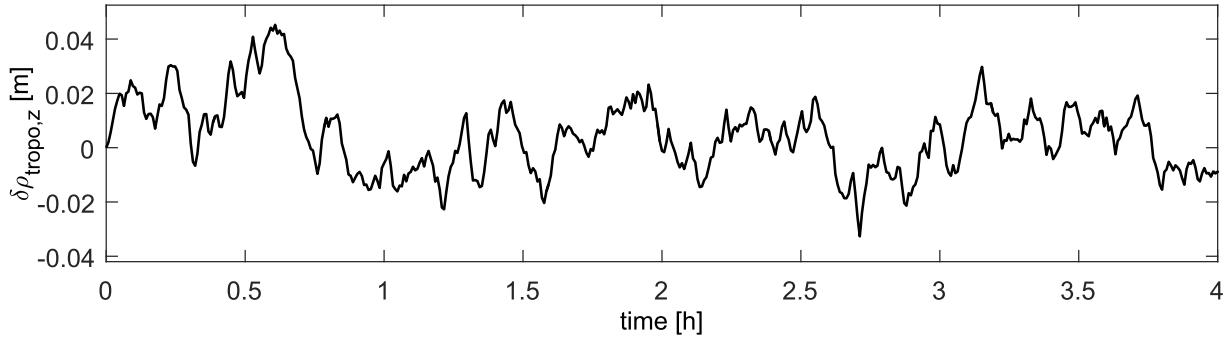


Figure 3-89: Tropospheric wet delay residual error realization for $v = 10 \text{ m/s}$

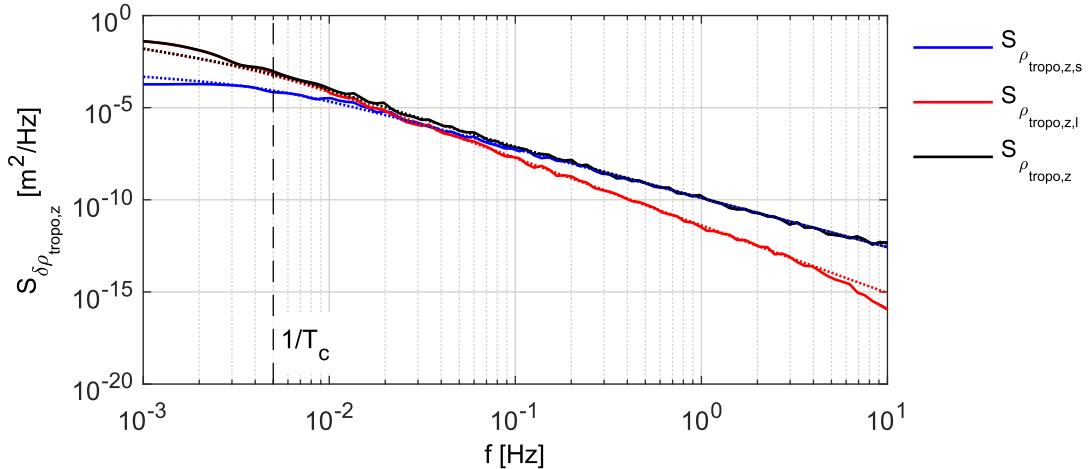


Figure 3-90: PSD of the tropospheric wet delay residual error for $v = 10 \text{ m/s}$

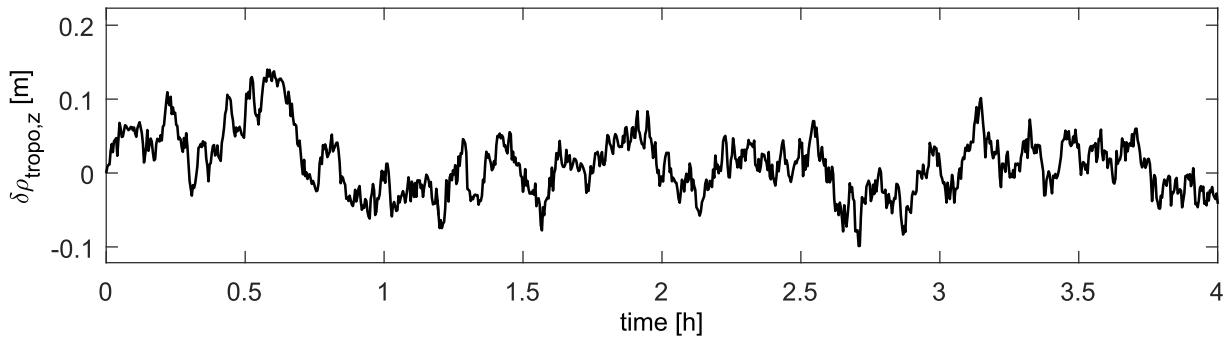


Figure 3-91: Tropospheric wet delay residual error realization for $v = 250 \text{ m/s}$

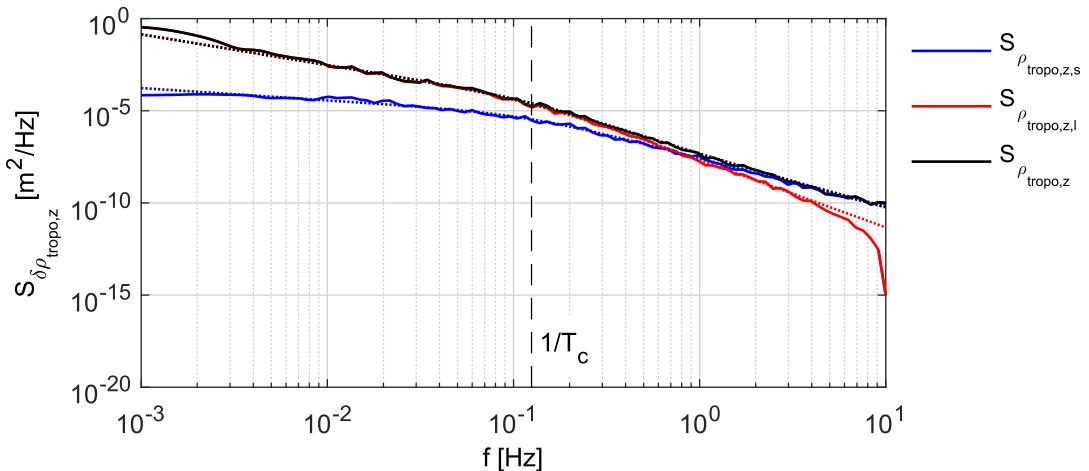


Figure 3-92: PSD of the tropospheric wet delay residual error for $v = 250 \text{ m/s}$

3.5.5 Receiver Clock Error

3.5.5.1 Motivation

A precise clock, that is frequency generating oscillator, is fundamental for the proper functioning of any GNSS receiver. Basically, satellite navigation makes use of the time of flight measurements of the radio signals between the satellites and the user antenna. All frequencies that are required by the GNSS receiver for the down-conversion of the carrier frequency, for the numerically controlled oscillators (NCO) steering the carrier and code replica generation and for the interrupts needed by the navigation processing are derived from a central precise crystal oscillator with a nominal frequency in the range from 10 to 50 MHz.

Figure 3-93 shows a schematic of a typical HF frontend and the digital signal processing (DSP) part. The received satellite signal is down-converted from carrier frequency to intermediate frequency by three subsequent stages, sampled by the ADC and finally forwarded to the tracking channels in the DSP. The three frequencies f_{LO_1} , f_{LO_2} and f_{LO_3} as well as the reference frequency for the clock generator in the DSP are derived from the central reference oscillator f_{ref} by the frequency synthesizer. The highest required frequency for the first down-conversion stage (here 1.4 GHz) is generated by a phase lock loop (PLL) that controls the frequency of a voltage controlled oscillator (VCO). All other lower frequencies are generated by appropriate frequency division. In this design example, the sample frequency f_s for the ADC and the carrier and code NCOs are provided by the clock generator within the DSP. Since ADC and NCOs are steered by the same frequency, a new correlation value can be computed with each input sample. The reference oscillator frequency error δf_{ref} plus the frequency error of the VCO δf_{vco} affect the intermediate frequency, resulting in a slightly shifted intermediate frequency and sampling frequency. The effective influence of the overall oscillator error on the tracking loops is considered by the frequency error inputs in the phase and delay lock loops as shown in Figure 3-115.

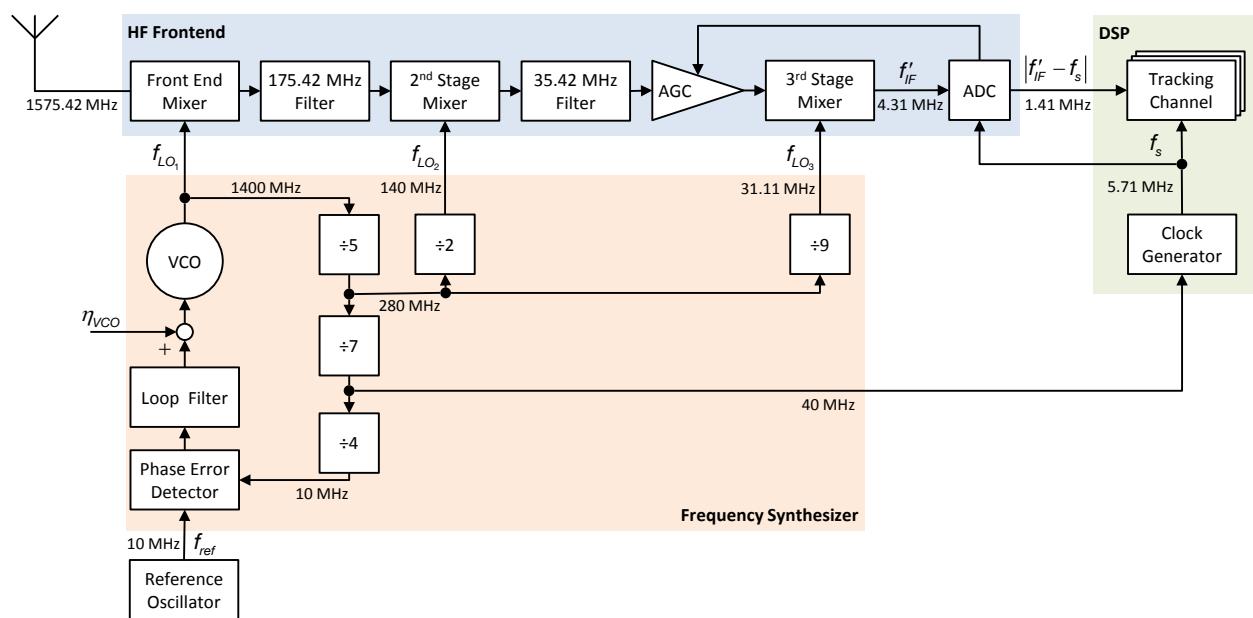


Figure 3-93: HF frontend, frequency synthesizer and DSP of a Zarlink GP2015-based receiver [85] [86]

Oscillator types

Different kinds of oscillators are typically found in GNSS receivers, depending amongst others on the accuracy requirement. All applied oscillators use quartz crystals (X) as resonator. It is distinct between simple crystal oscillators (XO), temperature compensated crystal oscillators (TCXO), voltage controlled crystal oscillators (VCXO) and oven controlled crystal oscillators (OCXO). The frequency stability of quartz oscillators is mainly affected by temperature variations, vibrations and ageing. XOs are low cost and cheap and get along without any temperature compensation. TCXOs largely compensate frequency errors due to temperature variations by means of a temperature sensor and an electronic circuitry. VCXOs feature the possibility to control the oscillation frequency by an applied voltage. OCXOs are the most precise of the oscillators that are applied in mobile GNSS receivers. The temperature of the OCXO quartz is precisely controlled by a crystal oven and thus kept within a very narrow temperature range. OCXOs are very precise with stabilities less than 1 part per billion (ppb). They are generally more expensive, larger of size and consume more power than TCXOs and VCXOs.

Oscillator errors

It is distinct between bias-like and noise-like oscillator errors or in clock typical words, between the drift and the short-term stability [87]. The drift is defined as the systematic change in frequency due to ageing and environmental factors. Ageing designates the long-term change of the frequency in the range of years due to material changes. The decisive environmental factor is the temperature that causes a slow drift when changing. Depending on the age of the crystal and the current temperature T , a certain frequency f , which is offset from the nominal local oscillator frequency f_{LO} , will appear when the oscillator is powered and may slowly change with temperature afterwards.

The temperature influence on the oscillator frequency primarily depends on the cutting angle of the crystal. The typical trend of the frequency deviation δf with temperature T can be well described by a cubic polynomial with the Bechmann coefficients a , b , c

$$\frac{\delta f}{f_{LO}} = a(T - T_0) + b(T - T_0)^2 + c(T - T_0)^3 \quad (3.177)$$

where T_0 is the reference temperature. Due to imperfections at production the calibration values \tilde{a} , \tilde{b} , \tilde{c} , which are implemented as constants in the electronic circuitry of a TCXO, deviate from the true coefficients a , b , c and the resulting errors δa , δb , δc cause a drift when the temperature changes. Uncompensated and compensated frequency deviations are exemplarily illustrated in Figure 3-94.

The short-term, noise-like errors feature shorter correlation times. They are mainly thermal noise as well as acceleration and random vibration induced frequency errors. The thermal noise can be best characterized by its power spectral density. Accelerations and vibrations can strain the crystal, which leads to frequency deviations. There is typically a predominant direction, described by the acceleration sensitivity vector \mathbf{m}_f , in which the frequency deviation δf due to the specific force \mathbf{f}_b on the crystal is maximum. The normalized frequency deviation due to acceleration is given as

$$\frac{\delta f}{f_{LO}} = \mathbf{m}_f^\top (\mathbf{f}_b + \mathbf{R}_{nb}^\top \mathbf{Y}_n) \quad (3.178)$$

The effect is linear for accelerations up to 50 g [87]. Most often, the predominant direction is not known and only a scalar sensitivity m_f is given in the datasheet of the oscillator. Then, the error is assumed to be isotropic

$$\frac{\delta f}{f_{LO}} = m_f \left\| \mathbf{f}_b + \mathbf{R}_{nb}^T \mathbf{Y}_n \right\| \quad (3.179)$$

Frequency errors due to vibrations have basically the same origin, only that the signal power is at higher frequencies, either at distinct frequencies if the vibration has harmonic character or spread over a broader frequency band if the vibration has more random character. Random vibration is as in the case of the thermal noise best described by its power spectral density.

Table 3-33 lists some of the key performance parameters as typically given in the specification of a TCXO that is adequate for GNSS receivers. As example, the Rakon IT225B TCXO has been chosen. In order to compare the performance of different oscillators more easily, the IEEE standard 1139-2008 [15] recommends which parameters shall be given in the specification.

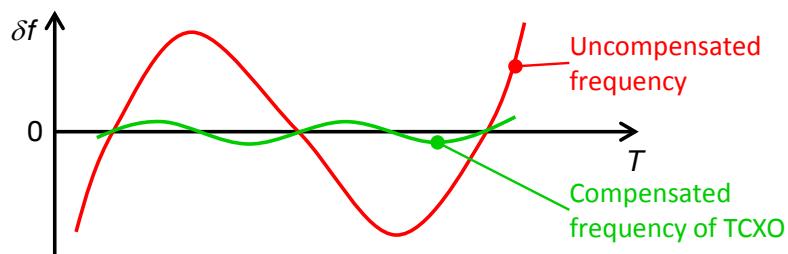


Figure 3-94: Oscillator frequency error characteristics with respect to temperature changes [87]

Table 3-33: Specification Rakon IT225B [88]

Nominal frequency	10 MHz	
Frequency stability over temperature	±2 ppm	±599.5 m/s
Temperature range	-30 – 75 °C	
Frequency slope of perturbations	0.5 ppm/°C	149.9 m/s/°C
Root Allan variance	1.0 ppb	0.3 m/s
Long-term stability (drift over one year)	±1.0 ppm	299.8 m/s
G sensitivity	2.0 ppb/g	0.6 m/s/g

Influence of the oscillator frequency error on the pseudorange measurements

Frequency errors of the carrier NCO lead to a biased estimation of the Doppler carrier frequency and thus affect the range rate measurements. Frequency errors of the code NCO lead to a biased estimation of the Doppler code chip rate and, integrated over time, end up in biased pseudorange measurements. The focus of this section is on the delay lock loop (DLL) and the resulting pseudorange errors. The influence of the oscillator error on the PLL is more complex because not only the carrier NCO is concerned but also the down-conversion of the carrier in the HF frontend.

Since simultaneous measurements to the satellites in view are likewise affected by the oscillator error, the expected receiver clock bias and drift can be well estimated together with the position and velocity by the snapshot least-squares solver or by the integrated navigation filter. The stochastic component of the frequency error, which corresponds to the colored phase noise of the oscillator, however, cannot be estimated but influences the accuracy of the pseudorange measurements.

Receiver clock error model

In this section a two-state clock error model, which also considers the stochastic noise parts of the frequency noise, is presented. The model can be later integrated into the navigation error filter. It is remarked that the oscillator noise is part of the receiver noise and is thus estimated by the receiver with the carrier-to-noise ratio C/N_0 . If the pseudorange measurement noise is derived from the C/N_0 value that is output by the receiver the oscillator phase noise does not have to be separately considered (see section 3.5.6). The RTCA standard DO-316, “Minimum Operational Performance Standards for Global Positioning System/Aircraft Based Augmentation System Airborne Equipment” [7], requires in section R.2.2.4 that “The receiver clock frequency random walk 1-sigma shall [GPS 286] not exceed 1 feet/s/sqrt(s) under steady state thermal conditions. The frequency drift shall [GPS 287] not exceed 3 ppm/ $^{\circ}$ C under transient thermal conditions.” It is noted that “A significantly lower temperature sensitivity than 3 ppm/ $^{\circ}$ C would require the use of an oven-controlled crystal oscillator (OCXO) and exclude the use of a temperature compensated crystal oscillator (TCXO) in the GPS receiver.” Concerning the integrity of the integrated navigation solution, it is stated that “If receiver clock aiding is used to enhance integrity, the algorithms that perform calibration shall [GPS 288] be designed to prevent the satellite failure itself from affecting the integrity of the calibration. Conventional Kalman filter integrations using clock states for offset and drift rate with no further enhancements to protect these states in a failure situation will not meet this requirement.”

3.5.5.2 Model

The two-state continuous-time receiver clock error model is given with

$$\begin{aligned}\dot{\mathbf{z}}_{\Delta t_R} &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \mathbf{z}_{\Delta t_R} + \frac{f_{code}}{f_{LO}} \begin{pmatrix} \eta_{WF} + \omega_{FF} \\ \eta_{RWF} \end{pmatrix} + f_{code} \begin{pmatrix} \delta a(T - T_0) + m_f \|\mathbf{f}_b + \mathbf{R}_{nb}^T \mathbf{Y}_n\| + \varepsilon \\ 0 \end{pmatrix} \\ \Delta t_R &= (1 \ 0) \mathbf{z}_{\Delta t_R} + \frac{f_{code}}{f_{LO}} (\eta_{WP} + \omega_{FP}) \quad \mathbf{z}(t=0) = \begin{pmatrix} \Delta t_{R,0} \\ \Delta \dot{t}_{R,0} \end{pmatrix}\end{aligned}\quad (3.180)$$

where Δt_R is the receiver clock bias. The white phase noise η_{WP} , flicker phase noise ω_{FP} , white frequency noise η_{WF} , flicker frequency noise ω_{FF} and the random walk frequency input noise η_{RWF} are the short-term thermal noise components of the oscillator. The frequency noises are given in s/s and the phase noises are in s . They have to be converted from the nominal local oscillator frequency f_{LO} to the actual code chip rate f_{code} with the scaling factor f_{code}/f_{LO} . That is because the frequency noise is divided just as the nominal frequency in an ideal frequency divider. Moreover, the linear temperature calibration error δa and the isotropic acceleration dependent error are considered. Higher order terms are combined in ε . The normalized temperature and acceleration dependent errors have to be scaled by the nominal code chip rate f_{code} . The turn-on frequency offset due to ageing is contained in $\Delta \dot{t}_{R,0}$. The initial clock bias $\Delta t_{R,0}$ depends on how the receiver calculates the pseudorange from the code delay measurement and can be large. The power spectral densities of the separate noise components feature power law behavior, as discussed in section 3.1.4.

The discrete-time correspondent of (3.180) is straightforward

$$\begin{aligned}\mathbf{z}_{\Delta t_R,k} &= \begin{pmatrix} 1 & \Delta T \\ 0 & 1 \end{pmatrix} \mathbf{z}_{\Delta t_R,k-1} + \Delta T \frac{f_{code}}{f_{LO}} \begin{pmatrix} \eta_{WF,k-1} + \omega_{FF,k-1} \\ \eta_{RWF,k-1} \end{pmatrix} \\ &\quad + \Delta T f_{code} \begin{pmatrix} \delta a(T_{k-1} - T_0) + m_f \|\mathbf{f}_{b,k-1} + \mathbf{R}_{nb,k-1}^T \mathbf{Y}_{n,k-1}\| \\ 0 \end{pmatrix}, \quad \mathbf{z}_0 = \begin{pmatrix} \Delta t_{R,0} \\ \Delta \dot{t}_{R,0} \end{pmatrix}\end{aligned}\quad (3.181)$$

$$\begin{aligned}\Delta t_{R,k} &= (1 \ 0) \mathbf{z}_{\Delta t_{R,k}} + \frac{f_{code}}{f_{LO}} (\eta_{WP,k} + \omega_{FP,k}) \\ \Delta \dot{t}_{R,k} &= (0 \ 1) \mathbf{z}_{\Delta t_{R,k}} + \frac{f_{code}}{f_{LO}} \left(\eta_{WF,k} + \omega_{FF,k} + \frac{\eta_{WP,k} - \eta_{WP,k-1}}{\Delta T} + \frac{\omega_{FP,k} - \omega_{FP,k-1}}{\Delta T} \right) \\ &\quad + f_{code} \left(\delta a(T_{k-1} - T_0) + m_f \|\mathbf{f}_{b,k-1} + \mathbf{R}_{nb,k-1}^T \mathbf{Y}_{n,k-1}\| \right)\end{aligned}$$

Be aware not to mix up the clock error Δt_R , propagation sample time ΔT and temperature T . Discrete-time models for the white frequency noise ω_{WF} , flicker frequency noise ω_{FF} , random walk frequency noise ω_{RWF} , white phase noise ω_{WP} and the flicker phase noise ω_{FP} are according to Table 3-7 in section 3.1.4.3. Note that the difference quotients of the white and flicker phase noises ω_{WP} and ω_{FP} have to be added to the clock drift output equation. The parameters h_α , $\alpha = \{-2, \dots, 2\}$, are to be estimated from the phase noise power spectral density $S_\phi(f)$ from recorded oscillator voltages or from available specifications from the manufacturer. How to determine these parameters is shown in the next section. The short-term noises on the receiver clock bias and drift are part of the pseudorange and range rate measurement noises that will be discussed in section 3.5.6. They are affected by the tracking filters. In this section, only the long-term, random walk influences on the receiver clock bias and drift are of interest. Therefore, the influence of the white and flicker phase noises ω_{WP} and ω_{FP} on the receiver clock bias and drift and of the white and flicker frequency noises ω_{WF} and ω_{FF} on the receiver clock drift are omitted. The output equations of (3.181) simplify to

$$\begin{aligned}\Delta t_{R,k} &= (1 \ 0) \mathbf{z}_{\Delta t_{R,k}} \\ \Delta \dot{t}_{R,k} &= (0 \ 1) \mathbf{z}_{\Delta t_{R,k}}\end{aligned}\tag{3.182}$$

neglecting also the temperature and acceleration effect. Fortunately, the difference quotients in the clock drift output equation, which are difficult to handle, can thus be avoided, too.

3.5.5.3 Parameter Estimation

The IEEE standard 1139-2008 [15] proposes to characterize the short-term stability of oscillators either in time domain with the (overlapping) Allan deviation $\sigma_{avar,y}(\tau)$ of the fractional frequency $y = f/f_{LO}$ or in frequency domain with the two-sided PSD of the phase deviations $S_\phi(f)$. If the short-term stability shall be estimated from recorded oscillator voltages, the overlapping Allan deviation is the first choice. Before the Allan deviation can be calculated, the systematic drift has to be removed from the fractional frequencies. Alternatively, the Hadamard deviation can be applied being insensitive to drifts. If specifications of the phase PSDs $S_\phi(f)$ are available from the manufacturer, these can be directly used to derive the model parameters. Here, the second variant is presented. The single sideband spectral density $S_{SSB}(f)$, which is often found in the manufacturer's datasheets, corresponds to the two-sided spectral density of the phase deviations $S_\phi(f)$

$$S_{SSB}(f) = S_\phi(f)\tag{3.183}$$

Novatel's OEM628 receiver uses a 20 MHz VCXO (Filtronetics ON-169A) and DLR's Phoenix GPS receiver bases upon a 10 MHz TCXO (Rakon IT225B). The specification of the phase deviation PSD of both oscillators is given in Table 3-34. It has to be noted that the phase noise of the VCO in the frequency synthesizer has to be additionally considered. In the case of the Zarlink GP2015, the noise of the incorporated VCO is much higher than the noise of the external Rakon IT225B reference oscillator [85]. Since the bandwidth of the frequency synthesizer PLL is rather high (15 kHz for the Zarlink GP2015), the filter effect of the PLL can be neglected. The PLL thus follows directly the

frequency of the reference oscillator and adds the noise of the VCO straight. For the illustration of the identification approach, the focus is here on the reference oscillator. The phase noise of the VCO can be obtained in the same way and has simply to be added to the noise of the reference oscillator. The PSD of the phase deviation $S_\phi(f)$ is given by

$$S_\phi(f) = f_{LO}^2 \left(\underbrace{\frac{1}{2} \frac{h_{-2}}{f^4}}_{\text{Random Walk FM}} + \underbrace{\frac{1}{2} \frac{h_{-1}}{f^3}}_{\text{Flicker FM}} + \underbrace{\frac{1}{2} \frac{h_0}{f^2}}_{\text{White FM}} + \underbrace{\frac{1}{2} \frac{h_1}{f}}_{\text{Flicker PM}} + \underbrace{\frac{1}{2} h_2}_{\text{White PM}} \right) \quad (3.184)$$

where

$$S_\phi(f) = (2\pi f_{LO})^2 S_x(f) = \frac{f_{LO}^2}{f^2} S_y(f) \quad (3.185)$$

and

$$S_y(f) = \underbrace{\frac{1}{2} \frac{h_{-2}}{f^2}}_{\text{Random Walk FM}} + \underbrace{\frac{1}{2} \frac{h_{-1}}{f^1}}_{\text{Flicker FM}} + \underbrace{\frac{1}{2} h_0}_{\text{White FM}} + \underbrace{\frac{1}{2} h_1 f}_{\text{Flicker PM}} + \underbrace{\frac{1}{2} h_2 f^2}_{\text{White PM}} \quad (3.186)$$

is the PSD of the fractional frequency noise. From the given PSD $S_\phi(f)$, for example taken from the performance specification of the oscillator, the parameter vector \mathbf{a} containing

$$\mathbf{a}^\top = (h_{-2} \ h_{-1} \ h_0 \ h_1 \ h_2) \quad (3.187)$$

can be easily estimated by means of the least squares method with the vector \mathbf{s} of the n available PSD values $S_\phi(f_i)$, $i=1\dots n$,

$$\mathbf{s}^\top = (S_\phi(f_1) \ \dots \ S_\phi(f_n)) \quad (3.188)$$

and the coefficient matrix \mathbf{A} with the corresponding frequencies

$$\mathbf{A} = \begin{pmatrix} \frac{1}{2} f_{LO}^2 f_1^{-4} & \frac{1}{2} f_{LO}^2 f_1^{-3} & \frac{1}{2} f_{LO}^2 f_1^{-2} & \frac{1}{2} f_{LO}^2 f_1^{-1} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{1}{2} f_{LO}^2 f_n^{-4} & \frac{1}{2} f_{LO}^2 f_n^{-3} & \frac{1}{2} f_{LO}^2 f_n^{-2} & \frac{1}{2} f_{LO}^2 f_n^{-1} & 1 \end{pmatrix} \quad (3.189)$$

As was done with the IMU parameter estimation, the sum of the squares of the differences of the common logarithms shall be minimized by the estimator

$$\arg \left\{ \min_{\hat{\mathbf{a}}} \left((\log(\mathbf{s}) - \log(\mathbf{A} \hat{\mathbf{a}}))^\top (\log(\mathbf{s}) - \log(\mathbf{A} \hat{\mathbf{a}})) \right) \right\} \quad (3.190)$$

Due to numerical reasons it is advisable to estimate the parameters scaled by the squared nominal oscillator frequency, $\bar{h}_\alpha = f_{LO}^2 h_\alpha$, since the parameter values h_α are very small.

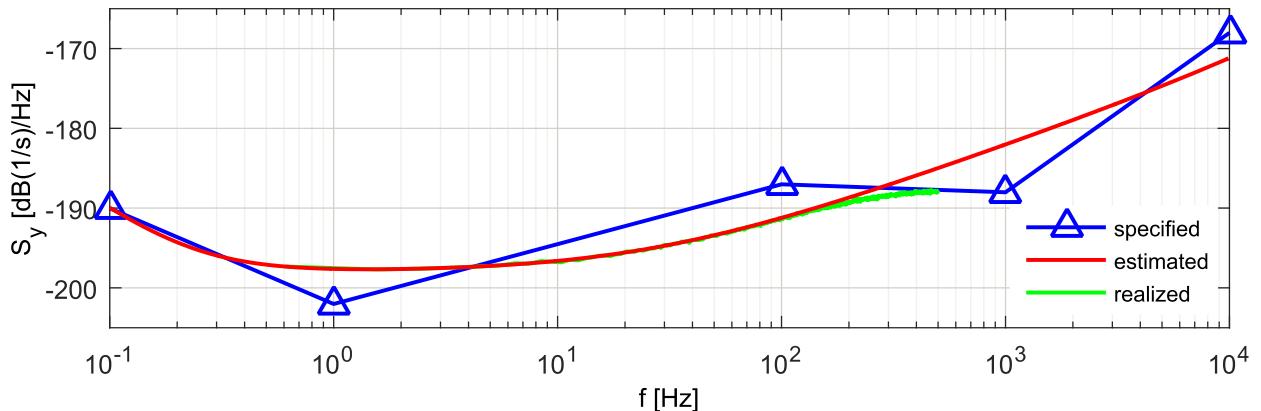
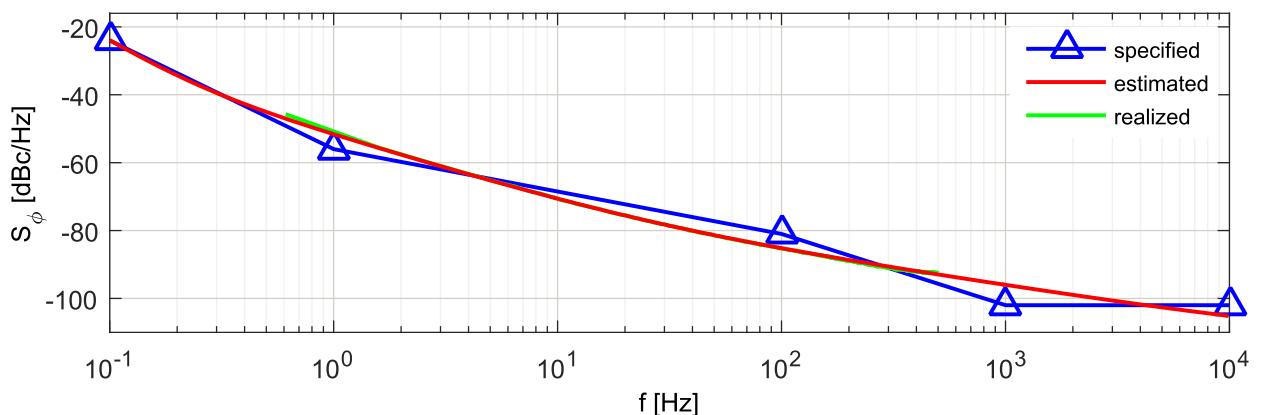
The parameters that have been estimated for the ON-169A and IT225B oscillators are listed in Table 3-35. The specified, estimated and realized PSDs of the fractional frequency of the ON-169A oscillator are plotted in Figure 3-95 and the corresponding PSDs of the phase deviation in Figure 3-96. Figure 3-97 shows a typical realization of the clock drift of the ON-169A oscillator and Figure 3-98 illustrates six exemplary curves of the clock bias. Figure 3-99 to Figure 3-102 are the corresponding figures for the IT225B oscillator.

Table 3-34: Phase deviation power spectral density specifications of ON-169A and IT225B oscillators

Frequency f [Hz]	Two-sided PSD S_ϕ [dBc/Hz] ($= S_{SSB}$)	
	Filtronetics ON-169A [89] VCXO (Novatel OEM628)	Rakon IT225B [88] TCXO (Phoenix GPS)
0.1	-24	
1	-56	-55
10		-85
100	-81	-110
1000	-102	-125
10000	-102	-140

Table 3-35: Estimated parameters for ON-169A and IT225B oscillators

Parameter	Unit	ON-169A	IT225B
\bar{h}_2	Hz ² Hz	6.825177804170940e-07	7.392439921073599e-10
\bar{h}_1	Hz ²	2.095611517694396e-11	5.348045043004299e-06
\bar{h}_0	Hz ² /Hz	1.266266277147555e-05	1.912900765736840e-07
\bar{h}_1	Hz ² /Hz ²	4.759222884490891e-07	2.955711832121236e-10
\bar{h}_2	Hz ² /Hz ³	1.328758637139857e-11	7.829645179372844e-20

**Figure 3-95: Fractional frequency power spectral density of the ON-169A oscillator****Figure 3-96: Phase deviation power spectral density of the ON-169A oscillator**

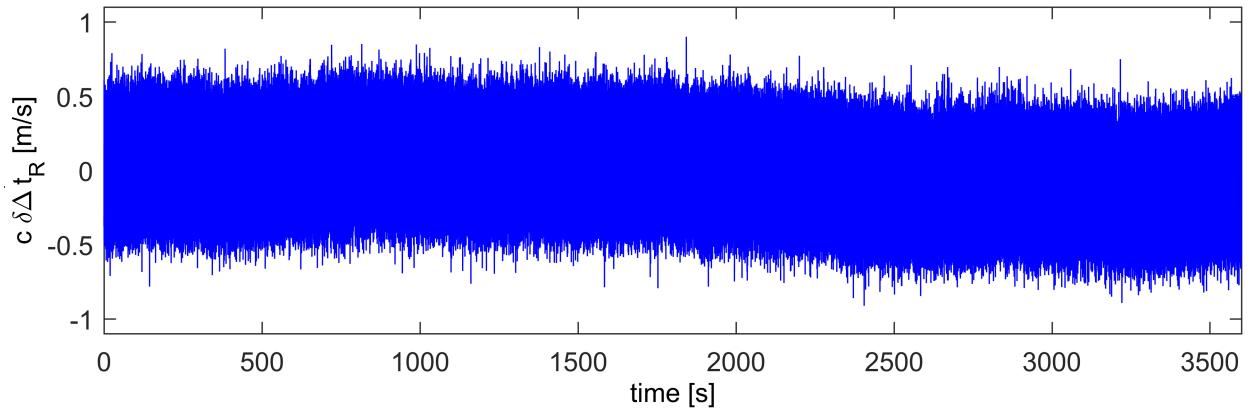


Figure 3-97: Exemplary clock drift of the ON-169A oscillator

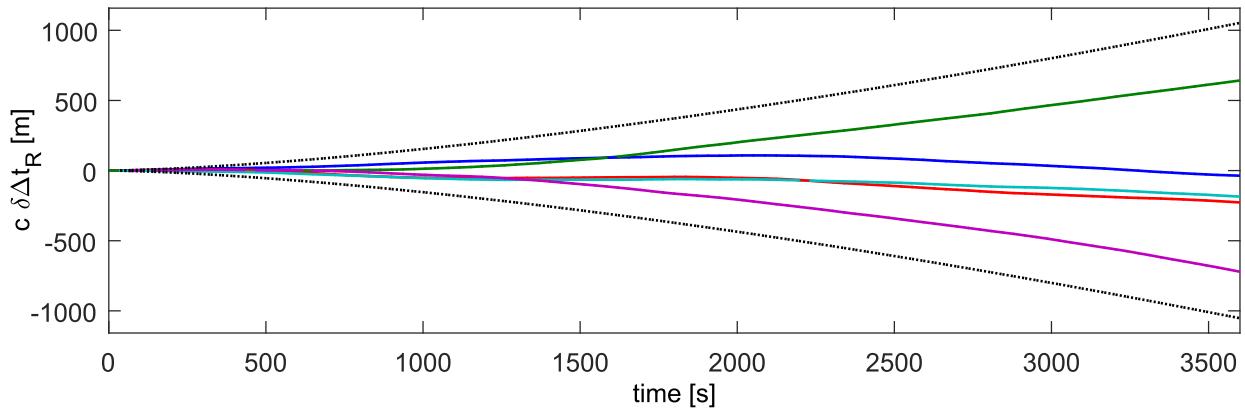


Figure 3-98: Six exemplary clock bias curves of the ON-169A oscillator and predicted 3σ boundaries

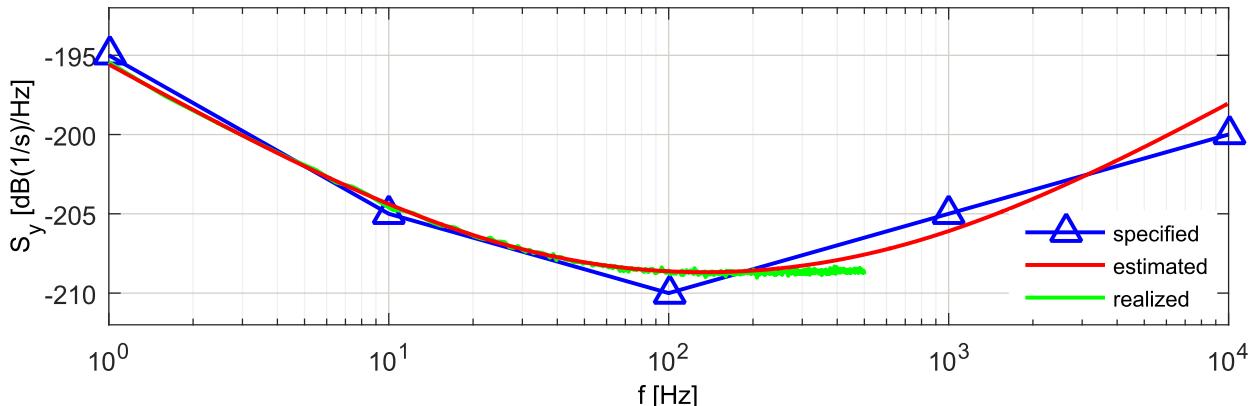


Figure 3-99: Fractional frequency power spectral density of the IT225B oscillator

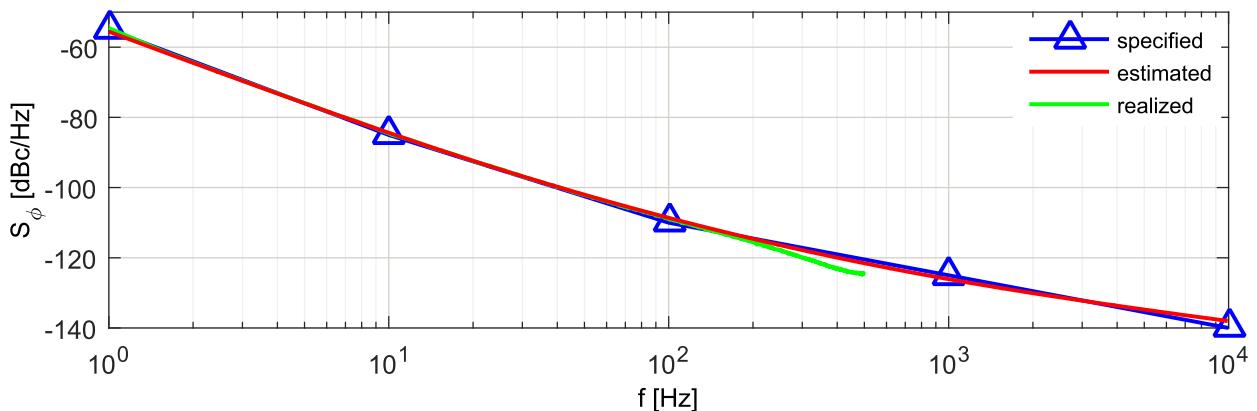


Figure 3-100: Phase deviation power spectral density of the IT225B oscillator

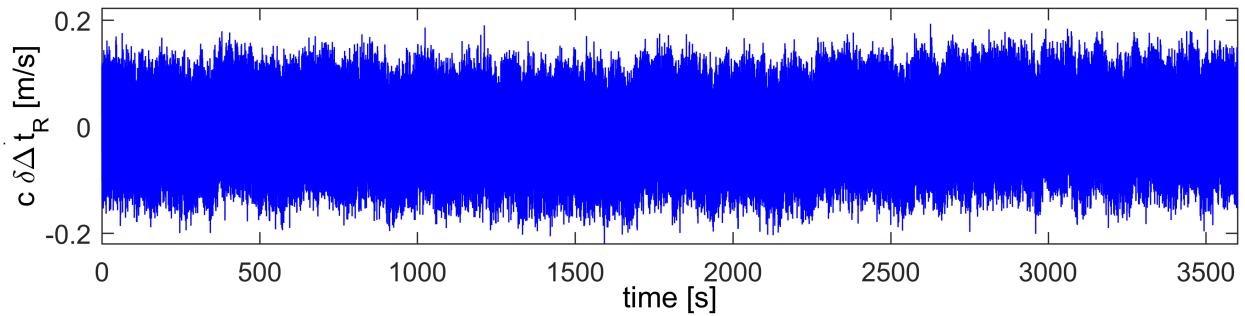


Figure 3-101: Exemplary clock drift of the IT225B oscillator

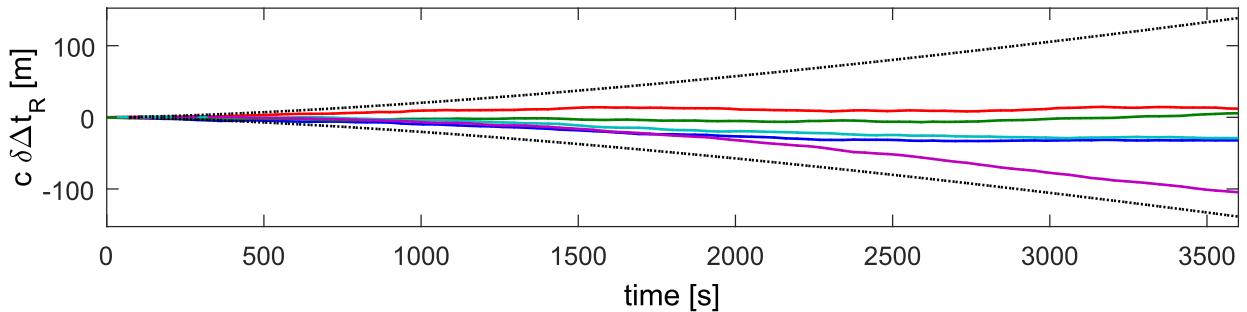
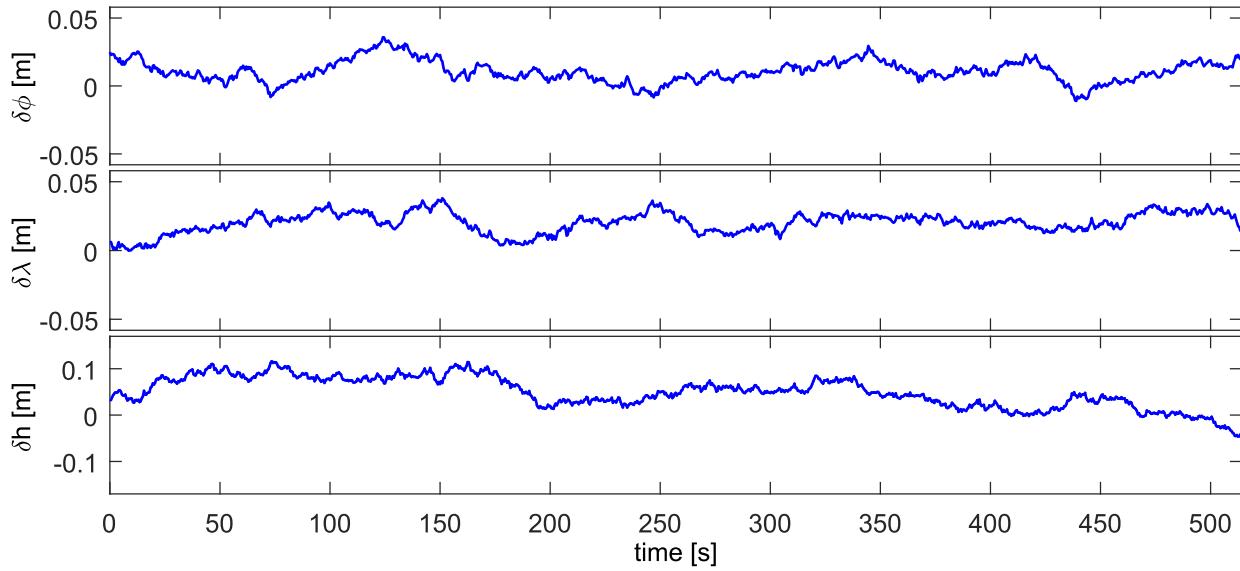
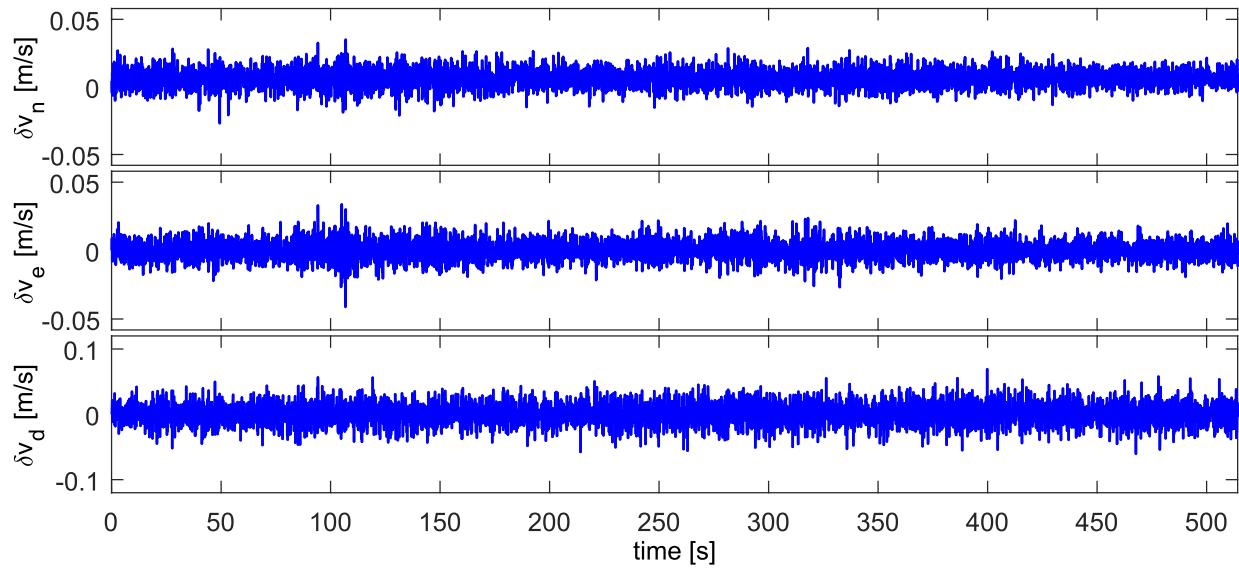
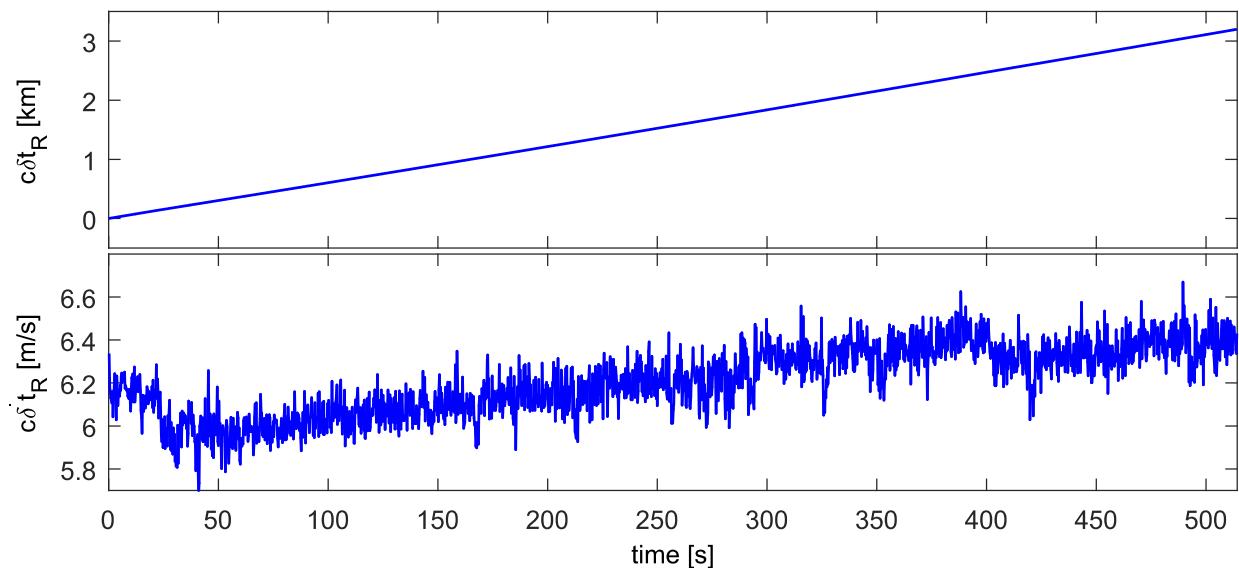


Figure 3-102: Six exemplary clock bias curves of the IT225B oscillator and predicted 3σ boundaries

3.5.5.4 Example: Novatel OEM628

In order to verify the results, laboratory tests with a Novatel OEM628 GNSS receiver [90] incorporating the ON-169A oscillator and a Spectracom GSG-5 GNSS signal simulator have been done. A stationary user antenna position has been used for the analysis. All errors like ionosphere and troposphere delays have been switched off. It is assumed that the oscillator error of the signal simulator is at least one magnitude lower than that of the GNSS receiver, such that the resulting clock error primarily stems from the receiver. Correction of the pseudorange measurements with the estimated clock error in the receiver has been deactivated. The carrier-to-noise ratio is set to constant 49 dB/Hz in all channels. Figure 3-103 shows the position error of the PVT solution. It is below 4 cm in horizontal direction and remains well below 15 cm in vertical direction during the simulation time. In Figure 3-104 the velocity error is plotted. It is lower than 3 cm/s in horizontal direction. The estimated receiver clock bias and drift are illustrated in Figure 3-105. It has to be noted that the receiver clock noise is filtered by the tracking loops in contrast to the pure receiver clock error shown in Figure 3-97. Therefore, higher frequencies above the tracking loop bandwidth are suppressed. The magnitudes and time correlation agree quite well if the filtering effect is taken into account. In Figure 3-106, the constant clock drift has been removed. The receiver clock bias characteristic in the upper subplot is as expected from the previous parameter estimation. The pseudorange measurements to the satellites in view are corrected by the estimated receiver clock bias in order to obtain the pseudorange residuals. They are shown in Figure 3-107. The additionally plotted 3σ standard deviations are output by the receiver. Since the carrier-to-noise ratio is equal for all measurements, all standard deviations are the same. The standard deviations match well to the actual pseudorange errors. It can be observed that the single pseudorange residuals are not noticeably correlated among each other. It is concluded that the pseudorange measurements can be handled as statistically independent. The range rate residuals are given in Figure 3-108. As is the case with the pseudorange residuals, the range rate residuals are not correlated.

**Figure 3-103: Position error****Figure 3-104: Velocity error****Figure 3-105: Receiver clock bias and drift error**

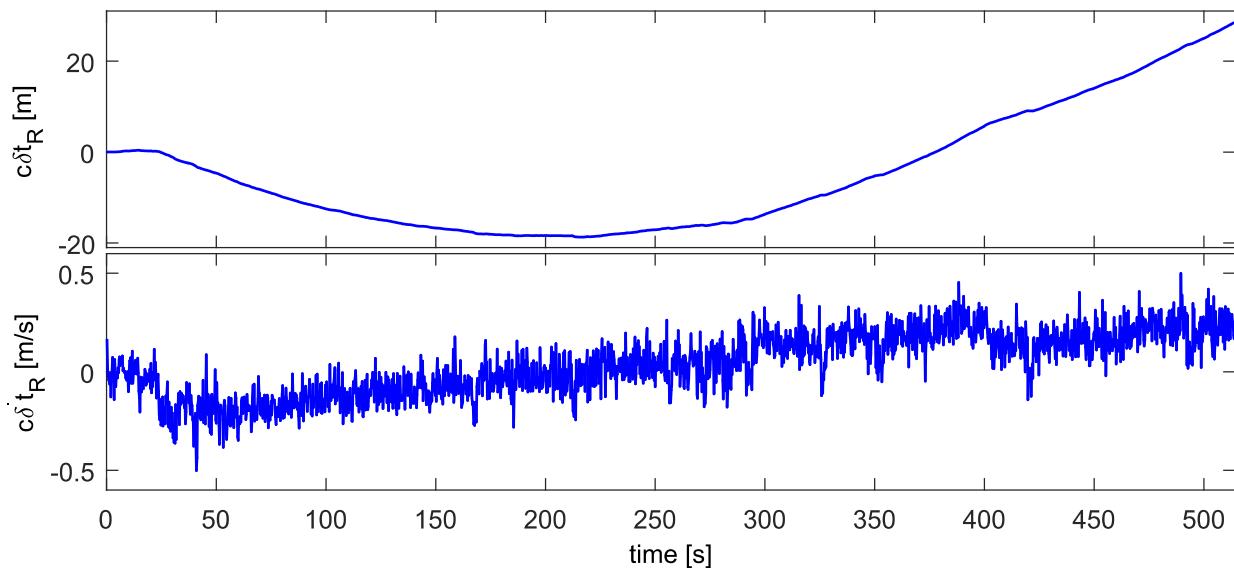


Figure 3-106: Detrended receiver clock bias and drift error

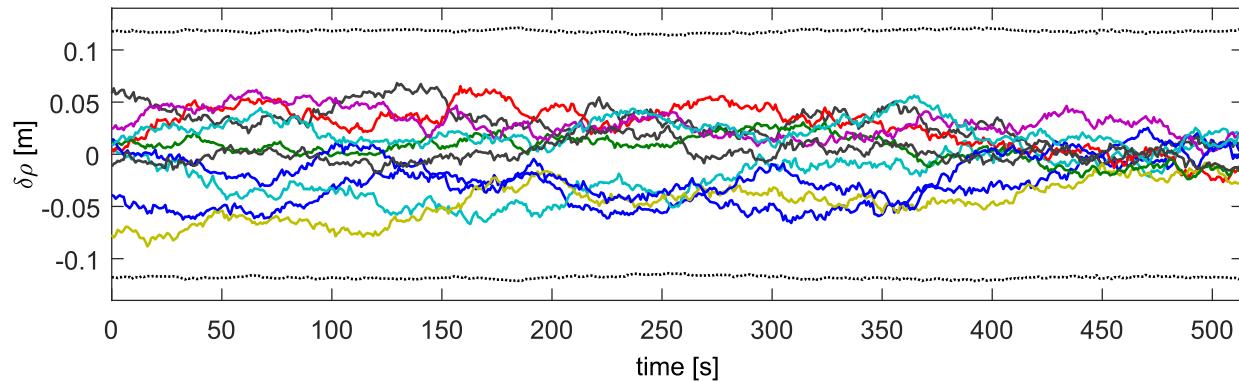


Figure 3-107: Pseudorange residuals and 3σ standard deviations (dotted lines) as output by the receiver

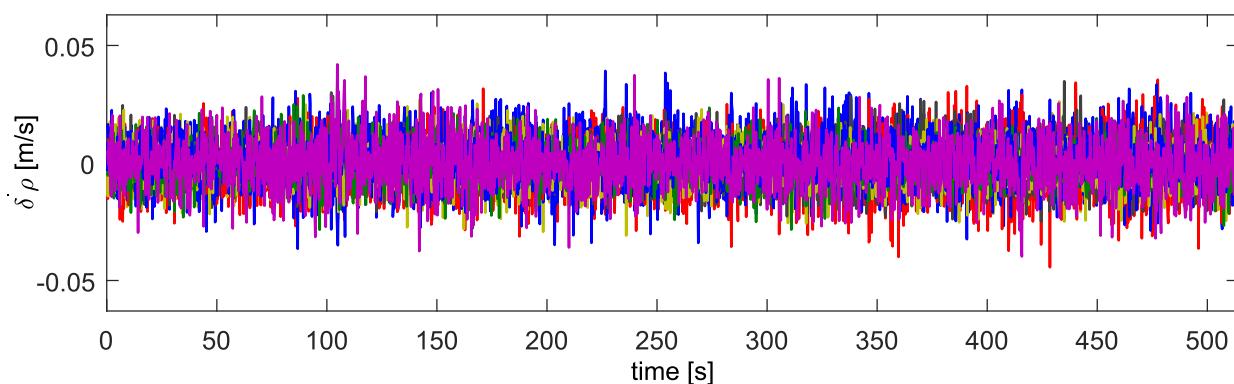


Figure 3-108: Range rate residuals

3.5.5.5 Remarks on the Integration in the Navigation Filter

It has to be noted that some GNSS receivers adjust the NCO frequency according to the currently estimated receiver clock drift of the computed PVT solution or correct the raw pseudorange measurements with the estimated receiver clock bias continuously or from time-to-time if a specified threshold (for example 1 millisecond) is exceeded. If the receiver uses a VCXO like Novatel's OEM628, the receiver can also continuously adapt the frequency steering voltage of the oscillator. The understanding of this behavior is critical for the performance of the integrated navigation system from different aspects. First, sudden pseudorange measurement errors like for example due to the malfunction of a satellite will deteriorate the internally estimated PVT solution including the receiver clock error if the receiver autonomous integrity monitor (RAIM) does not reliably detect and isolate this error with the predicted probability (if RAIM is available at all). This can lead to an additional bias on the pseudorange measurements besides the corrupted measurement of the malfunctioning satellite. The integrity monitor of the integrated navigation system must be able to cope with this special situation. Second, the integrated navigation filter has to be aware that there might be sudden steps in the pseudorange measurements. The system has to detect and appropriately react on occurring steps. Otherwise, if the estimated receiver clock bias and drift are already settled and the corresponding covariances are accordingly small, these measurements will henceforth be refused by innovation based integrity monitors because the actual innovation is much larger than expected, hence violating the statistics. In high-end receivers the automatic correction of the raw pseudorange and range rate measurements can be disabled. Due to the ageing of the crystal oscillator the influence of the phase noise can grow with time. In cases requiring very high accuracy, the oscillator should be recalibrated at regular intervals. Alternatively, the noise statistics upon which the navigation error filter bases can be adapted accordingly. The receiver clock error model will be integrated into the navigation filter in section 4.6.7.

3.5.6 Receiver Noise

3.5.6.1 Motivation

For the statistically correct fusion of the pseudorange and range rate measurements within the integrated navigation filter the knowledge of the variances and correlations of the single measurement errors amongst each other and with time is important. Often, receivers do not output standard deviations of the raw measurements, but provide only rudimentary statements about the expected error of the PVT solution. In this case the user has to assess the current magnitudes of the measurement noises himself.

Under favorable environmental conditions the thermal noise constitutes the main part of the measurement noise. Besides, the noise of the NCOs and the quantization noise of the ADC play a role. Under unfavorable conditions multipath or other interferences may exceed the thermal noise. Since the focus lies on applications for aircraft and on nominal operation, these disturbances are not considered.

[64] and [91] give approximations for the standard deviations of the pseudorange and carrier phase measurement noise in dependence on the carrier-to-noise ratio and the tracking loop filter bandwidths. However, these formulas are derived for one certain discriminator and tracking loop structure. If other designs are implemented in the GNSS receiver, the resulting measurement noises may turn out to be different. Moreover, no statement is made about the correlations of the measurements among each other and with time.

Providing that the user has insight into the tracking loop filter design and knows the selected filter parameters, he can exactly predict the expected measurement noise variances and the corresponding correlations if assumptions are made about the noise of the incoming signal at the antenna input and the other noise sources. Various tracking loop filter designs and discriminator variants can be found in the literature. Therefore, the approach is exemplarily outlined by means of a common design – a third order frequency-assisted PLL and a carrier-aided DLL. The provided statements can however be analogously derived for any other arbitrary filter structure.

3.5.6.2 Satellite Signal Model

A general receiver structure is shown in Figure 3-109. First, the GNSS signal is fed through a low noise amplifier to increase the signal amplitude. Second, the signal carrier frequency f is down-converted to the lower intermediate frequency f_{IF} and subsequently digitized by an ADC with the sample rate f_s . The ADC additionally steers the automatic gain control that adapts the signal amplitude before sampling such that the ADC solution is optimally used. The digital signal samples are processed by the tracking channels, which generate the pseudorange, range rate and carrier phase measurements and extract the navigation data. In the final navigation processing block the actual PVT solution and corresponding covariances are computed from the raw measurements. All required local frequencies and sample rates are derived from the reference frequency f_{ref} of a common stable oscillator by the frequency synthesizer.

The signal at the antenna input at reception time t is mathematically described by

$$s(t) = A c(t - \tau) d(t - \tau) \cos(2\pi f(t) + \phi(t)) + \eta(t) \quad (3.191)$$

Therein, A is the peak amplitude of the unmodulated carrier wave. c is the ranging code and d the navigation data that are modulated on the carrier signal. τ is the true travel time of the signal and $t - \tau$ the true broadcast time at the satellite antenna. f is the frequency and ϕ the phase of the carrier signal at reception time. η is the thermal noise. It is assumed to be white and Gaussian distributed with two-sided power spectral density N_0 in Watts/Hertz. The power of the unmodulated carrier signal C is proportional to the squared root mean square amplitude $A/\sqrt{2}$

$$C = \left(\frac{A}{\sqrt{2}} \right)^2 \quad (3.192)$$

The amplitude A is thus expressed by the signal power C

$$A = \sqrt{2C} \quad (3.193)$$

The quotient of the unmodulated carrier signal power C and the noise power spectral density N_0 is the carrier-to-noise ratio C/N_0 , which is a common measure for the quality of the received signal. $s(t)$ actually consists of a mixture of the signals of all satellites in view. However, since the signal of a certain satellite can be uniquely identified later in the correlator, it is sufficient to write the formulas for one single satellite.

The incoming signal passes several mixer and band-pass filter stages in the receiver's frontend. In the end, the down-converted continuous-time signal at the output of the frontend is given with

$$s'(t) = \sqrt{2C} c(t - \tau) d(t - \tau) \cos(2\pi f'_{IF}(t) + \phi(t)) + \eta'(t) \quad (3.194)$$

f'_{IF} depicts the intermediate carrier frequency of the signal after the last mixer stage. The white, Gaussian distributed noise is now band-limited

$$\eta'(t) \sim WN(0, BN_0(t)) \quad (3.195)$$

with variance $\sigma_\eta^2 = BN_0(t)$. B is the effective bandwidth of the baseband equivalent band-pass filter and is specified in Hertz.

A low-resolution ADC finally samples the down-converted signal (3.194) with the intermediate carrier frequency f'_{IF} with the sampling rate f_s . Automatic gain control amplifiers in the frontend adapt the signal magnitude and thus ensure that the range of the ADC is optimally used. The discrete-time signal after ADC sampling is described by

$$s_k = \sqrt{2C} c_k d_k \cos(2\pi f'_{IF,k} t_k + \phi_k) + \eta_k \quad (3.196)$$

The effective intermediate frequency of the digitized signal $f'_{IF,k}$ results from the frequency of the analog signal f'_{IF} and the sampling rate f_s due to the aliasing effect

$$f'_{IF,k} = |f'_{IF,k} - f_s| \quad (3.197)$$

The power and variance of the noise η_k is $\sigma_{\eta,k}^2 = BN_{0,k}$. The signal-to-noise ratio (SNR) after ADC sampling is the quotient of the carrier power C and the noise power $\sigma_{\eta,k}^2$ in Watts/Watts

$$SNR_k = \frac{C}{\sigma_{\eta,k}^2} = \frac{C}{BN_{0,k}} \quad (3.198)$$

The discrete-time signal is then forwarded to the correlator. Zarlink's GP2015 is a typical integrated circuit frontend for GPS L1 signals [85].

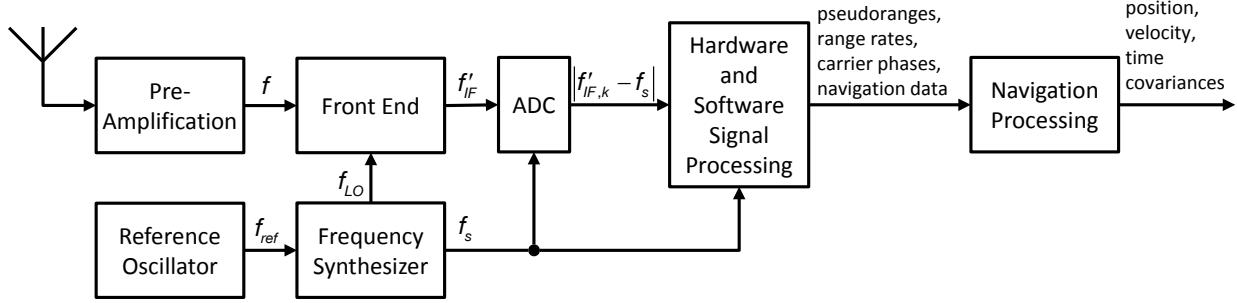


Figure 3-109: General receiver structure

3.5.6.3 Correlator Model

Figure 3-110 shows the hardware signal processing part of a tracking channel. The digital signal s_k is first multiplied with a replica of the carrier wave, which is generated by the carrier NCO, to obtain the in-phase branch and a 90 degree phase delayed replica to obtain the quadrature branch. Second, the in-phase and quadrature signals are correlated with the early, prompt and late replica of the ranging codes of the different satellites that have been identified in the signal mix in the acquisition. The replica code generator is triggered by the code NCO. With the number of summed samples m , the signal-to-noise ratio can be adjusted in accordance with the expected dynamics of the platform on which the receiver is applied.

The obtained in-phase and quadrature signals $\tilde{I}_{E,k}$, $\tilde{I}_{P,k}$, $\tilde{I}_{L,k}$, $\tilde{Q}_{E,k}$, $\tilde{Q}_{P,k}$ and $\tilde{Q}_{L,k}$ are then fed into the discriminators to estimate the current frequency offset between the digitized signal and the carrier replica and the current code chip offset between the ranging code that is modulated on the incoming signal and the local code replica. Zarlink's GP2021 is an example for a twelve channel correlator for GPS L1 signals [92].

The prompt in-phase correlator output is [91]

$$\tilde{I}_{P,k} = \frac{\sqrt{2C}}{2} f_s T R(\tau_k) \operatorname{sinc}(\delta f_k T) \cos \delta \phi_k + \eta_{I_{P,k}} \quad (3.199)$$

The output of the prompt quadrature phase correlator is analogous to the in-phase correlator output, only that the cosine of the phase error is replaced by the sine

$$\tilde{Q}_{P,k} = \frac{\sqrt{2C}}{2} f_s T R(\tau_k) \operatorname{sinc}(\delta f_k T) \sin \delta \phi_k + \eta_{Q_{P,k}} \quad (3.200)$$

T is the integration time of the correlator. $\operatorname{sinc}(\delta f_k T)$ is the cardinal sine function which is defined as

$$\operatorname{sinc}(\delta f_k T) = \frac{\sin(\pi \delta f_k T)}{\pi \delta f_k T} \quad (3.201)$$

$R(\tau_k)$ is the autocorrelation function of the spreading code, where τ_k is the chip lag between the incoming and the replica codes. In the case of the C/A code of GPS the autocorrelation function is given with

$$R(\tau_k) = \begin{cases} 1 - |\tau_k| & , |\tau_k| \leq 1 \\ 0 & , |\tau_k| > 1 \end{cases} \quad (3.202)$$

Note that the autocorrelation function is slightly altered by the frontend band-pass filter. Since the changes are only small, the effect is neglected here. δf_k is the difference between the currently estimated intermediate carrier frequency $\tilde{f}_{IF,k}$ and the true intermediate carrier frequency of the incoming signal $f_{IF,k}$

$$\delta f_k = \tilde{f}_{IF,k} - f_{IF,k} \quad (3.203)$$

$\delta\phi_k$ represents the corresponding difference between the phases of both signals

$$\delta\phi_k = \tilde{\phi}_k - \phi_k \quad (3.204)$$

The in-phase and quadrature phase noises are white and normally distributed and have the covariance

$$E[\eta_{I_p,k}^2] = E[\eta_{Q_p,k}^2] = \frac{T f_s}{2} BN_{0,k} \quad (3.205)$$

Since receivers often output a current estimate of the carrier-to-noise ratio, it is suitable to scale the correlator equations by the square root of the noise variance

$$\sqrt{\frac{T f_s BN_{0,k}}{2}} \quad (3.206)$$

Then, the scaled early, prompt and late in-phase correlator outputs are

$$\tilde{I}_{E,k} = \sqrt{2(C/N_0)_k T} \sqrt{\frac{f_s}{B}} R\left(\tau_k - \frac{\Delta_{EL}}{2}\right) \operatorname{sinc}(\delta f_k T) \cos \delta\phi_k + \eta_{I_E,k} \quad (3.207)$$

$$\tilde{I}_{P,k} = \sqrt{2(C/N_0)_k T} \sqrt{\frac{f_s}{B}} R(\tau_k) \operatorname{sinc}(\delta f_k T) \cos \delta\phi_k + \eta_{I_P,k} \quad (3.208)$$

$$\tilde{I}_{L,k} = \sqrt{2(C/N_0)_k T} \sqrt{\frac{f_s}{B}} R\left(\tau_k + \frac{\Delta_{EL}}{2}\right) \operatorname{sinc}(\delta f_k T) \cos \delta\phi_k + \eta_{I_L,k} \quad (3.209)$$

The scaled early, prompt and late quadrature phase correlator outputs are

$$\tilde{Q}_{E,k} = \sqrt{2(C/N_0)_k T} \sqrt{\frac{f_s}{B}} R\left(\tau_k - \frac{\Delta_{EL}}{2}\right) \operatorname{sinc}(\delta f_k T) \sin \delta\phi_k + \eta_{Q_E,k} \quad (3.210)$$

$$\tilde{Q}_{P,k} = \sqrt{2(C/N_0)_k T} \sqrt{\frac{f_s}{B}} R(\tau_k) \operatorname{sinc}(\delta f_k T) \sin \delta\phi_k + \eta_{Q_P,k} \quad (3.211)$$

$$\tilde{Q}_{L,k} = \sqrt{2(C/N_0)_k T} \sqrt{\frac{f_s}{B}} R\left(\tau_k + \frac{\Delta_{EL}}{2}\right) \operatorname{sinc}(\delta f_k T) \sin \delta\phi_k + \eta_{Q_L,k} \quad (3.212)$$

Δ_{EL} is the chip spacing between the early and the late correlator. It is usually 0.5 or smaller in modern receivers which are more robust against multipath. The noise of the early, prompt and late branches of the in-phase as well as quadrature phase correlators is now normalized

$$\eta_{X_i} \sim WN(0, 1), \quad E[\eta_{X_i,l} \eta_{X_i,k}] = 0 \quad \forall k \neq l, \quad X = \{I, Q\}, \quad i = \{E, P, L\} \quad (3.213)$$

Although the noise of the in-phase and quadrature correlators stem from the same thermal noise on the input signal, it is assumed that the in-phase and quadrature noise are not correlated

$$E[\eta_{I_i,l} \eta_{Q_i,k}] = 0 \quad \forall k, l, \quad i = \{E, P, L\} \quad (3.214)$$

However, the early, prompt and late noises are correlated according to [93]

$$\begin{aligned} \mathbb{E}[\eta_{I_E,k} \eta_{I_L,k}] &= R(\Delta_{EL}) = 1 - \Delta_{EL} \\ \mathbb{E}[\eta_{I_E,k} \eta_{I_P,k}] &= R\left(\frac{\Delta_{EL}}{2}\right) = 1 - \frac{\Delta_{EL}}{2} \\ \mathbb{E}[\eta_{I_P,k} \eta_{I_L,k}] &= R\left(\frac{\Delta_{EL}}{2}\right) = 1 - \frac{\Delta_{EL}}{2} \end{aligned} \quad (3.215)$$

The carrier-to-noise ratio has to be inserted in the formulas with the unit Hz. If the value is given in dB-Hz, it has to be converted at first by

$$\frac{C}{N_0} [\text{Hz}] = 10^{\frac{C/N_0 [\text{dB-Hz}]}{10}} \quad (3.216)$$

The receiver estimates the current carrier-to-noise ratio from the prompt in-phase and quadrature phase samples. The estimate contains already all noise sources like the thermal noise, the clock noise or the quantization noise. They do not have to be considered separately.

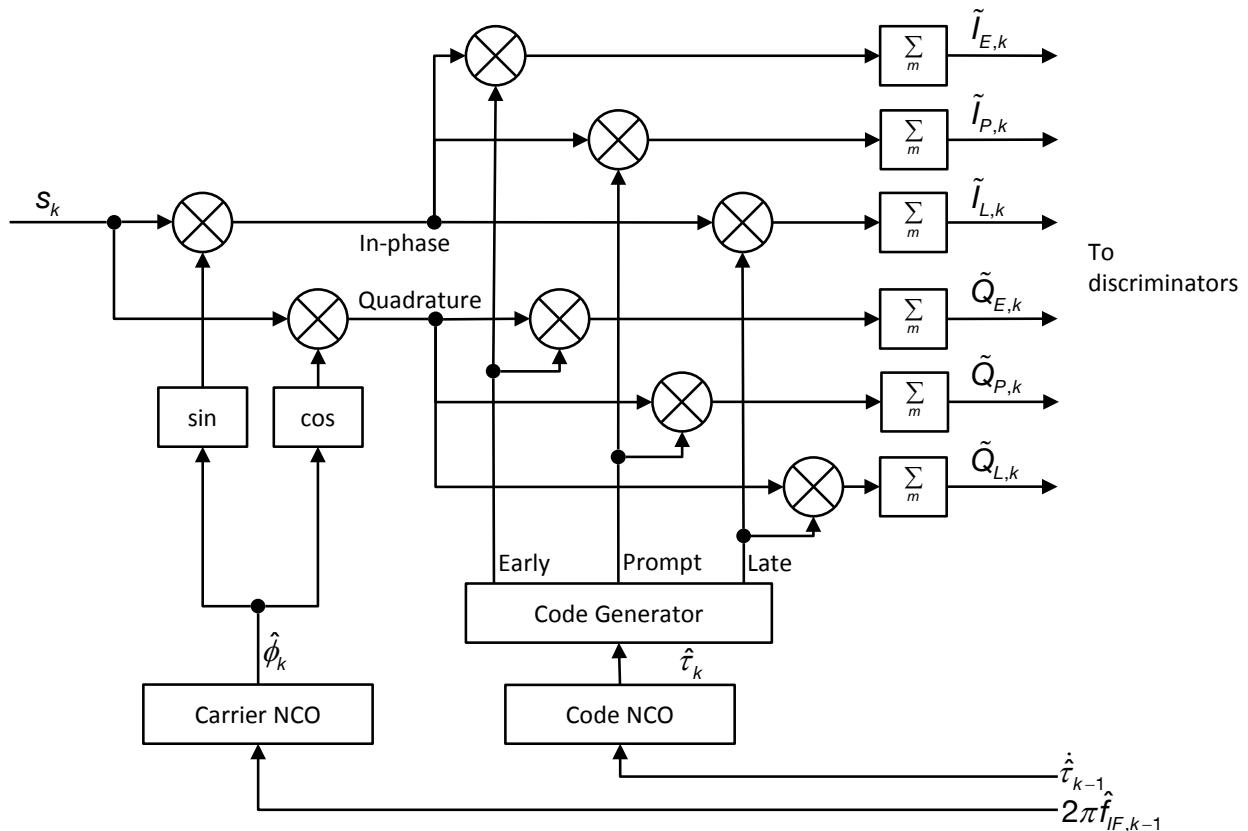


Figure 3-110: Correlator

3.5.6.4 Discriminators

The correlator outputs the accumulated early, prompt and late in-phase and quadrature samples $\tilde{I}_{E,k}$, $\tilde{I}_{P,k}$, $\tilde{I}_{L,k}$, $\tilde{Q}_{E,k}$, $\tilde{Q}_{P,k}$ and $\tilde{Q}_{L,k}$. Discriminators are used to extract the phase error $\delta\phi_k$, frequency error δf_k and code error $\delta\tau_k$ from the early, prompt and late in-phase and quadrature phase correlator outputs. Common discriminators are listed in [64].

3.5.6.4.1 Phase

A popular discriminator for the phase error $\delta\phi_k$ is the two-quadrant arctangent discriminator

$$\tilde{d}_{\phi,k} = \arctan \frac{\tilde{Q}_{P,k}}{\tilde{I}_{P,k}} \quad (3.217)$$

which requires the prompt in-phase and quadrature phase correlator outputs. The phase discriminator is a non-linear function of the in-phase and quadrature phase samples $\tilde{I}_{P,k}$ and $\tilde{Q}_{P,k}$. In order to be able to use the discriminator later in the linear tracking loop filter, it has to be linearized. For that, the involved values are replaced by the sums of noise-free values and pure noise

$$d_{\phi,k} + \eta_{d_{\phi,k}} = \arctan \frac{Q_{P,k} + \eta_{Q_{P,k}}}{I_{P,k} + \eta_{I_{P,k}}} \quad (3.218)$$

where $d_{\phi,k} = E[\tilde{d}_{\phi,k}]$, $Q_{P,k} = E[\tilde{Q}_{P,k}]$ and $I_{P,k} = E[\tilde{I}_{P,k}]$. With the first order Taylor series expansion of the arctangent function, the discriminator can be rewritten as

$$d_{\phi,k} + \eta_{d_{\phi,k}} = \arctan \frac{Q_{P,k}}{I_{P,k}} + \frac{I_{P,k}}{I_{P,k}^2 + Q_{P,k}^2} \eta_{Q_{P,k}} - \frac{Q_{P,k}}{I_{P,k}^2 + Q_{P,k}^2} \eta_{I_{P,k}} + h.o.t. \quad (3.219)$$

When the expected values of the prompt correlator outputs (3.208) and (3.211) are inserted, it can be seen that the discriminator directly represents the carrier phase error

$$d_{\phi,k} = \arctan \frac{Q_{P,k}}{I_{P,k}} = \arctan \frac{\sqrt{2(C/N_0)_k T} R(\tau_k) \text{sinc}(\delta f_k T) \sin \delta\phi_k}{\sqrt{2(C/N_0)_k T} R(\tau_k) \text{sinc}(\delta f_k T) \cos \delta\phi_k} = \delta\phi_k \quad (3.220)$$

The phase discriminator noise is, considering only the linear terms in (3.219)

$$\eta_{d_{\phi,k}} \doteq \frac{I_{P,k}}{I_{P,k}^2 + Q_{P,k}^2} \eta_{Q_{P,k}} - \frac{Q_{P,k}}{I_{P,k}^2 + Q_{P,k}^2} \eta_{I_{P,k}} \quad (3.221)$$

The variance of the discriminator noise is calculated by

$$Q_{d_{\phi,k}} = E[\eta_{d_{\phi,k}}^2] = \frac{I_{P,k}^2 E[\eta_{Q_{P,k}}^2] + Q_{P,k}^2 E[\eta_{I_{P,k}}^2] - I_{P,k} Q_{P,k} E[\eta_{Q_{P,k}} \eta_{I_{P,k}}]}{(I_{P,k}^2 + Q_{P,k}^2)^2} \quad (3.222)$$

With (3.213) and (3.214), the variance simplifies to

$$Q_{d_{\phi,k}} = E[\eta_{d_{\phi,k}}^2] = \frac{1}{I_{P,k}^2 + Q_{P,k}^2} \quad (3.223)$$

With the correlator outputs (3.208) and (3.211), the denominator becomes

$$I_{P,k}^2 + Q_{P,k}^2 = 2(C/N_0)_k T R^2(\tau_k) \text{sinc}^2(\delta f_k T) \quad (3.224)$$

The cardinal sine function can be approximated by one, $\text{sinc}^2(\delta f_k T) \approx 1$, since the frequency deviation δf_k is assumed to be small if the tracking loops are settled. The same is valid for the square of the correlation function. It can be replaced by one, $R^2(\tau_k) = (1 - |\tau_k|)^2 \approx 1$, because the code deviation τ_k is small. Hence

$$I_{P,k}^2 + Q_{P,k}^2 \approx 2(C/N_0)_k T \quad (3.225)$$

The phase discriminator noise variance finally becomes

$$Q_{d_\phi,k} \approx \frac{1}{2(C/N_0)_k T} \quad (3.226)$$

3.5.6.4.2 Frequency

The difference quotient of subsequent phase discriminators $d_{\phi,k}$ and $d_{\phi,k-1}$ can be simply used as frequency discriminator

$$\tilde{d}_{f,k} = \frac{\tilde{d}_{\phi,k} - \tilde{d}_{\phi,k-1}}{T} = \frac{1}{T} \left(\arctan \frac{\tilde{Q}_{P,k}}{\tilde{I}_{P,k}} - \arctan \frac{\tilde{Q}_{P,k-1}}{\tilde{I}_{P,k-1}} \right) \quad (3.227)$$

The values are separated in the expected values and noise

$$d_{f,k} + \eta_{d_{f,k}} = \frac{1}{T} \left(\arctan \frac{Q_{P,k} + \eta_{Q_{P,k}}}{I_{P,k} + \eta_{I_{P,k}}} - \arctan \frac{Q_{P,k-1} + \eta_{Q_{P,k-1}}}{I_{P,k-1} + \eta_{I_{P,k-1}}} \right) \quad (3.228)$$

The expected value of the frequency discriminator is the difference quotient of the phase error

$$d_{f,k} = \frac{1}{T} \left(\arctan \frac{Q_{P,k}}{I_{P,k}} - \arctan \frac{Q_{P,k-1}}{I_{P,k-1}} \right) = \frac{1}{T} (\delta\phi_k - \delta\phi_{k-1}) \quad (3.229)$$

and the frequency discriminator noise is

$$\eta_{d_{f,k}} \doteq \frac{1}{T} \left(\frac{I_{P,k}}{I_{P,k}^2 + Q_{P,k}^2} \eta_{Q_{P,k}} - \frac{Q_{P,k}}{I_{P,k}^2 + Q_{P,k}^2} \eta_{I_{P,k}} - \frac{I_{P,k-1}}{I_{P,k-1}^2 + Q_{P,k-1}^2} \eta_{Q_{P,k-1}} + \frac{Q_{P,k-1}}{I_{P,k-1}^2 + Q_{P,k-1}^2} \eta_{I_{P,k-1}} \right) \quad (3.230)$$

Analogous to (3.223), the variance of the frequency discriminator noise is given with

$$Q_{d_{f,k}} = E[\eta_{d_{f,k}}^2] = \frac{1}{T^2} \left(\frac{1}{I_{P,k}^2 + Q_{P,k}^2} + \frac{1}{I_{P,k-1}^2 + Q_{P,k-1}^2} \right) \quad (3.231)$$

and with the signal power (3.225) it is approximated by

$$Q_{d_{f,k}} \approx \frac{1}{(C/N_0)_k T^3} \quad (3.232)$$

if it is assumed that the carrier-to-noise ratio does not change much between the subsequent correlator time steps.

3.5.6.4.3 Code

For the determination of the code error τ the quasi-coherent dot product power is proposed

$$\tilde{d}_{\tau,k} = \frac{1}{N} \left[(\tilde{I}_{E,k} - \tilde{I}_{L,k}) \tilde{I}_{P,k} + (\tilde{Q}_{E,k} - \tilde{Q}_{L,k}) \tilde{Q}_{P,k} \right] \quad (3.233)$$

The scale factor N can be the current signal power $\tilde{I}_{P,k}^2 + \tilde{Q}_{P,k}^2$ or a constant that represents the mean signal power. Separation of the involved values in expected values and noise gives

$$\begin{aligned} \tilde{d}_{\tau,k} + \eta_{d_{\tau,k}} &= \frac{1}{N} \left[(I_{E,k} + \eta_{I_{E,k}} - I_{L,k} - \eta_{I_{L,k}})(I_{P,k} + \eta_{I_{P,k}}) \right. \\ &\quad \left. + (Q_{E,k} + \eta_{Q_{E,k}} - Q_{L,k} - \eta_{Q_{L,k}})(Q_{P,k} + \eta_{Q_{P,k}}) \right] \end{aligned} \quad (3.234)$$

where $d_{\tau,k} = E[\tilde{d}_{\tau,k}]$. With the expected values of the correlator outputs (3.207) – (3.212), the expected code discriminator value becomes

$$d_{\tau,k} = \frac{1}{N} \cdot 2(C/N_0)_k T \operatorname{sinc}^2(\delta f_k T) \left(R\left(\tau_k - \frac{\Delta_{EL}}{2}\right) - R\left(\tau_k + \frac{\Delta_{EL}}{2}\right) \right) R(\tau_k) \quad (3.235)$$

Inserting the correlation function (3.202) and approximating $\operatorname{sinc}^2(\delta f_k T) \approx 1$ yields

$$d_{\tau,k} \approx \frac{1}{N} \cdot 2(C/N_0)_k T \left(\left| \tau_k + \frac{\Delta_{EL}}{2} \right| - \left| \tau_k - \frac{\Delta_{EL}}{2} \right| \right) (1 - |\tau_k|) \quad (3.236)$$

Moreover, the absolute code deviation $|\tau_k|$ is assumed to be small under settled conditions so that it is definitely smaller than half of the early late correlator spacing, $|\tau_k| < \Delta_{EL}/2$. The lag of the late correlator is consequently positive and the lag of the early correlator is negative and the difference of the absolute values in (3.236) can be replaced by

$$\left| \tau + \frac{\Delta_{EL}}{2} \right| - \left| \tau - \frac{\Delta_{EL}}{2} \right| = \tau + \frac{\Delta_{EL}}{2} - \left(-\tau + \frac{\Delta_{EL}}{2} \right) = 2\tau \quad (3.237)$$

The code discriminator approximation (3.236) becomes

$$d_{\tau,k} \approx \frac{1}{N} \cdot 4(C/N_0)_k T \tau_k (1 - |\tau_k|) \quad (3.238)$$

With the small code deviation assumption, $\tau_k |\tau_k| \approx 0$ is valid

$$d_{\tau,k} \approx \frac{1}{N} \cdot 4(C/N_0)_k T \tau_k \quad (3.239)$$

The code discriminator $d_{\tau,k}$ thus depends linearly on the code deviation τ_k . It is multiplied by the gain

$$K_{d_{\tau,k}} = \frac{1}{N} \cdot 4(C/N_0)_k T \approx 2 \quad (3.240)$$

The code discriminator noise considering only linear terms is

$$\eta_{d_{\tau,k}} \doteq \frac{1}{N} \left[I_{P,k} (\eta_{I_E,k} - \eta_{I_L,k}) + Q_{P,k} (\eta_{Q_E,k} - \eta_{Q_L,k}) + (I_{E,k} - I_{L,k}) \eta_{I_P,k} + (Q_{E,k} - Q_{L,k}) \eta_{Q_P,k} \right] \quad (3.241)$$

The variance of the code discriminator noise is calculated by

$$\begin{aligned} Q_{d_{\tau,k}} = E[\eta_{d_{\tau,k}}^2] &= \frac{1}{N^2} \left[I_{P,k}^2 \left(E[\eta_{I_E,k}^2] + E[\eta_{I_L,k}^2] - 2E[\eta_{I_E,k} \eta_{I_L,k}] \right) \right. \\ &\quad + Q_{P,k}^2 \left(E[\eta_{Q_E,k}^2] + E[\eta_{Q_L,k}^2] - 2E[\eta_{Q_E,k} \eta_{Q_L,k}] \right) \\ &\quad + (I_{E,k} - I_{L,k})^2 E[\eta_{I_P,k}^2] + (Q_{E,k} - Q_{L,k})^2 E[\eta_{Q_P,k}^2] \\ &\quad \left. + 2I_{P,k} (I_{E,k} - I_{L,k}) E[\eta_{I_E,k} \eta_{I_P,k}] - 2I_{P,k} (I_{E,k} - I_{L,k}) E[\eta_{I_P,k} \eta_{I_L,k}] \right. \\ &\quad \left. + 2Q_{P,k} (Q_{E,k} - Q_{L,k}) E[\eta_{Q_E,k} \eta_{Q_P,k}] - 2Q_{P,k} (Q_{E,k} - Q_{L,k}) E[\eta_{Q_P,k} \eta_{Q_L,k}] \right] \end{aligned} \quad (3.242)$$

With (3.213) and (3.215), the variance simplifies to

$$Q_{d_{\tau,k}} = \frac{1}{N^2} \left[2I_{P,k}^2 (1 - R(\Delta_{EL})) + 2Q_{P,k}^2 (1 - R(\Delta_{EL})) + (I_{E,k} - I_{L,k})^2 + (Q_{E,k} - Q_{L,k})^2 \right] \quad (3.243)$$

The correlation function (3.202) is inserted

$$Q_{d_r,k} = \frac{1}{N^2} \left[2 \Delta_{EL} (I_{P,k}^2 + Q_{P,k}^2) + (I_{E,k} - I_{L,k})^2 + (Q_{E,k} - Q_{L,k})^2 \right] \quad (3.244)$$

With (3.225) and the early and late correlator outputs (3.207), (3.209), (3.210) and (3.212) and the assumptions $\text{sinc}^2(\delta f_k T) \approx 1$ and $R^2(\tau_k) = (1 - |\tau_k|)^2 \approx 1$, the code discriminator noise variance is approximated by

$$Q_{d_r,k} \approx \frac{1}{N^2} \left[4 \Delta_{EL} (C/N_0)_k T + 2 (C/N_0)_k T \left(R\left(\tau_k - \frac{\Delta_{EL}}{2}\right) - R\left(\tau_k + \frac{\Delta_{EL}}{2}\right) \right)^2 \right] \quad (3.245)$$

or with the correlation function (3.202) inserted

$$Q_{d_r,k} \approx \frac{1}{N^2} \left[4 \Delta_{EL} (C/N_0)_k T + 2 (C/N_0)_k T \left(\left| \tau_k + \frac{\Delta_{EL}}{2} \right| - \left| \tau_k - \frac{\Delta_{EL}}{2} \right| \right)^2 \right] \quad (3.246)$$

With (3.237), the variance reduces to

$$Q_{d_r,k} \approx \frac{1}{N^2} (4 \Delta_{EL} (C/N_0)_k T + 8 (C/N_0)_k T \tau_k^2) \quad (3.247)$$

and can finally be approximated by

$$Q_{d_r,k} \approx \frac{4 \Delta_{EL}}{N^2} (C/N_0)_k T \quad (3.248)$$

if again the code error τ_k is assumed to be small, $\tau_k^2 \approx 0$.

3.5.6.4.4 Cross-Correlation

For the state-space model of the closed tracking loops, the cross-correlations between the phase discriminator noise η_{d_ϕ} , the frequency discriminator noise η_{d_f} and the code discriminator noise η_{d_r} will be required. From (3.221) and (3.230), the cross-correlation between η_{d_ϕ} and η_{d_r} is derived analogous to (3.222) and (3.223) as

$$\mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] = \frac{1}{T} \frac{1}{I_{P,k}^2 + Q_{P,k}^2} \quad (3.249)$$

and becomes with (3.225)

$$\mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] \approx \frac{1}{2(C/N_0)_k T^2} \quad (3.250)$$

The cross-correlation between η_{d_ϕ} and η_{d_f} is with (3.221) and (3.241)

$$\begin{aligned} \mathbb{E}[\eta_{d_\phi,k} \eta_{d_f,k}] &= \frac{1}{N(I_{P,k}^2 + Q_{P,k}^2)} \left[I_{P,k} (Q_{E,k} - Q_{L,k}) \mathbb{E}[\eta_{Q_P,k}^2] + Q_{P,k} (I_{E,k} - I_{L,k}) \mathbb{E}[\eta_{I_P,k}^2] \right. \\ &\quad \left. + I_{P,k} Q_{P,k} (\mathbb{E}[\eta_{Q_E,k} \eta_{Q_P,k}] - \mathbb{E}[\eta_{Q_P,k} \eta_{Q_L,k}] - \mathbb{E}[\eta_{I_E,k} \eta_{I_P,k}] + \mathbb{E}[\eta_{I_P,k} \eta_{I_L,k}]) \right] \end{aligned} \quad (3.251)$$

With (3.213) and (3.215), the cross-correlation simplifies to

$$\mathbb{E}[\eta_{d_\phi,k} \eta_{d_f,k}] = \frac{1}{N} \frac{I_{P,k} (Q_{E,k} - Q_{L,k}) + Q_{P,k} (I_{E,k} - I_{L,k})}{I_{P,k}^2 + Q_{P,k}^2} \quad (3.252)$$

The numerator is just the expected value of the quasi-coherent dot product power discriminator (3.233). It can be approximated by (3.239). The denominator is the expected signal power (3.225). The cross-correlation thus becomes approximately

$$\mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] \approx \frac{2 \tau_k}{N} \quad (3.253)$$

The remaining cross-correlation between η_{d_r} and η_{d_r} is with (3.230) and (3.241)

$$\begin{aligned} \mathbb{E}[\eta_{d_r,k} \eta_{d_r,k}] &= \frac{1}{TN(I_{P,k}^2 + Q_{P,k}^2)} [I_{P,k}(Q_{E,k} - Q_{L,k})\mathbb{E}[\eta_{Q_P,k}^2] + Q_{P,k}(I_{E,k} - I_{L,k})\mathbb{E}[\eta_{I_P,k}^2]] \\ &\quad + I_{P,k}Q_{P,k}(\mathbb{E}[\eta_{Q_E,k} \eta_{Q_P,k}] - \mathbb{E}[\eta_{Q_P,k} \eta_{Q_L,k}] - \mathbb{E}[\eta_{I_E,k} \eta_{I_P,k}] + \mathbb{E}[\eta_{I_P,k} \eta_{I_L,k}]) \end{aligned} \quad (3.254)$$

Compared with (3.251), the cross-correlation is analogous to (3.253)

$$\mathbb{E}[\eta_{d_r,k} \eta_{d_r,k}] = \frac{1}{T} \mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] \approx \frac{2 \tau_k}{TN} \quad (3.255)$$

Since the code deviation τ_k is small, the cross-correlations (3.253) and (3.255) can be assumed to be zero.

Furthermore, the correlations $\mathbb{E}[\eta_{d_\phi,k-1} \eta_{d_r,k}]$, $\mathbb{E}[\eta_{d_r,k-1} \eta_{d_r,k}]$ and $\mathbb{E}[\eta_{d_r,k-1} \eta_{d_r,k}]$ between two subsequent correlator time steps are necessary for the state-space model of the closed tracking loop. The first cross-correlation is derived as

$$\mathbb{E}[\eta_{d_\phi,k-1} \eta_{d_r,k}] = -\mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] \approx -\frac{1}{2(C/N_0)_{k-1} T^2} \quad (3.256)$$

The second correlation and its approximation are given with

$$\mathbb{E}[\eta_{d_r,k-1} \eta_{d_r,k}] = -\frac{1}{T} \mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] \approx -\frac{1}{2(C/N_0)_{k-1} T^3} \quad (3.257)$$

and finally the third correlation is

$$\mathbb{E}[\eta_{d_r,k-1} \eta_{d_r,k}] = \frac{1}{T} \mathbb{E}[\eta_{d_\phi,k} \eta_{d_r,k}] \approx \frac{2 \tau_k}{TN} \quad (3.258)$$

3.5.6.5 Tracking Loop Model

3.5.6.5.1 Structure

The discriminator outputs are fed into the PLL filter to estimate the current intermediate frequency of the incoming signal $\hat{f}_{IF,k}$ on the one hand and into the DLL filter to estimate the current ranging code rate $\dot{\hat{r}}_k$ on the other hand. The estimated frequency is used to steer the NCO of the carrier signal replica and the estimated code rate adapts the NCO of the code replica. Mathematically spoken, the NCOs are numerical integrators. The phase lock and delay lock (control) loops are closed via the correlation of the incoming signal with the signal replica. The overall structure of the tracking loops is illustrated in Figure 3-111.

For the PLL and DLL IIR filter from zeroth to third order are applied. The poles of the closed loops are chosen such that the filtered noise is as low as possible, but the filter does not constrain the signal

dynamics owing to the actual motion of the receiver. The effective cut-off frequencies of the filters thus have to be selected in accordance with the expected dynamics of the carrier platform.

Tracking loops are generally designed in continuous-time and transformed to discrete-time afterwards. For this purpose, the continuous-time integrator is replaced by a forward Euler, backward Euler or trapezoidal discrete-time integrator. The latter method is also known as bilinear transformation.

For the design of the tracking loops, a simple linear model is used. The structure of the PLL and the DLL is principally the same. The discriminator is represented by a linear gain, the NCO by an integrator and the tracking loop filter by the transfer function $F(s)$. The closed-loop transfer function including discriminator gain, tracking loop filter transfer function and NCO integrator is $H(s)$.

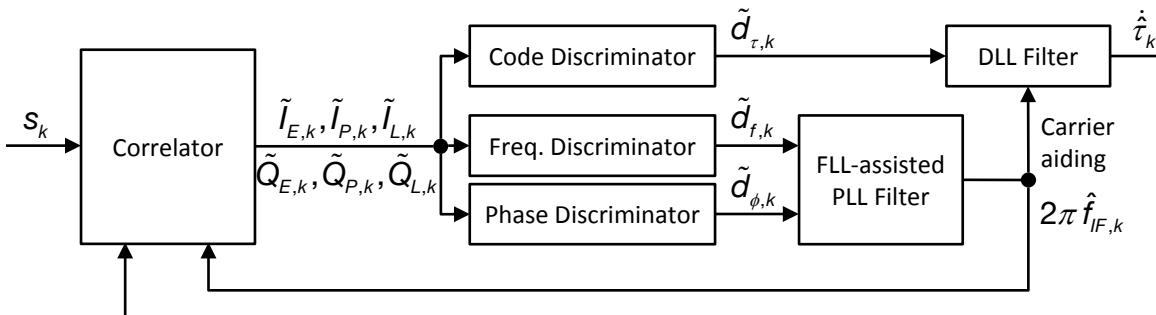


Figure 3-111: Tracking loop structure

3.5.6.5.2 Frequency-Assisted Phase Lock Loop

The block diagram of the third order PLL that is assisted by a second order frequency lock loop (FLL) is shown in Figure 3-112. A specialty of the controller design is that the plant is a simple integrator, which is the carrier NCO. The poles of the closed loop transfer function can hence be arbitrarily placed by the user.

The continuous-time transfer function of the PLL filter

$$\dot{\phi}_\phi(s) = F_\phi(s) d_\phi(s) \quad (3.259)$$

is given by

$$F_\phi(s) = \frac{a_{0,\phi} s^2 + a_{1,\phi} s + a_{2,\phi}}{s^2} \quad (3.260)$$

The continuous-time transfer function of the FLL filter

$$\dot{\phi}_f(s) = F_f(s) d_f(s) \quad (3.261)$$

is given by

$$F_f(s) = \frac{a_{0,f} s + a_{1,f}}{s} \quad (3.262)$$

The combined PLL/FLL filter is obtained by superposing the outputs of both single filters

$$\begin{aligned} \dot{\phi}(s) &= \dot{\phi}_\phi(s) + \dot{\phi}_f(s) \\ &= F_\phi(s) d_\phi(s) + F_f(s) d_f(s) \end{aligned} \quad (3.263)$$

The closed-loop transfer function including the carrier NCO integrator and phase and frequency discriminators is given by

$$\begin{aligned} H(s) &= \frac{F_f(s)K_{d_f}s + F_\phi(s)K_{d_\phi}}{(1+F_f(s)K_{d_f})s + F_\phi(s)K_{d_\phi}} \\ &= \frac{a_{0,f}K_{d_f}s^3 + (a_{1,f}K_{d_f} + a_{0,\phi}K_{d_\phi})s^2 + a_{1,\phi}K_{d_\phi}s + a_{2,\phi}K_{d_\phi}}{s^4 + a_{0,f}K_{d_f}s^3 + (a_{1,f}K_{d_f} + a_{0,\phi}K_{d_\phi})s^2 + a_{1,\phi}K_{d_\phi}s + a_{2,\phi}K_{d_\phi}} \end{aligned} \quad (3.264)$$

For the implementation in program code, the continuous-time tracking loop transfer function is discretized next. This is accomplished here by replacing the continuous-time integrators by simple forward Euler integrators

$$y[n] = y[n-1] + T u[n-1] \quad (3.265)$$

or Z-transformed

$$Y(z) = \frac{Tz^{-1}}{1-z^{-1}} U(z) \quad (3.266)$$

The frequency-assisted PLL of Figure 3-112, now with discrete-time integrators and a difference quotient instead of the derivative, is shown in Figure 3-113.

The discrete-time correspondent of the closed-loop transfer function $H(s)$ of (3.264), represented as discrete-time state-space model, is

$$\mathbf{z}_{PLL,k} = \underbrace{\begin{pmatrix} 1-Ta_{0,\phi}K_{d_\phi} & T & 0 & 0 \\ -Ta_{1,\phi}K_{d_\phi} - a_{0,f}K_{d_f} & 1 & T & a_{0,f}K_{d_f} \\ -Ta_{2,\phi}K_{d_\phi} - a_{1,f}K_{d_f} & 0 & 1 & a_{1,f}K_{d_f} \\ 1 & 0 & 0 & 0 \end{pmatrix}}_{\Phi_{PLL}} \mathbf{z}_{PLL,k-1} + \underbrace{\begin{pmatrix} Ta_{0,\phi} \\ Ta_{1,\phi} \\ Ta_{2,\phi} \\ 0 \end{pmatrix}}_{\Gamma_{PLL,d_\phi}} \eta_{d_\phi,k-1} + \underbrace{\begin{pmatrix} 0 \\ Ta_{0,f} \\ Ta_{1,f} \\ 0 \end{pmatrix}}_{\Gamma_{PLL,d_f}} \eta_{d_f,k-1} \quad (3.267)$$

where the state vector is defined as

$$\mathbf{z}_{PLL,k} = \begin{pmatrix} z_{\phi,k} \\ z_{\omega,k} \\ z_{\dot{\omega},k} \\ z_{\phi,k-1} \end{pmatrix} \quad (3.268)$$

Therein, the last element $z_{\phi,k-1}$ is the estimated carrier phase error of the previous time step. It is required for the frequency discriminator difference quotient. The estimated carrier phase error output equation is given with

$$\delta\phi_k = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \end{pmatrix}}_{\mathbf{H}_\phi} \mathbf{z}_{PLL,k} \quad (3.269)$$

The estimated carrier frequency error output equation is

$$\delta f_k = \underbrace{\frac{1}{2\pi} \begin{pmatrix} -a_{0,\phi}K_{d_\phi} & 1 & 0 & 0 \end{pmatrix}}_{\mathbf{H}_f} \mathbf{z}_{PLL,k} + \underbrace{\frac{1}{2\pi} a_{0,\phi} \eta_{d_\phi,k}}_{D_f} \quad (3.270)$$

Usually, the filter coefficients are chosen as

$$\begin{aligned} a_{2,\phi} &= \omega_{0,\phi}^3, & a_{1,\phi} &= 2\omega_{0,\phi}^2, & a_{0,\phi} &= 2\omega_{0,\phi} \\ a_{1,f} &= \omega_{0,f}^2, & a_{0,f} &= \sqrt{2}\omega_{0,f} \end{aligned} \quad (3.271)$$

where $\omega_{0,\phi}$ is the natural frequency of the PLL part and $\omega_{0,f}$ is the natural frequency of the FLL part. The poles of the PLL transfer function lie on the negative real part axis and on lines with inclination $\pm\sqrt{3}$ in the stable left hand side of the complex plane and move with varying $\omega_{0,\phi}$. The poles of the FLL transfer function lie on lines with inclination $\pm\sqrt{2}$ and move with varying $\omega_{0,f}$. The locations of the poles are illustrated in Figure 3-114 for different frequencies. If the filter coefficients are chosen according to (3.271), the 3dB bandwidth B_{PLL} (cut-off frequency specified in Hz) of the third order PLL part is $B_{PLL} = 5/6\omega_{0,\phi}$ [64]; the corresponding relation of the second order FLL part is $B_{FLL} = 4/3 \cdot \sqrt{2} \omega_{0,f}$.

Since the filter has been designed in continuous-time, but the filter has to be converted to discrete-time by means of the forward Euler integrator, the locations of the poles and hence the stability of the filter has to be additionally proven in discrete-time. In the case of the discriminators (3.220) and (3.229), the gains K_{d_ϕ} and K_{d_f} are one.

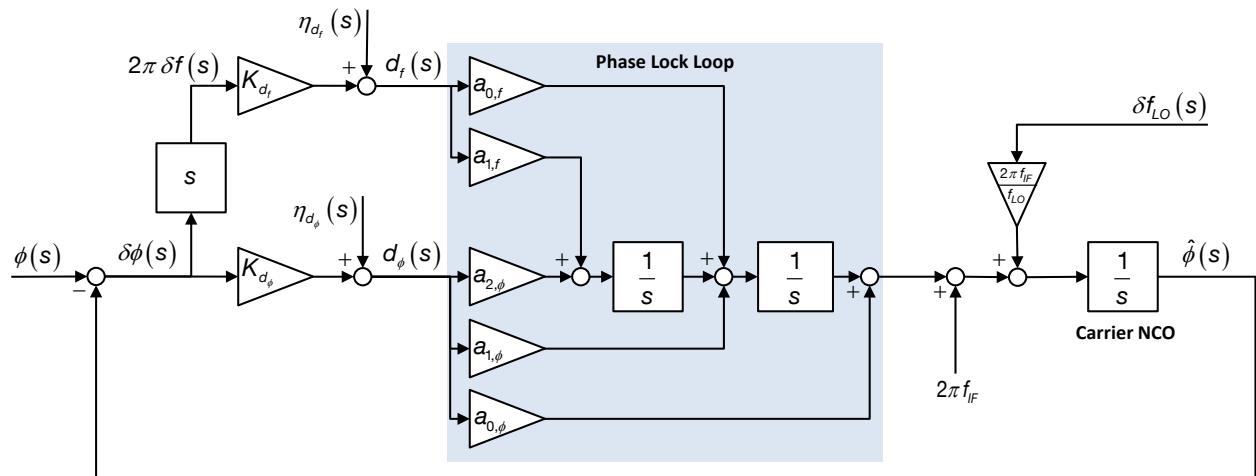


Figure 3-112: Continuous-time frequency-assisted PLL

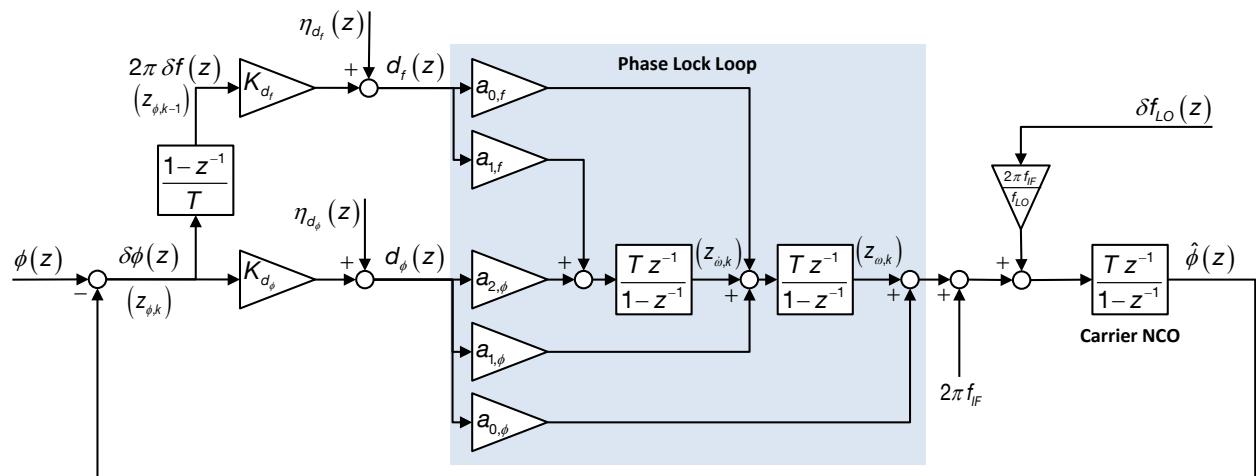


Figure 3-113: Discrete-time frequency-assisted PLL

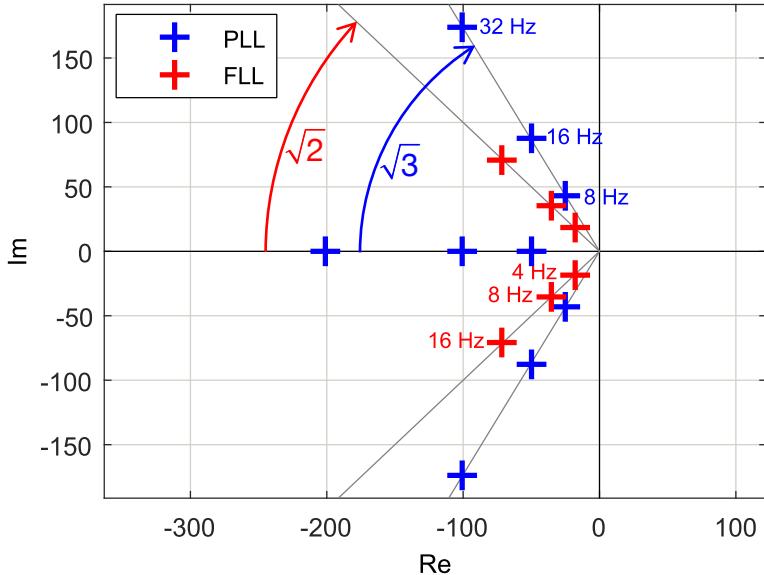


Figure 3-114: Poles of the continuous-time filter loop transfer function

3.5.6.5.3 Carrier-Aided Delay Lock Loop

The task of the DLL is to align the code replica with the code of the incoming signal to obtain the pseudorange measurement. If the PLL is in lock, the expected code delay can be calculated by means of the estimated intermediate frequency of the carrier wave and used as feed-forward for the DLL. The DLL controller then only has to diminish the residual error that stems for example from the limited frequency resolution of the NCOs and a zeroth order tracking loop is therefore sufficient. The structure of the carrier-aided DLL together with the third order frequency-assisted PLL of the previous section is illustrated in Figure 3-115.

The discrete-time state space model of the closed-loop DLL is

$$\begin{aligned} z_{DLL,k} &= \underbrace{\left(1 - T a_{0,\tau} K_{d_\tau}\right)}_{\Phi_{DLL}} z_{DLL,k-1} + \underbrace{T a_{0,\tau} \eta_{d_\tau,k-1}}_{\Gamma_{DLL,d_\tau}} + T r (\hat{f}_{IF,k-1} - f_{IF}) \\ \delta\tau_k &= z_{DLL,k} \end{aligned} \quad (3.272)$$

$\hat{f}_{IF,k}$ is the current estimate of the intermediate frequency of the PLL. $r (\hat{f}_{IF,k-1} - f_{IF})$ is the feed-forward branch of the DLL controller. Therein

$$r = \frac{f_{code}}{f_{carrier}} \quad (3.273)$$

is the relation between the nominal code chip rate f_{code} and the carrier signal frequency $f_{carrier}$. In the case of the GPS L1 signal and C/A code, the code rate is $f_{code} = 1.023 \text{ MHz}$, the carrier frequency is $f_{carrier} = 1575.42 \text{ MHz}$ and the relation r is hence $r = 1/1540$. It scales the estimated frequency deviation of the carrier down to code frequency. The filter coefficient $a_{0,\tau}$ corresponds to the natural frequency of the DLL $\omega_{0,\tau}$. It is related to the 3dB bandwidth B_{DLL} by $\omega_{0,\tau} = 4B_{DLL}$ [64]. (3.274) becomes

$$\begin{aligned} z_{DLL,k} &= \Phi_{DLL} z_{DLL,k-1} + \Gamma_{DLL,d_\tau} \eta_{d_\tau,k-1} + T r \delta\tau_k \\ &= \Phi_{DLL} z_{DLL,k-1} + \Gamma_{DLL,d_\tau} \eta_{d_\tau,k-1} + T r (\mathbf{H}_f \mathbf{z}_{PLL,k} + D_f \eta_{d_\phi,k}) \end{aligned} \quad (3.274)$$

3.5.6.5.4 Complete Tracking Loop

The overall tracking loop model is composed of the frequency-assisted carrier tracking loop model (3.267) and the code tracking loop model (3.272)

$$\underbrace{\begin{pmatrix} \mathbf{z}_{PLL,k} \\ \mathbf{z}_{DLL,k} \end{pmatrix}}_{\mathbf{z}_k} = \underbrace{\begin{pmatrix} \Phi_{PLL} & 0 \\ Tr\mathbf{H}_f & \Phi_{DLL} \end{pmatrix}}_{\Phi} \begin{pmatrix} \mathbf{z}_{PLL,k-1} \\ \mathbf{z}_{DLL,k-1} \end{pmatrix} + \underbrace{\begin{pmatrix} \boldsymbol{\Gamma}_{PLL,d_\phi} \\ TrD_f \end{pmatrix}}_{\boldsymbol{\Gamma}_{d_\phi}} \boldsymbol{\eta}_{d_\phi,k-1} + \underbrace{\begin{pmatrix} \boldsymbol{\Gamma}_{PLL,d_f} \\ 0 \end{pmatrix}}_{\boldsymbol{\Gamma}_{d_f}} \boldsymbol{\eta}_{d_f,k-1} + \underbrace{\begin{pmatrix} 0 \\ \boldsymbol{\Gamma}_{DLL,d_\tau} \end{pmatrix}}_{\boldsymbol{\Gamma}_{d_\tau}} \boldsymbol{\eta}_{d_\tau,k-1} \quad (3.275)$$

The covariance propagation equation is given with

$$\mathbf{P}_k = \Phi \mathbf{P}_{k-1} \Phi^T + \mathbf{Q}_k, \quad \mathbf{P}_0 = 0 \quad (3.276)$$

where the process noise covariance matrix \mathbf{Q}_k is composed of

$$\begin{aligned} \mathbf{Q}_k = & \boldsymbol{\Gamma}_{d_\phi} \mathbf{Q}_{d_\phi,k} \boldsymbol{\Gamma}_{d_\phi}^T + \boldsymbol{\Gamma}_{d_f} \mathbf{Q}_{d_f,k} \boldsymbol{\Gamma}_{d_f}^T + \boldsymbol{\Gamma}_{d_\tau} \mathbf{Q}_{d_\tau,k} \boldsymbol{\Gamma}_{d_\tau}^T \\ & + \Phi \mathbf{E}[\mathbf{z}_k \boldsymbol{\eta}_{d_f,k}] \boldsymbol{\Gamma}_{d_f}^T + \boldsymbol{\Gamma}_{d_f} \mathbf{E}[\boldsymbol{\eta}_{d_f,k} \mathbf{z}_k^T] \Phi^T \\ & + \boldsymbol{\Gamma}_{d_\phi} \mathbf{E}[\boldsymbol{\eta}_{d_\phi,k} \boldsymbol{\eta}_{d_f,k}] \boldsymbol{\Gamma}_{d_f}^T + \boldsymbol{\Gamma}_{d_f} \mathbf{E}[\boldsymbol{\eta}_{d_f,k} \boldsymbol{\eta}_{d_\phi,k}] \boldsymbol{\Gamma}_{d_\phi}^T \\ & + \boldsymbol{\Gamma}_{d_\phi} \mathbf{E}[\boldsymbol{\eta}_{d_\phi,k} \boldsymbol{\eta}_{d_\tau,k}] \boldsymbol{\Gamma}_{d_\tau}^T + \boldsymbol{\Gamma}_\tau \mathbf{E}[\boldsymbol{\eta}_{d_\tau,k} \boldsymbol{\eta}_{d_\phi,k}] \boldsymbol{\Gamma}_{d_\phi}^T \\ & + \boldsymbol{\Gamma}_{d_f} \mathbf{E}[\boldsymbol{\eta}_{d_f,k} \boldsymbol{\eta}_{d_\tau,k}] \boldsymbol{\Gamma}_{d_\tau}^T + \boldsymbol{\Gamma}_\tau \mathbf{E}[\boldsymbol{\eta}_{d_\tau,k} \boldsymbol{\eta}_{d_f,k}] \boldsymbol{\Gamma}_{d_f}^T \end{aligned} \quad (3.277)$$

The cross-correlations between the discriminator noises have already been derived and are given in (3.250), (3.253) and (3.255). Solely the correlation between the state and the frequency discriminator noise remains to be addressed. It is

$$\mathbf{E}[\mathbf{z}_k \boldsymbol{\eta}_{d_f,k}] = \mathbf{E}\left[\left(\Phi \mathbf{z}_{k-1} + \boldsymbol{\Gamma}_{d_\phi} \boldsymbol{\eta}_{d_\phi,k-1} + \boldsymbol{\Gamma}_{d_f} \boldsymbol{\eta}_{d_f,k-1} + \boldsymbol{\Gamma}_{d_\tau} \boldsymbol{\eta}_{d_\tau,k-1}\right) \boldsymbol{\eta}_{d_f,k}\right] \quad (3.278)$$

As has been seen in section 3.5.6.4.4, the discriminator noises of the previous time step and the frequency discriminator noise of the current time step are correlated

$$\mathbf{E}[\mathbf{z}_k \boldsymbol{\eta}_{d_f,k}] = \boldsymbol{\Gamma}_{d_\phi} \mathbf{E}[\boldsymbol{\eta}_{d_\phi,k-1} \boldsymbol{\eta}_{d_f,k}] + \boldsymbol{\Gamma}_{d_f} \mathbf{E}[\boldsymbol{\eta}_{d_f,k-1} \boldsymbol{\eta}_{d_f,k}] + \boldsymbol{\Gamma}_{d_\tau} \mathbf{E}[\boldsymbol{\eta}_{d_\tau,k-1} \boldsymbol{\eta}_{d_f,k}] \quad (3.279)$$

With (3.256), (3.257) and (3.258), the cross-correlation is approximated by

$$\mathbf{E}[\mathbf{z}_k \boldsymbol{\eta}_{d_f,k}] \approx -\left(\boldsymbol{\Gamma}_{d_\phi} + \frac{1}{T} \boldsymbol{\Gamma}_{d_f}\right) \frac{1}{2(C/N_0)_{k-1} T^2} + \frac{2 \tau_k}{TN} \quad (3.280)$$

Comprehensive simulations, however, have revealed that the cross-correlations are quite small and can normally be neglected. The process noise covariance matrix \mathbf{Q}_k can be well approximated by

$$\mathbf{Q}_k \approx \boldsymbol{\Gamma}_{d_\phi} \mathbf{Q}_{d_\phi,k} \boldsymbol{\Gamma}_{d_\phi}^T + \boldsymbol{\Gamma}_{d_f} \mathbf{Q}_{d_f,k} \boldsymbol{\Gamma}_{d_f}^T + \boldsymbol{\Gamma}_{d_\tau} \mathbf{Q}_{d_\tau,k} \boldsymbol{\Gamma}_{d_\tau}^T \quad (3.281)$$

The propagation has to be done with the integration time T of the receiver. It would be best to run the propagation within the navigation processing task onboard of the receiver. However, this is a very high computational burden and often not feasible since the user does not have access to the navigation algorithm on the receiver processor. The sample rate of the pseudorange and range rate measurements is usually between 1 and 10 Hertz. If the carrier-to-noise ratio does not change too fast with time, it is sufficient to assume a quasi-stationary state. Then, stationary solutions can be computed at the instances of pseudorange and range rate measurements. In order to obtain the stationary covariance

$\Phi_s = \text{const}$, which is established after some settling time under stationary conditions, the Sylvester equation

$$\Phi^{-1} \mathbf{P}_s - \mathbf{P}_s \Phi^T = \mathbf{Q} \quad (3.282)$$

has to be solved. It is mathematically rewritten by using the Kronecker product and vec-operator

$$(\mathbf{I}_5 \otimes \Phi^{-1} + \Phi \otimes \mathbf{I}_5) \text{vec}(\mathbf{P}_s) = \text{vec}(\mathbf{Q}) \quad (3.283)$$

and solved

$$\text{vec}(\mathbf{P}_s) = (\mathbf{I}_5 \otimes \Phi^{-1} + \Phi \otimes \mathbf{I}_5)^{-1} \text{vec}(\mathbf{Q}) \quad (3.284)$$

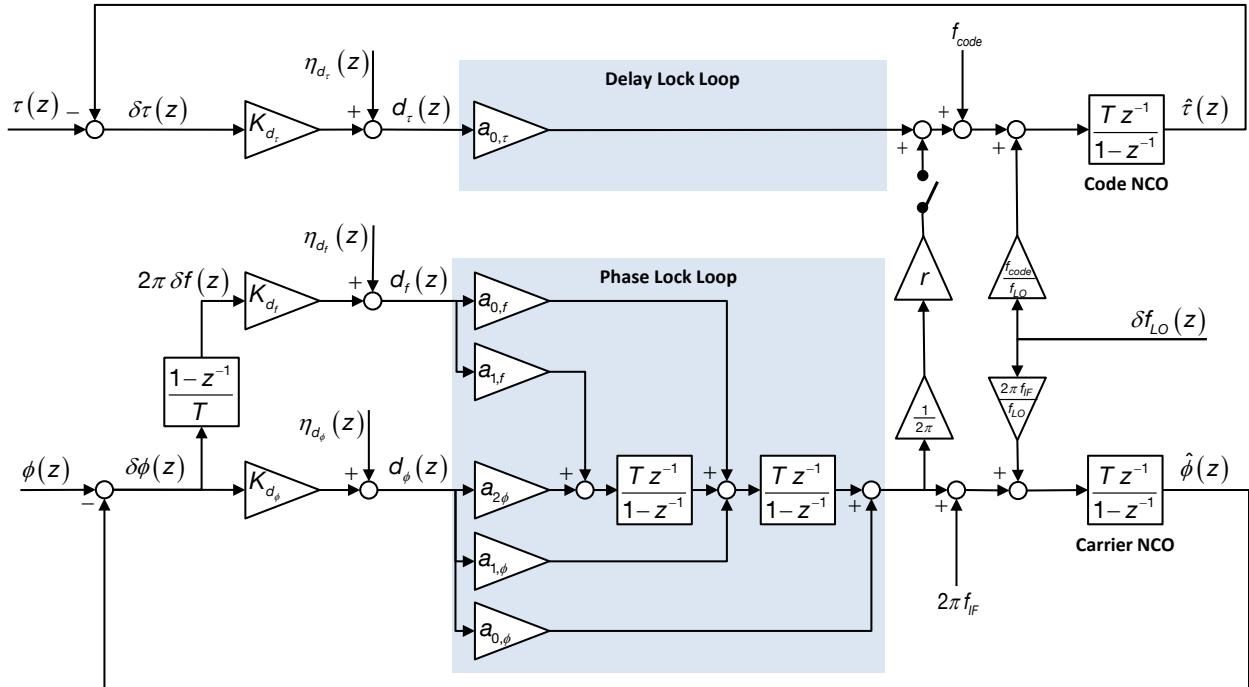


Figure 3-115: Discrete-time frequency-assisted PLL and carrier-aided DLL

3.5.6.5.5 Pseudorange and Range Rate Measurement Variances

The pseudorange measurement variance, which is required for the pseudorange innovation in the integrated navigation filter, is given with

$$R_{\rho,k} = \frac{c^2}{f_{code}^2} (\mathbf{0}_{1 \times 4} \quad 1) \mathbf{P}_k (\mathbf{0}_{1 \times 4} \quad 1)^T \quad (3.285)$$

The range rate measurement variance is

$$R_{\dot{\rho},k} = \frac{c^2}{f_{IF}^2} \left((\mathbf{H}_f \quad 0) \mathbf{P}_k (\mathbf{H}_f \quad 0)^T + D_f^2 Q_{\eta_{d,f},k} \right) \quad (3.286)$$

The noise of the pseudorange and range rate measurements to the different satellites in view is assumed to be independent.

The receiver noise model will be integrated into the navigation error filter and further analyzed in section 4.6.8.

4 Efficiently Increasing Navigation Filter Consistency

4.1 Inertial Navigation Algorithms

4.1.1 Motivation

As has been described in section 3.2, integrating IMU measure time integrals of accelerations $\mathbf{f}_b(t)$ and angular rates $\boldsymbol{\omega}_{ib}(t)$ between two subsequent sample times. Since the conventional strapdown algorithms like (2.1) or (2.3) require the accelerations \mathbf{f}_b and angular rates $\boldsymbol{\omega}_{ib}$ as input instead of the increments $\Delta\mathbf{v}_b$ and $\Delta\boldsymbol{\theta}_{ib}$, special algorithms for integrating IMU are necessary.

In this section, different strapdown navigation algorithms for integrating IMU are presented and compared with respect to numerical accuracy, robustness against noise and vibrations. These are the well-established dual frequency algorithms for orientation, the conversion-integration-extrapolation (CIE) algorithms and a novel approach, which – due to its inherent ability to propagate navigation states at the same frequency as the integrating IMU is providing measurements – will be called single frequency approach. In line with the accuracy vs. efficiency trade-off it is discussed in which cases higher order integration algorithms, coming along with higher computational expense, are justified and when it is sufficient to revert to a simpler, less demanding algorithm.

4.1.2 Simple Averaging Algorithm

The inertial navigation differential equations (2.1) or (2.3) can be applied with integrating IMU measurements if the required non-integrating IMU inputs are obtained by forming average accelerations and angular rates from the increments that are output by the integrating IMU

$$\mathbf{f}_b(t) = \frac{\Delta\mathbf{v}_b(t - \Delta t, t)}{\Delta t} + O(\Delta t), \quad \boldsymbol{\omega}_{ib}(t) = \frac{\Delta\boldsymbol{\theta}_{ib}(t - \Delta t, t)}{\Delta t} + O(\Delta t) \quad (4.1)$$

The approach is illustrated in Figure 4-1. The calculated accelerations and angular rates are however only of 0th order accuracy. High order integration schemes (like 4th order Runge-Kutta) are useless if inputs are of low accuracy.

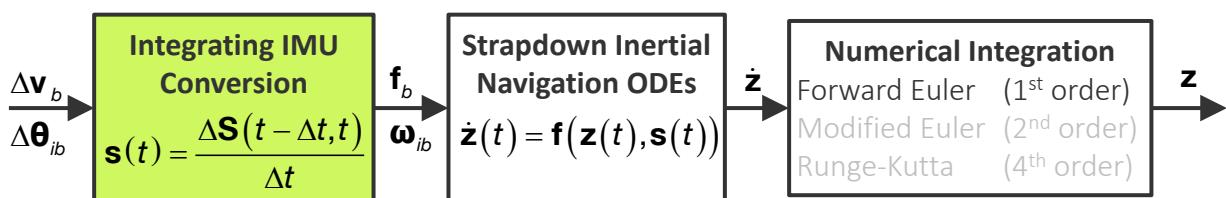


Figure 4-1: Averaging approach

4.1.3 Dual Frequency Algorithms for Orientation

Besides this simple averaging algorithm inertial navigation experts have derived dual frequency strapdown algorithms particularly for the orientation, [94], [95], [96], [97], [98] and [99]. These algorithms are “dual frequency” because the output rate of the propagated orientation state is lower than the IMU measurement input rate by factor three or four. The orientation is represented by Laning-Bortz parameters [100], which are well suited for handling angle increments as provided by an integrating gyroscope. Background information can be found in section C.1.1 in the appendix. Template 4-1 lists Miller’s, Lee’s and Gusinsky’s dual frequency algorithms. Miller’s algorithm needs three measurement inputs for one orientation output and Lee’s and Gusinsky’s algorithms require four inputs for one output value. The 3:1 and 4:1 integrating gyroscope measurement sampling is illustrated in Figure 4-2. The Laning-Bortz orientation vector σ_{ib} can be converted into an orientation quaternion \bar{q}_{ib} by

$$\bar{q}_{ib} = \left(\cos \frac{\|\sigma_{ib}\|}{2}, \sin \frac{\|\sigma_{ib}\|}{2} \frac{\sigma_{ib}^T}{\|\sigma_{ib}\|} \right)^T \quad (4.2)$$

Template 4-1: Dual frequency algorithms

Miller (3 : 1, optimized coefficients)

$$\sigma_{ib}(t, t + \Delta t) = \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + 0.45 \Delta\theta_1 \times \Delta\theta_3 + 0.675 \Delta\theta_2 \times (\Delta\theta_3 - \Delta\theta_1)$$

$$\text{where } \Delta\theta_1 = \Delta\theta_{ib}(t, t + \frac{\Delta t}{3}), \Delta\theta_2 = \Delta\theta_{ib}(t + \frac{\Delta t}{3}, t + \frac{2\Delta t}{3}), \Delta\theta_3 = \Delta\theta_{ib}(t + \frac{2\Delta t}{3}, t + \Delta t)$$

Lee et al. (4 : 1, optimized coefficients)

$$\begin{aligned} \sigma_{ib}(t, t + \Delta t) = & \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + \Delta\theta_4 + \frac{214}{105} (a_1 \Delta\theta_1 \times \Delta\theta_2 + a_2 \Delta\theta_2 \times \Delta\theta_3 + a_3 \Delta\theta_3 \times \Delta\theta_4) \\ & + \frac{92}{105} (b_1 \Delta\theta_1 \times \Delta\theta_3 + b_2 \Delta\theta_2 \times \Delta\theta_4) + \frac{54}{105} \Delta\theta_1 \times \Delta\theta_4 \end{aligned}$$

$$\text{where } a_1 + a_2 + a_3 = 1, \quad b_1 + b_2 = 1$$

$$\Delta\theta_1 = \Delta\theta_{ib}(t, t + \frac{\Delta t}{4}), \Delta\theta_2 = \Delta\theta_{ib}(t + \frac{\Delta t}{4}, t + \frac{\Delta t}{2}), \Delta\theta_3 = \Delta\theta_{ib}(t + \frac{\Delta t}{2}, t + \frac{3\Delta t}{4}), \Delta\theta_4 = \Delta\theta_{ib}(t + \frac{3\Delta t}{4}, t + \Delta t)$$

Gusinsky et al. (4 : 1, noise smoothing)

$$\begin{aligned} \sigma_{ib}(t, t + \Delta t) = & \Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + \Delta\theta_4 + \frac{52}{105} \Delta\theta_1 \times \Delta\theta_2 + \frac{212}{315} \Delta\theta_1 \times \Delta\theta_3 + \frac{112}{315} \Delta\theta_1 \times \Delta\theta_4 \\ & + \frac{184}{315} \Delta\theta_2 \times \Delta\theta_3 + \frac{212}{315} \Delta\theta_2 \times \Delta\theta_4 + \frac{52}{105} \Delta\theta_3 \times \Delta\theta_4 \end{aligned}$$

$$\text{where } \Delta\theta_1 = \Delta\theta_{ib}(t, t + \frac{\Delta t}{4}), \Delta\theta_2 = \Delta\theta_{ib}(t + \frac{\Delta t}{4}, t + \frac{\Delta t}{2}), \Delta\theta_3 = \Delta\theta_{ib}(t + \frac{\Delta t}{2}, t + \frac{3\Delta t}{4}), \Delta\theta_4 = \Delta\theta_{ib}(t + \frac{3\Delta t}{4}, t + \Delta t)$$

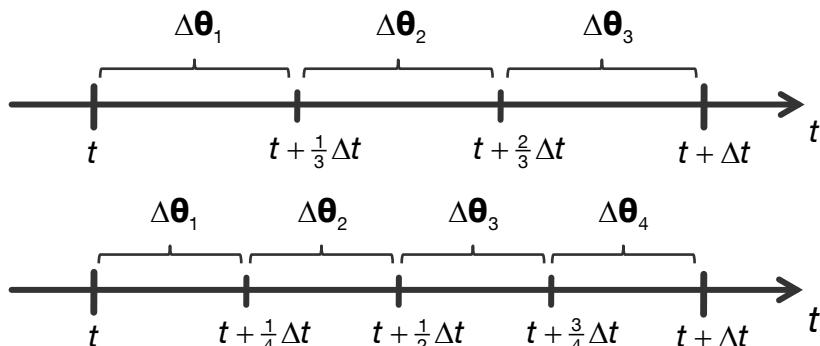


Figure 4-2: 3:1 and 4:1 sampling of the integrating gyroscope. Δt is the output sample time

4.1.4 Conversion-Integration-Extrapolation Algorithms

4.1.4.1 Conversion from Integrating to Non-Integrating IMU Measurements

The error order of the averaging algorithm can be increased by using higher order conversion schemes instead of the 0th order averaging (4.1) to convert the velocity increments $\Delta\mathbf{v}_b$ and angle increments $\Delta\boldsymbol{\theta}_{ib}$ of integrating IMU to accelerations \mathbf{f}_b and angular rates $\boldsymbol{\omega}_{ib}$ of non-integrating IMU. In the following, higher order conversion formulae will be derived by means of Taylor series expansion.

Measured angle increments over different epochs in the past or in the future are defined as follows

$$\Delta\boldsymbol{\theta}_{ib}^{--} = \int_{t-2\Delta t}^{t-\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta\boldsymbol{\theta}_{ib}^- = \int_{t-\Delta t}^t \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta\boldsymbol{\theta}_{ib}^+ = \int_t^{t+\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta\boldsymbol{\theta}_{ib}^{++} = \int_{t+\Delta t}^{t+2\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau \quad (4.3)$$

The sum of two subsequent increments $\Delta\boldsymbol{\theta}_{ib}^{--}$ and $\Delta\boldsymbol{\theta}_{ib}^-$ is defined as

$$\Delta\boldsymbol{\theta}_{ib}^= = \int_{t-2\Delta t}^t \boldsymbol{\omega}_{ib}(\tau) d\tau = \Delta\boldsymbol{\theta}_{ib}^{--} + \Delta\boldsymbol{\theta}_{ib}^- \quad (4.4)$$

The sequence of the single measurements is illustrated in Figure 4-3. The same definitions are valid for the integrating accelerometer measurements $\Delta\mathbf{v}_b^{--}$, $\Delta\mathbf{v}_b^-$, $\Delta\mathbf{v}_b^+$, $\Delta\mathbf{v}_b^{++}$ and $\Delta\mathbf{v}_b^=$. The integrating measurements (4.3) and (4.4) are expanded in Taylor series considering terms up to 4th order

$$\begin{aligned} \Delta\boldsymbol{\theta}_{ib}^+ &= \boldsymbol{\omega}_{ib}(t) \cdot \Delta t + \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^2}{2!} + \ddot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^3}{3!} + \dddot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^4}{4!} + O(\Delta t^5) \\ \Delta\boldsymbol{\theta}_{ib}^- &= \boldsymbol{\omega}_{ib}(t) \cdot \Delta t - \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^2}{2!} + \ddot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^3}{3!} - \dddot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^4}{4!} + O(\Delta t^5) \\ \Delta\boldsymbol{\theta}_{ib}^= &= 2 \boldsymbol{\omega}_{ib}(t) \cdot \Delta t - 4 \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^2}{2!} + 8 \ddot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^3}{3!} - 16 \dddot{\boldsymbol{\omega}}_{ib}(t) \cdot \frac{\Delta t^4}{4!} + O(\Delta t^5) \end{aligned} \quad (4.5)$$

Now, different linear combinations of these angle increments can be formed and solved for the angular rate $\boldsymbol{\omega}_{ib}$. Depending on the number of involved measurements, different orders of the accuracy can be achieved.

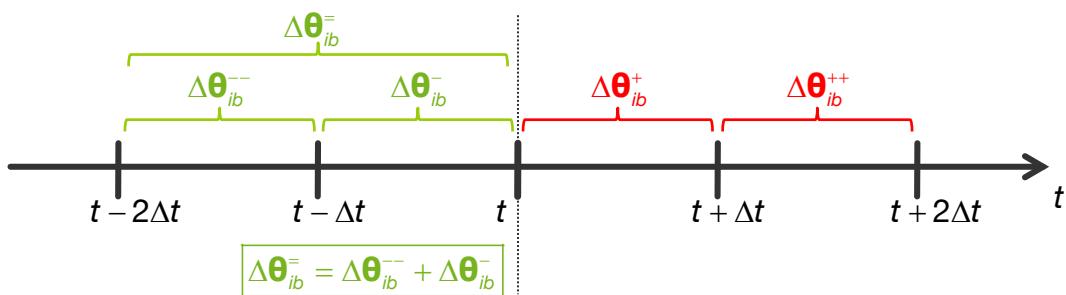


Figure 4-3: Integrating IMU measurement sequence

1 left / 1 right

1st order accuracy can be attained if two central measurements are used for the conversion. The Taylor series expansions of $\Delta\boldsymbol{\theta}_{ib}^+$ and $\Delta\boldsymbol{\theta}_{ib}^-$ written in matrix vector form

$$\begin{pmatrix} \Delta\boldsymbol{\theta}_{ib}^+ \\ \Delta\boldsymbol{\theta}_{ib}^- \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{2} \\ 1 & -\frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\omega}_{ib}(t) \cdot \Delta t \\ \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \Delta t^2 \end{pmatrix} + O(\Delta t^3) \quad (4.6)$$

are solved for the angular rate and its derivative

$$\begin{pmatrix} \boldsymbol{\omega}_{ib}(t) \cdot \Delta t \\ \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \Delta t^2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \Delta\theta_{ib}^+ \\ \Delta\theta_{ib}^- \end{pmatrix} + O(\Delta t^3) \quad (4.7)$$

The 1st order conversion formulae for the acceleration and angular rate are hence given by

$$\begin{aligned} \mathbf{f}_b(t) &= \frac{1}{2\Delta t} (\Delta\mathbf{v}_b^- + \Delta\mathbf{v}_b^+) + O(\Delta t^2) \\ \boldsymbol{\omega}_{ib}(t) &= \frac{1}{2\Delta t} (\Delta\theta_{ib}^- + \Delta\theta_{ib}^+) + O(\Delta t^2) \end{aligned} \quad (4.8)$$

The conversion is termed “1 left / 1 right” and is illustrated in Figure 4-4.

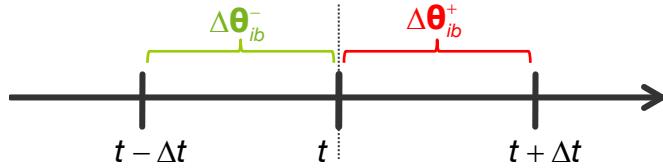


Figure 4-4: 1st order conversion: “1 left / 1 right”

2 left / 1 right

2nd order accuracy can be attained if the two central measurements $\Delta\theta_{ib}^+$ and $\Delta\theta_{ib}^-$ and one additional measurement of the past, $\Delta\theta_{ib}^-$, are used for the conversion. The corresponding Taylor series expansions are written in matrix vector form

$$\begin{pmatrix} \Delta\theta_{ib}^+ \\ \Delta\theta_{ib}^- \\ \Delta\theta_{ib}^- \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{6} \\ 1 & -\frac{1}{2} & \frac{1}{6} \\ 2 & -2 & \frac{4}{3} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\omega}_{ib}(t) \cdot \Delta t \\ \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \Delta t^2 \\ \ddot{\boldsymbol{\omega}}_{ib}(t) \cdot \Delta t^3 \end{pmatrix} + O(\Delta t^4) \quad (4.9)$$

and solved for the angular rate and its derivatives

$$\begin{pmatrix} \boldsymbol{\omega}_{ib}(t) \cdot \Delta t \\ \dot{\boldsymbol{\omega}}_{ib}(t) \cdot \Delta t^2 \\ \ddot{\boldsymbol{\omega}}_{ib}(t) \cdot \Delta t^3 \end{pmatrix} = \begin{pmatrix} \frac{1}{3} & 1 & -\frac{1}{6} \\ 1 & -1 & 0 \\ 1 & -3 & 1 \end{pmatrix} \cdot \begin{pmatrix} \Delta\theta_{ib}^+ \\ \Delta\theta_{ib}^- \\ \Delta\theta_{ib}^- \end{pmatrix} + O(\Delta t^4) \quad (4.10)$$

The 2nd order conversion formulae for the acceleration and the angular rate are finally

$$\begin{aligned} \mathbf{f}_b(t) &= \frac{1}{6\Delta t} (2\Delta\mathbf{v}_b^+ + 5\Delta\mathbf{v}_b^- - \Delta\mathbf{v}_b^{--}) + O(\Delta t^3) \\ \boldsymbol{\omega}_{ib}(t) &= \frac{1}{6\Delta t} (2\Delta\theta_{ib}^+ + 5\Delta\theta_{ib}^- - \Delta\theta_{ib}^{--}) + O(\Delta t^3) \end{aligned} \quad (4.11)$$

The conversion is termed “2 left / 1 right” and is illustrated in Figure 4-5. Other linear combinations involving further measurements can be derived and are summarized in Template 4-2. Therein, $\Delta\theta_{ib}^{---} = \int_{t-3\Delta t}^{t-2\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau$ and $\Delta\theta_{ib}^{----} = \int_{t-4\Delta t}^{t-3\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau$.

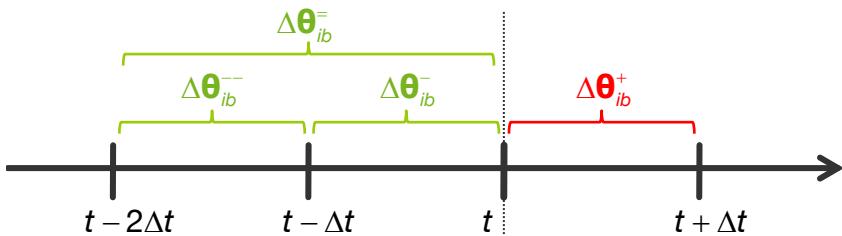
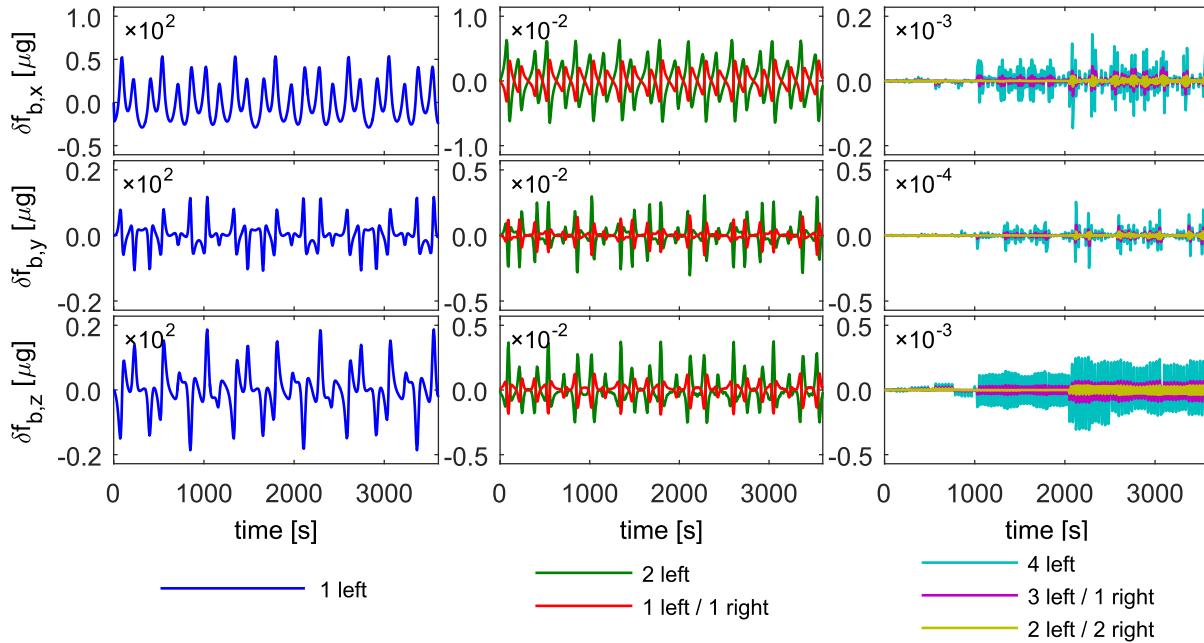
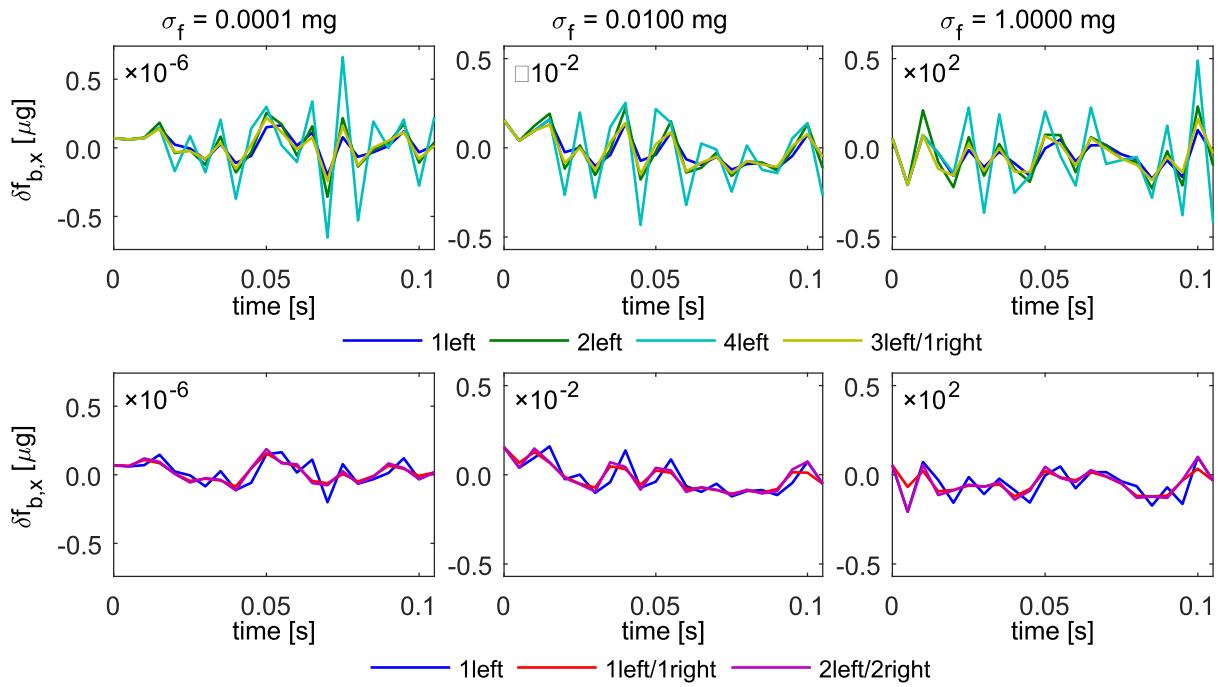


Figure 4-5: 2nd order conversion: “2 left / 1 right”

Template 4-2: Integrating IMU conversion formulae

Method	Formula	Order
2 left / 2 right	$\mathbf{f}_b(t) = \frac{1}{12\Delta t}(-\Delta \mathbf{v}_b^{++} + 7\Delta \mathbf{v}_b^+ + 7\Delta \mathbf{v}_b^- - \Delta \mathbf{v}_b^{--})$ $\boldsymbol{\omega}_{ib}(t) = \frac{1}{12\Delta t}(-\Delta \boldsymbol{\theta}_{ib}^{++} + 7\Delta \boldsymbol{\theta}_{ib}^+ + 7\Delta \boldsymbol{\theta}_{ib}^- - \Delta \boldsymbol{\theta}_{ib}^{--})$	$O(\Delta t^4)$
1 left / 1 right	$\mathbf{f}_b(t) = \frac{1}{2\Delta t}(\Delta \mathbf{v}_b^- + \Delta \mathbf{v}_b^+)$ $\boldsymbol{\omega}_{ib}(t) = \frac{1}{2\Delta t}(\Delta \boldsymbol{\theta}_{ib}^- + \Delta \boldsymbol{\theta}_{ib}^+)$	$O(\Delta t^2)$
3 left / 1 right	$\mathbf{f}_b(t) = \frac{1}{12\Delta t}(3\Delta \mathbf{v}_b^+ + 13\Delta \mathbf{v}_b^- - 5\Delta \mathbf{v}_b^{--} + \Delta \mathbf{v}_b^{---})$ $\boldsymbol{\omega}_{ib}(t) = \frac{1}{12\Delta t}(3\Delta \boldsymbol{\theta}_{ib}^+ + 13\Delta \boldsymbol{\theta}_{ib}^- - 5\Delta \boldsymbol{\theta}_{ib}^{--} + \Delta \boldsymbol{\theta}_{ib}^{---})$	$O(\Delta t^4)$
2 left	$\mathbf{f}_b(t) = \frac{1}{2\Delta t}(3\Delta \mathbf{v}_b^- - \Delta \mathbf{v}_b^{--})$ $\boldsymbol{\omega}_{ib}(t) = \frac{1}{2\Delta t}(3\Delta \boldsymbol{\theta}_{ib}^- - \Delta \boldsymbol{\theta}_{ib}^{--})$	$O(\Delta t^2)$
1 left	$\mathbf{f}_b(t) = \frac{1}{\Delta t}\Delta \mathbf{v}_b^-$ $\boldsymbol{\omega}_{ib}(t) = \frac{1}{\Delta t}\Delta \boldsymbol{\theta}_{ib}^-$	$O(\Delta t)$
4 left	$\mathbf{f}_b(t) = \frac{1}{12\Delta t}(25\Delta \mathbf{v}_b^- - 23\Delta \mathbf{v}_b^{--} + 13\Delta \mathbf{v}_b^{---} - 3\Delta \mathbf{v}_b^{---})$ $\boldsymbol{\omega}_{ib}(t) = \frac{1}{12\Delta t}(25\Delta \boldsymbol{\theta}_{ib}^- - 23\Delta \boldsymbol{\theta}_{ib}^{--} + 13\Delta \boldsymbol{\theta}_{ib}^{---} - 3\Delta \boldsymbol{\theta}_{ib}^{---})$	$O(\Delta t^4)$

The different conversion methods shall be analyzed by simulation. The converted integrating IMU values are compared to the non-integrating IMU values. The conversion method is the better, the smaller the error. First, the conversions are applied to error-free IMU measurements. This analysis should reveal that the accuracy of the conversion increases with the order of the conversion method. The Generic Lissajous trajectory (section E.1.1) is used to generate the integrating IMU measurements by numerical integration. In Figure 4-6, the conversion errors of the accelerometer measurements are plotted. It can be stated that the errors of the conversion methods are compliant with the corresponding truncation orders. The errors of the central conversion schemes are generally smaller than of the non-central ones of same order. Measurements, however, are always affected by noise of different colors. The conversions are hence repeated with noisy measurements. For that, pure white Gaussian distributed noise, that is velocity/angle white noise, is added to the error-free integrating IMU measurements. The results are illustrated in Figure 4-7 for accelerometer measurements with three different noise standard deviations σ_f . It can be observed that the central integrating IMU conversions provide better results than the non-central conversions and that the influence of the conversion scheme order is much smaller compared to the error-free case even with the very low noise. If non-central IMU conversions have to be used in order to keep up real-time capability, it is recommended to use “1 left”, “2 left” or “3 left / 1 right” conversions. The 3rd order conversion “4 left” provides worst results. This is an indication for lower robustness against noise.

**Figure 4-6: Conversion of error-free integrating accelerometer measurements****Figure 4-7: Conversion of noisy integrating accelerometer measurements**

4.1.4.2 Conversion-Integration-Extrapolation

After having converted the integrating IMU measurements to non-integrating values with an appropriate method from the previous section, the inertial navigation ODE (2.1) or (2.3) for non-integrating IMU can be used to calculate the navigation solution. It can be chosen from different numerical integration schemes with different accuracy orders (for example forward Euler, modified Euler, Runge-Kutta). If conversions are used that are not real-time capable the solution can only be computed with a delay of some sample times and has then to be extrapolated to the current time. The processing sequence is depicted in Figure 4-8.

The order of the overall error is primarily defined by the lowest order component. If, for example, a low order conversion is connected in series with a high order integrator, the order of the complete approach is still low and vice versa. It is recommended to choose the order of the integrating IMU conversion in correspondence to the truncation order of the applied integration scheme. If the navigation differential equations are integrated, for example, with forward Euler, a 0th order or 1st order conversion is appropriate. In the following, suitable combinations of conversion and integration methods are presented.

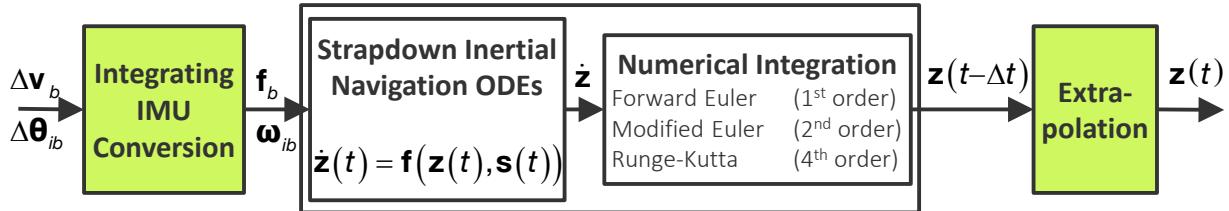


Figure 4-8: Processing sequence of a Conversion-Integration-Extrapolation (CIE) algorithm

Forward Euler CIE

This approach is the simplest CIE algorithm. It is of 0th order accuracy and was presented as averaging approach in 4.1.2. The method is real-time capable and extrapolation is not required. The method is illustrated in Figure 4-9. \mathbf{s} represents the non-integrating sensor input, $\mathbf{s} = (\mathbf{f}_b^\top, \boldsymbol{\omega}_{ib}^\top)^\top$, and $\Delta\mathbf{S}$ is the integrating sensor input, $\Delta\mathbf{S} = (\Delta\mathbf{v}_b^\top, \Delta\boldsymbol{\omega}_{ib}^\top)^\top$.

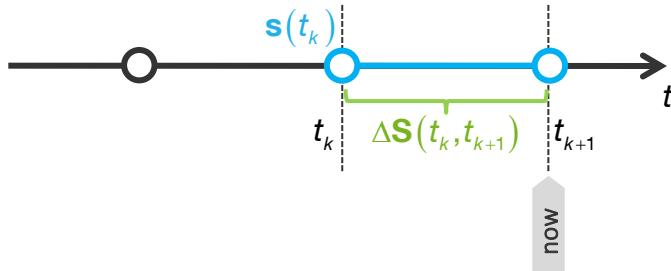


Figure 4-9: "1 left" conversion and forward Euler integration

Single Frequency 1st Order CIE

The accuracy can be increased if the forward Euler integrator is replaced by a modified Euler integrator that evaluates the derivatives at t_k and $t_k + \frac{\Delta t}{2}$ and if another measurement is added. The required state $\mathbf{z}(t_k + \frac{\Delta t}{2})$ has to be calculated by extrapolation of half a time step. The conversion uses (4.8) for the derivative at t_k and (4.1) for the derivative at $t_k + \frac{\Delta t}{2}$

$$\begin{aligned} \mathbf{s}(t_k) &= \frac{1}{2\Delta t} (\Delta\mathbf{S}(t_{k-1}, t_k) + \Delta\mathbf{S}(t_k, t_{k+1})) + O(\Delta t^2) && \text{"1left/1right"} \\ \mathbf{s}(t_k + \frac{\Delta t}{2}) &= \frac{\Delta\mathbf{S}(t_k, t_{k+1})}{\Delta t} + O(\Delta t^2) && \text{"1central"} \end{aligned} \quad (4.12)$$

It can be shown that the second conversion whose validity time is assumed in the middle of the sampling interval at $t_k + \frac{\Delta t}{2}$ is of 1st order accuracy instead of 0th order of the conversion (4.1) whose validity time is assumed to be at t_k . The modified Euler integrator is

$$\begin{aligned}
 \mathbf{k}_1 &= \mathbf{f}(\mathbf{z}(t_k), \mathbf{s}(t_k)) \\
 \mathbf{k}_2 &= \mathbf{f}\left(\mathbf{z}(t_k) + \frac{\Delta t}{2} \cdot \mathbf{k}_1, \mathbf{s}(t_k + \frac{\Delta t}{2})\right) \\
 \mathbf{z}(t_{k+1}) &= \mathbf{z}(t_k) + \Delta t \cdot \mathbf{k}_2 + O(\Delta t^3)
 \end{aligned} \tag{4.13}$$

The method is of 1st order accuracy, real-time capable and extrapolation is not required. The method is illustrated in Figure 4-10. It has to be noted that the method is a single frequency algorithm meaning that the input rate is the same as the output rate.

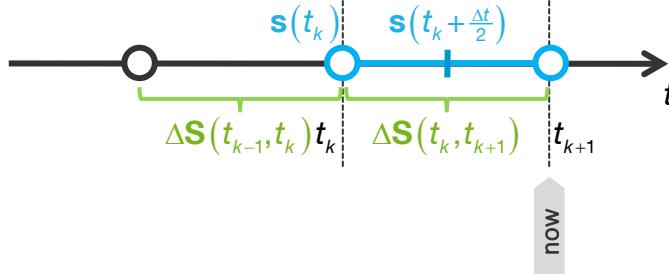


Figure 4-10: “1 left / 1 right” conversion and modified Euler integration

Modified Euler CIE

The modified Euler integrator is a dual frequency integrator. The input rate is twice the output rate. For both required derivatives at t_k and $t_k + \frac{\Delta t}{2}$ the central “1 left / 1 right” conversion schemes (4.8), which are particularly robust against noise, are selected

$$\begin{aligned}
 \mathbf{s}(t_k) &= \frac{1}{2\Delta t} \left(\Delta \mathbf{S}(t_k - \frac{\Delta t}{2}, t_k) + \Delta \mathbf{S}(t_k, t_k + \frac{\Delta t}{2}) \right) + O(\Delta t^2) && \text{"1left/1right"} \\
 \mathbf{s}(t_k + \frac{\Delta t}{2}) &= \frac{1}{2\Delta t} \left(\Delta \mathbf{S}(t_k, t_k + \frac{\Delta t}{2}) + \Delta \mathbf{S}(t_k + \frac{\Delta t}{2}, t_{k+1}) \right) + O(\Delta t^2) && \text{"1left/1right"}
 \end{aligned} \tag{4.14}$$

The integration is accomplished with the modified Euler integrator scheme (4.13). The algorithm is of 1st order accuracy, real-time capable and extrapolation is not required. The method is illustrated in Figure 4-11.

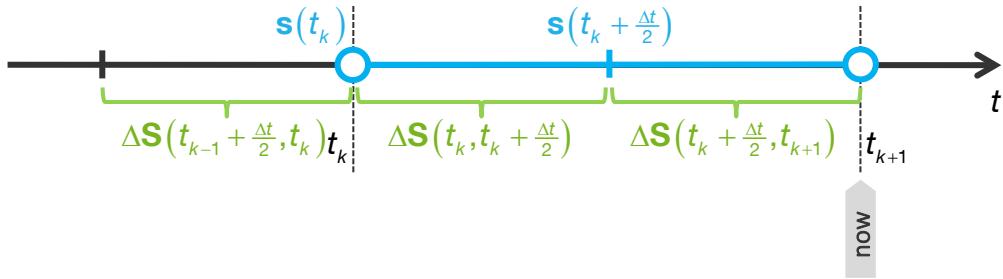


Figure 4-11: “1 left / 1 right” conversion and modified Euler integration

Runge-Kutta CIE

The Runge-Kutta integrator is of 4th order accuracy and is like the modified Euler integrator a dual frequency integrator. 3rd order “2 left / 2 right” conversions are applied for the measurements at the three subsequent time steps

$$\begin{aligned}
 \mathbf{s}(t_{k-1}) &= \frac{1}{12\Delta t} \left(-\Delta \mathbf{S}(t_{k-2} + \frac{\Delta t}{2}) + 7\Delta \mathbf{S}(t_{k-1}) + 7\Delta \mathbf{S}(t_{k-1} + \frac{\Delta t}{2}) - \Delta \mathbf{S}(t_k) \right) + O(\Delta t^4) && \text{"2left/2right"} \\
 \mathbf{s}(t_{k-1} + \frac{\Delta t}{2}) &= \frac{1}{12\Delta t} \left(-\Delta \mathbf{S}(t_{k-1}) + 7\Delta \mathbf{S}(t_{k-1} + \frac{\Delta t}{2}) + 7\Delta \mathbf{S}(t_k) - \Delta \mathbf{S}(t_k + \frac{\Delta t}{2}) \right) + O(\Delta t^4) && \text{"2left/2right"} \\
 \mathbf{s}(t_k) &= \frac{1}{12\Delta t} \left(-\Delta \mathbf{S}(t_{k-1} + \frac{\Delta t}{2}) + 7\Delta \mathbf{S}(t_k) + 7\Delta \mathbf{S}(t_k + \frac{\Delta t}{2}) - \Delta \mathbf{S}(t_{k+1}) \right) + O(\Delta t^4) && \text{"2left/2right"}
 \end{aligned} \tag{4.15}$$

The Runge-Kutta integration scheme is

$$\begin{aligned}\mathbf{k}_1 &= \mathbf{f}(\mathbf{z}(t_{k-1}), \mathbf{s}(t_{k-1})), & \mathbf{k}_3 &= \mathbf{f}\left(\mathbf{z}(t_{k-1}) + \frac{\Delta t}{2} \cdot \mathbf{k}_2, \mathbf{s}(t_{k-1} + \frac{\Delta t}{2})\right) \\ \mathbf{k}_2 &= \mathbf{f}\left(\mathbf{z}(t_{k-1}) + \frac{\Delta t}{2} \cdot \mathbf{k}_1, \mathbf{s}(t_{k-1} + \frac{\Delta t}{2})\right), & \mathbf{k}_4 &= \mathbf{f}\left(\mathbf{z}(t_{k-1}) + \Delta t \cdot \mathbf{k}_3, \mathbf{s}(t_k)\right) \\ \mathbf{z}(t_k) &= \mathbf{z}(t_{k-1}) + \frac{\Delta t}{6} \cdot (\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) + O(\Delta t^5)\end{aligned}\quad (4.16)$$

The method is not real-time capable. The solution is calculated with a delay of one sample time step (two IMU cycles) and afterwards extrapolated to the current time

$$\mathbf{z}(t_{k+1}) = \mathbf{z}(t_k) + \dot{\mathbf{z}}(t_k) \cdot \Delta t \quad (4.17)$$

The method is illustrated in Figure 4-12.

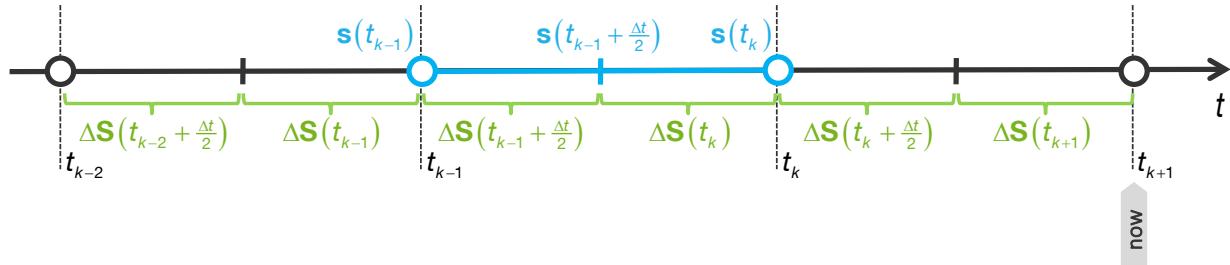


Figure 4-12: “2 left / 2 right” conversion with Runge-Kutta integration

4.1.5 Single Frequency Approach

Instead of converting integrating IMU measurements and solving the strapdown inertial navigation ODE (2.1) or (2.3) with an appropriate numerical integration scheme, an algorithm is sought that can directly process integrating IMU measurements. Unlike the dual frequency approach the algorithm shall work with one single frequency, that means output a solution for each input, and provide position, velocity and orientation. This simplifies the implementation of the algorithm. In the navigation difference equations to be derived, integrating IMU measurements of the epochs $[t - \Delta t, t]$ and $[t, t + \Delta t]$ shall be used to compute the solution at the current time $t + \Delta t$, as illustrated in Figure 4-13.

The derivation of the single frequency approach has been presented in [101] and is repeated in the appendix in section C.1.2. In [101], the direction cosine matrix is used to represent the orientation. Here, the orientation difference equation is alternatively derived for quaternions. The equations of the single frequency approach are summarized in Template 4-3. The orientation is of 3rd order accuracy. The equations can be alternatively derived for the n -frame with the latitude ϕ , longitude λ , height h , velocity \mathbf{v}_n and orientation quaternion \tilde{q}_{nb} between n - and b -frame as states. They are given in Template C-1 in section C.1.2 in the appendix.

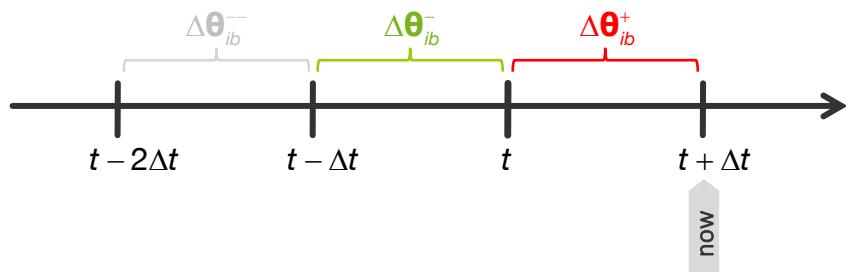


Figure 4-13: Involved measurements of the single frequency approach

Template 4-3: Single frequency approach (e-frame, 3rd order orientation)

$$\begin{aligned}\mathbf{x}_e(t + \Delta t) &= \mathbf{x}_e(t) + \mathbf{v}_e(t) \Delta t + \dot{\mathbf{v}}_e(t) \frac{\Delta t^2}{2!} + \ddot{\mathbf{v}}_e(t) \frac{\Delta t^3}{3!} + O(\Delta t^4) \\ \mathbf{v}_e(t + \Delta t) &= \mathbf{v}_e(t) + \dot{\mathbf{v}}_e(t) \Delta t + \ddot{\mathbf{v}}_e(t) \frac{\Delta t^2}{2!} + O(\Delta t^3) \\ \check{q}_{eb}(t + \Delta t) &= \check{p}_e^{-1} \check{q}_{eb}(t) \check{p}_b(t) + O(\Delta t^4)\end{aligned}$$

where

$$\begin{aligned}\dot{\mathbf{v}}_e(t) &= \mathbf{R}(\check{q}_{eb}(t)) \frac{\Delta \mathbf{v}_b^+ + \Delta \mathbf{v}_b^-}{2\Delta t} + \mathbf{Y}_e(\mathbf{x}_e(t)) - 2\boldsymbol{\omega}_{ie} \times \mathbf{v}_e(t) \\ \ddot{\mathbf{v}}_e(t) &= \mathbf{R}(\check{q}_{eb}(t)) \left(\boldsymbol{\omega}_{eb}(t) \times \frac{\Delta \mathbf{v}_b^+ + \Delta \mathbf{v}_b^-}{2\Delta t} + \frac{\Delta \mathbf{v}_b^+ - \Delta \mathbf{v}_b^-}{\Delta t^2} \right) + \frac{\partial \mathbf{Y}_e(\mathbf{x}_e(t))}{\partial \mathbf{x}_e^\top} \mathbf{v}_e(t) - 2\boldsymbol{\omega}_{ie} \times \dot{\mathbf{v}}_e(t) \\ \boldsymbol{\omega}_{eb}(t) &= \frac{1}{2\Delta t} (\Delta \boldsymbol{\Theta}_{ib}^+ + \Delta \boldsymbol{\Theta}_{ib}^-) - \mathbf{R}^\top(\check{q}_{eb}(t)) \boldsymbol{\omega}_{ie} \\ \check{p}_e &= \left(\cos\left(\omega_{ie} \frac{\Delta t}{2}\right), \sin\left(\omega_{ie} \frac{\Delta t}{2}\right) \frac{\boldsymbol{\omega}_{ie}}{\omega_{ie}} \right) \\ \check{p}_b(t) &= \left(1 - \frac{1}{8} \|\Delta \boldsymbol{\Theta}_{ib}^+\|^2, \frac{1}{2} \left(1 - \frac{1}{24} \|\Delta \boldsymbol{\Theta}_{ib}^+\|^2 \right) \Delta \boldsymbol{\Theta}_{ib}^+ - \frac{1}{24} \Delta \boldsymbol{\Theta}_{ib}^+ \times \Delta \boldsymbol{\Theta}_{ib}^- \right)\end{aligned}$$

IMU Measurements :

$$\Delta \boldsymbol{\Theta}_{ib}^+ = \int_t^{t+\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta \boldsymbol{\Theta}_{ib}^- = \int_{t-\Delta t}^t \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta \mathbf{v}_b^+ = \int_t^{t+\Delta t} \mathbf{f}_b(\tau) d\tau, \quad \Delta \mathbf{v}_b^- = \int_{t-\Delta t}^t \mathbf{f}_b(\tau) d\tau$$

4.1.6 Numerical Analysis

In this section the performance of the CIE algorithms, the single frequency approach and the dual frequency algorithms is analyzed by means of numerical simulation. Especially accuracy, robustness against vibrations and robustness against noise are considered. At the end recommendations for different environments are given. Table 4-1 lists the inertial navigation algorithms to be compared. The Generic Lissajous scenario (section E.1.1) is used.

Table 4-1: Algorithms to be compared

Name	Conversion	Integration	Extrapolation
Forward Euler CIE (Forw. Euler)	1 left (0 th order)	Forward Euler (1 st order)	no
Single Frequency CIE (Single Freq. CIE)	1 left / 1 right (1 st order) 1 central (1 st order)	Modified Euler (2 nd order)	no
Modified Euler CIE (Modified Euler)	1 left / 1 right (1 st order)	Modified Euler (2 nd order)	no
Runge-Kutta CIE (Runge-K.)	2 left / 2 right (3 rd order)	Runge-Kutta (4 th order)	2 IMU time steps
Single Frequency approach, e-frame, 3 rd order (Single Freq.)			
Lee (orientation only)			
Gusinsky (orientation only)			

Accuracy

First, the accuracy of the algorithms is analyzed by processing error-free IMU measurements. Figure 4-14 shows the position error and Figure 4-15 the orientation error for the different algorithms to be compared.

It can be seen that the accuracy of the methods corresponds to their theoretical truncation order. The height error at the end of the trajectory is in the range of 10^6 m with the Forward Euler CIE method, 10^1 m with the Single Frequency CIE and Modified Euler CIE methods, 10^{-1} m with the Single Frequency approach and 10^{-5} m with the Runge-Kutta CIE method. The orientation error is in the range of 10^{-1} deg with the Forward Euler CIE method, 10^{-5} deg with the Single Frequency CIE and Modified Euler CIE methods, 10^{-7} deg with the Single Frequency approach and 10^{-10} deg with the Runge-Kutta CIE, Lee and Gusinsky methods.

Vibrations

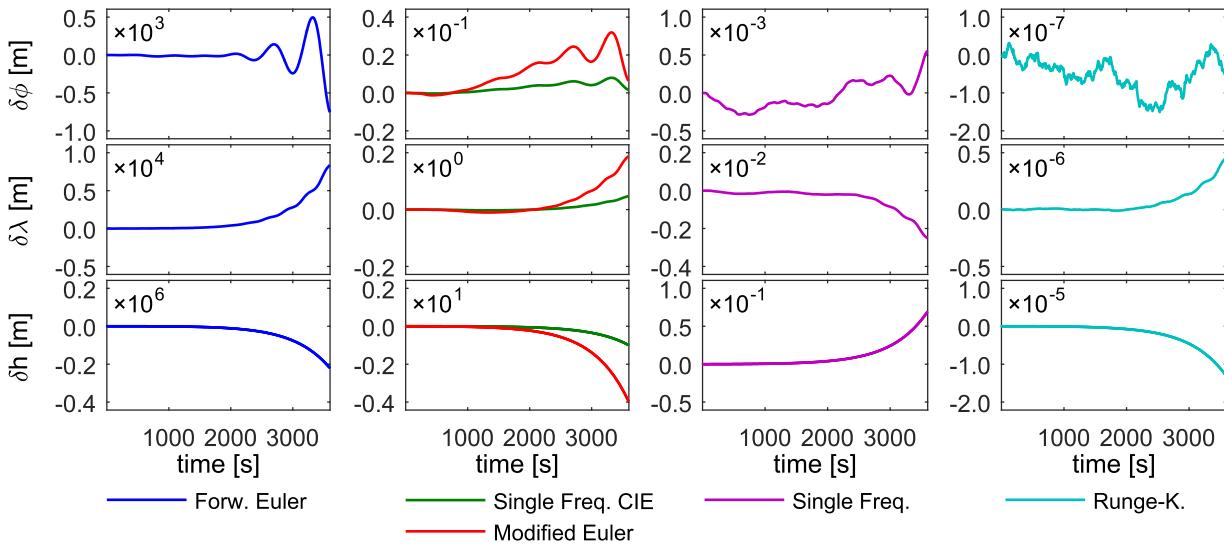
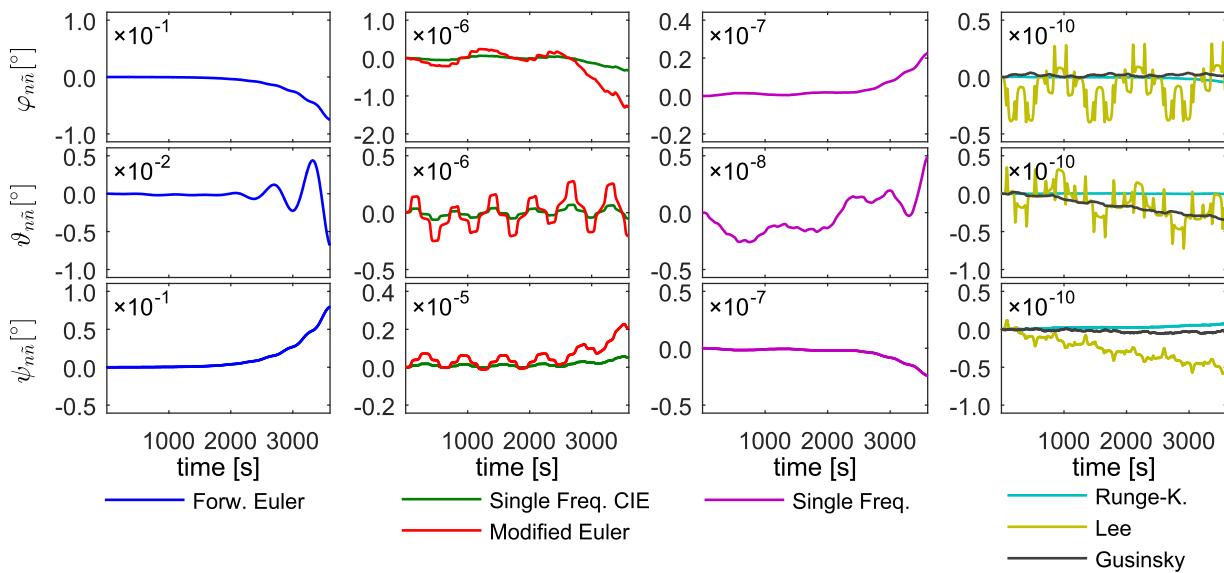
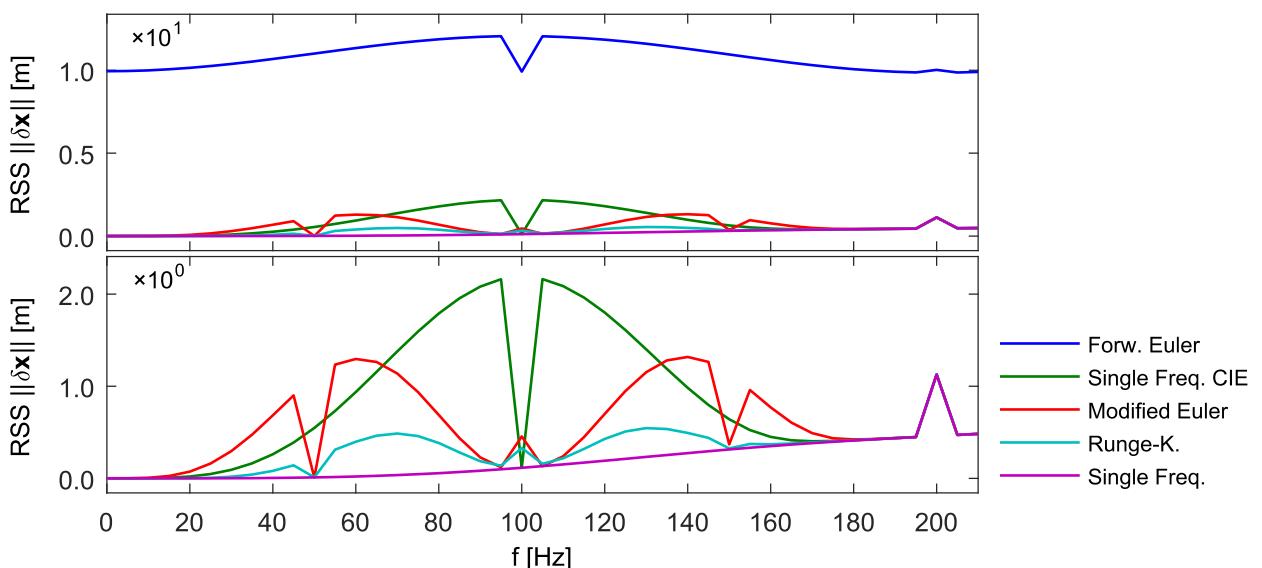
The IMU may be exposed to vibrations. To analyze the robustness of the algorithms against vibrations, the error-free kinematic trajectory is superimposed by vibrations represented by sine functions with amplitudes of $1 \mu\text{m}$ on the position and 1 mdeg on the orientation, respectively, and distinct frequencies between $10 - 215 \text{ Hz}$. In Figure 4-16, the root mean squares of the position errors and in Figure 4-17 the root mean squares of the orientation errors after 300 seconds are shown for different vibration frequencies.

The position error increases with vibration frequency with all methods. As expected, the position error of the Forward Euler method is largest for all vibration frequencies. The position errors of the Single Frequency CIE, Modified Euler CIE and Runge-Kutta CIE methods vary with frequency. The magnitudes of the variations behave in accordance with the order of the methods. The position error of the Single Frequency approach does not exhibit these variations but grows monotonously with vibration frequency. The characteristics of the orientation errors are principally similar to the position errors. Lee and Gusinsky are very robust against vibrations. The large orientation error peaks of the Lee method at 100 and 200 Hz are remarkable.

Note that the error particularly grows if the frequency and phase of the vibration are prevalent and stable for a longer period. In reality, this is rarely the case. This simulation rather reveals inherent shortcomings of some methods, which are less apparent when processing real data.

Measurement Noise

Next, the IMU measurements are affected by velocity/angular random walk noise N . Very low noise level are chosen: the power spectral density of the accelerometer noise is $1 \mu\text{g}/\sqrt{\text{Hz}}$, the power spectral density of the gyroscope noise is $1 \text{ mdeg}/\sqrt{\text{h}}$. The assumed noise levels are in the range of highly performant navigation grade IMU. The position errors of the different approaches along the one hour long Generic Lissajous trajectory are plotted in Figure 4-18. The corresponding orientation errors are illustrated in Figure 4-19. It is remarkable that all methods except for the Forward Euler CIE algorithm feature the same position error curves. The same is valid for the orientation errors. The orientation errors of Lee and Gusinsky differ from the Runge-Kutta CIE method but are of the same order. The example clearly demonstrates that even low-level noise overrides the theoretical numerical truncation order of the methods. The 1st to 4th order methods yield same results, only the navigation errors of the 0th order algorithm are worse.

**Figure 4-14: Position error****Figure 4-15: Orientation error****Figure 4-16: Influence of vibrations on position error after 300 s**

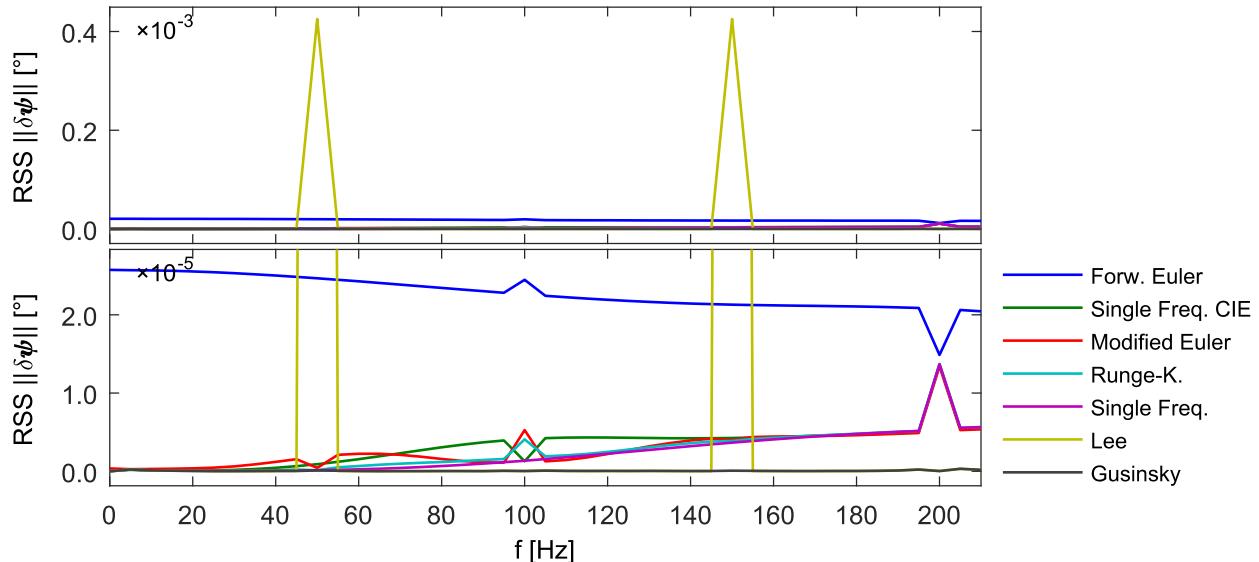


Figure 4-17: Influence of vibrations on orientation error after 300 s

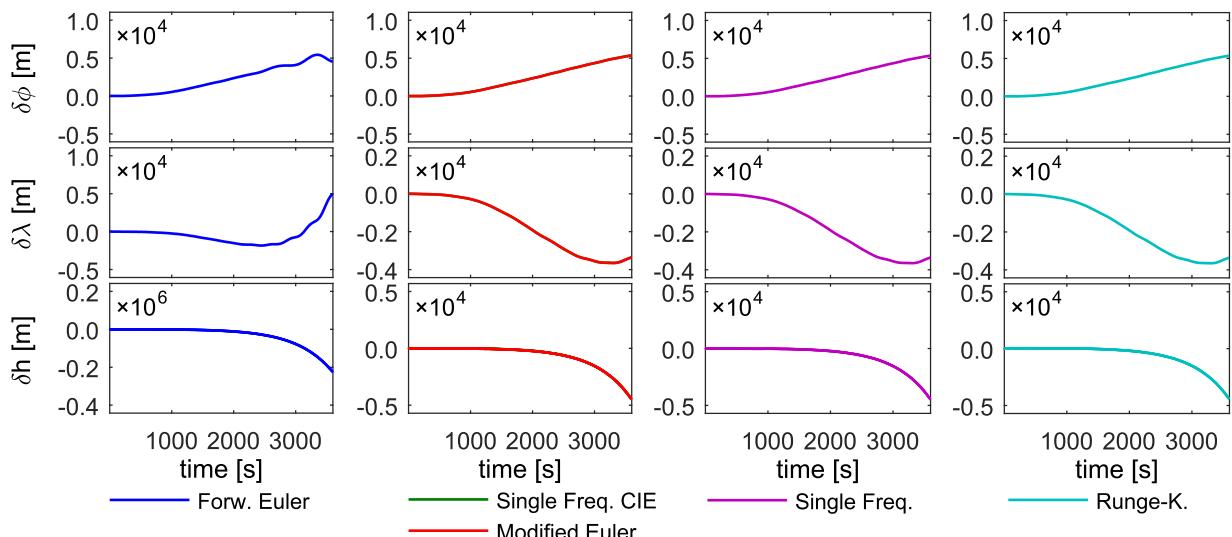


Figure 4-18: Influence of velocity random walk noise on the position error

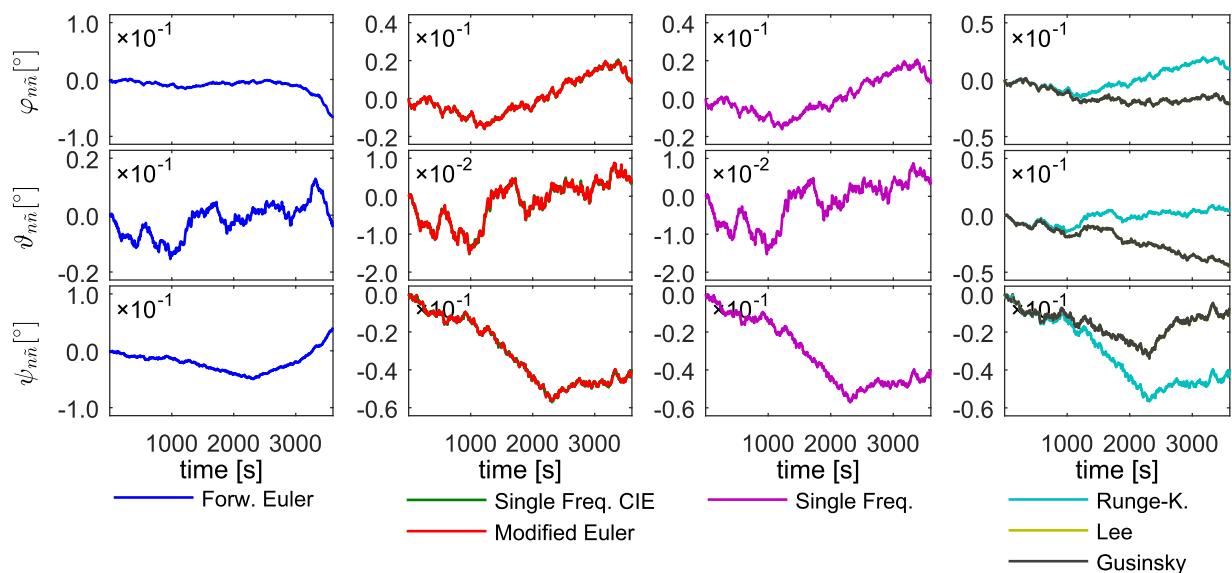


Figure 4-19: Influence of angular random walk noise on the orientation error

4.1.7 Conclusion

The simple averaging algorithm (4.1.2) is suitable for processing highly noisy measurements. In fact, high noise destroys the accuracy that is achievable by more sophisticated, higher order methods like the CIE methods.

For less noisy measurements the CIE methods (4.1.4) as well as the Single Frequency approach (4.1.5) provide more accurate results with comparably low additional effort. By appropriately combining conversion and integration, the CIE algorithms can be optimally adopted to the operational environment. Central integrating IMU conversion schemes offer high robustness against vibration and noise. Especially the Single Frequency 1st order CIE uses the same measurements and works with the same rate as the simple averaging algorithm, but provides a higher error order. Therefore, this algorithm should be always preferred to the averaging algorithm. Indeed, this algorithm is a good tradeoff between accuracy increase on the one hand and additional effort on the other hand.

The Single Frequency approach is a veritable alternative to CIE algorithms and the dual frequency algorithms (4.1.3). It is highly accurate, robust against vibrations and noise and can be used in all operational environments without adoptions. The single frequency approach captivates by its high error order and by its single sample rate.

The Lee and Gusinsky dual frequency algorithms revealed higher accuracy and higher robustness against noise compared to the Single Frequency approach. These algorithms are computationally very efficient. A disadvantage is that the algorithms provide only an output value every fourth input measurement value. Therefore, a combination of the dual frequency orientation algorithms, augmented by interim orientation extrapolation for example with the Single Frequency 1st order CIE, together with the position and velocity propagation equations of the Single Frequency approach would be a good possibility.

4.2 Navigation Filter Initialization

4.2.1 Motivation

At navigation system start-up the expected values of position, velocity and orientation as well as the corresponding error covariances are required to initialize the integrators of the inertial navigation equations and the navigation error filter. It is distinguished between stationary and in-flight initialization.

The stationary initialization is the easiest and most accurate way to set up the navigation filter. The carrier platform is kept at rest during the initialization period. The initial position is either known because the carrier platform is at a surveyed point or is provided by the GNSS receiver, the initial velocity is zero by definition and the initial orientation angles are obtained by observing Earth's gravity and angular rate. It has to be paid attention that disturbing motion caused for example by wind gusts or service operations may degrade the accuracy particularly of the initial orientation angles. After initialization, before takeoff or launch, the navigation filter may switch to interim modes like the zero velocity update mode or heading angle aiding mode to bound the navigation error growth if the observability (especially of the heading angle error) is constrained due to missing motion.

The in-flight initialization is very important from an operational point of view. In order to increase the availability of the navigation system, it has to be capable to restart and reinitialize after a service interruption. This might be the case if the navigation system has to be rebooted after a short power failure or a hang-up of the embedded computing module. It is necessary that reliable navigation information is provided to the flight guidance and control again as soon as possible after the interruption. In the in-flight initialization phase, it has to be accounted for that the carrier platform moves and is not stationary.

The RTCA standard DO-334, “Minimum Operational Performance Standards (MOPS) for Strapdown Attitude and Heading Reference Systems (AHRS)” [5], gives the requirements for the stationary initialization in section 2.2.1.2, “The equipment shall start and provide valid attitude outputs within 3 minutes after normal rated power is applied. [...] For heading category H1 [which is related to non-magnetic heading determination by gyrocompassing with an accuracy of 2° under dynamic and flight conditions], the equipment shall start and provide valid heading output within 10 minutes after normal rated power is applied, when operating in latitudes less than $\pm 60^\circ$.“ Additionally, it is required in section 2.2.1.3 that “In-flight or in-motion alignment should be supported. When supported, restrictions on the conditions where in-motion alignment is available and the subsequent performance that is provided shall be specified.”

The initial position, velocity and orientation error covariance matrix is usually set up with a diagonal matrix. The diagonal elements are chosen conservatively large because it is assumed that the filter will estimate the correct covariance in the initial settlement phase. The correlation between the single position and velocity components as well as the orientation angles are mostly neglected. Moreover, the correlation between the initial orientation angle errors and the accelerometer turn-on biases, which is negative since both errors are only observable in linear combination, is not considered.

In this section, methods for the stationary and in-flight initialization are presented. The focus is put on the estimation of the initial orientation angles and the corresponding, statistically correct covariance since the initial position and velocity are easily obtained from the GNSS receiver. Furthermore, the

correlation between the orientation angles and the IMU errors like turn-on biases, scale factor errors and misalignments will be considered. It will be analyzed if a statistically correct initial covariance matrix is beneficial compared to a diagonal substitute covariance matrix. The importance of the statistically correct initialization for consistency is outlined in [102].

4.2.2 Stationary Navigation Filter Initialization

4.2.2.1 Orientation Angle Expected Values

The derivation of the formulas for the orientation angle expected values is given in the appendix in section C.2.1. The roll angle φ_{nb} and the pitch angle ϑ_{nb} are calculated as

$$\varphi_{nb} = \text{atan2}(-f_{b,y}, -f_{b,z}) \quad \in [-\pi, \pi) \quad (4.18)$$

$$\vartheta_{nb} = \arcsin \frac{f_{b,x}}{\gamma} \quad \in \left[-\frac{\pi}{2}, \frac{\pi}{2} \right] \quad (4.19)$$

The heading angle ψ_{nb} is found by calculating the double-argument arctangent

$$\psi_{nb} = \text{atan2}(-\omega_{ib,y} \cos \varphi_{nb} + \omega_{ib,z} \sin \varphi_{nb}, \dots \\ \omega_{ib,x} \cos \vartheta_{nb} + \omega_{ib,y} \sin \varphi_{nb} \sin \vartheta_{nb} + \omega_{ib,z} \cos \varphi_{nb} \sin \vartheta_{nb}) \quad \in [-\pi, \pi) \quad (4.20)$$

It is remarkable that only the entries of the measured angular rate vector $\boldsymbol{\omega}_{ib}$ are required for the gyrocompassing, but not the Earth rate $\boldsymbol{\omega}_{ie}$.

4.2.2.2 Orientation Angle Error Equations

In order to determine the errors of the estimated initial roll, pitch and heading angles, the orientation angle error equations for $\delta\varphi_{\tilde{nb}}$, $\delta\vartheta_{\tilde{nb}}$ and $\delta\psi_{\tilde{nb}}$ have to be derived.

The true, error-free roll angle φ_{nb} and the erroneous roll angle $\varphi_{\tilde{nb}}$ are according to (4.18)

$$\varphi_{nb} = \text{atan2}(-f_{b,y}, -f_{b,z}), \quad \varphi_{\tilde{nb}} = \text{atan2}(-\tilde{f}_{b,y}, -\tilde{f}_{b,z}) \quad (4.21)$$

Perturbation of both sides of the equation above yields

$$\varphi_{\tilde{nb}} + \delta\varphi_{\tilde{nb}} = \varphi_{nb} + \left. \frac{\partial \varphi_{nb}}{\partial \mathbf{f}_b^\top} \right|_{\sim} \delta\mathbf{f}_b + h.o.t. \quad (4.22)$$

Using the derivative of the arctangent function [103]

$$\frac{d}{dx} \arctan x = \begin{cases} \frac{1}{1+x^2} & x \neq 0 \\ 0 & x = 0 \end{cases} \quad (4.23)$$

the linearized roll angle error is

$$\delta\varphi_{\tilde{nb}} \doteq \left. \frac{\partial \varphi_{nb}}{\partial \mathbf{f}_b^\top} \right|_{\sim} \delta\mathbf{f}_b = \frac{1}{\tilde{f}_{b,y}^2 + \tilde{f}_{b,z}^2} \begin{pmatrix} 0 \\ \tilde{f}_{b,z} \\ -\tilde{f}_{b,y} \end{pmatrix}^\top \delta\mathbf{f}_b \quad (4.24)$$

The true, error-free pitch angle ϑ_{nb} and the erroneous pitch angle $\vartheta_{\tilde{nb}}$ are from (4.19)

$$\vartheta_{nb} = \arcsin \frac{f_{b,x}}{\gamma}, \quad \vartheta_{\tilde{nb}} = \arcsin \frac{\tilde{f}_{b,x}}{\tilde{\gamma}} \quad (4.25)$$

Using the definition of the arcsine derivative [103]

$$\frac{d}{dx} \arcsin x = \frac{1}{\sqrt{1-x^2}} \quad (4.26)$$

the linearized pitch angle error $\delta\vartheta_{\tilde{n}b}$ is then, analogous to the roll angle error $\delta\varphi_{\tilde{n}b}$ (4.24),

$$\delta\vartheta_{\tilde{n}b} \doteq \frac{\partial\vartheta_{nb}}{\partial\mathbf{f}_b^\top} \left| \delta\mathbf{f}_b + \frac{\partial\vartheta_{nb}}{\partial\mathbf{Y}_n^\top} \right| \delta\mathbf{Y}_n = \frac{1}{\sqrt{\tilde{\gamma}^2 - \tilde{f}_{b,x}^2}} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}^\top \delta\mathbf{f}_b - \frac{\tilde{f}_{b,x}}{\tilde{\gamma}\sqrt{\tilde{\gamma}^2 - \tilde{f}_{b,x}^2}} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}^\top \delta\mathbf{Y}_n \quad (4.27)$$

Usually, the gravity model error $\delta\mathbf{Y}_n$ is very small compared to the acceleration error $\delta\mathbf{f}_b$.

The true, error-free roll angle ψ_{nb} and the erroneous heading angle $\psi_{\tilde{n}b}$ are according to (4.20)

$$\begin{aligned} \psi_{nb} &= \text{atan2}(R_{nb,21}, R_{nb,11}) \\ &= \text{atan2}(-\omega_{ib,y} \cos \varphi_{nb} + \omega_{ib,z} \sin \varphi_{nb}, \omega_{ib,x} \cos \vartheta_{nb} + \omega_{ib,y} \sin \varphi_{nb} \sin \vartheta_{nb} + \omega_{ib,z} \cos \varphi_{nb} \sin \vartheta_{nb}) \\ \psi_{\tilde{n}b} &= \text{atan2}(R_{\tilde{n}b,21}, R_{\tilde{n}b,11}) \\ &= \text{atan2}(-\tilde{\omega}_{ib,y} \cos \varphi_{\tilde{n}b} + \tilde{\omega}_{ib,z} \sin \varphi_{\tilde{n}b}, \tilde{\omega}_{ib,x} \cos \vartheta_{\tilde{n}b} + \tilde{\omega}_{ib,y} \sin \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} + \tilde{\omega}_{ib,z} \cos \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b}) \end{aligned} \quad (4.28)$$

The linearized heading error $\delta\psi_{\tilde{n}b}$ is

$$\begin{aligned} \delta\psi_{\tilde{n}b} &\doteq \frac{\partial\psi_{nb}}{\partial\mathbf{\omega}_{ib}^\top} \left| \delta\mathbf{\omega}_{ib} + \frac{\partial\psi_{nb}}{\partial\varphi_{nb}} \right| \delta\varphi_{\tilde{n}b} + \frac{\partial\psi_{nb}}{\partial\vartheta_{nb}} \left| \delta\vartheta_{\tilde{n}b} \right| \\ &= \frac{\partial\psi_{nb}}{\partial\mathbf{\omega}_{ib}^\top} \left| \delta\mathbf{\omega}_{ib} + \left(\frac{\partial\psi_{nb}}{\partial\varphi_{nb}} \left| \frac{\partial\varphi_{nb}}{\partial\mathbf{f}_b^\top} \right| + \frac{\partial\psi_{nb}}{\partial\vartheta_{nb}} \left| \frac{\partial\vartheta_{nb}}{\partial\mathbf{f}_b^\top} \right| \right) \delta\mathbf{f}_b + \frac{\partial\psi_{nb}}{\partial\vartheta_{nb}} \left| \frac{\partial\vartheta_{nb}}{\partial\mathbf{Y}_n^\top} \right| \delta\mathbf{Y}_n \right| \end{aligned} \quad (4.29)$$

Using again (4.23), the derivatives are

$$\frac{\partial\psi_{nb}}{\partial\mathbf{\omega}_{ib}^\top} \left| = \frac{1}{R_{\tilde{n}b,11}^2 + R_{\tilde{n}b,21}^2} \begin{pmatrix} -R_{\tilde{n}b,21} \cos \vartheta_{\tilde{n}b} \\ -R_{\tilde{n}b,11} \cos \varphi_{\tilde{n}b} - R_{\tilde{n}b,21} \sin \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} \\ R_{\tilde{n}b,11} \sin \varphi_{\tilde{n}b} - R_{\tilde{n}b,21} \cos \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} \end{pmatrix}^\top \right. \quad (4.30)$$

and

$$\begin{aligned} \frac{\partial\psi_{nb}}{\partial\varphi_{nb}} \left| &= \frac{R_{\tilde{n}b,11}(\tilde{\omega}_{ib,y} \sin \varphi_{\tilde{n}b} + \tilde{\omega}_{ib,z} \cos \varphi_{\tilde{n}b}) - R_{\tilde{n}b,21}(\tilde{\omega}_{ib,y} \cos \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} - \tilde{\omega}_{ib,z} \sin \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b})}{R_{\tilde{n}b,11}^2 + R_{\tilde{n}b,21}^2} \\ \frac{\partial\psi_{nb}}{\partial\vartheta_{nb}} \left| &= \frac{R_{\tilde{n}b,21}(\tilde{\omega}_{ib,x} \sin \vartheta_{\tilde{n}b} - \tilde{\omega}_{ib,y} \sin \varphi_{\tilde{n}b} \cos \vartheta_{\tilde{n}b} - \tilde{\omega}_{ib,z} \cos \varphi_{\tilde{n}b} \cos \vartheta_{\tilde{n}b})}{R_{\tilde{n}b,11}^2 + R_{\tilde{n}b,21}^2} \end{aligned} \quad (4.31)$$

Note that the roll angle error $\delta\varphi_{\tilde{n}b}$, pitch angle error $\delta\vartheta_{\tilde{n}b}$ and heading error angle $\delta\psi_{\tilde{n}b}$ are not the orientation errors $\varphi_{n\tilde{n}}$, $\vartheta_{n\tilde{n}}$ and $\psi_{n\tilde{n}}$ that are used in the navigation error equations (2.7) within the navigation filter. The relationship between the angle errors $\delta\varphi_{\tilde{n}b}$, $\delta\vartheta_{\tilde{n}b}$ and $\delta\psi_{\tilde{n}b}$ and the orientation error vector $\mathbf{\Psi}_{n\tilde{n}} = (\varphi_{n\tilde{n}}, \vartheta_{n\tilde{n}}, \psi_{n\tilde{n}})^\top$ is given with

$$\begin{aligned} \mathbf{\Psi}_{n\tilde{n}} &= \begin{pmatrix} \varphi_{n\tilde{n}} \\ \vartheta_{n\tilde{n}} \\ \psi_{n\tilde{n}} \end{pmatrix} = \mathbf{R}_3(\psi_{\tilde{n}b}) \mathbf{R}_2(\vartheta_{\tilde{n}b}) \begin{pmatrix} \delta\varphi_{\tilde{n}b} \\ 0 \\ 0 \end{pmatrix} + \mathbf{R}_3(\psi_{\tilde{n}b}) \begin{pmatrix} 0 \\ \delta\vartheta_{\tilde{n}b} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \delta\psi_{\tilde{n}b} \end{pmatrix} \\ &= \begin{pmatrix} \cos \vartheta_{\tilde{n}b} \cos \psi_{\tilde{n}b} & -\sin \psi_{\tilde{n}b} & 0 \\ \cos \vartheta_{\tilde{n}b} \sin \psi_{\tilde{n}b} & \cos \psi_{\tilde{n}b} & 0 \\ -\sin \vartheta_{\tilde{n}b} & 0 & 1 \end{pmatrix} \begin{pmatrix} \delta\varphi_{\tilde{n}b} \\ \delta\vartheta_{\tilde{n}b} \\ \delta\psi_{\tilde{n}b} \end{pmatrix} \end{aligned} \quad (4.32)$$

The orientation error $\Psi_{n\bar{n}}$ depends on the acceleration error $\delta\mathbf{f}_b$, the angular rate error $\delta\omega_{ib}$ and the gravity model error $\delta\mathbf{y}_n$. The dependence is expressed with the general mapping matrices \mathbf{H}_f , \mathbf{H}_ω and \mathbf{H}_γ

$$\Psi_{n\bar{n}} = \mathbf{H}_f \delta\mathbf{f}_b + \mathbf{H}_\omega \delta\omega_{ib} + \mathbf{H}_\gamma \delta\mathbf{y}_n \quad (4.33)$$

which are specified by

$$\mathbf{H}_f = \left[\begin{pmatrix} \cos \vartheta_{nb} \cos \psi_{nb} \\ \cos \vartheta_{nb} \sin \psi_{nb} \\ -\sin \vartheta_{nb} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \right] \frac{\partial \vartheta_{nb}}{\partial \mathbf{f}_b^\top} + \left[\begin{pmatrix} -\sin \psi_{nb} \\ \cos \psi_{nb} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \vartheta_{nb}}{\partial \mathbf{g}_{nb}} \right] \frac{\partial \mathbf{g}_{nb}}{\partial \mathbf{f}_b^\top} \quad (4.34)$$

$$\mathbf{H}_\omega = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \psi_{nb}}{\partial \omega_{ib}^\top} \quad (4.35)$$

$$\mathbf{H}_\gamma = \left[\begin{pmatrix} -\sin \psi_{nb} \\ \cos \psi_{nb} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \vartheta_{nb}}{\partial \mathbf{y}_n^\top} \right] \frac{\partial \vartheta_{nb}}{\partial \mathbf{y}_n^\top} \quad (4.36)$$

4.2.2.3 Orientation Angle Error Covariance

Assuming that the mean value of the orientation error is zero, $E[\Psi_{n\bar{n}}] = 0$, the orientation error covariance is simply the expectation value of the dyadic product of the orientation error vector

$$\text{Cov}(\Psi_{n\bar{n}}) = E[\Psi_{n\bar{n}} \Psi_{n\bar{n}}^\top] \quad (4.37)$$

Inserting the orientation error equation (4.33) and presuming that all three errors are not cross-correlated, that is $E[\delta\mathbf{f}_b \delta\omega_{ib}^\top] = 0$, $E[\delta\mathbf{f}_b \delta\mathbf{y}_n^\top] = 0$ and $E[\delta\omega_{ib} \delta\mathbf{y}_n^\top] = 0$, gives

$$E[\Psi_{n\bar{n}} \Psi_{n\bar{n}}^\top] = \mathbf{H}_f E[\delta\mathbf{f}_b \delta\mathbf{f}_b^\top] \mathbf{H}_f^\top + \mathbf{H}_\omega E[\delta\omega_{ib} \delta\omega_{ib}^\top] \mathbf{H}_\omega^\top + \mathbf{H}_\gamma E[\delta\mathbf{y}_n \delta\mathbf{y}_n^\top] \mathbf{H}_\gamma^\top \quad (4.38)$$

Furthermore, the correlation of the orientation error $\Psi_{n\bar{n}}$ and the acceleration error $\delta\mathbf{f}_b$

$$E[\Psi_{n\bar{n}} \delta\mathbf{f}_b^\top] = \mathbf{H}_f E[\delta\mathbf{f}_b \delta\mathbf{f}_b^\top] \quad (4.39)$$

as well as the correlation of the orientation error $\Psi_{n\bar{n}}$ and the angular rate error $\delta\omega_{ib}$

$$E[\Psi_{n\bar{n}} \delta\omega_{ib}^\top] = \mathbf{H}_\omega E[\delta\omega_{ib} \delta\omega_{ib}^\top] \quad (4.40)$$

are later required for the initialization of the navigation error filter. The cross-covariances between the orientation error vector and the explicit IMU errors (turn-on biases, scale factor errors and misalignments) are given in section C.2.2 in the appendix.

Furthermore, in order to reduce the influence of noise-like IMU errors, the IMU measurements have to be averaged over a certain timespan. As described in section C.2.3 in the appendix this can be done recursively, updating the averaged IMU measurements with each new sample. Averaging can thus be cancelled if the variances of the averaged values go below a given upper boundary.

4.2.2.4 Pre-Flight Calibration of Downward Gyroscope Error

The downward angular rate component (in the direction of the gravity vector) is not required for the initialization of the orientation angles, in particular not for the heading angle ψ_{nb} . This can be easily verified with equation (4.20) for a horizontally leveled platform with $\varphi_{nb} = 0$ and $\vartheta_{nb} = 0$. It can be seen that $\omega_{ib,z}$ is not required in this equation. The bias-like gyroscope errors in downward direction can therefore be pre-flight calibrated. This is pretty advantageous because it is just this component of the rotational degrees of freedom that is worst observable during flight. The downward angular rate component is given with

$$\omega_{ib_{n,z}} = \begin{pmatrix} -\sin \vartheta_{nb} \\ \sin \varphi_{nb} \cos \vartheta_{nb} \\ \cos \varphi_{nb} \cos \vartheta_{nb} \end{pmatrix}^T \bar{\boldsymbol{\omega}}_{ib} \quad (4.41)$$

It is equal with

$$\omega_{ib_{n,z}} = -\sin \phi \omega_{ie} \quad (4.42)$$

If the pre-flight calibration is done after the recursive averaging of the angular rate measurements, all noise-like errors have been removed and only the bias-like error components remain. With the observation matrix

$$\mathbf{H}_\omega = \begin{pmatrix} -\sin \vartheta_{nb} \\ \sin \varphi_{nb} \cos \vartheta_{nb} \\ \cos \varphi_{nb} \cos \vartheta_{nb} \end{pmatrix}^T (\mathbf{I}_3, \text{ diag}(\bar{\boldsymbol{\omega}}_{ib}), \mathbf{F}_{M_\omega}) \quad (4.43)$$

and covariance matrix

$$\mathbf{P}_\omega = \begin{pmatrix} \mathbb{E}[\mathbf{b}_{0,\omega} \mathbf{b}_{0,\omega}^T] & 0 & 0 \\ 0 & \mathbb{E}[\mathbf{s}_{0,\omega} \mathbf{s}_{0,\omega}^T] & 0 \\ 0 & 0 & \mathbb{E}[\mathbf{m}_\omega \mathbf{m}_\omega^T] \end{pmatrix} \quad (4.44)$$

and gain matrix

$$\mathbf{K} = \mathbf{P}_\omega \mathbf{H}_\omega^T (\mathbf{H}_\omega \mathbf{P}_\omega \mathbf{H}_\omega^T)^{-1} \quad (4.45)$$

the calibration values for turn-on bias $\mathbf{b}_{0,\omega}$, scale factor error $\mathbf{s}_{0,\omega}$ and misalignments \mathbf{m}_ω and the corresponding covariance are estimated with

$$\begin{aligned} \begin{pmatrix} \hat{\mathbf{b}}_{0,\omega} \\ \hat{\mathbf{s}}_{0,\omega} \\ \hat{\mathbf{m}}_\omega \end{pmatrix} &= -\mathbf{K} \left(\sin \tilde{\phi} \omega_{ie} + \begin{pmatrix} -\sin \vartheta_{nb} \\ \sin \varphi_{nb} \cos \vartheta_{nb} \\ \cos \varphi_{nb} \cos \vartheta_{nb} \end{pmatrix}^T \bar{\boldsymbol{\omega}}_{ib} \right) \\ \mathbf{P}_{\omega,cal} &= (\mathbf{I}_{12} - \mathbf{K} \mathbf{H}_\omega) \mathbf{P}_\omega \end{aligned} \quad (4.46)$$

All other bias-like errors are neglected here but can be easily added in the same manner if required. Note that the covariance matrix now also has off-diagonal entries representing the negative correlations between the single errors due to the inability to uniquely separate the errors. This cross-correlations should be considered in the initial covariance matrix of the navigation filter.

4.2.2.5 Template

The equations for the expected values of the roll angle $\varphi_{\tilde{nb}}$, pitch angle $\vartheta_{\tilde{nb}}$ and heading angle $\psi_{\tilde{nb}}$ are summarized in Template 4-4. The required mean values of the accelerometer and gyroscope measurements, the corresponding covariances and the cross-covariances of the averaged sensor values and the single sensor error components can be found in Template 4-5. The orientation error covariance and the cross-covariances between the orientation error and the accelerometer error, gyroscope error and gravity model error are recapitulated in Template 4-6. The required sensitivities are given in Template 4-7.

Template 4-4: Orientation angle expected values

Leveling	$\varphi_{\tilde{nb}} = \text{atan}2(-\bar{f}_{b,y}, -\bar{f}_{b,z}) \in [-\pi, \pi]$	$\vartheta_{\tilde{nb}} = \arcsin \frac{\bar{f}_{b,x}}{\tilde{\gamma}} \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$
Gyro Compassing		
	$\psi_{\tilde{nb}} = \text{atan}2(-\bar{\omega}_{ib,y} \cos \varphi_{\tilde{nb}} + \bar{\omega}_{ib,z} \sin \varphi_{\tilde{nb}}, \bar{\omega}_{ib,x} \cos \vartheta_{\tilde{nb}} + \bar{\omega}_{ib,y} \sin \varphi_{\tilde{nb}} \sin \vartheta_{\tilde{nb}} + \bar{\omega}_{ib,z} \cos \varphi_{\tilde{nb}} \sin \vartheta_{\tilde{nb}})$	$\in [-\pi, \pi)$

Template 4-5: Recursive averaging

Recursive Averaging	$\bar{\mathbf{f}}_{b,k} = \frac{k-1}{k} \bar{\mathbf{f}}_{b,k-1} + \frac{1}{k} \tilde{\mathbf{f}}_{b,k}$	$k = 1, 2, \dots$
	$E[\delta \bar{\mathbf{f}}_{b,k} \delta \bar{\mathbf{f}}_{b,k}^T] = \left(\frac{k-1}{k}\right)^2 E[\delta \bar{\mathbf{f}}_{b,k-1} \delta \bar{\mathbf{f}}_{b,k-1}^T]$	$E[\delta \bar{\mathbf{f}}_{b,0} \delta \bar{\mathbf{f}}_{b,0}^T] = 0$
	$+ \frac{1}{k^2} \left(\sum_{i \in \{N, B\}} \mathbf{Q}_{b_i,f} + E[\mathbf{b}_{0,f} \mathbf{b}_{0,f}^T] + \text{diag}(\tilde{\mathbf{f}}_{b,k}) E[\mathbf{s}_{0,f} \mathbf{s}_{0,f}^T] \text{diag}(\tilde{\mathbf{f}}_{b,k}) + \mathbf{F}_{M_f} E[\mathbf{m}_f \mathbf{m}_f^T] \mathbf{F}_{M_f}^T \right)$	
	$+ \frac{k-1}{k^2} \left(E[\mathbf{b}_{0,f} \delta \bar{\mathbf{f}}_{b,k-1}^T] + \text{diag}(\tilde{\mathbf{f}}_{b,k}) E[\mathbf{s}_{0,f} \delta \bar{\mathbf{f}}_{b,k-1}^T] + \mathbf{F}_{M_f} E[\mathbf{m}_f \delta \bar{\mathbf{f}}_{b,k-1}^T]$	
	$+ E[\delta \bar{\mathbf{f}}_{b,k-1} \mathbf{b}_{0,f}^T] + E[\delta \bar{\mathbf{f}}_{b,k-1} \mathbf{s}_{0,f}^T] \text{diag}(\tilde{\mathbf{f}}_{b,k}) + E[\delta \bar{\mathbf{f}}_{b,k-1} \mathbf{m}_f^T] \mathbf{F}_{M_f}^T \right)$	
	$E[\delta \bar{\mathbf{f}}_{b,k} \mathbf{b}_{0,f}^T] = \frac{k-1}{k} E[\delta \bar{\mathbf{f}}_{b,k-1} \mathbf{b}_{0,f}^T] + \frac{1}{k} E[\mathbf{b}_{0,f} \mathbf{b}_{0,f}^T]$	$E[\delta \bar{\mathbf{f}}_{b,0} \mathbf{b}_{0,f}^T] = 0$
	$E[\delta \bar{\mathbf{f}}_{b,k} \mathbf{s}_{0,f}^T] = \frac{k-1}{k} E[\delta \bar{\mathbf{f}}_{b,k-1} \mathbf{s}_{0,f}^T] + \frac{1}{k} \text{diag}(\tilde{\mathbf{f}}_{b,k}) E[\mathbf{s}_{0,f} \mathbf{s}_{0,f}^T]$	$E[\delta \bar{\mathbf{f}}_{b,0} \mathbf{s}_{0,f}^T] = 0$
	$E[\delta \bar{\mathbf{f}}_{b,k} \mathbf{m}_f^T] = \frac{k-1}{k} E[\delta \bar{\mathbf{f}}_{b,k-1} \mathbf{m}_f^T] + \frac{1}{k} \mathbf{F}_{M_f} E[\mathbf{m}_f \mathbf{m}_f^T]$	$E[\delta \bar{\mathbf{f}}_{b,0} \mathbf{m}_f^T] = 0$
Analog for angular rate measurements $\bar{\boldsymbol{\omega}}_{ib}$.		

Template 4-6: Orientation error covariance and IMU error cross-covariances**Orientation error covariance**

$$E[\Psi_{\tilde{n}\tilde{n}} \Psi_{\tilde{n}\tilde{n}}^T] = \mathbf{H}_f E[\delta \tilde{\mathbf{f}}_b \delta \tilde{\mathbf{f}}_b^T] \mathbf{H}_f^T + \mathbf{H}_\omega E[\delta \bar{\boldsymbol{\omega}}_{ib} \delta \bar{\boldsymbol{\omega}}_{ib}^T] \mathbf{H}_\omega^T + \mathbf{H}_\gamma E[\delta \mathbf{Y}_n \delta \mathbf{Y}_n^T] \mathbf{H}_\gamma^T$$

Orientation error and IMU error cross-covariances

$$E[\Psi_{\tilde{n}\tilde{n}} \mathbf{b}_{0,f}^T] = \mathbf{H}_f E[\delta \tilde{\mathbf{f}}_b \mathbf{b}_{0,f}^T]$$

$$E[\Psi_{\tilde{n}\tilde{n}} \mathbf{b}_{0,\omega}^T] = \mathbf{H}_\omega E[\delta \bar{\boldsymbol{\omega}}_{ib} \mathbf{b}_{0,\omega}^T]$$

$$E[\Psi_{\tilde{n}\tilde{n}} \mathbf{s}_{0,f}^T] = \mathbf{H}_f E[\delta \tilde{\mathbf{f}}_b \mathbf{s}_{0,f}^T]$$

$$E[\Psi_{\tilde{n}\tilde{n}} \mathbf{s}_{0,\omega}^T] = \mathbf{H}_\omega E[\delta \bar{\boldsymbol{\omega}}_{ib} \mathbf{s}_{0,\omega}^T]$$

$$E[\Psi_{\tilde{n}\tilde{n}} \mathbf{m}_f^T] = \mathbf{H}_f E[\delta \tilde{\mathbf{f}}_b \mathbf{m}_f^T]$$

$$E[\Psi_{\tilde{n}\tilde{n}} \mathbf{m}_\omega^T] = \mathbf{H}_\omega E[\delta \bar{\boldsymbol{\omega}}_{ib} \mathbf{m}_\omega^T]$$

Mapping matrices

$$\mathbf{H}_f = \left[\begin{pmatrix} \cos \vartheta_{\tilde{n}b} & \cos \psi_{\tilde{n}b} \\ \cos \vartheta_{\tilde{n}b} & \sin \psi_{\tilde{n}b} \\ -\sin \vartheta_{\tilde{n}b} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \right] \frac{\partial \vartheta_{nb}}{\partial \tilde{\mathbf{f}}_b^T} + \left[\begin{pmatrix} -\sin \psi_{\tilde{n}b} \\ \cos \psi_{\tilde{n}b} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \right] \frac{\partial \vartheta_{nb}}{\partial \tilde{\mathbf{f}}_b^T}$$

$$\mathbf{H}_\omega = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \psi_{nb}}{\partial \bar{\boldsymbol{\omega}}_{ib}^T}, \quad \mathbf{H}_\gamma = \left[\begin{pmatrix} -\sin \psi_{\tilde{n}b} \\ \cos \psi_{\tilde{n}b} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \right] \frac{\partial \vartheta_{nb}}{\partial \mathbf{Y}_n^T}$$

Template 4-7: Orientation error sensitivities

	$\frac{1}{\partial \tilde{\mathbf{f}}_b^T}$	$\frac{1}{\partial \mathbf{Y}_n^T}$	$\frac{1}{\partial \bar{\boldsymbol{\omega}}_{ib}^T}$
$\partial \vartheta_{nb}$	$\frac{1}{\tilde{f}_{b,y}^2 + \tilde{f}_{b,z}^2} \begin{pmatrix} 0 \\ \tilde{f}_{b,z} \\ -\tilde{f}_{b,y} \end{pmatrix}^T$		
$\partial \vartheta_{nb}$	$\frac{1}{\sqrt{\tilde{\gamma}^2 - \tilde{f}_{b,x}^2}} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}^T$	$-\frac{\tilde{f}_{b,x}}{\tilde{\gamma} \sqrt{\tilde{\gamma}^2 - \tilde{f}_{b,x}^2}} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}^T$	
$\partial \psi_{nb}$	$\frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \frac{\partial \vartheta_{nb}}{\partial \tilde{\mathbf{f}}_b^T} + \frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \frac{\partial \vartheta_{nb}}{\partial \mathbf{Y}_n^T}$	$\frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} \frac{\partial \vartheta_{nb}}{\partial \mathbf{Y}_n^T}$	$\frac{1}{R_{\tilde{n}b,11}^2 + R_{\tilde{n}b,21}^2} \begin{pmatrix} -R_{\tilde{n}b,21} \cos \vartheta_{\tilde{n}b} \\ -R_{\tilde{n}b,11} \cos \varphi_{\tilde{n}b} - R_{\tilde{n}b,21} \sin \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} \\ R_{\tilde{n}b,11} \sin \varphi_{\tilde{n}b} - R_{\tilde{n}b,21} \cos \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} \end{pmatrix}^T$
$\partial \psi_{nb}$	$\frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} = \frac{R_{\tilde{n}b,11} (\tilde{\omega}_{ib,y} \sin \varphi_{\tilde{n}b} + \tilde{\omega}_{ib,z} \cos \varphi_{\tilde{n}b}) - R_{\tilde{n}b,21} (\tilde{\omega}_{ib,y} \cos \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b} - \tilde{\omega}_{ib,z} \sin \varphi_{\tilde{n}b} \sin \vartheta_{\tilde{n}b})}{R_{\tilde{n}b,11}^2 + R_{\tilde{n}b,21}^2}$		
	$\frac{\partial \psi_{nb}}{\partial \vartheta_{nb}} = \frac{R_{\tilde{n}b,21} (\tilde{\omega}_{ib,x} \sin \vartheta_{\tilde{n}b} - \tilde{\omega}_{ib,y} \sin \varphi_{\tilde{n}b} \cos \vartheta_{\tilde{n}b} - \tilde{\omega}_{ib,z} \cos \varphi_{\tilde{n}b} \cos \vartheta_{\tilde{n}b})}{R_{\tilde{n}b,11}^2 + R_{\tilde{n}b,21}^2}$		
	$R_{\tilde{n}b,11} = \cos \vartheta_{\tilde{n}b} \cos \psi_{\tilde{n}b}$		
	$R_{\tilde{n}b,21} = \sin \vartheta_{\tilde{n}b} \cos \vartheta_{\tilde{n}b} \sin \psi_{\tilde{n}b}$		

4.2.2.6 Example

The General Aviation Aircraft scenario (section E.1.2) and IMU 4 as specified in Table E-5 are used for the example. The sample rate is 100 Hz. The inertial navigation algorithm is initialized during the first 60 seconds of the trajectory in which the aircraft is still at standstill. After the initialization period the integrated navigation solution is computed for 120 seconds using 1 Hz pseudorange measurements to six satellites in view. Take-off time is at June 22, 2012 at 7pm. The predicted covariance of the orientation error is compared with the actual covariance that is calculated from 500 Monte Carlo runs. Figure 4-20 shows the 3σ standard deviations of the roll, pitch and heading angle errors using the formulas of the previously given templates. The blue dotted lines are the predicted 3σ standard deviations and the grey dotted lines represent the 3σ boundaries as actually realized in the Monte Carlo simulation. There is a good coincidence between predicted and true statistics. In Figure 4-21, the standard deviations of the orientation errors are shown when the correlations between the IMU and orientation errors are neglected at filter initialization at second 60. The black lines are the reference standard deviation of the first case, which considers the initial correlations. It can be seen that the roll and pitch angle error standard deviations are underestimated between second 60 and second 130. Later on, after second 140, the predicted standard deviations of the roll and pitch angle errors indeed correspond to those of the first case, but the actual standard deviations are slightly larger than the predicted ones. The predicted and true standard deviations of the heading angle error are statistically consistent but larger than in the first case. In Figure 4-22, the initial correlations are neglected as in the previous case and additionally the fully populated initial orientation error covariance matrix is substituted by an over-bounded diagonal covariance matrix. The initial roll and pitch angle error uncertainties are chosen as 0.1° and the initial heading angle uncertainty as 1° (1σ values). It can be observed that it takes some time until the orientation angle errors settle to the values of the first case. The roll and pitch angle errors are settled roughly at second 140. However, as in the second case, the standard deviations are slightly larger than in the first case after second 140. The heading error is worse observable and does not go down to the standard deviation of the first case until second 180. Figure 4-23 depicts the correlation between the single orientation errors for the first case considering the initial correlations (blue), the second case not considering the initial correlations (red) and the third case not considering the initial correlations and with over-bounded diagonal initial orientation error covariance matrix (green). The correlation between the separate orientation errors is zero in the initialization period except between the roll and the heading angle errors. At the end of the simulation, the correlations of all three cases are at least qualitatively the same. Figure 4-24 illustrates the cross-correlation between the orientation angle errors and the turn-on biases of the accelerometer for the three cases. The roll and pitch angle errors are considerably correlated with the turn-on biases of the accelerometer triad. In the second and third cases, the correlations are established in the first 60 seconds after initialization. Figure 4-25 finally shows the cross-correlation between the orientation errors and the gyroscope turn-on biases for the three cases. In contrast to the accelerometer turn-on bias, the heading angle error is mainly affected. It is confirmed that the roll and pitch angles are not affected by gyroscope errors. The aircraft is horizontally leveled during initialization and the IMU z-axis points in down direction. Since only the x - and y -axes components of the angular rate go into the heading angle, the z -axis gyroscope bias does not influence the heading angle, too. The influence of the gyroscope scale factor error and the accelerometer and gyroscope misalignments is small and can be neglected.

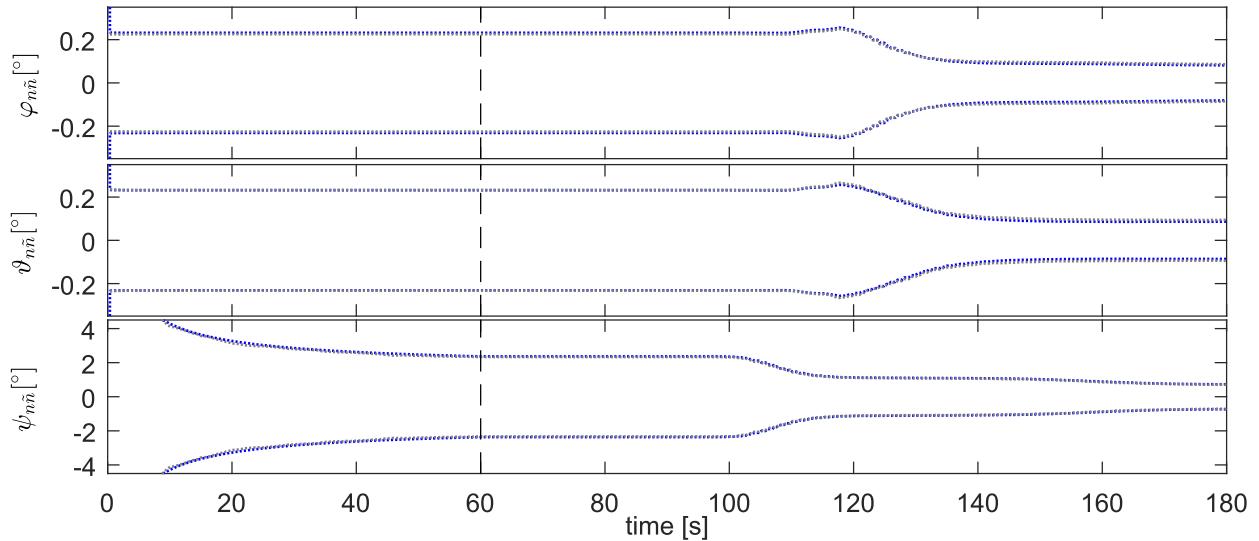


Figure 4-20: Orientation error 3σ boundaries (blue: predicted, grey: Monte Carlo simulation)

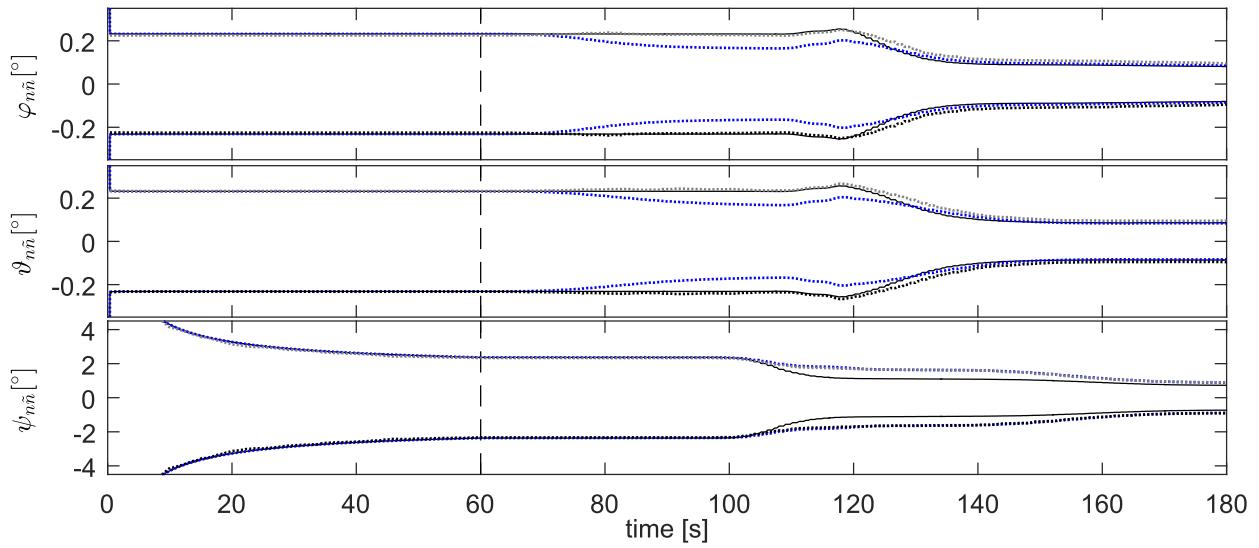


Figure 4-21: Orientation error 3σ boundaries without consideration of correlations (blue: predicted, grey: Monte Carlo simulation, black: reference)

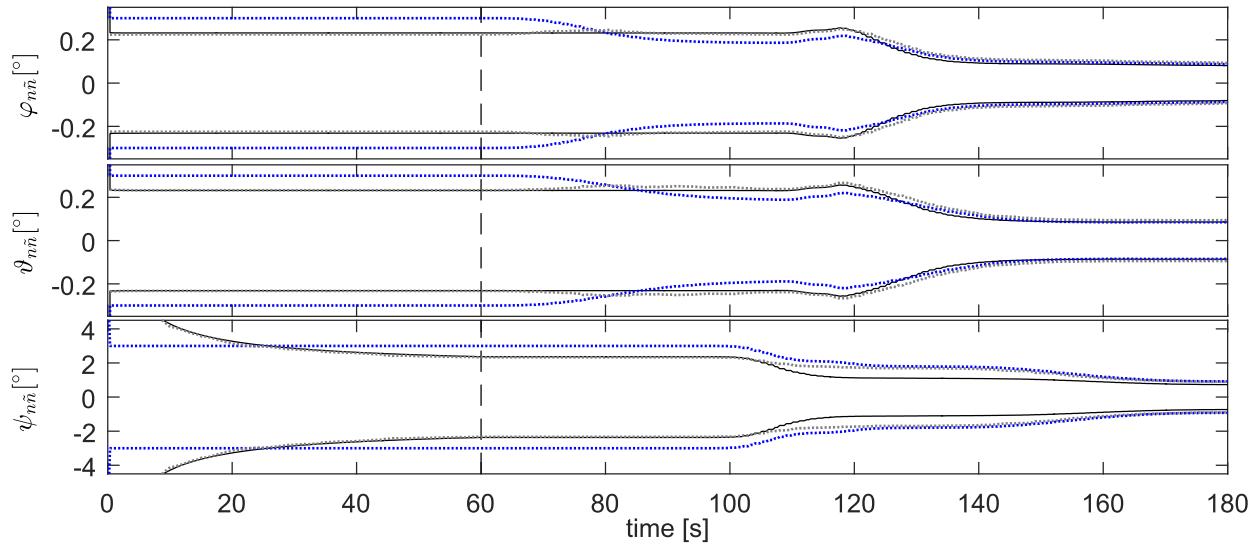
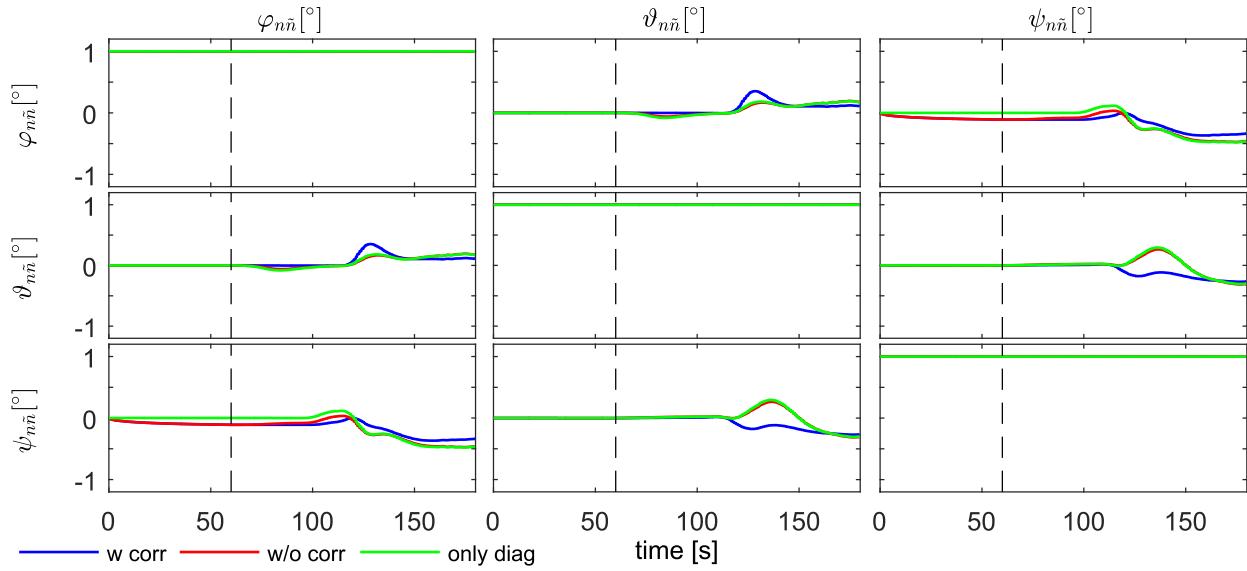
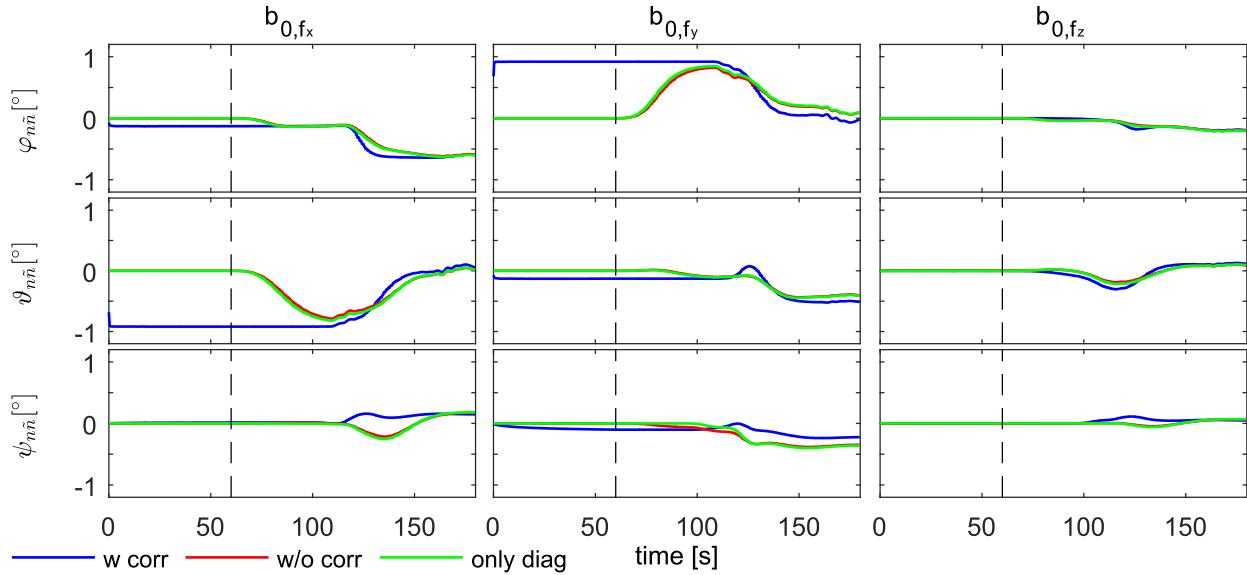
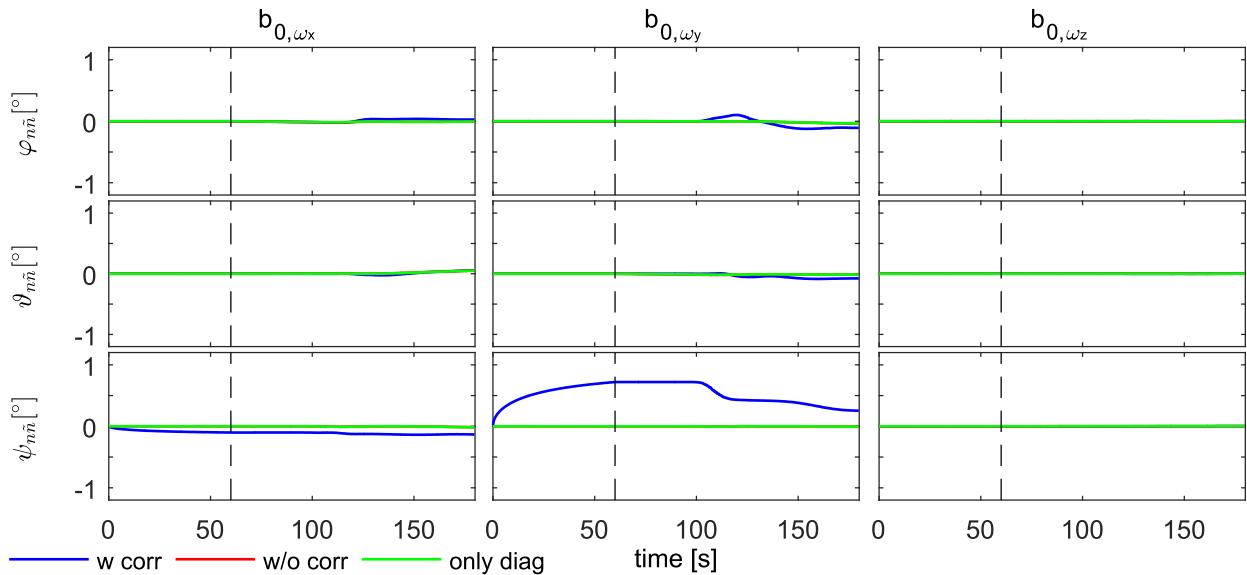


Figure 4-22: Orientation error 3σ boundaries with diagonal covariance initialization (blue: predicted, grey: Monte Carlo simulation, black: reference)

**Figure 4-23: Orientation error correlation****Figure 4-24: Orientation error and accelerometer turn-on bias correlation****Figure 4-25: Orientation error and gyroscope turn-on bias correlation**

4.2.3 In-Flight Navigation Filter Initialization

4.2.3.1 Orientation Angle Expected Values

The more challenging task is to initialize the orientation angles during flight, for example if the navigation system has been rebooted after a failure. Initial values for position and velocity and the corresponding covariances are easily obtained from the GNSS receiver. A stationary alignment for the initial orientation angles is out of question since the aircraft is moving in the initialization phase. Even if the aircraft flies straight and leveled, disturbances like aerodynamic turbulence may change the orientation of the platform during initialization and distort the estimated orientation angles and the corresponding covariance.

The Three-Axis Attitude Determination (TRIAD) method, which was originally presented by Black [104] for satellites and was, for example, applied by [105], will be used in the following for the in-flight orientation initialization. The roll angle φ_{nb} , pitch angle ϑ_{nb} and heading angle ψ_{nb} are estimated by means of two arbitrary vectors that are simultaneously observed in the n - and in the b -frame. It will be clarified later which vectors come into question for the in-flight orientation initialization. We start the derivation with two general vectors \mathbf{a} and \mathbf{b} given in the n - and b -frames

$$\begin{aligned} \mathbf{a}_n, \mathbf{a}_b \\ \mathbf{b}_n, \mathbf{b}_b \end{aligned} \quad (4.47)$$

The vectors do not have to be orthogonal but must not be co-linear. Then, the transformation matrix \mathbf{R}_{nb} is given with

$$\mathbf{R}_{nb} = \mathbf{A}\mathbf{B}^{-1} = \mathbf{B}^T \mathbf{A}^{-T} \quad (4.48)$$

where

$$\begin{aligned} \mathbf{A} &= (\mathbf{a}_n, \mathbf{b}_n, \mathbf{c}_n) & \mathbf{c}_n &= \mathbf{a}_n \times \mathbf{b}_n \\ \mathbf{B} &= (\mathbf{a}_b, \mathbf{b}_b, \mathbf{c}_b) & \mathbf{c}_b &= \mathbf{a}_b \times \mathbf{b}_b \end{aligned} \quad (4.49)$$

are the non-orthogonal spaces spanned by the vectors \mathbf{a}_n and \mathbf{b}_n and by the vectors \mathbf{a}_b and \mathbf{b}_b , respectively. The \mathbf{c} vectors are introduced as abbreviation for the cross products. The second equality in (4.48) is valid since transformation matrices belong to the SO(3) group, meaning that $\mathbf{R}_{nb}^T \mathbf{R}_{nb} = \mathbf{I}_3$. The Euler angles φ_{nb} , ϑ_{nb} and ψ_{nb} can be calculated from the entries of the transformation matrix \mathbf{R}_{nb} by

$$\begin{aligned} \varphi_{nb} &= \text{atan2}(R_{nb,32}, R_{nb,33}) \\ \vartheta_{nb} &= -\arcsin R_{nb,31} \\ \psi_{nb} &= \text{atan2}(R_{nb,21}, R_{nb,11}) \end{aligned} \quad (4.50)$$

The question how the entries of the transformation matrix $R_{nb,jj}$ relate to the vectors \mathbf{a} and \mathbf{b} is explained in the following. We consider the first variant in (4.48), now with erroneous values

$$\mathbf{R}_{\tilde{n}b} = \tilde{\mathbf{A}}\tilde{\mathbf{B}}^{-1} \quad (4.51)$$

The inverse of the $\tilde{\mathbf{B}}$ matrix is calculated and abridged with

$$\tilde{\mathbf{B}}^{-1} = \frac{1}{\det(\tilde{\mathbf{B}})} \begin{pmatrix} (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)^T \\ (\tilde{\mathbf{c}}_b \times \tilde{\mathbf{a}}_b)^T \\ (\tilde{\mathbf{a}}_b \times \tilde{\mathbf{b}}_b)^T \end{pmatrix} =: \begin{pmatrix} \tilde{\mathbf{u}}_b^T \\ \tilde{\mathbf{v}}_b^T \\ \tilde{\mathbf{w}}_b^T \end{pmatrix} \quad (4.52)$$

Inserting into the direction cosine matrix (4.51) yields

$$\mathbf{R}_{\tilde{n}b} = \tilde{\mathbf{a}}_n \tilde{\mathbf{u}}_b^\top + \tilde{\mathbf{b}}_n \tilde{\mathbf{v}}_b^\top + \tilde{\mathbf{c}}_n \tilde{\mathbf{w}}_b^\top \quad (4.53)$$

The explicit orientation angle equations (4.50) then become

$$\begin{aligned}\varphi_{\tilde{n}b} &= \text{atan2}\left(\tilde{a}_{n,z} \tilde{u}_{b,y} + \tilde{b}_{n,z} \tilde{v}_{b,y} + \tilde{c}_{n,z} \tilde{w}_{b,y}, \tilde{a}_{n,z} \tilde{u}_{b,z} + \tilde{b}_{n,z} \tilde{v}_{b,z} + \tilde{c}_{n,z} \tilde{w}_{b,z}\right) \\ \theta_{\tilde{n}b} &= \arcsin\left(-\tilde{a}_{n,z} \tilde{u}_{b,x} - \tilde{b}_{n,z} \tilde{v}_{b,x} - \tilde{c}_{n,z} \tilde{w}_{b,x}\right) \\ \psi_{\tilde{n}b} &= \text{atan2}\left(\tilde{a}_{n,y} \tilde{u}_{b,x} + \tilde{b}_{n,y} \tilde{v}_{b,x} + \tilde{c}_{n,y} \tilde{w}_{b,x}, \tilde{a}_{n,x} \tilde{u}_{b,x} + \tilde{b}_{n,x} \tilde{v}_{b,x} + \tilde{c}_{n,x} \tilde{w}_{b,x}\right)\end{aligned} \quad (4.54)$$

4.2.3.2 Orientation Error Equations

In this section the orientation error equations are derived.

The linearized roll angle error $\delta\varphi_{\tilde{n}b}$ depends on the errors of the six involved vectors

$$\delta\varphi_{\tilde{n}b} \doteq \frac{\partial\varphi_{nb}}{\partial\mathbf{a}_n^\top} \left|_{\sim} \right. \delta\mathbf{a}_n + \frac{\partial\varphi_{nb}}{\partial\mathbf{b}_n^\top} \left|_{\sim} \right. \delta\mathbf{b}_n + \frac{\partial\varphi_{nb}}{\partial\mathbf{c}_n^\top} \left|_{\sim} \right. \delta\mathbf{c}_n + \frac{\partial\varphi_{nb}}{\partial\mathbf{u}_b^\top} \left|_{\sim} \right. \delta\mathbf{u}_b + \frac{\partial\varphi_{nb}}{\partial\mathbf{v}_b^\top} \left|_{\sim} \right. \delta\mathbf{v}_b + \frac{\partial\varphi_{nb}}{\partial\mathbf{w}_b^\top} \left|_{\sim} \right. \delta\mathbf{w}_b \quad (4.55)$$

Using the derivative of the double-argument arctangent function (4.23), the Euler angle equations (4.50) and the explicit Euler angle equations (4.54), the single derivatives are

$$\begin{aligned}\frac{\partial\varphi_{nb}}{\partial\mathbf{a}_n^\top} \left|_{\sim} \right. &= \frac{1}{R_{\tilde{n}b,32}^2 + R_{\tilde{n}b,33}^2} \begin{pmatrix} 0 \\ 0 \\ \tilde{u}_{b,y} R_{\tilde{n}b,33} - \tilde{u}_{b,z} R_{\tilde{n}b,32} \end{pmatrix}^\top, \quad \frac{\partial\varphi_{nb}}{\partial\mathbf{u}_b^\top} \left|_{\sim} \right. = \frac{1}{R_{\tilde{n}b,32}^2 + R_{\tilde{n}b,33}^2} \begin{pmatrix} 0 \\ \tilde{a}_{n,z} R_{\tilde{n}b,33} \\ -\tilde{a}_{n,z} R_{\tilde{n}b,32} \end{pmatrix}^\top \\ \frac{\partial\varphi_{nb}}{\partial\mathbf{b}_n^\top} \left|_{\sim} \right. &= \frac{1}{R_{\tilde{n}b,32}^2 + R_{\tilde{n}b,33}^2} \begin{pmatrix} 0 \\ 0 \\ \tilde{v}_{b,y} R_{\tilde{n}b,33} - \tilde{v}_{b,z} R_{\tilde{n}b,32} \end{pmatrix}^\top, \quad \frac{\partial\varphi_{nb}}{\partial\mathbf{v}_b^\top} \left|_{\sim} \right. = \frac{1}{R_{\tilde{n}b,32}^2 + R_{\tilde{n}b,33}^2} \begin{pmatrix} 0 \\ \tilde{b}_{n,z} R_{\tilde{n}b,33} \\ -\tilde{b}_{n,z} R_{\tilde{n}b,32} \end{pmatrix}^\top \\ \frac{\partial\varphi_{nb}}{\partial\mathbf{c}_n^\top} \left|_{\sim} \right. &= \frac{1}{R_{\tilde{n}b,32}^2 + R_{\tilde{n}b,33}^2} \begin{pmatrix} 0 \\ 0 \\ \tilde{w}_{b,y} R_{\tilde{n}b,33} - \tilde{w}_{b,z} R_{\tilde{n}b,32} \end{pmatrix}^\top, \quad \frac{\partial\varphi_{nb}}{\partial\mathbf{w}_b^\top} \left|_{\sim} \right. = \frac{1}{R_{\tilde{n}b,32}^2 + R_{\tilde{n}b,33}^2} \begin{pmatrix} 0 \\ \tilde{c}_{n,z} R_{\tilde{n}b,33} \\ -\tilde{c}_{n,z} R_{\tilde{n}b,32} \end{pmatrix}^\top\end{aligned} \quad (4.56)$$

The linearized pitch angle error equation $\delta\theta_{\tilde{n}b}$ is analogous

$$\delta\theta_{\tilde{n}b} \doteq \frac{\partial\theta_{nb}}{\partial\mathbf{a}_n^\top} \left|_{\sim} \right. \delta\mathbf{a}_n + \frac{\partial\theta_{nb}}{\partial\mathbf{b}_n^\top} \left|_{\sim} \right. \delta\mathbf{b}_n + \frac{\partial\theta_{nb}}{\partial\mathbf{c}_n^\top} \left|_{\sim} \right. \delta\mathbf{c}_n + \frac{\partial\theta_{nb}}{\partial\mathbf{u}_b^\top} \left|_{\sim} \right. \delta\mathbf{u}_b + \frac{\partial\theta_{nb}}{\partial\mathbf{v}_b^\top} \left|_{\sim} \right. \delta\mathbf{v}_b + \frac{\partial\theta_{nb}}{\partial\mathbf{w}_b^\top} \left|_{\sim} \right. \delta\mathbf{w}_b \quad (4.57)$$

With the derivative of the arcsine function (4.26), the Euler angle equations (4.50) and the explicit Euler angle equations (4.54), the single derivatives are

$$\begin{aligned}\frac{\partial\theta_{nb}}{\partial\mathbf{a}_n^\top} \left|_{\sim} \right. &= \frac{-1}{\sqrt{1-R_{\tilde{n}b,31}^2}} \begin{pmatrix} 0 \\ 0 \\ \tilde{u}_{b,x} \end{pmatrix}^\top, \quad \frac{\partial\theta_{nb}}{\partial\mathbf{u}_b^\top} \left|_{\sim} \right. = \frac{-1}{\sqrt{1-R_{\tilde{n}b,31}^2}} \begin{pmatrix} \tilde{a}_{n,z} \\ 0 \\ 0 \end{pmatrix}^\top \\ \frac{\partial\theta_{nb}}{\partial\mathbf{b}_n^\top} \left|_{\sim} \right. &= \frac{-1}{\sqrt{1-R_{\tilde{n}b,31}^2}} \begin{pmatrix} 0 \\ 0 \\ \tilde{v}_{b,x} \end{pmatrix}^\top, \quad \frac{\partial\theta_{nb}}{\partial\mathbf{v}_b^\top} \left|_{\sim} \right. = \frac{-1}{\sqrt{1-R_{\tilde{n}b,31}^2}} \begin{pmatrix} \tilde{b}_{n,z} \\ 0 \\ 0 \end{pmatrix}^\top \\ \frac{\partial\theta_{nb}}{\partial\mathbf{c}_n^\top} \left|_{\sim} \right. &= \frac{-1}{\sqrt{1-R_{\tilde{n}b,31}^2}} \begin{pmatrix} 0 \\ 0 \\ \tilde{w}_{b,x} \end{pmatrix}^\top, \quad \frac{\partial\theta_{nb}}{\partial\mathbf{w}_b^\top} \left|_{\sim} \right. = \frac{-1}{\sqrt{1-R_{\tilde{n}b,31}^2}} \begin{pmatrix} \tilde{c}_{n,z} \\ 0 \\ 0 \end{pmatrix}^\top\end{aligned} \quad (4.58)$$

The linearized heading angle error equation $\delta\psi_{\tilde{n}b}$ is finally

$$\delta\psi_{\tilde{n}b} \doteq \frac{\partial\psi_{nb}}{\partial\mathbf{a}_n^T} \Big|_{\sim} \delta\mathbf{a}_n + \frac{\partial\psi_{nb}}{\partial\mathbf{b}_n^T} \Big|_{\sim} \delta\mathbf{b}_n + \frac{\partial\psi_{nb}}{\partial\mathbf{c}_n^T} \Big|_{\sim} \delta\mathbf{c}_n + \frac{\partial\psi_{nb}}{\partial\mathbf{u}_b^T} \Big|_{\sim} \delta\mathbf{u}_b + \frac{\partial\psi_{nb}}{\partial\mathbf{v}_b^T} \Big|_{\sim} \delta\mathbf{v}_b + \frac{\partial\psi_{nb}}{\partial\mathbf{w}_b^T} \Big|_{\sim} \delta\mathbf{w}_b \quad (4.59)$$

with the derivatives

$$\begin{aligned} \frac{\partial\psi_{nb}}{\partial\mathbf{a}_n^T} \Big|_{\sim} &= \frac{1}{R_{\tilde{n}b,21}^2 + R_{\tilde{n}b,11}^2} \begin{pmatrix} -\tilde{u}_{b,x} R_{\tilde{n}b,21} \\ \tilde{u}_{b,x} R_{\tilde{n}b,11} \\ 0 \end{pmatrix}^T, & \frac{\partial\psi_{nb}}{\partial\mathbf{u}_b^T} \Big|_{\sim} &= \frac{1}{R_{\tilde{n}b,21}^2 + R_{\tilde{n}b,11}^2} \begin{pmatrix} \tilde{a}_{n,y} R_{\tilde{n}b,11} - \tilde{a}_{n,x} R_{\tilde{n}b,21} \\ 0 \\ 0 \end{pmatrix}^T \\ \frac{\partial\psi_{nb}}{\partial\mathbf{b}_n^T} \Big|_{\sim} &= \frac{1}{R_{\tilde{n}b,21}^2 + R_{\tilde{n}b,11}^2} \begin{pmatrix} -\tilde{v}_{b,x} R_{\tilde{n}b,21} \\ \tilde{v}_{b,x} R_{\tilde{n}b,11} \\ 0 \end{pmatrix}^T, & \frac{\partial\psi_{nb}}{\partial\mathbf{v}_b^T} \Big|_{\sim} &= \frac{1}{R_{\tilde{n}b,21}^2 + R_{\tilde{n}b,11}^2} \begin{pmatrix} \tilde{b}_{n,y} R_{\tilde{n}b,11} - \tilde{b}_{n,x} R_{\tilde{n}b,21} \\ 0 \\ 0 \end{pmatrix}^T \\ \frac{\partial\psi_{nb}}{\partial\mathbf{c}_n^T} \Big|_{\sim} &= \frac{1}{R_{\tilde{n}b,21}^2 + R_{\tilde{n}b,11}^2} \begin{pmatrix} -\tilde{w}_{b,x} R_{\tilde{n}b,21} \\ \tilde{w}_{b,x} R_{\tilde{n}b,11} \\ 0 \end{pmatrix}^T, & \frac{\partial\psi_{nb}}{\partial\mathbf{w}_b^T} \Big|_{\sim} &= \frac{1}{R_{\tilde{n}b,21}^2 + R_{\tilde{n}b,11}^2} \begin{pmatrix} \tilde{c}_{n,y} R_{\tilde{n}b,11} - \tilde{c}_{n,x} R_{\tilde{n}b,21} \\ 0 \\ 0 \end{pmatrix}^T \end{aligned} \quad (4.60)$$

With (4.32), (4.55), (4.57) and (4.59), the orientation error vector $\Psi_{n\tilde{n}}$ is written as

$$\begin{aligned} \Psi_{n\tilde{n}} &= \begin{pmatrix} \varphi_{n\tilde{n}} \\ \vartheta_{n\tilde{n}} \\ \psi_{n\tilde{n}} \end{pmatrix} = \mathbf{R}_3(\psi_{\tilde{n}b}) \mathbf{R}_2(\vartheta_{\tilde{n}b}) \begin{pmatrix} \delta\varphi_{\tilde{n}b} \\ 0 \\ 0 \end{pmatrix} + \mathbf{R}_3(\psi_{\tilde{n}b}) \begin{pmatrix} 0 \\ \delta\vartheta_{\tilde{n}b} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \delta\psi_{\tilde{n}b} \end{pmatrix} \\ &= \mathbf{C}_{\psi a_n} \delta\mathbf{a}_n + \mathbf{C}_{\psi b_n} \delta\mathbf{b}_n + \mathbf{C}_{\psi c_n} \delta\mathbf{c}_n + \mathbf{C}_{\psi u_b} \delta\mathbf{u}_b + \mathbf{C}_{\psi v_b} \delta\mathbf{v}_b + \mathbf{C}_{\psi w_b} \delta\mathbf{w}_b \end{aligned} \quad (4.61)$$

where the coefficient matrices are defined as follows

$$\begin{aligned} \mathbf{C}_{\psi a_n} &= \left(\begin{array}{c} \cos\vartheta_{\tilde{n}b} \cos\psi_{\tilde{n}b} \\ \cos\vartheta_{\tilde{n}b} \sin\psi_{\tilde{n}b} \\ -\sin\vartheta_{\tilde{n}b} \end{array} \right) \frac{\partial\varphi_{nb}}{\partial\mathbf{a}_n^T} \Big|_{\sim} + \left(\begin{array}{c} -\sin\psi_{\tilde{n}b} \\ \cos\psi_{\tilde{n}b} \\ 0 \end{array} \right) \frac{\partial\vartheta_{nb}}{\partial\mathbf{a}_n^T} \Big|_{\sim} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial\psi_{nb}}{\partial\mathbf{a}_n^T} \Big|_{\sim} \\ \mathbf{C}_{\psi b_n} &= \left(\begin{array}{c} \cos\vartheta_{\tilde{n}b} \cos\psi_{\tilde{n}b} \\ \cos\vartheta_{\tilde{n}b} \sin\psi_{\tilde{n}b} \\ -\sin\vartheta_{\tilde{n}b} \end{array} \right) \frac{\partial\varphi_{nb}}{\partial\mathbf{b}_n^T} \Big|_{\sim} + \left(\begin{array}{c} -\sin\psi_{\tilde{n}b} \\ \cos\psi_{\tilde{n}b} \\ 0 \end{array} \right) \frac{\partial\vartheta_{nb}}{\partial\mathbf{b}_n^T} \Big|_{\sim} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial\psi_{nb}}{\partial\mathbf{b}_n^T} \Big|_{\sim} \\ \mathbf{C}_{\psi c_n} &= \left(\begin{array}{c} \cos\vartheta_{\tilde{n}b} \cos\psi_{\tilde{n}b} \\ \cos\vartheta_{\tilde{n}b} \sin\psi_{\tilde{n}b} \\ -\sin\vartheta_{\tilde{n}b} \end{array} \right) \frac{\partial\varphi_{nb}}{\partial\mathbf{c}_n^T} \Big|_{\sim} + \left(\begin{array}{c} -\sin\psi_{\tilde{n}b} \\ \cos\psi_{\tilde{n}b} \\ 0 \end{array} \right) \frac{\partial\vartheta_{nb}}{\partial\mathbf{c}_n^T} \Big|_{\sim} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial\psi_{nb}}{\partial\mathbf{c}_n^T} \Big|_{\sim} \\ \mathbf{C}_{\psi u_b} &= \left(\begin{array}{c} \cos\vartheta_{\tilde{n}b} \cos\psi_{\tilde{n}b} \\ \cos\vartheta_{\tilde{n}b} \sin\psi_{\tilde{n}b} \\ -\sin\vartheta_{\tilde{n}b} \end{array} \right) \frac{\partial\varphi_{nb}}{\partial\mathbf{u}_b^T} \Big|_{\sim} + \left(\begin{array}{c} -\sin\psi_{\tilde{n}b} \\ \cos\psi_{\tilde{n}b} \\ 0 \end{array} \right) \frac{\partial\vartheta_{nb}}{\partial\mathbf{u}_b^T} \Big|_{\sim} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial\psi_{nb}}{\partial\mathbf{u}_b^T} \Big|_{\sim} \\ \mathbf{C}_{\psi v_b} &= \left(\begin{array}{c} \cos\vartheta_{\tilde{n}b} \cos\psi_{\tilde{n}b} \\ \cos\vartheta_{\tilde{n}b} \sin\psi_{\tilde{n}b} \\ -\sin\vartheta_{\tilde{n}b} \end{array} \right) \frac{\partial\varphi_{nb}}{\partial\mathbf{v}_b^T} \Big|_{\sim} + \left(\begin{array}{c} -\sin\psi_{\tilde{n}b} \\ \cos\psi_{\tilde{n}b} \\ 0 \end{array} \right) \frac{\partial\vartheta_{nb}}{\partial\mathbf{v}_b^T} \Big|_{\sim} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial\psi_{nb}}{\partial\mathbf{v}_b^T} \Big|_{\sim} \\ \mathbf{C}_{\psi w_b} &= \left(\begin{array}{c} \cos\vartheta_{\tilde{n}b} \cos\psi_{\tilde{n}b} \\ \cos\vartheta_{\tilde{n}b} \sin\psi_{\tilde{n}b} \\ -\sin\vartheta_{\tilde{n}b} \end{array} \right) \frac{\partial\varphi_{nb}}{\partial\mathbf{w}_b^T} \Big|_{\sim} + \left(\begin{array}{c} -\sin\psi_{\tilde{n}b} \\ \cos\psi_{\tilde{n}b} \\ 0 \end{array} \right) \frac{\partial\vartheta_{nb}}{\partial\mathbf{w}_b^T} \Big|_{\sim} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\partial\psi_{nb}}{\partial\mathbf{w}_b^T} \Big|_{\sim} \end{aligned} \quad (4.62)$$

The equation for the orientation error vector $\Psi_{n\tilde{n}}$ (4.61) depends on the error vectors $\delta\mathbf{u}_b$, $\delta\mathbf{v}_b$ and $\delta\mathbf{w}_b$. They are derived next. For this purpose the equation for the inverse of the $\tilde{\mathbf{B}}$ matrix (4.52) with the determinant $\det(\tilde{\mathbf{B}}) = \tilde{\mathbf{a}}_b^T (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)$ is perturbed on the left and right hand sides

$$\tilde{\mathbf{B}}^{-1} + \delta\mathbf{B}^{-1} = \frac{1}{(\tilde{\mathbf{a}}_b + \delta\mathbf{a}_b)^\top [(\tilde{\mathbf{b}}_b + \delta\mathbf{b}_b) \times (\tilde{\mathbf{c}}_b + \delta\mathbf{c}_b)]} \begin{pmatrix} [(\tilde{\mathbf{b}}_b + \delta\mathbf{b}_b) \times (\tilde{\mathbf{c}}_b + \delta\mathbf{c}_b)]^\top \\ [(\tilde{\mathbf{c}}_b + \delta\mathbf{c}_b) \times (\tilde{\mathbf{a}}_b + \delta\mathbf{a}_b)]^\top \\ [(\tilde{\mathbf{a}}_b + \delta\mathbf{a}_b) \times (\tilde{\mathbf{b}}_b + \delta\mathbf{b}_b)]^\top \end{pmatrix} \quad (4.63)$$

Bringing the denominator to the left side, expanding the cross products, neglecting bilinear error terms and assuming that $\tilde{\mathbf{a}}_b^\top (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b) \tilde{\mathbf{B}}^{-1} = (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b, \tilde{\mathbf{c}}_b \times \tilde{\mathbf{a}}_b, \tilde{\mathbf{a}}_b \times \tilde{\mathbf{b}}_b)$ yields an equation that is linear in the error terms

$$\begin{aligned} & \delta\mathbf{a}_b^\top (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b) \tilde{\mathbf{B}}^{-1} + \tilde{\mathbf{a}}_b^\top (\delta\mathbf{b}_b \times \tilde{\mathbf{c}}_b + \tilde{\mathbf{b}}_b \times \delta\mathbf{c}_b) \tilde{\mathbf{B}}^{-1} + \tilde{\mathbf{a}}_b^\top (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b) \delta\mathbf{B}^{-1} \\ & \doteq \begin{pmatrix} (\delta\mathbf{b}_b \times \tilde{\mathbf{c}}_b + \tilde{\mathbf{b}}_b \times \delta\mathbf{c}_b)^\top \\ (\delta\mathbf{c}_b \times \tilde{\mathbf{a}}_b + \tilde{\mathbf{c}}_b \times \delta\mathbf{a}_b)^\top \\ (\delta\mathbf{a}_b \times \tilde{\mathbf{b}}_b + \tilde{\mathbf{a}}_b \times \delta\mathbf{b}_b)^\top \end{pmatrix} \end{aligned} \quad (4.64)$$

Next, the error matrix $\delta\mathbf{B}^{-1}$ is isolated

$$\delta\mathbf{B}^{-1} \doteq \frac{1}{\tilde{\mathbf{a}}_b^\top (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)} \begin{pmatrix} (\delta\mathbf{b}_b \times \tilde{\mathbf{c}}_b + \tilde{\mathbf{b}}_b \times \delta\mathbf{c}_b)^\top \\ (\delta\mathbf{c}_b \times \tilde{\mathbf{a}}_b + \tilde{\mathbf{c}}_b \times \delta\mathbf{a}_b)^\top \\ (\delta\mathbf{a}_b \times \tilde{\mathbf{b}}_b + \tilde{\mathbf{a}}_b \times \delta\mathbf{b}_b)^\top \end{pmatrix} - \frac{\delta\mathbf{a}_b^\top (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b) + \tilde{\mathbf{a}}_b^\top (\delta\mathbf{b}_b \times \tilde{\mathbf{c}}_b) + \tilde{\mathbf{a}}_b^\top (\tilde{\mathbf{b}}_b \times \delta\mathbf{c}_b)}{\tilde{\mathbf{a}}_b^\top (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)} \tilde{\mathbf{B}}^{-1} \quad (4.65)$$

The searched error vectors $\delta\mathbf{u}_b$, $\delta\mathbf{v}_b$ and $\delta\mathbf{w}_b$ are just the row vectors of the error matrix $\delta\mathbf{B}^{-1}$

$$\delta\mathbf{B}^{-1} = \begin{pmatrix} \delta\mathbf{u}_b^\top \\ \delta\mathbf{v}_b^\top \\ \delta\mathbf{w}_b^\top \end{pmatrix} \quad (4.66)$$

The determinant is again replaced, the cross products in the first term are substituted by matrix products and the inverse $\tilde{\mathbf{B}}^{-1}$ matrix is replaced by the $\tilde{\mathbf{u}}_b$, $\tilde{\mathbf{v}}_b$ and $\tilde{\mathbf{w}}_b$ vectors. Then, the linearized equations for $\delta\mathbf{u}_b$, $\delta\mathbf{v}_b$ and $\delta\mathbf{w}_b$ are

$$\begin{aligned} \delta\mathbf{u}_b & \doteq \mathbf{C}_{u_b a_b} \delta\mathbf{a}_b + \mathbf{C}_{u_b b_b} \delta\mathbf{b}_b + \mathbf{C}_{u_b c_b} \delta\mathbf{c}_b \\ \delta\mathbf{v}_b & \doteq \mathbf{C}_{v_b a_b} \delta\mathbf{a}_b + \mathbf{C}_{v_b b_b} \delta\mathbf{b}_b + \mathbf{C}_{v_b c_b} \delta\mathbf{c}_b \\ \delta\mathbf{w}_b & \doteq \mathbf{C}_{w_b a_b} \delta\mathbf{a}_b + \mathbf{C}_{w_b b_b} \delta\mathbf{b}_b + \mathbf{C}_{w_b c_b} \delta\mathbf{c}_b \end{aligned} \quad (4.67)$$

where the coefficient matrices are given with

$$\begin{aligned} \mathbf{C}_{u_b a_b} &= \frac{-\tilde{\mathbf{u}}_b (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)^\top}{\det(\tilde{\mathbf{B}})} & \mathbf{C}_{v_b a_b} &= \frac{\text{veck}^{-1}(\tilde{\mathbf{c}}_b) - \tilde{\mathbf{v}}_b (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)^\top}{\det(\tilde{\mathbf{B}})} \\ \mathbf{C}_{u_b b_b} &= \frac{-(\mathbf{I}_3 - \tilde{\mathbf{u}}_b \tilde{\mathbf{a}}_b^\top) \text{veck}^{-1}(\tilde{\mathbf{c}}_b)}{\det(\tilde{\mathbf{B}})} & \mathbf{C}_{v_b b_b} &= \frac{\tilde{\mathbf{v}}_b \tilde{\mathbf{a}}_b^\top \text{veck}^{-1}(\tilde{\mathbf{c}}_b)}{\det(\tilde{\mathbf{B}})} \\ \mathbf{C}_{u_b c_b} &= \frac{(\mathbf{I}_3 - \tilde{\mathbf{u}}_b \tilde{\mathbf{c}}_b^\top) \text{veck}^{-1}(\tilde{\mathbf{b}}_b)}{\det(\tilde{\mathbf{B}})} & \mathbf{C}_{v_b c_b} &= \frac{-\text{veck}^{-1}(\tilde{\mathbf{a}}_b) - \tilde{\mathbf{v}}_b \tilde{\mathbf{a}}_b^\top \text{veck}^{-1}(\tilde{\mathbf{b}}_b)}{\det(\tilde{\mathbf{B}})} \\ \mathbf{C}_{w_b a_b} &= \frac{-\text{veck}^{-1}(\tilde{\mathbf{b}}_b) - \tilde{\mathbf{w}}_b (\tilde{\mathbf{b}}_b \times \tilde{\mathbf{c}}_b)^\top}{\det(\tilde{\mathbf{B}})} \end{aligned} \quad (4.68)$$

$$\begin{aligned}\mathbf{C}_{w_b b_b} &= \frac{\text{veck}^{-1}(\tilde{\mathbf{a}}_b) + \tilde{\mathbf{w}}_b \tilde{\mathbf{a}}_b^\top \text{veck}^{-1}(\tilde{\mathbf{c}}_b)}{\det(\tilde{\mathbf{B}})} \\ \mathbf{C}_{w_b c_b} &= \frac{-\tilde{\mathbf{w}}_b \tilde{\mathbf{a}}_b^\top \text{veck}^{-1}(\tilde{\mathbf{b}}_b)}{\det(\tilde{\mathbf{B}})}\end{aligned}$$

The orientation error vector $\Psi_{n\bar{n}}$ (4.61) can thus be expressed with the $\delta\mathbf{a}_b$, $\delta\mathbf{b}_b$ and $\delta\mathbf{c}_b$ error vectors instead of the $\delta\mathbf{u}_b$, $\delta\mathbf{v}_b$ and $\delta\mathbf{w}_b$ error vectors

$$\Psi_{n\bar{n}} \doteq \mathbf{C}_{\psi a_n} \delta\mathbf{a}_n + \mathbf{C}_{\psi b_n} \delta\mathbf{b}_n + \mathbf{C}_{\psi c_n} \delta\mathbf{c}_n + \mathbf{C}_{\psi a_b} \delta\mathbf{a}_b + \mathbf{C}_{\psi b_b} \delta\mathbf{b}_b + \mathbf{C}_{\psi c_b} \delta\mathbf{c}_b \quad (4.69)$$

with the additional coefficient matrices

$$\begin{aligned}\mathbf{C}_{\psi a_b} &= \mathbf{C}_{\psi u_b} \mathbf{C}_{u_b a_b} + \mathbf{C}_{\psi v_b} \mathbf{C}_{v_b a_b} + \mathbf{C}_{\psi w_b} \mathbf{C}_{w_b a_b} \\ \mathbf{C}_{\psi b_b} &= \mathbf{C}_{\psi u_b} \mathbf{C}_{u_b b_b} + \mathbf{C}_{\psi v_b} \mathbf{C}_{v_b b_b} + \mathbf{C}_{\psi w_b} \mathbf{C}_{w_b b_b} \\ \mathbf{C}_{\psi c_b} &= \mathbf{C}_{\psi u_b} \mathbf{C}_{u_b c_b} + \mathbf{C}_{\psi v_b} \mathbf{C}_{v_b c_b} + \mathbf{C}_{\psi w_b} \mathbf{C}_{w_b c_b}\end{aligned} \quad (4.70)$$

The derivation is concluded by inserting the linearized error of the \mathbf{c} vectors into (4.69)

$$\begin{aligned}\delta\mathbf{c}_n &\doteq -\text{veck}^{-1}(\tilde{\mathbf{b}}_n) \delta\mathbf{a}_n + \text{veck}^{-1}(\tilde{\mathbf{a}}_n) \delta\mathbf{b}_n \\ \delta\mathbf{c}_b &\doteq -\text{veck}^{-1}(\tilde{\mathbf{b}}_b) \delta\mathbf{a}_b + \text{veck}^{-1}(\tilde{\mathbf{a}}_b) \delta\mathbf{b}_b\end{aligned} \quad (4.71)$$

yielding

$$\Psi_{n\bar{n}} = \mathbf{H}_{a_n} \delta\mathbf{a}_n + \mathbf{H}_{a_b} \delta\mathbf{a}_b + \mathbf{H}_{b_n} \delta\mathbf{b}_n + \mathbf{H}_{b_b} \delta\mathbf{b}_b \quad (4.72)$$

with the four coefficient matrices

$$\begin{aligned}\mathbf{H}_{a_n} &= \mathbf{C}_{\psi a_n} - \mathbf{C}_{\psi c_n} \text{veck}^{-1}(\tilde{\mathbf{b}}_n) & \mathbf{H}_{b_n} &= \mathbf{C}_{\psi b_n} + \mathbf{C}_{\psi c_n} \text{veck}^{-1}(\tilde{\mathbf{a}}_n) \\ \mathbf{H}_{a_b} &= \mathbf{C}_{\psi a_b} - \mathbf{C}_{\psi c_b} \text{veck}^{-1}(\tilde{\mathbf{b}}_b) & \mathbf{H}_{b_b} &= \mathbf{C}_{\psi b_b} + \mathbf{C}_{\psi c_b} \text{veck}^{-1}(\tilde{\mathbf{a}}_b)\end{aligned} \quad (4.73)$$

4.2.3.3 Orientation Error Covariance

For the initialization of the navigation error filter the covariance matrix of the initial orientation error (4.72) is required. It is given by

$$\begin{aligned}\mathbb{E}[\Psi_{n\bar{n}} \Psi_{n\bar{n}}^\top] &= \mathbf{H}_{a_n} \mathbb{E}[\delta\mathbf{a}_n \delta\mathbf{a}_n^\top] \mathbf{H}_{a_n}^\top + \mathbf{H}_{a_b} \mathbb{E}[\delta\mathbf{a}_b \delta\mathbf{a}_b^\top] \mathbf{H}_{a_b}^\top \\ &\quad + \mathbf{H}_{b_n} \mathbb{E}[\delta\mathbf{b}_n \delta\mathbf{b}_n^\top] \mathbf{H}_{b_n}^\top + \mathbf{H}_{b_b} \mathbb{E}[\delta\mathbf{b}_b \delta\mathbf{b}_b^\top] \mathbf{H}_{b_b}^\top \\ &\quad + \mathbf{H}_{a_b} \mathbb{E}[\delta\mathbf{a}_b \delta\mathbf{b}_b^\top] \mathbf{H}_{b_b}^\top + \mathbf{H}_{b_b} \mathbb{E}[\delta\mathbf{b}_b \delta\mathbf{a}_b^\top] \mathbf{H}_{a_b}^\top\end{aligned} \quad (4.74)$$

Therein, it is assumed that the vector pairs $\delta\mathbf{a}_n / \delta\mathbf{a}_b$, $\delta\mathbf{b}_n / \delta\mathbf{b}_b$ and $\delta\mathbf{a}_n / \delta\mathbf{b}_n$ are statistically independent but $\delta\mathbf{a}_b$ and $\delta\mathbf{b}_b$ may be correlated. Besides the orientation error covariance itself, the cross-covariances between the orientation error $\Psi_{n\bar{n}}$ and the observation vector errors $\delta\mathbf{a}_n$, $\delta\mathbf{b}_n$, $\delta\mathbf{a}_b$ and $\delta\mathbf{b}_b$ are needed if models of the observation vector errors are incorporated into the navigation error filter. The single cross-covariances are

$$\begin{aligned}\mathbb{E}[\Psi_{n\bar{n}} \delta\mathbf{a}_n^\top] &= \mathbf{H}_{a_n} \mathbb{E}[\delta\mathbf{a}_n \delta\mathbf{a}_n^\top] \\ \mathbb{E}[\Psi_{n\bar{n}} \delta\mathbf{b}_n^\top] &= \mathbf{H}_{b_n} \mathbb{E}[\delta\mathbf{b}_n \delta\mathbf{b}_n^\top] \\ \mathbb{E}[\Psi_{n\bar{n}} \delta\mathbf{a}_b^\top] &= \mathbf{H}_{a_b} \mathbb{E}[\delta\mathbf{a}_b \delta\mathbf{a}_b^\top] \\ \mathbb{E}[\Psi_{n\bar{n}} \delta\mathbf{b}_b^\top] &= \mathbf{H}_{b_b} \mathbb{E}[\delta\mathbf{b}_b \delta\mathbf{b}_b^\top]\end{aligned} \quad (4.75)$$

4.2.3.4 Integrated Acceleration Observation

Expected Value

In the previous sections the initial orientation angles and the corresponding covariances have been derived for two general vector observations \mathbf{a} and \mathbf{b} . Now, the method shall be concretized for the in-flight orientation initialization application. In this section the first of the two vector observations is presented: the integrated acceleration observation.

In the case of the stationary initialization the acceleration $\dot{\mathbf{v}}_n$ on the left hand side and the velocity \mathbf{v}_n on the right hand side of the velocity differential equation (C.35) could be set to zero due to the stationary assumption. This is not possible in the case of the in-flight initialization because the velocity \mathbf{v}_n cannot be kept constant such that the acceleration $\dot{\mathbf{v}}_n$ is strictly zero during the whole initialization period. The velocity differential equation (C.35) solved for the acceleration $\mathbf{R}_{nb} \mathbf{f}_b$ becomes

$$\mathbf{R}_{nb} \mathbf{f}_b = \dot{\mathbf{v}}_n - \mathbf{Y}_n + (2 \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}) \times \mathbf{v}_n \quad (4.76)$$

In contrast to the stationary initialization case none of the terms vanishes. Indeed, the left hand side as well as the right hand side changes with time during the initialization period. The change of the direction cosine matrix \mathbf{R}_{nb} with time due to orientation changes of the aircraft in the initialization phase is expressed on the one hand by the change of the n -frame with time, \mathbf{R}_{nn_0} , and on the other hand by the change of the b -frame with time, \mathbf{R}_{b_0b} , starting from the initial direction cosine matrix $\mathbf{R}_{n_0b_0}$,

$$\mathbf{R}_{nb}(t) = \mathbf{R}_{nn_0}(t) \mathbf{R}_{n_0b_0} \mathbf{R}_{b_0b}(t) \quad (4.77)$$

Inserting the split transformation matrix in (4.76) and multiplying with $\mathbf{R}_{nn_0}^T = \mathbf{R}_{n_0n}$ from left yields

$$\mathbf{R}_{n_0b_0} \mathbf{R}_{b_0b} \mathbf{f}_b = \mathbf{R}_{n_0n} [\dot{\mathbf{v}}_n - \mathbf{Y}_n + (2 \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}) \times \mathbf{v}_n] \quad (4.78)$$

Next, both sides of the equation are integrated with time t , beginning with the initial time t_0 ,

$$\mathbf{R}_{n_0b_0} \int_{t_0}^t \mathbf{R}_{b_0b} \mathbf{f}_b d\tau = \mathbf{R}_{n_0n} \left[\dot{\mathbf{v}}_n - \mathbf{Y}_n + (2 \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}) \times \mathbf{v}_n \right] dt \quad (4.79)$$

With the integration by parts rule [103]

$$\int_a^b f(x) g'(x) dx = [f(x) g(x)]_a^b - \int_a^b f'(x) g(x) dx \quad (4.80)$$

the first term of the integral of the right hand side becomes

$$\begin{aligned} \int_{t_0}^t \mathbf{R}_{n_0n} \dot{\mathbf{v}}_n d\tau &= [\mathbf{R}_{n_0n} \mathbf{v}_n]_{t_0}^t - \int_{t_0}^t \dot{\mathbf{R}}_{n_0n} \mathbf{v}_n d\tau \\ &= \mathbf{R}_{n_0n} \mathbf{v}_n(t_0) - \mathbf{v}_n(t_0) - \int_{t_0}^t \mathbf{R}_{n_0n} \boldsymbol{\Omega}_{n_0n} \mathbf{v}_n d\tau \end{aligned} \quad (4.81)$$

With (4.81) the integral on the right hand side of (4.79) becomes

$$\begin{aligned} \mathbf{R}_{n_0n} \mathbf{v}_n(t_0) - \mathbf{v}_n(t_0) - \int_{t_0}^t \mathbf{R}_{n_0n} \boldsymbol{\Omega}_{n_0n} \mathbf{v}_n d\tau + \int_{t_0}^t \mathbf{R}_{n_0n} [-\mathbf{Y}_n + (2 \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}) \times \mathbf{v}_n] d\tau \\ = \mathbf{R}_{n_0n} \mathbf{v}_n(t_0) + \int_{t_0}^t \mathbf{R}_{n_0n} [-\mathbf{Y}_n + (2 \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en} - \boldsymbol{\omega}_{n_0n}) \times \mathbf{v}_n] d\tau \end{aligned} \quad (4.82)$$

With $\boldsymbol{\omega}_{en} - \boldsymbol{\omega}_{n_0n} = \mathbf{R}_{n_0n}^\top \boldsymbol{\omega}_{en_0}$, the integral further simplifies to

$$\mathbf{R}_{n_0n} \mathbf{v}_n - \mathbf{v}_n(t_0) + \int_{t_0}^t \mathbf{R}_{n_0n} \left[-\dot{\mathbf{Y}}_n + (2\mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \mathbf{R}_{n_0n}^\top \boldsymbol{\omega}_{en_0}) \times \mathbf{v}_n \right] d\tau \quad (4.83)$$

The differential equations of the three direction cosine matrices $\mathbf{R}_{nb}(t)$, $\mathbf{R}_{nn_0}(t)$ and $\mathbf{R}_{b_0b}(t)$ are given with

$$\begin{aligned} \dot{\mathbf{R}}_{nb}(t) &= \mathbf{R}_{nb}(t) \boldsymbol{\Omega}_{ib}(t) - \boldsymbol{\Omega}_{in}(t) \mathbf{R}_{nb}(t), & \mathbf{R}_{nb}(t_0) &= \mathbf{R}_{n_0b_0} \\ \dot{\mathbf{R}}_{n_0n}(t) &= \mathbf{R}_{n_0n}(t) \boldsymbol{\Omega}_{in}(t) - \boldsymbol{\Omega}_{in_0} \mathbf{R}_{n_0n}(t), & \mathbf{R}_{n_0n}(t_0) &= \mathbf{I}_3 \\ \dot{\mathbf{R}}_{b_0b}(t) &= \mathbf{R}_{b_0b}(t) \boldsymbol{\Omega}_{ib}(t) - \boldsymbol{\Omega}_{ib_0} \mathbf{R}_{b_0b}(t), & \mathbf{R}_{n_0n}(t_0) &= \mathbf{I}_3 \end{aligned} \quad (4.84)$$

where

$$\begin{aligned} \boldsymbol{\Omega}_{ib}(t) &= \text{veck}^{-1}(\boldsymbol{\omega}_{ib}(t)) \\ \boldsymbol{\Omega}_{in}(t) &= \text{veck}^{-1}(\boldsymbol{\omega}_{in}(t)) \\ \boldsymbol{\Omega}_{ib_0} &= \text{veck}^{-1}(\boldsymbol{\omega}_{ib}(t_0)) \\ \boldsymbol{\Omega}_{in_0} &= \text{veck}^{-1}(\boldsymbol{\omega}_{ib}(t_0)) \end{aligned} \quad (4.85)$$

It is assumed that the Earth rate $\boldsymbol{\omega}_{ie}$ and the transport rate $\boldsymbol{\omega}_{en_0}$ only have minor influence due to the short initialization time and can therefore be neglected, that is $\boldsymbol{\omega}_{ie} = \mathbf{0}$, $\boldsymbol{\omega}_{en_0} = \mathbf{0}$. Then, \mathbf{R}_{n_0n} does not change with time and is simply $\mathbf{R}_{n_0n} = \mathbf{I}_3$. Furthermore, the gravity can be assumed to be constant, that is $\dot{\mathbf{Y}}_n = \mathbf{0}$. Thus, the integrated acceleration observation equation is finally

$$\mathbf{R}_{n_0b_0} \mathbf{a}_{b_0} = \mathbf{a}_{n_0} \quad (4.86)$$

where

$$\begin{aligned} \mathbf{a}_{b_0} &= \int_{t_0}^t \mathbf{R}_{b_0b} \mathbf{f}_b d\tau \\ \mathbf{R}_{b_0b} &= \mathbf{R}(\bar{\mathbf{q}}_{b_0b}) \\ \dot{\bar{\mathbf{q}}}_{b_0b} &= \frac{1}{2} (\bar{\mathbf{q}}_{b_0b} \bar{\boldsymbol{\omega}}_{ib} - \bar{\boldsymbol{\omega}}_{ib}(t_0) \bar{\mathbf{q}}_{b_0b}), \quad \bar{\mathbf{q}}_{b_0b}(t_0) = (\mathbf{1}, \mathbf{0}) \\ \mathbf{a}_{n_0} &= \mathbf{v}_n - \mathbf{v}_{n,0} - \mathbf{Y}_n \cdot (t - t_0) \end{aligned} \quad (4.87)$$

The matrix differential equation for \mathbf{R}_{b_0b} has been replaced by the corresponding quaternion differential equation. The quaternion differential equation as well as the acceleration integral are computed by means of a numerical integration scheme, for example 4th order Runge-Kutta.

Error Equations

In order to compute the orientation error vector $\Psi_{n\tilde{n}}$ according to (4.72), the error vectors $\delta\mathbf{a}_{n_0}$ and $\delta\mathbf{a}_{b_0}$ of the vectors \mathbf{a}_{n_0} and \mathbf{a}_{b_0} in (4.86) are required. First, the $\delta\mathbf{a}_{n_0}$ vector is derived.

The \mathbf{a}_{n_0} vector in (4.87) is perturbed

$$\tilde{\mathbf{a}}_{n_0} + \delta\mathbf{a}_{n_0} = \tilde{\mathbf{v}}_n + \delta\mathbf{v}_n - \tilde{\mathbf{v}}_{n,0} - \delta\mathbf{v}_{n,0} - \mathbf{Y}_n \cdot (t - t_0) \quad (4.88)$$

Assuming that $\tilde{\mathbf{a}}_{n_0} = \tilde{\mathbf{v}}_n - \tilde{\mathbf{v}}_{n,0} - \mathbf{Y}_n \cdot (t - t_0)$ results in the error vector

$$\delta\mathbf{a}_{n_0} = \delta\mathbf{v}_n - \delta\mathbf{v}_n(t_0) \quad (4.89)$$

The time derivative of the \mathbf{a}_{b_0} vector in (4.87)

$$\dot{\mathbf{a}}_{b_0} = \mathbf{R}_{b_0b} \mathbf{f}_b \quad (4.90)$$

is perturbed on the left and right hand sides

$$\dot{\tilde{\mathbf{a}}}_{b_0} + \delta\dot{\mathbf{a}}_{b_0} = \mathbf{R}_{b_0\tilde{b}_0} \mathbf{R}_{\tilde{b}_0b} (\tilde{\mathbf{f}}_b + \delta\mathbf{f}_b) \quad (4.91)$$

The small angle rotation matrix $\mathbf{R}_{b_0\tilde{b}_0}$ between the true initial b_0 -frame and the assumed initial \tilde{b}_0 -frame is replaced by $\mathbf{I}_3 + \Psi_{b_0\tilde{b}_0} + h.o.t.$

$$\dot{\tilde{\mathbf{a}}}_{b_0} + \delta\dot{\mathbf{a}}_{b_0} = (\mathbf{I}_3 + \Psi_{b_0\tilde{b}_0} + h.o.t.) \mathbf{R}_{\tilde{b}_0b} (\tilde{\mathbf{f}}_b + \delta\mathbf{f}_b) \quad (4.92)$$

Neglecting bilinear and higher order error terms and assuming that $\dot{\tilde{\mathbf{a}}}_{b_0} = \mathbf{R}_{\tilde{b}_0b} \tilde{\mathbf{f}}_b$ yields

$$\delta\dot{\mathbf{a}}_{b_0} \doteq -\text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0b} \tilde{\mathbf{f}}_b) \Psi_{b_0\tilde{b}_0} + \mathbf{R}_{\tilde{b}_0b} \delta\mathbf{f}_b \quad (4.93)$$

Perturbing the direction cosine matrix differential equation for \mathbf{R}_{b_0b} in (4.84) with $\mathbf{R}_{b_0b} = \mathbf{R}_{b_0\tilde{b}_0} \mathbf{R}_{\tilde{b}_0b}$ makes

$$\dot{\mathbf{R}}_{b_0\tilde{b}_0} \mathbf{R}_{\tilde{b}_0b} + \mathbf{R}_{b_0\tilde{b}_0} \dot{\mathbf{R}}_{\tilde{b}_0b} = \mathbf{R}_{b_0\tilde{b}_0} \mathbf{R}_{\tilde{b}_0b} (\tilde{\boldsymbol{\Omega}}_{ib} + \delta\boldsymbol{\Omega}_{ib}) - (\tilde{\boldsymbol{\Omega}}_{ib,0} + \delta\boldsymbol{\Omega}_{ib,0}) \mathbf{R}_{b_0\tilde{b}_0} \mathbf{R}_{\tilde{b}_0b} \quad (4.94)$$

Assuming that $\dot{\mathbf{R}}_{\tilde{b}_0b} = \mathbf{R}_{\tilde{b}_0b} \tilde{\boldsymbol{\Omega}}_{ib} - \tilde{\boldsymbol{\Omega}}_{ib,0} \mathbf{R}_{\tilde{b}_0b}$ leads to the matrix differential equation

$$\dot{\mathbf{R}}_{b_0\tilde{b}_0} = \mathbf{R}_{b_0\tilde{b}_0} \underbrace{(\mathbf{R}_{\tilde{b}_0b} \delta\boldsymbol{\Omega}_{ib} \mathbf{R}_{\tilde{b}_0b}^T + \tilde{\boldsymbol{\Omega}}_{ib,0})}_{\boldsymbol{\Omega}_1} - \underbrace{(\tilde{\boldsymbol{\Omega}}_{ib,0} + \delta\boldsymbol{\Omega}_{ib,0})}_{\boldsymbol{\Omega}_2} \mathbf{R}_{b_0\tilde{b}_0} \quad (4.95)$$

and collapsed to the three orientation errors

$$\dot{\Psi}_{b_0\tilde{b}_0} = \begin{pmatrix} \dot{\phi}_{b_0\tilde{b}_0} \\ \dot{\theta}_{b_0\tilde{b}_0} \\ \dot{\psi}_{b_0\tilde{b}_0} \end{pmatrix} = \begin{pmatrix} 0 & \omega_{1_z} & -\omega_{2_y} \\ -\omega_{1_z} & 0 & \omega_{2_x} \\ \omega_{1_y} & -\omega_{2_x} & 0 \end{pmatrix} \begin{pmatrix} \phi_{b_0\tilde{b}_0} \\ \theta_{b_0\tilde{b}_0} \\ \psi_{b_0\tilde{b}_0} \end{pmatrix} + \begin{pmatrix} \omega_{1_x} - \omega_{2_x} \\ \omega_{1_y} - \omega_{2_y} \\ \omega_{1_z} - \omega_{2_z} \end{pmatrix}, \quad \Psi_{b_0\tilde{b}_0}(t_0) = 0 \quad (4.96)$$

If the bilinear error terms are neglected, the differential equation simplifies to

$$\dot{\Psi}_{b_0\tilde{b}_0} \doteq -\tilde{\boldsymbol{\Omega}}_{ib,0} \Psi_{b_0\tilde{b}_0} + \mathbf{R}_{\tilde{b}_0b} \delta\boldsymbol{\omega}_{ib} - \delta\boldsymbol{\omega}_{ib,0} \quad (4.97)$$

(4.93) and (4.97) are combined in a state-space model

$$\begin{aligned} \begin{pmatrix} \delta \dot{\mathbf{a}}_{b_0} \\ \Psi_{b_0 \tilde{b}_0} \end{pmatrix} &= \begin{pmatrix} 0 & -\text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0 b} \tilde{\mathbf{f}}_b) \\ 0 & -\tilde{\boldsymbol{\Omega}}_{ib,0} \end{pmatrix} \begin{pmatrix} \delta \mathbf{a}_{b_0} \\ \Psi_{b_0 \tilde{b}_0} \end{pmatrix} + \begin{pmatrix} \mathbf{R}_{\tilde{b}_0 b} & 0 \\ 0 & \mathbf{R}_{\tilde{b}_0 b} \end{pmatrix} \begin{pmatrix} \delta \mathbf{f}_b \\ \delta \boldsymbol{\omega}_{ib} \end{pmatrix} - \begin{pmatrix} 0 \\ \delta \boldsymbol{\omega}_{ib,0} \end{pmatrix} \\ \delta \mathbf{a}_{b_0} &= (\mathbf{I}_3 \quad 0) \begin{pmatrix} \delta \mathbf{a}_{b_0} \\ \Psi_{b_0 \tilde{b}_0} \end{pmatrix} \quad \begin{pmatrix} \delta \mathbf{a}_{b_0,0} \\ \Psi_{b_0 \tilde{b}_0,0} \end{pmatrix} = \mathbf{0}_{6 \times 1} \end{aligned} \quad (4.98)$$

The discrete-time version of the state-space model with sample time Δt is given by

$$\begin{aligned} \begin{pmatrix} \delta \mathbf{a}_{b_0,k} \\ \Psi_{b_0 \tilde{b}_0,k} \end{pmatrix} &= \begin{pmatrix} \mathbf{I}_3 & -\Delta t \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0 b, k-1} \tilde{\mathbf{f}}_{b,k-1}) \\ 0 & \mathbf{I}_3 + \Delta t \tilde{\boldsymbol{\Omega}}_{ib,0} \end{pmatrix} \begin{pmatrix} \delta \mathbf{a}_{b_0,k-1} \\ \Psi_{b_0 \tilde{b}_0,k-1} \end{pmatrix} \\ &\quad + \begin{pmatrix} \Delta t \mathbf{R}_{\tilde{b}_0 b, k-1} & 0 \\ 0 & \Delta t \mathbf{R}_{\tilde{b}_0 b, k-1} \end{pmatrix} \begin{pmatrix} \delta \mathbf{f}_{b,k-1} \\ \delta \boldsymbol{\omega}_{ib,k-1} \end{pmatrix} - \begin{pmatrix} 0 \\ \Delta t \delta \boldsymbol{\omega}_{ib,0} \end{pmatrix} \\ \delta \mathbf{a}_{b_0,k} &= (\mathbf{I}_3 \quad 0) \begin{pmatrix} \delta \mathbf{a}_{b_0,k} \\ \Psi_{b_0 \tilde{b}_0,k} \end{pmatrix} \quad \begin{pmatrix} \delta \mathbf{a}_{b_0,0} \\ \Psi_{b_0 \tilde{b}_0,0} \end{pmatrix} = \mathbf{0}_{6 \times 1} \end{aligned} \quad (4.99)$$

Augmentation by IMU Measurement Error Models

Neglecting the temperature dependent bias $\mathbf{b}_{T,f}$ and scale factor error $\mathbf{s}_{T,f}$ as well as the quantization noise $\mathbf{b}_{Q,f}$ and considering only the turn-on bias $\mathbf{b}_{0,f}$, scale factor error $\mathbf{s}_{0,f}$, white noise $\mathbf{b}_{N,f}(t)$, bias instability $\mathbf{b}_{B,f}(t)$ and the misalignments \mathbf{m}_f , the accelerometer error model (3.46) can be written as

$$\begin{aligned} \begin{pmatrix} \mathbf{b}_{0,f,k} \\ \mathbf{s}_{0,f,k} \\ \mathbf{m}_{f,k} \\ \mathbf{z}_{B_{f,x},k} \\ \mathbf{z}_{B_{f,y},k} \\ \mathbf{z}_{B_{f,z},k} \end{pmatrix} &= \underbrace{\begin{pmatrix} \mathbf{I}_3 & 0 & & & & \\ 0 & \mathbf{I}_3 & 0 & & & \\ 0 & 0 & \mathbf{I}_6 & 0 & & \\ & & & \Phi_B & 0 & \\ & & & 0 & \Phi_B & 0 \\ & & & & 0 & \Phi_B \end{pmatrix}}_{\Phi_f} \underbrace{\begin{pmatrix} \mathbf{b}_{0,f,k-1} \\ \mathbf{s}_{0,f,k-1} \\ \mathbf{m}_{f,k-1} \\ \mathbf{z}_{B_{f,x},k-1} \\ \mathbf{z}_{B_{f,y},k-1} \\ \mathbf{z}_{B_{f,z},k-1} \end{pmatrix}}_{\mathbf{z}_{f,k-1}} + \underbrace{\begin{pmatrix} 0 & & & & & \\ & \mathbf{I}_3 & 0 & & & \\ & 0 & \mathbf{I}_3 & 0 & & \\ & & & \Gamma_B & 0 & \\ & & & 0 & \Gamma_B & 0 \\ & & & & 0 & \Gamma_B \end{pmatrix}}_{\Gamma_f} \underbrace{\begin{pmatrix} \boldsymbol{\eta}_{N_f,k-1} \\ \boldsymbol{\eta}_{B_{f,x},k-1} \\ \boldsymbol{\eta}_{B_{f,y},k-1} \\ \boldsymbol{\eta}_{B_{f,z},k-1} \\ \boldsymbol{\eta}_{f,k-1} \end{pmatrix}}_{\boldsymbol{\eta}_{f,k-1}}, \quad \mathbf{z}_{f,0} = \begin{pmatrix} \mathbf{b}_{0,f} \\ \mathbf{s}_{0,f} \\ \mathbf{m}_f \\ \mathbf{0}_{p \times 1} \\ \mathbf{0}_{p \times 1} \\ \mathbf{0}_{p \times 1} \end{pmatrix} \\ \delta \mathbf{f}_{b,k} &= \underbrace{\begin{pmatrix} \mathbf{I}_3 & \text{diag}(\tilde{\mathbf{f}}_{b,k}) & \mathbf{F}_{M_f,k} & \mathbf{H}_B & 0 \\ & & & 0 & \mathbf{H}_B \\ & & & & \mathbf{H}_B \end{pmatrix}}_{\mathbf{H}_f} \mathbf{z}_{f,k} + \underbrace{\begin{pmatrix} \mathbf{I}_3 & D_B & 0 \\ 0 & \mathbf{I}_3 & D_B \\ & 0 & D_B \end{pmatrix}}_{\mathbf{D}_f} \boldsymbol{\eta}_{f,k} \end{aligned} \quad (4.100)$$

$\mathbf{b}_{0,f}$, $\mathbf{s}_{0,f}$ and \mathbf{m}_f are modeled as random constants. The bias instabilities on the x -, y - and z -axis are represented by the models (3.27). An analogous model can be set up for the gyroscope measurement error

$$\begin{aligned} \mathbf{z}_{\omega,k} &= \boldsymbol{\Phi}_{\omega} \mathbf{z}_{\omega,k-1} + \boldsymbol{\Gamma}_{\omega} \boldsymbol{\eta}_{\omega,k-1} \\ \delta \boldsymbol{\omega}_{ib,k} &= \mathbf{H}_{\omega} \mathbf{z}_{\omega,k} + \mathbf{D}_{\omega} \boldsymbol{\eta}_{\omega,k} \end{aligned} \quad (4.101)$$

The state-space model for the $\delta \mathbf{a}_{b_0,k}$ vector (4.99) is augmented by the accelerometer error model (4.100) and the gyroscope error model (4.101)

$$\begin{aligned}
\begin{pmatrix} \delta \mathbf{a}_{b_0,k} \\ \Psi_{b_0\tilde{b}_0,k} \\ \mathbf{z}_{f,k} \\ \mathbf{z}_{\omega,k} \\ \mathbf{z}_{a_{b_0},k} \end{pmatrix} &= \underbrace{\begin{pmatrix} \mathbf{I}_3 & -\Delta t \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0b,k-1} \tilde{\mathbf{f}}_{b,k-1}) & \Delta t \mathbf{R}_{\tilde{b}_0b,k-1} \mathbf{H}_f & 0 \\ 0 & \mathbf{I}_3 + \Delta t \boldsymbol{\Omega}_{ib,0} & 0 & \Delta t \mathbf{R}_{\tilde{b}_0b,k-1} \mathbf{H}_\omega \\ 0 & 0 & \Phi_f & 0 \\ 0 & 0 & 0 & \Phi_\omega \end{pmatrix}}_{\Phi_{a_{b_0},k-1}} \begin{pmatrix} \delta \mathbf{a}_{b_0,k-1} \\ \Psi_{b_0\tilde{b}_0,k-1} \\ \mathbf{z}_{f,k-1} \\ \mathbf{z}_{\omega,k-1} \\ \mathbf{z}_{a_{b_0},k-1} \end{pmatrix} \\
&+ \underbrace{\begin{pmatrix} \Delta t \mathbf{R}_{\tilde{b}_0b,k-1} \mathbf{D}_f & 0 \\ 0 & \Delta t \mathbf{R}_{\tilde{b}_0b,k-1} \mathbf{D}_\omega \\ \Gamma_f & 0 \\ 0 & \Gamma_\omega \end{pmatrix}}_{\Gamma_{a_{b_0},k-1}} \begin{pmatrix} \mathbf{n}_{f,k-1} \\ \mathbf{n}_{\omega,k-1} \end{pmatrix} - \begin{pmatrix} 0 \\ \Delta t (\mathbf{H}_\omega \mathbf{z}_{\omega,0} + \mathbf{D}_\omega \mathbf{n}_{\omega,0}) \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \delta \mathbf{a}_{b_0,0} \\ \Psi_{b_0\tilde{b}_0,0} \\ \mathbf{z}_{f,0} \\ \mathbf{z}_{\omega,0} \end{pmatrix} = \begin{pmatrix} \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 1} \\ \mathbf{z}_{f,0} \\ \mathbf{z}_{\omega,0} \end{pmatrix} \\
\delta \mathbf{a}_{b_0,k} &= \underbrace{(\mathbf{I}_3 \quad 0 \quad 0 \quad 0)}_{\mathbf{H}_{a_{b_0}}} \mathbf{z}_{a_{b_0},k}
\end{aligned} \tag{4.102}$$

Covariance

For the orientation error covariance (4.74) the covariances of the $\delta \mathbf{a}_{n_0}$ and $\delta \mathbf{a}_{b_0}$ vectors are required

$$\begin{aligned}
\mathbb{E}[\delta \mathbf{a}_{n_0,k} \delta \mathbf{a}_{n_0,k}^\top] &=: \mathbf{P}_{a_{n_0},k} = \mathbf{R}_{v,k} + \mathbf{R}_{v,0} \\
\mathbb{E}[\mathbf{z}_{a_{b_0},k} \mathbf{z}_{a_{b_0},k}^\top] &=: \mathbf{P}_{z_{a_{b_0},k}} = \Phi_{a_{b_0},k-1} \mathbf{P}_{z_{a_{b_0},k-1}} \Phi_{a_{b_0},k-1}^\top + \Gamma_{a_{b_0},k-1} \mathbf{Q}_{k-1} \Gamma_{a_{b_0},k-1}^\top \\
\mathbb{E}[\delta \mathbf{a}_{b_0,k} \delta \mathbf{a}_{b_0,k}^\top] &=: \mathbf{P}_{a_{b_0},k} = \mathbf{H}_{a_{b_0}} \mathbf{P}_{z_{a_{b_0},k}} \mathbf{H}_{a_{b_0}}^\top
\end{aligned} \tag{4.103}$$

$\mathbf{R}_{v,k}$ is the covariance matrix of the velocity measurement. It is assumed that the error of the velocity measurement is not correlated with time but purely white. The initial covariance is

$$\mathbf{P}_{z_{a_{b_0}},0} = \begin{pmatrix} \mathbf{Q}_{b_0,f} & 0 & 0 & 0 \\ 0 & \mathbf{Q}_{S_0,f} & 0 & 0 \\ 0 & 0 & \mathbf{Q}_{M_f} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{P}_{z_f,0} = \begin{pmatrix} \mathbf{Q}_{b_0,\omega} & 0 & 0 & 0 \\ 0 & \mathbf{Q}_{S_0,\omega} & 0 & 0 \\ 0 & 0 & \mathbf{Q}_{M_\omega} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.104}$$

4.2.3.5 Magnetic Field Observation

Besides the integrated acceleration observation (4.86) a second vector pair is required to apply the TRIAD method. If a magnetometer is available, the magnetic field vector \mathbf{m}_b , measured in the body-fixed frame, could be compared to the expected vector \mathbf{m}_n provided, for example, by the World Magnetic Model (WMM), given in the local north-east-down frame. The vector observation at the initial time t_0 is given with

$$\mathbf{R}_{n_0b_0} \mathbf{m}_{b_0} = \mathbf{m}_{n_0} \tag{4.105}$$

If the magnetic field measurement at the current time t and not at the initial time t_0 is used, the vectors have additionally to be transformed to the initial n_0 - and b_0 -frames, respectively,

$$\mathbf{R}_{n_0b_0} \underbrace{\mathbf{R}_{b_0b} \mathbf{m}_b}_{\mathbf{b}_{b_0}} = \underbrace{\mathbf{R}_{n_0n} \mathbf{m}_n}_{\mathbf{b}_{n_0}} \tag{4.106}$$

\mathbf{R}_{n_0n} is again assumed to be the identity matrix, $\mathbf{R}_{n_0n} = \mathbf{I}_3$. For the right hand side simply holds

$$\tilde{\mathbf{b}}_{n_0} = \tilde{\mathbf{m}}_n, \quad \delta\mathbf{b}_{n_0} = \delta\mathbf{m}_n \quad (4.107)$$

The perturbation and linearization of the \mathbf{b}_{b_0} vector yields

$$\tilde{\mathbf{b}}_{b_0} + \delta\mathbf{b}_{b_0} = (\mathbf{I}_3 + \boldsymbol{\Psi}_{b_0\tilde{b}_0} + h.o.t.) \mathbf{R}_{\tilde{b}_0b} (\tilde{\mathbf{m}}_b + \delta\mathbf{m}_b) \quad (4.108)$$

which gives with $\tilde{\mathbf{b}}_{b_0} = \mathbf{R}_{\tilde{b}_0b} \tilde{\mathbf{m}}_b$ the linear error vector

$$\delta\mathbf{b}_{b_0} \doteq -\text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0b} \tilde{\mathbf{m}}_b) \boldsymbol{\Psi}_{b_0\tilde{b}_0} + \mathbf{R}_{\tilde{b}_0b} \delta\mathbf{m}_b \quad (4.109)$$

The covariances are accordingly

$$\mathbb{E}[\delta\mathbf{b}_{n_0} \delta\mathbf{b}_{n_0}^T] = \mathbb{E}[\delta\mathbf{m}_n \delta\mathbf{m}_n^T] \quad (4.110)$$

and

$$\begin{aligned} \mathbb{E}[\delta\mathbf{b}_{b_0} \delta\mathbf{b}_{b_0}^T] &= \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0b} \tilde{\mathbf{m}}_b) \mathbb{E}[\boldsymbol{\Psi}_{b_0\tilde{b}_0} \boldsymbol{\Psi}_{b_0\tilde{b}_0}^T] \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0b} \tilde{\mathbf{m}}_b)^T \\ &\quad + \mathbf{R}_{\tilde{b}_0b} \mathbb{E}[\delta\mathbf{m}_b \delta\mathbf{m}_b^T] \mathbf{R}_{\tilde{b}_0b}^T \end{aligned} \quad (4.111)$$

Note that the $\delta\mathbf{a}_{b_0}$ vector of the integrated acceleration observation (4.99) and the $\delta\mathbf{b}_{b_0}$ vector of the magnetic field observation (4.109) are correlated, that is to say by

$$\mathbb{E}[\delta\mathbf{a}_{b_0,k} \delta\mathbf{b}_{b_0,k}^T] = -\mathbf{H}_{a_{b_0}} \mathbf{P}_{z_{a_{b_0}},k} \underbrace{(\mathbf{0} \quad \mathbf{I}_3 \quad | \quad \mathbf{0} \quad \mathbf{0})^T}_{\mathbf{H}_{\psi_{b_0\tilde{b}_0}}} \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0b,k} \tilde{\mathbf{m}}_{b,k})^T \quad (4.112)$$

This correlation has to be considered when the covariance of the initial orientation angle error is computed according to (4.74).

However, magnetic field measurements have to be regarded with caution because they are often subject to electro-magnetic interferences in the environment of the sensor. Before magnetic field sensors can be reliably used, they have to be calibrated right at the intended installation location in the aircraft with all systems in the surrounding that might potentially disturb switched on. Special calibration chambers are required in which the magnetic field can be actively controlled, which however limits the maximum size of the carrier platform.

4.2.3.6 Template

The equations for the orientation angle expected values from two arbitrary vector observations are summarized in Template 4-8. The corresponding covariance and cross-covariances are listed in Template 4-9 and the required matrices in Template 4-10 and Template 4-11. Template 4-12 gives the equations for the integrated acceleration observation. Template 4-13 recapitulates the magnetic field observation.

Template 4-8: Orientation angle expected values from two vector observations

Input vectors	$\tilde{\mathbf{a}}_{n_0}, \tilde{\mathbf{b}}_{n_0}, \tilde{\mathbf{a}}_{b_0}, \tilde{\mathbf{b}}_{b_0}$
	$\tilde{\mathbf{c}}_{n_0} = \tilde{\mathbf{a}}_{n_0} \times \tilde{\mathbf{b}}_{n_0}, \quad \tilde{\mathbf{c}}_{b_0} = \tilde{\mathbf{a}}_{b_0} \times \tilde{\mathbf{b}}_{b_0}$
Scaling	$\bar{\mathbf{a}}_{n_0} = \frac{\tilde{\mathbf{a}}_{n_0}}{\ \tilde{\mathbf{a}}_{n_0}\ }, \quad \bar{\mathbf{b}}_{n_0} = \frac{\tilde{\mathbf{b}}_{n_0}}{\ \tilde{\mathbf{b}}_{n_0}\ }, \quad \bar{\mathbf{c}}_{n_0} = \frac{\tilde{\mathbf{c}}_{n_0}}{\ \tilde{\mathbf{a}}_{n_0}\ \ \tilde{\mathbf{b}}_{n_0}\ }$ $\bar{\mathbf{a}}_{b_0} = \frac{\tilde{\mathbf{a}}_{b_0}}{\ \tilde{\mathbf{a}}_{b_0}\ }, \quad \bar{\mathbf{b}}_{b_0} = \frac{\tilde{\mathbf{b}}_{b_0}}{\ \tilde{\mathbf{b}}_{b_0}\ }, \quad \bar{\mathbf{c}}_{b_0} = \frac{\tilde{\mathbf{c}}_{b_0}}{\ \tilde{\mathbf{a}}_{b_0}\ \ \tilde{\mathbf{b}}_{b_0}\ }$
Direction cosine matrix	$\mathbf{A} = (\bar{\mathbf{a}}_{n_0}, \bar{\mathbf{b}}_{n_0}, \bar{\mathbf{c}}_{n_0})$ $\bar{\mathbf{u}}_b = \frac{1}{D} (\bar{\mathbf{b}}_{b_0} \times \bar{\mathbf{c}}_{b_0}), \quad \bar{\mathbf{v}}_{b_0} = \frac{1}{D} (\bar{\mathbf{c}}_{b_0} \times \bar{\mathbf{a}}_{b_0}), \quad \bar{\mathbf{w}}_{b_0} = \frac{1}{D} (\bar{\mathbf{a}}_{b_0} \times \bar{\mathbf{b}}_{b_0})$ $\mathbf{B}^{-1} = \begin{pmatrix} \bar{\mathbf{u}}_{b_0}^T \\ \bar{\mathbf{v}}_{b_0}^T \\ \bar{\mathbf{w}}_{b_0}^T \end{pmatrix}$ $\mathbf{R}_{n_0 b_0} = \mathbf{A} \mathbf{B}^{-1}$
Orientation angles	$\varphi_{n_0 b_0} = \text{atan} 2(R_{n_0 b_0, 32}, R_{n_0 b_0, 33}), \quad \vartheta_{n_0 b_0} = \arcsin(-R_{n_0 b_0, 31})$ $\psi_{n_0 b_0} = \text{atan} 2(R_{n_0 b_0, 21}, R_{n_0 b_0, 11})$ $D = \bar{\mathbf{a}}_{b_0}^T (\bar{\mathbf{b}}_{b_0} \times \bar{\mathbf{c}}_{b_0})$

Template 4-9: Orientation error covariance**Scaled observation vector covariances**

$$\begin{aligned}\bar{\mathbf{P}}_{a_{n_0}} &= \frac{1}{\|\tilde{\mathbf{a}}_{n_0}\|^2} \mathbb{E} \left[\delta \mathbf{a}_{n_0} \delta \mathbf{a}_{n_0}^\top \right] & \bar{\mathbf{P}}_{b_{n_0}} &= \frac{1}{\|\tilde{\mathbf{b}}_{n_0}\|^2} \mathbb{E} \left[\delta \mathbf{b}_{n_0} \delta \mathbf{b}_{n_0}^\top \right] \\ \bar{\mathbf{P}}_{a_{b_0}} &= \frac{1}{\|\tilde{\mathbf{a}}_{n_0}\|^2} \mathbb{E} \left[\delta \mathbf{a}_{b_0} \delta \mathbf{a}_{b_0}^\top \right] & \bar{\mathbf{P}}_{b_{b_0}} &= \frac{1}{\|\tilde{\mathbf{b}}_{n_0}\|^2} \mathbb{E} \left[\delta \mathbf{b}_{b_0} \delta \mathbf{b}_{b_0}^\top \right] & \bar{\mathbf{P}}_{a_{b_0} b_{b_0}} &= \frac{1}{\|\tilde{\mathbf{a}}_{n_0}\| \|\tilde{\mathbf{b}}_{n_0}\|} \mathbb{E} \left[\delta \mathbf{a}_{b_0} \delta \mathbf{b}_{b_0}^\top \right]\end{aligned}$$

Orientation error covariance

$$\begin{aligned}\mathbb{E} \left[\Psi_{n\bar{n}} \Psi_{n\bar{n}}^\top \right] &= \mathbf{H}_{a_{n_0}} \bar{\mathbf{P}}_{a_{n_0}} \mathbf{H}_{a_{n_0}}^\top + \mathbf{H}_{a_{b_0}} \bar{\mathbf{P}}_{a_{b_0}} \mathbf{H}_{a_{b_0}}^\top + \mathbf{H}_{b_{n_0}} \bar{\mathbf{P}}_{b_{n_0}} \mathbf{H}_{b_{n_0}}^\top + \mathbf{H}_{b_{b_0}} \bar{\mathbf{P}}_{b_{b_0}} \mathbf{H}_{b_{b_0}}^\top \\ &\quad + \mathbf{H}_{a_{b_0}} \bar{\mathbf{P}}_{a_{b_0} b_{b_0}} \mathbf{H}_{b_{b_0}}^\top + \mathbf{H}_{b_{b_0}} \bar{\mathbf{P}}_{b_{b_0} a_{b_0}} \mathbf{H}_{a_{b_0}}^\top\end{aligned}$$

Template 4-10: Required matrices 1

$$\begin{aligned}\mathbf{C}_{\psi a_n} &= \left(\begin{array}{c} \cos \vartheta_{\bar{n}b} \cos \psi_{\bar{n}b} \\ \cos \vartheta_{\bar{n}b} \sin \psi_{\bar{n}b} \\ -\sin \vartheta_{\bar{n}b} \end{array} \right) \frac{\partial \varphi_{nb}}{\partial \mathbf{a}_n} + \left(\begin{array}{c} -\sin \psi_{\bar{n}b} \\ \cos \psi_{\bar{n}b} \\ 0 \end{array} \right) \frac{\partial \vartheta_{nb}}{\partial \mathbf{a}_n} + \left(\begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \frac{\partial \psi_{nb}}{\partial \mathbf{a}_n} \\ \mathbf{C}_{\psi b_n} &= \left(\begin{array}{c} \cos \vartheta_{\bar{n}b} \cos \psi_{\bar{n}b} \\ \cos \vartheta_{\bar{n}b} \sin \psi_{\bar{n}b} \\ -\sin \vartheta_{\bar{n}b} \end{array} \right) \frac{\partial \varphi_{nb}}{\partial \mathbf{b}_n} + \left(\begin{array}{c} -\sin \psi_{\bar{n}b} \\ \cos \psi_{\bar{n}b} \\ 0 \end{array} \right) \frac{\partial \vartheta_{nb}}{\partial \mathbf{b}_n} + \left(\begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \frac{\partial \psi_{nb}}{\partial \mathbf{b}_n} \\ \mathbf{C}_{\psi c_n} &= \left(\begin{array}{c} \cos \vartheta_{\bar{n}b} \cos \psi_{\bar{n}b} \\ \cos \vartheta_{\bar{n}b} \sin \psi_{\bar{n}b} \\ -\sin \vartheta_{\bar{n}b} \end{array} \right) \frac{\partial \varphi_{nb}}{\partial \mathbf{c}_n} + \left(\begin{array}{c} -\sin \psi_{\bar{n}b} \\ \cos \psi_{\bar{n}b} \\ 0 \end{array} \right) \frac{\partial \vartheta_{nb}}{\partial \mathbf{c}_n} + \left(\begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \frac{\partial \psi_{nb}}{\partial \mathbf{c}_n} \\ \mathbf{C}_{\psi u_b} &= \left(\begin{array}{c} \cos \vartheta_{\bar{n}b} \cos \psi_{\bar{n}b} \\ \cos \vartheta_{\bar{n}b} \sin \psi_{\bar{n}b} \\ -\sin \vartheta_{\bar{n}b} \end{array} \right) \frac{\partial \varphi_{nb}}{\partial \mathbf{u}_b} + \left(\begin{array}{c} -\sin \psi_{\bar{n}b} \\ \cos \psi_{\bar{n}b} \\ 0 \end{array} \right) \frac{\partial \vartheta_{nb}}{\partial \mathbf{u}_b} + \left(\begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \frac{\partial \psi_{nb}}{\partial \mathbf{u}_b} \\ \mathbf{C}_{\psi v_b} &= \left(\begin{array}{c} \cos \vartheta_{\bar{n}b} \cos \psi_{\bar{n}b} \\ \cos \vartheta_{\bar{n}b} \sin \psi_{\bar{n}b} \\ -\sin \vartheta_{\bar{n}b} \end{array} \right) \frac{\partial \varphi_{nb}}{\partial \mathbf{v}_b} + \left(\begin{array}{c} -\sin \psi_{\bar{n}b} \\ \cos \psi_{\bar{n}b} \\ 0 \end{array} \right) \frac{\partial \vartheta_{nb}}{\partial \mathbf{v}_b} + \left(\begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \frac{\partial \psi_{nb}}{\partial \mathbf{v}_b} \\ \mathbf{C}_{\psi w_b} &= \left(\begin{array}{c} \cos \vartheta_{\bar{n}b} \cos \psi_{\bar{n}b} \\ \cos \vartheta_{\bar{n}b} \sin \psi_{\bar{n}b} \\ -\sin \vartheta_{\bar{n}b} \end{array} \right) \frac{\partial \varphi_{nb}}{\partial \mathbf{w}_b} + \left(\begin{array}{c} -\sin \psi_{\bar{n}b} \\ \cos \psi_{\bar{n}b} \\ 0 \end{array} \right) \frac{\partial \vartheta_{nb}}{\partial \mathbf{w}_b} + \left(\begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \frac{\partial \psi_{nb}}{\partial \mathbf{w}_b}\end{aligned}$$

$$\begin{aligned}\mathbf{C}_{u_b a_b} &= -\frac{1}{D} \bar{\mathbf{u}}_b (\bar{\mathbf{b}}_b \times \bar{\mathbf{c}}_b)^\top & \mathbf{C}_{v_b a_b} &= \frac{1}{D} \left(\text{veck}^{-1}(\bar{\mathbf{c}}_b) - \bar{\mathbf{v}}_b (\bar{\mathbf{b}}_b \times \bar{\mathbf{c}}_b)^\top \right) \\ \mathbf{C}_{u_b b_b} &= -\frac{1}{D} (\mathbf{I}_3 - \bar{\mathbf{u}}_b \bar{\mathbf{a}}_b^\top) \text{veck}^{-1}(\bar{\mathbf{c}}_b) & \mathbf{C}_{v_b b_b} &= \frac{1}{D} \bar{\mathbf{v}}_b \bar{\mathbf{a}}_b^\top \text{veck}^{-1}(\bar{\mathbf{c}}_b) \\ \mathbf{C}_{u_b c_b} &= \frac{1}{D} (\mathbf{I}_3 - \bar{\mathbf{u}}_b \bar{\mathbf{a}}_b^\top) \text{veck}^{-1}(\bar{\mathbf{b}}_b) & \mathbf{C}_{v_b c_b} &= -\frac{1}{D} \left(\text{veck}^{-1}(\bar{\mathbf{a}}_b) + \bar{\mathbf{v}}_b \bar{\mathbf{a}}_b^\top \text{veck}^{-1}(\bar{\mathbf{b}}_b) \right)\end{aligned}$$

$$\begin{aligned}\mathbf{C}_{w_b a_b} &= -\frac{1}{D} \left(\text{veck}^{-1}(\bar{\mathbf{b}}_b) + \bar{\mathbf{w}}_b (\bar{\mathbf{b}}_b \times \bar{\mathbf{c}}_b)^\top \right) \\ \mathbf{C}_{w_b b_b} &= \frac{1}{D} \left(\text{veck}^{-1}(\bar{\mathbf{a}}_b) + \bar{\mathbf{w}}_b \bar{\mathbf{a}}_b^\top \text{veck}^{-1}(\bar{\mathbf{c}}_b) \right) \\ \mathbf{C}_{w_b c_b} &= -\frac{1}{D} \bar{\mathbf{w}}_b \bar{\mathbf{a}}_b^\top \text{veck}^{-1}(\bar{\mathbf{b}}_b)\end{aligned}$$

$$\begin{aligned}\mathbf{C}_{\psi a_b} &= \mathbf{C}_{\psi u_b} \mathbf{C}_{u_b a_b} + \mathbf{C}_{\psi v_b} \mathbf{C}_{v_b a_b} + \mathbf{C}_{\psi w_b} \mathbf{C}_{w_b a_b} \\ \mathbf{C}_{\psi b_b} &= \mathbf{C}_{\psi u_b} \mathbf{C}_{u_b b_b} + \mathbf{C}_{\psi v_b} \mathbf{C}_{v_b b_b} + \mathbf{C}_{\psi w_b} \mathbf{C}_{w_b b_b} \\ \mathbf{C}_{\psi c_b} &= \mathbf{C}_{\psi u_b} \mathbf{C}_{u_b c_b} + \mathbf{C}_{\psi v_b} \mathbf{C}_{v_b c_b} + \mathbf{C}_{\psi w_b} \mathbf{C}_{w_b c_b}\end{aligned}$$

$$\begin{aligned}\mathbf{H}_{a_n} &= \mathbf{C}_{\psi a_n} - \mathbf{C}_{\psi c_n} \text{veck}^{-1}(\bar{\mathbf{b}}_n) & \mathbf{H}_{a_b} &= \mathbf{C}_{\psi a_b} - \mathbf{C}_{\psi c_b} \text{veck}^{-1}(\bar{\mathbf{b}}_b) \\ \mathbf{H}_{b_n} &= \mathbf{C}_{\psi b_n} + \mathbf{C}_{\psi c_n} \text{veck}^{-1}(\bar{\mathbf{a}}_n) & \mathbf{H}_{b_b} &= \mathbf{C}_{\psi b_b} + \mathbf{C}_{\psi c_b} \text{veck}^{-1}(\bar{\mathbf{a}}_b)\end{aligned}$$

Template 4-11: Required matrices 2

$$\begin{array}{lll}
\frac{\partial \varphi_{nb}}{\partial \mathbf{a}^T} = \frac{1}{R_{nb,32}^2 + R_{nb,33}^2} \begin{pmatrix} 0 \\ 0 \\ \bar{u}_{b,y} R_{nb,33} - \bar{u}_{b,z} R_{nb,32} \end{pmatrix}^T & \frac{\partial g_{nb}}{\partial \mathbf{a}^T} = \frac{-1}{\sqrt{1-R_{nb,31}^2}} \begin{pmatrix} 0 \\ 0 \\ \bar{u}_{b,x} \end{pmatrix}^T & \frac{\partial \psi_{nb}}{\partial \mathbf{a}^T} = \frac{1}{R_{nb,21}^2 + R_{nb,11}^2} \begin{pmatrix} -\bar{u}_{b,x} R_{nb,21} \\ \bar{u}_{b,x} R_{nb,11} \\ 0 \end{pmatrix}^T \\
\frac{\partial \varphi_{nb}}{\partial \mathbf{b}^T} = \frac{1}{R_{nb,32}^2 + R_{nb,33}^2} \begin{pmatrix} 0 \\ 0 \\ \bar{v}_{b,y} R_{nb,33} - \bar{v}_{b,z} R_{nb,32} \end{pmatrix}^T & \frac{\partial g_{nb}}{\partial \mathbf{b}^T} = \frac{-1}{\sqrt{1-R_{nb,31}^2}} \begin{pmatrix} 0 \\ 0 \\ \bar{v}_{b,x} \end{pmatrix}^T & \frac{\partial \psi_{nb}}{\partial \mathbf{b}^T} = \frac{1}{R_{nb,21}^2 + R_{nb,11}^2} \begin{pmatrix} -\bar{v}_{b,x} R_{nb,21} \\ \bar{v}_{b,x} R_{nb,11} \\ 0 \end{pmatrix}^T \\
\frac{\partial \varphi_{nb}}{\partial \mathbf{c}^T} = \frac{1}{R_{nb,32}^2 + R_{nb,33}^2} \begin{pmatrix} 0 \\ 0 \\ \bar{w}_{b,y} R_{nb,33} - \bar{w}_{b,z} R_{nb,32} \end{pmatrix}^T & \frac{\partial g_{nb}}{\partial \mathbf{c}^T} = \frac{-1}{\sqrt{1-R_{nb,31}^2}} \begin{pmatrix} 0 \\ 0 \\ \bar{w}_{b,x} \end{pmatrix}^T & \frac{\partial \psi_{nb}}{\partial \mathbf{c}^T} = \frac{1}{R_{nb,21}^2 + R_{nb,11}^2} \begin{pmatrix} -\bar{w}_{b,x} R_{nb,21} \\ \bar{w}_{b,x} R_{nb,11} \\ 0 \end{pmatrix}^T \\
\frac{\partial \varphi_{nb}}{\partial \mathbf{u}_b} = \frac{1}{R_{nb,32}^2 + R_{nb,33}^2} \begin{pmatrix} 0 \\ \bar{a}_{n,z} R_{nb,33} \\ -\bar{a}_{n,z} R_{nb,32} \end{pmatrix}^T & \frac{\partial g_{nb}}{\partial \mathbf{u}_b} = \frac{-1}{\sqrt{1-R_{nb,31}^2}} \begin{pmatrix} \bar{a}_{n,z} \\ 0 \\ 0 \end{pmatrix}^T & \frac{\partial \psi_{nb}}{\partial \mathbf{u}_b} = \frac{1}{R_{nb,21}^2 + R_{nb,11}^2} \begin{pmatrix} \bar{a}_{n,y} R_{nb,11} - \bar{a}_{n,z} R_{nb,21} \\ 0 \\ 0 \end{pmatrix}^T \\
\frac{\partial \varphi_{nb}}{\partial \mathbf{v}_b} = \frac{1}{R_{nb,32}^2 + R_{nb,33}^2} \begin{pmatrix} 0 \\ \bar{b}_{a,z} R_{nb,33} \\ -\bar{b}_{a,z} R_{nb,32} \end{pmatrix}^T & \frac{\partial g_{nb}}{\partial \mathbf{v}_b} = \frac{-1}{\sqrt{1-R_{nb,31}^2}} \begin{pmatrix} \bar{b}_{a,z} \\ 0 \\ 0 \end{pmatrix}^T & \frac{\partial \psi_{nb}}{\partial \mathbf{v}_b} = \frac{1}{R_{nb,21}^2 + R_{nb,11}^2} \begin{pmatrix} \bar{b}_{a,y} R_{nb,11} - \bar{b}_{a,z} R_{nb,21} \\ 0 \\ 0 \end{pmatrix}^T \\
\frac{\partial \varphi_{nb}}{\partial \mathbf{w}_b} = \frac{1}{R_{nb,32}^2 + R_{nb,33}^2} \begin{pmatrix} 0 \\ \bar{c}_{a,z} R_{nb,33} \\ -\bar{c}_{a,z} R_{nb,32} \end{pmatrix}^T & \frac{\partial g_{nb}}{\partial \mathbf{w}_b} = \frac{-1}{\sqrt{1-R_{nb,31}^2}} \begin{pmatrix} \bar{c}_{a,z} \\ 0 \\ 0 \end{pmatrix}^T & \frac{\partial \psi_{nb}}{\partial \mathbf{w}_b} = \frac{1}{R_{nb,21}^2 + R_{nb,11}^2} \begin{pmatrix} \bar{c}_{a,y} R_{nb,11} - \bar{c}_{a,z} R_{nb,21} \\ 0 \\ 0 \end{pmatrix}^T
\end{array}$$

Template 4-12: Integrated acceleration observation**Expected Values**

\mathbf{a}_{n_0}	$\tilde{\mathbf{a}}_{n_0,k} = \tilde{\mathbf{v}}_{n,k} - \tilde{\mathbf{v}}_{n,0} - \mathbf{y}_n \cdot (t_k - t_0)$
\mathbf{a}_{b_0}	$\dot{\tilde{q}}_{b_0 b} = \frac{1}{2} (\tilde{q}_{b_0 b} \tilde{\omega}_{ib} - \tilde{\omega}_{ib} (\tilde{q}_{b_0 b}))$, $\tilde{q}_{b_0 b}(t_0) = (1, \mathbf{0})$ $\mathbf{R}_{\tilde{b}_0 b, k} = \mathbf{R}(\tilde{q}_{\tilde{b}_0 b, k})$ $\tilde{\mathbf{a}}_{b_0, k} = \int_{t_0}^{t_k} \mathbf{R}_{\tilde{b}_0 b} \tilde{\mathbf{f}}_b d\tau$

Covariances

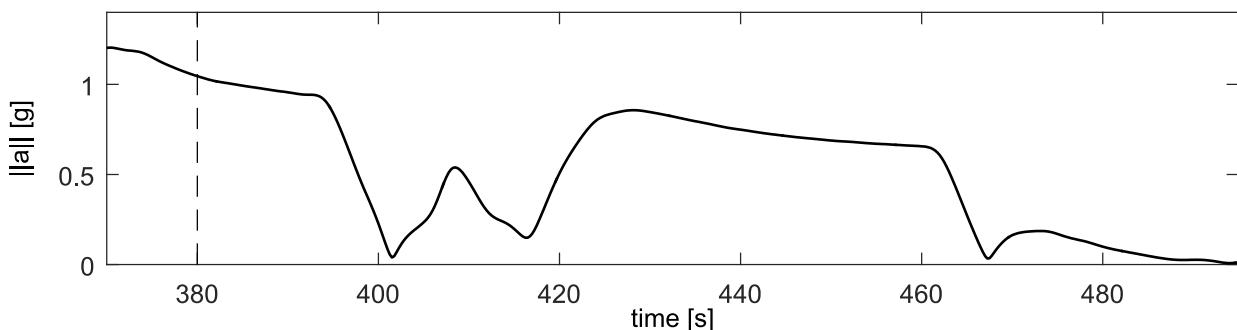
$\mathbf{P}_{a_{n_0}}$	$\mathbf{P}_{a_{n_0},k} = E[\delta \mathbf{v}_{n,k} \delta \mathbf{v}_{n,k}^T] + E[\delta \mathbf{v}_{n,0} \delta \mathbf{v}_{n,0}^T]$
	$\mathbf{P}_{z_{a_{b_0}},k} = \Phi_{a_{b_0},k-1} \mathbf{P}_{z_{a_{b_0}},k-1} \Phi_{a_{b_0},k-1}^T + \Gamma_{a_{b_0},k-1} \mathbf{Q}_{a_{b_0},k-1} \Gamma_{a_{b_0},k-1}^T$
	$\mathbf{P}_{a_{b_0},k} = \mathbf{H}_{a_{b_0}} \mathbf{P}_{z_{a_{b_0}},k} \mathbf{H}_{a_{b_0}}^T$
	$\Phi_{a_{b_0},k} = \begin{pmatrix} \mathbf{I}_3 & -\Delta t \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0 b, k-1} \tilde{\mathbf{f}}_{b,k}) & \Delta t \mathbf{R}_{\tilde{b}_0 b, k} \mathbf{H}_f & 0 \\ 0 & \mathbf{I}_3 & 0 & \Delta t \mathbf{R}_{\tilde{b}_0 b, k} \mathbf{H}_\omega \\ 0 & 0 & \Phi_f & 0 \\ 0 & 0 & 0 & \Phi_\omega \end{pmatrix}$, $\Gamma_{a_{b_0},k} = \begin{pmatrix} \Delta t \mathbf{R}_{\tilde{b}_0 b, k} \mathbf{D}_f & 0 \\ 0 & \Delta t \mathbf{R}_{\tilde{b}_0 b, k} \mathbf{D}_\omega \\ \Gamma_f & 0 \\ 0 & \Gamma_\omega \end{pmatrix}$
	$\mathbf{H}_{a_{b_0}} = (\mathbf{I}_3 \ 0 \mid 0 \ 0)$, $\mathbf{H}_{\psi_{b_0 b_0}} = (0 \ \mathbf{I}_3 \mid 0 \ 0)$
	$\mathbf{P}_{z_{a_{b_0}},0} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \mathbf{P}_{z_f,0} & 0 \\ 0 & 0 & \mathbf{P}_{z_\omega,0} \end{pmatrix}$, $\mathbf{Q}_{a_{b_0},k} = \begin{pmatrix} \mathbf{Q}_{f,k} & 0 \\ 0 & \mathbf{Q}_{\omega,k} \end{pmatrix}$

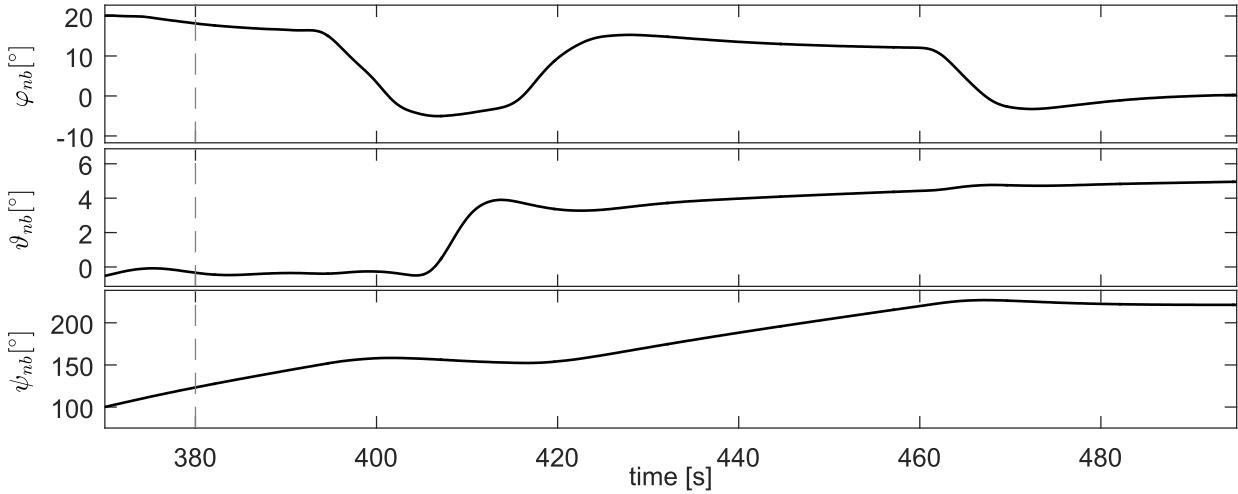
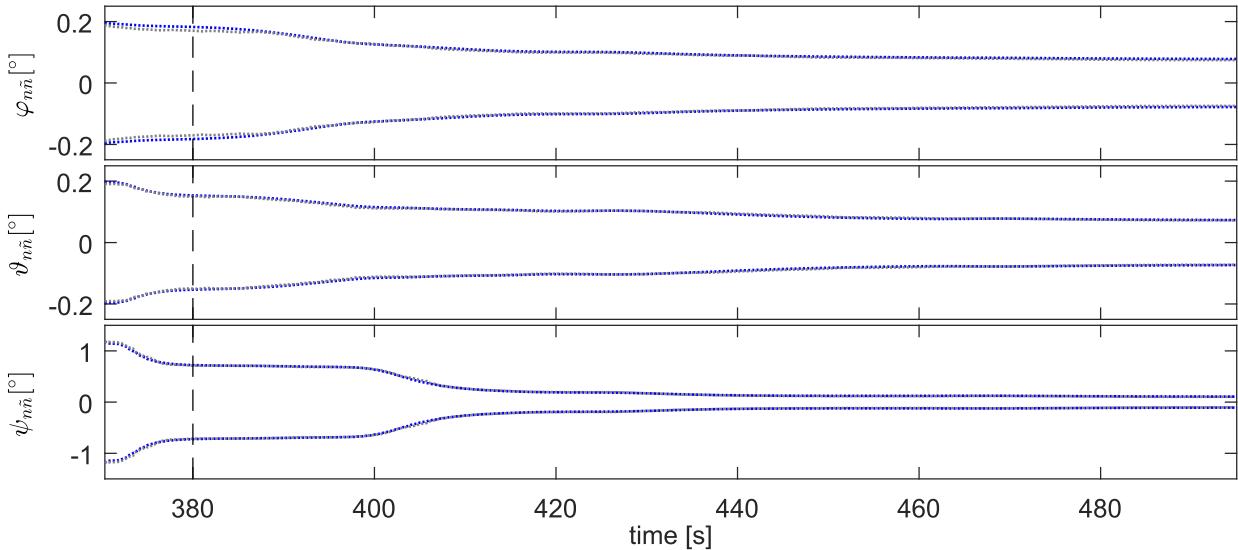
Template 4-13: Magnetic field observation

Expected Values	
\mathbf{b}_{n_0}	$\tilde{\mathbf{b}}_{n_0} = \tilde{\mathbf{m}}_n$
\mathbf{b}_{b_0}	$\tilde{\mathbf{b}}_{b_0,k} = \mathbf{R}_{\tilde{b}_0 b, k} \tilde{\mathbf{m}}_{b,k}$
Covariances	
$\mathbf{P}_{b_{n_0}}$	$\mathbf{P}_{b_{n_0}} = \mathbb{E}[\delta \mathbf{m}_n \delta \mathbf{m}_n^\top]$
$\mathbf{P}_{b_{b_0}}$	$\mathbf{P}_{b_{b_0},k} = \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0 b, k} \tilde{\mathbf{m}}_{b,k}) \mathbb{E}[\Psi_{b_0 \tilde{b}_0, k} \Psi_{b_0 \tilde{b}_0, k}^\top] \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0 b, k} \tilde{\mathbf{m}}_{b,k})^\top + \mathbf{R}_{\tilde{b}_0 b, k} \mathbb{E}[\delta \mathbf{m}_{b,k} \delta \mathbf{m}_{b,k}^\top] \mathbf{R}_{\tilde{b}_0 b, k}^\top$
Cross-Covariance	
$\mathbf{P}_{a_{b_0} b_{b_0}}$	$\mathbf{P}_{a_{b_0} b_{b_0},k} = -\mathbf{H}_{a_{b_0}} \mathbf{P}_{a_{b_0},k} \mathbf{H}_{b_{b_0}}^\top \text{veck}^{-1}(\mathbf{R}_{\tilde{b}_0 b, k} \tilde{\mathbf{m}}_{b,k})^\top$

4.2.3.7 Example

The in-flight initialization is demonstrated by means of an example. The same setup as in the example of the stationary initialization in section 4.2.2.5 is chosen. The navigation system shall be initialized in flight at second 375. The initialization period is five seconds. After five seconds, that is at second 380, the initial orientation at second 375 is calculated, the navigation filter is initialized with the estimated orientation and re-launched from second 375. The magnitude of the acceleration is shown in Figure 4-26. It can be seen that the acceleration is above one g during the initialization period. The corresponding orientation angles are illustrated in Figure 4-27. The roll angle is about 20° and pitch is slightly negative in the beginning. The initial orientation angles are estimated with the integrated acceleration observation on the one hand and magnetic field observation on the other hand. The sample rate of the velocity measurement is 5 Hz. The velocity measurement error is assumed to be purely white noise with 0.1 m/s standard deviation. The sample rate of the magnetic field measurement is 100 Hz. The magnetic field measurement error is assumed to be purely white noise with $5.4 \text{ nT}/\sqrt{\text{Hz}}$ standard deviation. The IMU is affected by velocity/angular random walk noise, turn-on bias, scale factor error and misalignments. Figure 4-28 shows the 3σ standard deviations of the orientation angles. The black solid line is the standard deviation as predicted by the integrated navigation filter. The grey dotted line represents the actual standard deviation from 500 Monte Carlo runs. Although the correlation of the measurement errors due to twice processing the same measurements once in the initial orientation angle estimation and once on the integrated navigation filter has been neglected, there is a very good statistical coherence.

**Figure 4-26: Acceleration magnitude**

**Figure 4-27: Orientation angles****Figure 4-28: Orientation error 3 σ boundaries with consideration of the correlation (blue: predicted, grey: Monte Carlo simulation)**

4.2.4 Conclusion

It has been shown how the initial orientation angle values and the corresponding covariance can be estimated if the aircraft is stationary or in flight. The bias-like IMU measurement errors, first of all the turn-on biases, affect the initial orientation angles. It can be seen that the estimated orientation angles and sensor errors are partially correlated. If this correlation is considered at the initialization of the navigation filter covariance matrix, the statistics are perfectly fulfilled immediately from the beginning. If the correlations are neglected and a diagonal and of course over-bounded orientation error covariance is chosen, the predicted roll and pitch angle error standard deviations will nevertheless settle quite fast to the true values, but the heading angle error might take longer to do so if the heading observability is not increased for example with a fine alignment maneuver. In this case accuracy is given away. As by-product, the downward pointing gyroscope error component can be calibrated during stationary initialization since this component does not go into the orientation angle estimates. For in-flight initialization, the integrated acceleration observation and a magnetic field measurement have been used to estimate the initial orientation. Simulation results have revealed that this initialization method works well even if the aircraft is accelerating during the initialization period. The acceleration direction can even change during the initialization process.

4.3 Out-of-Sequence Measurements

4.3.1 Motivation

Integrated navigation systems feature various sensors with different sample rates and measurement processing times. Accelerometer and gyroscope measurements are integrated and fused with external aiding measurements to reduce the accumulating navigation state errors. Common aiding sensors are, for example, GNSS receivers, barometer, magnetometer, cameras, radar altimeter or laser scanner.

The most eminent aiding sensor is certainly the GNSS receiver, which provides highly accurate absolute position and velocity or range and range rate measurements to the satellites in view, respectively. The output rate is usually between one and ten Hertz. Depending on the number of tracked satellites, the raw data processing can take up to 200 ms time. The RTCA DO-229D standard [3], which is effective for certified airborne GNSS receivers, actually requires maximum output delays of less than 200 ms in section 2.1.2.6.2. This comparatively long time interval is due to the amount of signal processing tasks that are necessary from the extraction of the actual measurements, the broadcast time of the signal, carrier frequency and phase angle from the correlators to the output of the readily computed PVT solution. Additional time is consumed when the data is transferred from the receiver interface to the navigation computer interface. In high dynamic applications, the navigation state can significantly change between the actual instant of the measurement and the moment when the processed measurement is available at the data fusion filter on the navigation computer.

Other aiding sensors like the barometric height sensor with shorter measurement processing times may deliver inputs in the meantime between the actual instant of a GNSS measurement and its arrival at the navigation filter. Processing these height measurements before the awaited GNSS measurement is available requires special out-of-sequence measurement algorithms. Other aiding methods like terrain aided navigation or image aided navigation, which are particularly applied when GNSS reception fails, usually have to process large amounts of data in real time. Methods that extract absolute or relative position information from surface range measurements with synthetic aperture radar (SAR) or laser scanner make use of high sensing frequencies in the range of Kilohertz. Dead reckoning aiding methods that use optical flow information have to evaluate a series of subsequent camera images to determine the flow vectors and the corresponding estimation covariances. In both cases, the processing of the collected raw data is very time consuming and the effective output delays may be even longer than those of the PVT solution of a GNSS receiver.

Figure 4-29 shows an example where the measurements of two sensors \tilde{y}_1 and \tilde{y}_2 are fused to obtain the state estimate $\hat{\mathbf{z}}$ of the true state \mathbf{z} . Measurement \tilde{y}_1 arrives regularly every fifth base time step at times $\dots, t_{k-2}, t_{k-1}, t_k, \dots$. It comes in without delay and can be directly processed by the filter. Measurement $\tilde{y}_{1,k}$ at time step t_k , which is related to the true state \mathbf{z}_k , for example, is immediately applied to obtain the state estimate $\hat{\mathbf{z}}_k^+$. The second measurement \tilde{y}_2 is, however, delayed by seven base time steps. The measurement $\tilde{y}_{2,j}$, which is made at time t_j and which is related to the true state \mathbf{z}_j , is not available before the current time t_k and not before the current measurement of the first sensor $\tilde{y}_{1,k}$ has been processed. Since the measurements $\tilde{y}_{1,k-1}$ and $\tilde{y}_{1,k}$ have already been incorporated into the filter to estimate $\hat{\mathbf{z}}_{k-1}^+$ and $\hat{\mathbf{z}}_k^+$, the measurement $\tilde{y}_{2,j}$ arrives out-of-sequence at the current time t_k . The innovation cannot be calculated by simply comparing the sensed value $\tilde{y}_{2,j}$

with the expected value from the current state estimate $h_2(\hat{\mathbf{z}}_k^+)$ because $\hat{\mathbf{z}}_k^+$ does not describe the platform state \mathbf{z}_j at time t_j seven time steps before the current time t_k but the platform state \mathbf{z}_k , which may have appreciably changed in the meantime. On the other hand, the measurement $\tilde{y}_{2,j}$ can also not directly be compared with the past state estimate $\hat{\mathbf{z}}_j^-$ because this estimate does not contain the information of the measurements $\tilde{y}_{1,k-1}$ and $\tilde{y}_{1,k}$ that have been processed by the filter meanwhile. The calculated innovation would not represent the innovative content of the measurement $\tilde{y}_{2,j}$ correctly.

In the following, three different methods are presented that can be used to handle out-of-sequence measurements in the navigation filter properly: First, the state backward propagation method that has been extensively analyzed by Bar-Shalom for tracking filters [106], second the measurement prediction method presented for example in [107] and last the history state update method. In the first method, the current state estimate $\hat{\mathbf{z}}_k^+$ and the corresponding state error covariance \mathbf{P}_k are back-propagated to the validity time t_j of the out-of-sequence measurement $\tilde{y}_{2,j}$ before the innovation is computed. In the second method, not the state estimate $\hat{\mathbf{z}}_k^+$ is back-propagated but the measurement $\tilde{y}_{2,j}$ is forward-propagated from the validity time t_j to the current time t_k using the state difference between $\hat{\mathbf{z}}_j^-$ and $\hat{\mathbf{z}}_k^+$. In the third method, the state estimate $\hat{\mathbf{z}}_j^-$ and the covariance \mathbf{P}_j are updated together with the current state when new measurements are available. All three methods base upon the same basic principle of the fusion of a delayed measurement, which will be explained in section 4.3.3. In the end, all three methods are compared with each other and the commonalities as well as the differences are outlined. It is shown how more than one delayed out-of-sequence measurement in parallel are treated. Finally, a subtleness concerning filter stability is addressed.

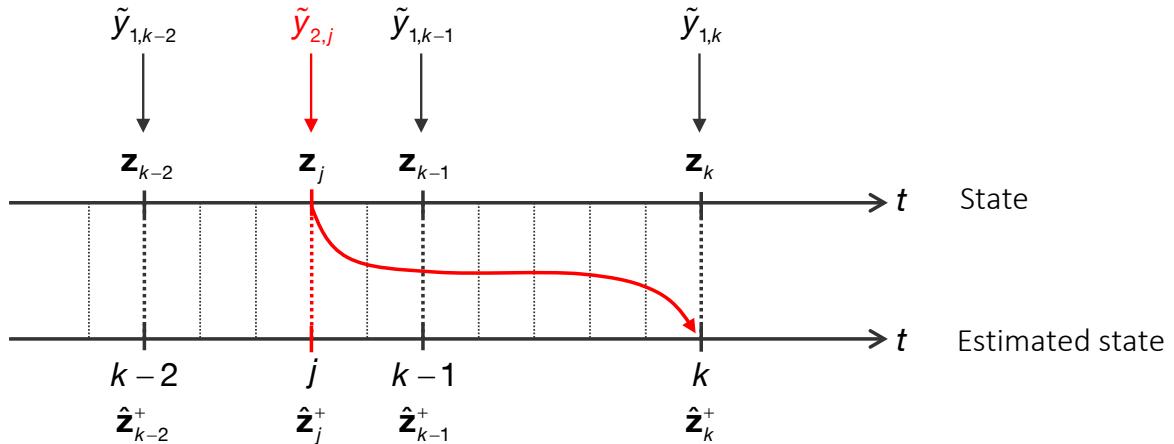


Figure 4-29: Delayed out-of-sequence measurement

4.3.2 Synchronization Concept

The various sensors of an integrated navigation system may work as master or slave. When operating as master, the sensor autonomously outputs measurements with a fix sample rate, which bases upon the frequency of its own oscillator. The measurements are stamped with the own sensor time, which can be the accumulated time since power-on, for example. Although the sensor acts independently of the navigation computer, the sensor clock may be synchronized with the central master clock of the system. For that purpose, the sensor may be supplied by a synchronization signal that is derived from the master oscillator. The measurements may still be triggered by the freely running sensor oscillator but the time stamping is now adjusted to the master time. Measurements that are tagged with timestamps from a common time base ease the fusion in the central navigation filter. In slave mode,

the navigation computer actively polls new measurements from the sensor by transmitting a polling signal. The navigation computer can request measurements from several sensors at the same moment. Simultaneously valid measurements are favorable for the data fusion in the integrated navigation filter. It is also possible that the polling signal is transmitted with a lower rate than the desired measurement rate (for example only once per second) and the interim measurement samples are derived from the sensor oscillator. The drift of the local time relative to the master time during this short period is negligible. If the system developer can influence the design of the integrated navigation system hardware, it is recommended to operate the sensors in slave mode and to timestamp all measurements with the same timescale. The GNSS receiver is often the most accurate time source in an integrated navigation system. Since the receiver is capable of estimating its own clock error as by-product of the position and velocity solution, the current absolute GPS time can be determined with an accuracy of ten to hundred nanoseconds. Moreover, since time stability is a central requirement, GNSS receivers generally incorporate crystal oscillators with higher time stability than run-of-the-mill quartz oscillators as already discussed in section 3.5.5. Therefore, it is reasonable to engage the GNSS receiver as master clock of the integrated navigation system.

Principally, two different scenarios amongst others are imaginable to handle time delays of sensor measurements. In the first case, a sensor indicates to the navigation computer when it takes a new raw measurement. The navigation computer reacts on the signal by remembering the current system time, state estimate and covariance and then waits until the actual measurement values are provided by the sensor. Or if the sensor is operated in slave mode, the navigation computer can simply remember the time when it polls the new measurement from the sensor. In the second case, a separate I/O controller manages the communication with the sensors. It is synchronized with the master clock, polls and receives sensor measurements and stamps all returned sensor values with the consistent measurement time. The measurement value and the corresponding timestamp are forwarded to the navigation computer as soon as the value is available. The navigation computer hence receives the measurement and timestamp all at once bundled in a single message but delayed by the processing and summed communication times.

4.3.3 Delayed Measurement Update

The principle upon which the state backward propagation, the measurement prediction and the history state update methods base is the processing of a single delayed measurement in the filter as illustrated in Figure 4-30. The measurement $\tilde{\mathbf{y}}_j$, which is valid at time t_j , is not available before the current time t_k and shall be used to obtain the optimal estimate of the current state $\hat{\mathbf{z}}_k^+$. The measurement $\tilde{\mathbf{y}}_j$ is related to the true state at time t_j by

$$\tilde{\mathbf{y}}_j = \mathbf{H}_j \mathbf{z}_j + \mathbf{v}_j \quad (4.113)$$

Therein, \mathbf{H}_j is the observation matrix and \mathbf{v}_j the measurement noise. \mathbf{v}_j is assumed to be zero mean, white and Gaussian distributed with covariance \mathbf{R}_j , $\mathbf{v}_j \sim WN(\mathbf{0}, \mathbf{R}_j)$.

In order to find the innovative content of the measurement $\tilde{\mathbf{y}}_j$, it has to be properly associated with the current a priori state estimate $\hat{\mathbf{z}}_k^-$. As has been introductory discussed and will be elaborated in the later sections, this can be done by propagating the state estimate back from the current time t_k to the measurement time t_j or vice versa by predicting the measurement from the measurement time t_j to the current time t_k . Independently of the chosen method, a state estimate $\hat{\mathbf{z}}_{j,k}$ that is valid at the measurement time t_j and that contains all preceding information up to the current time t_k will result.

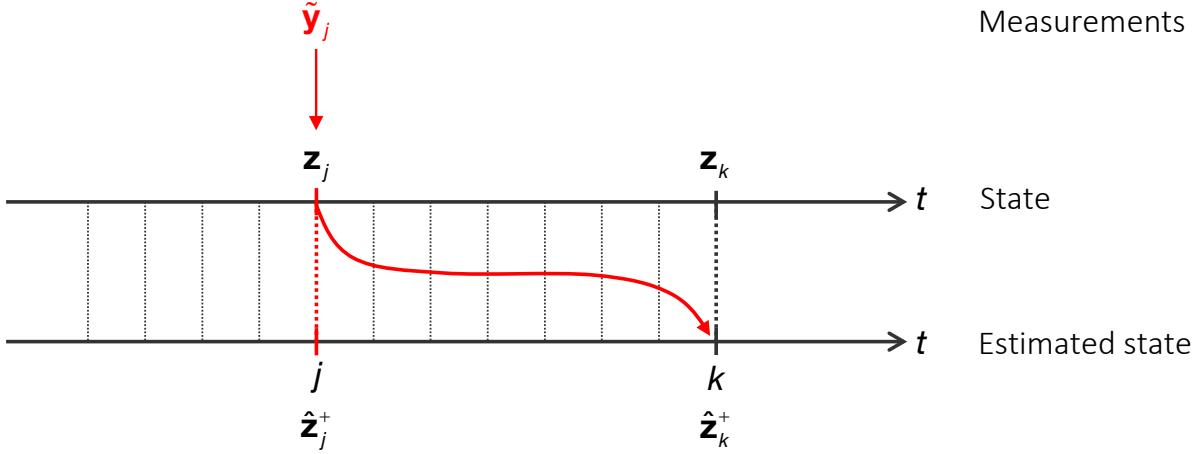


Figure 4-30: Delayed measurement

With this state estimate $\hat{\mathbf{z}}_{j,k}$ and the corresponding covariance $\mathbf{P}_{j,k} = E[\delta\mathbf{z}_{j,k}\delta\mathbf{z}_{j,k}^\top]$, the innovation $\mathbf{s}_{j,k}$ and innovation covariance $\mathbf{S}_{j,k}$ at time t_j can be formed, using the observation matrix \mathbf{H}_j and measurement covariance \mathbf{R}_j , each valid at time t_j ,

$$\begin{aligned}\mathbf{s}_{j,k} &= \tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k} \\ \mathbf{S}_{j,k} &= \mathbf{R}_j + \mathbf{H}_j \mathbf{P}_{j,k} \mathbf{H}_j^\top\end{aligned}\quad (4.114)$$

With the innovation $\mathbf{s}_{j,k}$ and the Kalman gain \mathbf{K}_k , which has yet to be determined for optimal fusion, the newest a priori state estimate $\hat{\mathbf{z}}_k^-$ is updated to get the best available estimate of the state at the current time $\hat{\mathbf{z}}_k^+$

$$\hat{\mathbf{z}}_k^+ = \hat{\mathbf{z}}_k^- + \mathbf{K}_k \mathbf{s}_{j,k} = \hat{\mathbf{z}}_k^- + \mathbf{K}_k (\tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k}) \quad (4.115)$$

The estimated state error after the update $\delta\mathbf{z}_k^+ = \mathbf{z}_k - \hat{\mathbf{z}}_k^+$ will be

$$\begin{aligned}\delta\mathbf{z}_k^+ &= \mathbf{z}_k - \hat{\mathbf{z}}_k^- - \mathbf{K}_k (\tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k}) \\ &= \delta\mathbf{z}_k^- - \mathbf{K}_k (\tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k})\end{aligned}\quad (4.116)$$

With the observation equation (4.113) the estimated posterior state error finally becomes

$$\begin{aligned}\delta\mathbf{z}_k^+ &= \delta\mathbf{z}_k^- - \mathbf{K}_k (\mathbf{H}_j \mathbf{z}_j + \mathbf{v}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k}) \\ &= \delta\mathbf{z}_k^- - \mathbf{K}_k \mathbf{H}_j \delta\mathbf{z}_{j,k} - \mathbf{K}_k \mathbf{v}_j\end{aligned}\quad (4.117)$$

Assuming that the mean value of the estimated state error is zero, $E[\delta\mathbf{z}_k^+] = \mathbf{0}$, the covariance of the estimated state error after the update is

$$\mathbf{P}_k^+ = E[\delta\mathbf{z}_k^+ \delta\mathbf{z}_k^{+\top}] \quad (4.118)$$

With the state error update relation (4.117) the covariance matrix is further

$$\begin{aligned}\mathbf{P}_k^+ &= E\left[(\delta\mathbf{z}_k^- - \mathbf{K}_k \mathbf{H}_j \delta\mathbf{z}_{j,k} - \mathbf{K}_k \mathbf{v}_j)(\delta\mathbf{z}_k^- - \mathbf{K}_k \mathbf{H}_j \delta\mathbf{z}_{j,k} - \mathbf{K}_k \mathbf{v}_j)^\top\right] \\ &= \mathbf{P}_k^- + \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{j,k} \mathbf{H}_j^\top \mathbf{K}_k^\top + \mathbf{K}_k \mathbf{R}_j \mathbf{K}_k^\top \\ &\quad - E[\delta\mathbf{z}_k^- \delta\mathbf{z}_{j,k}^\top] \mathbf{H}_j^\top \mathbf{K}_k^\top - \mathbf{K}_k \mathbf{H}_j E[\delta\mathbf{z}_{j,k} \delta\mathbf{z}_k^\top] - E[\delta\mathbf{z}_k^- \mathbf{v}_j^\top] \mathbf{K}_k^\top - \mathbf{K}_k E[\mathbf{v}_j \delta\mathbf{z}_k^\top] \\ &\quad + \mathbf{K}_k \mathbf{H}_j E[\delta\mathbf{z}_{j,k} \mathbf{v}_j^\top] \mathbf{K}_k^\top + \mathbf{K}_k E[\mathbf{v}_j \delta\mathbf{z}_{j,k}^\top] \mathbf{H}_j^\top \mathbf{K}_k^\top\end{aligned}\quad (4.119)$$

Since the measurement noise \mathbf{v}_j is not time correlated and the measurement $\tilde{\mathbf{y}}_j$ has not yet been processed and has thus not influenced the a priori state estimate error $\delta\mathbf{z}_k^-$ at the current time t_k , $\delta\mathbf{z}_k^-$ as well as $\delta\mathbf{z}_{j,k}$ are not correlated with the measurement noise \mathbf{v}_j , meaning

$$\mathbb{E}[\delta\mathbf{z}_k^- \mathbf{v}_j^\top] = 0, \quad \mathbb{E}[\delta\mathbf{z}_{j,k} \mathbf{v}_j^\top] = 0 \quad (4.120)$$

The covariance matrix (4.119) simplifies to

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbb{E}[\delta\mathbf{z}_k^- \delta\mathbf{z}_{j,k}^\top] \mathbf{H}_j^\top \mathbf{K}_k^\top - \mathbf{K}_k \mathbf{H}_j \mathbb{E}[\delta\mathbf{z}_{j,k} \delta\mathbf{z}_k^{-\top}] + \mathbf{K}_k (\mathbf{R}_j + \mathbf{H}_j \mathbf{P}_{j,k} \mathbf{H}_j^\top) \mathbf{K}_k^\top \quad (4.121)$$

The cross-covariance between the current a priori state estimate error $\delta\mathbf{z}_k^-$ and the error of the state estimate $\delta\mathbf{z}_{j,k}$ is defined as

$$\mathbf{P}_{jk}^- := \mathbb{E}[\delta\mathbf{z}_{j,k} \delta\mathbf{z}_k^{-\top}], \quad \mathbf{P}_{kj}^- := \mathbf{P}_{jk}^{-\top} \quad (4.122)$$

Note that the cross-covariance \mathbf{P}_{jk}^- is normally not zero because the state $\hat{\mathbf{z}}_{j,k}$ is either obtained by back-propagation of $\hat{\mathbf{z}}_k^-$ or by measurement prediction, which involves the difference between the state estimates $\hat{\mathbf{z}}_k^-$ and $\hat{\mathbf{z}}_j^+$ and is thus related to $\hat{\mathbf{z}}_k^-$. With the definition of the cross-covariance matrix \mathbf{P}_{jk}^- in (4.122) and the innovation covariance matrix $\mathbf{S}_{j,k}$ given in (4.114) the covariance update equation (4.121) becomes

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{P}_{kj}^- \mathbf{H}_j^\top \mathbf{K}_k^\top - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^- + \mathbf{K}_k \mathbf{S}_{j,k} \mathbf{K}_k^\top \quad (4.123)$$

This is Joseph's form of the covariance update equation. Next, the question arises how the Kalman gain has to be calculated in order to obtain an optimal result, that is minimum covariance matrix \mathbf{P}_k^+ after the update. Like in the derivation of the conventional Kalman filter, the trace of the updated covariance matrix \mathbf{P}_k^+ shall be minimized. The trace of \mathbf{P}_k^+ differentiated with respect to \mathbf{K}_k is

$$\frac{\partial \text{trace}(\mathbf{P}_k^+)}{\partial \mathbf{K}_k} = -2\mathbf{P}_{kj}^- \mathbf{H}_j^\top + 2\mathbf{K}_k \mathbf{S}_{j,k} \stackrel{!}{=} 0 \quad (4.124)$$

The optimal Kalman gain is then

$$\mathbf{K}_k = \mathbf{P}_{kj}^- \mathbf{H}_j^\top \mathbf{S}_{j,k}^{-1} \quad (4.125)$$

An alternative and shorter form of the covariance update equation (4.123) can be found when the Kalman gain (4.125) is inserted into (4.123)

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{P}_{kj}^- \mathbf{H}_j^\top \mathbf{S}_{j,k}^{-1} \mathbf{H}_j \mathbf{P}_{jk}^- - \mathbf{P}_{kj}^- \mathbf{H}_j^\top \mathbf{S}_{j,k}^{-1} \mathbf{H}_j \mathbf{P}_{jk}^- + \mathbf{P}_{kj}^- \mathbf{H}_j^\top \mathbf{S}_{j,k}^{-1} \underbrace{\mathbf{S}_{j,k} \mathbf{S}_{j,k}^{-1}}_{=I} \mathbf{H}_j \mathbf{P}_{jk}^- \quad (4.126)$$

Cancelling the double terms in (4.126) yields the shorter, but not symmetry conserving form of the covariance update equation

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^- \quad (4.127)$$

The Kalman filter update equations for delayed measurements are now readily derived and are recapitulated in Template 4-14. The Kalman filter propagation equations are not affected by the delayed processing of the measurement and are hence the same as in the CKF. The required inputs are

$$\hat{\mathbf{z}}_{j,k}, \quad \mathbf{P}_{j,k}, \quad \mathbf{P}_{jk}^- \quad (4.128)$$

The update procedure is the same for the state backward propagation, measurement prediction and history state update methods. The three methods merely differ in how the state estimate $\hat{\mathbf{z}}_{j,k}$ and the corresponding covariance $\mathbf{P}_{j,k}$ containing all information up to the current time t_k as well as the cross-covariance matrix \mathbf{P}_{jk} are determined.

Template 4-14: Kalman filter update for delayed measurements

$\mathbf{s}_{j,k} = \tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k}$	Innovation
$\mathbf{S}_{j,k} = \mathbf{R}_j + \mathbf{H}_j \mathbf{P}_{j,k} \mathbf{H}_j^T$	Innovation covariance
$\mathbf{K}_k = \mathbf{P}_{kj}^- \mathbf{H}_j^T \mathbf{S}_{j,k}^{-1}$	Kalman gain
$\hat{\mathbf{z}}_k^+ = \hat{\mathbf{z}}_k^- + \mathbf{K}_k \mathbf{s}_{j,k}$	Estimated state correction
$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^-$ $= \mathbf{P}_k^- - \mathbf{P}_{kj}^- \mathbf{H}_j^T \mathbf{K}_k^T - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^- + \mathbf{K}_k \mathbf{S}_{j,k} \mathbf{K}_k^T$	Estimated state error covariance correction (short and Joseph's form)

4.3.4 State Backward Propagation

The a priori state estimate $\hat{\mathbf{z}}_k^-$ and the corresponding covariance matrix \mathbf{P}_k^- are back-propagated from the current time t_k to the measurement time t_j to obtain the state estimate $\hat{\mathbf{z}}_{j,k}$ and covariance $\mathbf{P}_{j,k}$ that are relevant for forming the innovation $\mathbf{s}_{j,k}$ and $\mathbf{S}_{j,k}$ in (4.114). Additionally, the cross-covariance \mathbf{P}_{jk}^- between the back-propagated state $\hat{\mathbf{z}}_{j,k}$ and the current state estimate $\hat{\mathbf{z}}_k^-$ is required. The method has been elaborated in detail by Bar-Shalom et al. for usage in tracking filters [106]. The derivation is given in section C.3.1 in the appendix. The method is summarized in Template 4-15.

Template 4-15: State backward propagation method

Initialization (at time $t = t_j$)	
$\mathbf{P}_{jj}^- \mathbf{H}_j^T = \mathbf{P}_j^- \mathbf{H}_j^T$	State error cross-covariance initialization
$\Phi_{jj} = \mathbf{I}_n, \quad \Gamma_{jj} = \mathbf{0}, \quad \bar{\Gamma}_{jj} = \mathbf{0}$	Transition and input matrices initialization
Propagation (at time $t = t_i$)	
$\mathbf{P}_{ij}^- \mathbf{H}_j^T = \Phi_{i-1} \mathbf{P}_{(i-1)j}^+ \mathbf{H}_j^T$	State error cross-covariance propagation
$\Phi_{ij} = \Phi_{i-1} \Phi_{(i-1)j}$	Transition matrix propagation
$\Gamma_{ij} = \Phi_{i-1} \Gamma_{(i-1)j} + \Gamma_{i-1}$	Input matrix propagation
$\bar{\Gamma}_{ij} = \Phi_{i-1} \bar{\Gamma}_{(i-1)j}^+ + \Gamma_{i-1}$	Updated input matrix propagation
Interim update (other measurement at time $t = t_i$)	
$\mathbf{P}_{ij}^+ \mathbf{H}_j^T = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^- \mathbf{H}_j^T$	State error cross-covariance update
$\bar{\Gamma}_{ij}^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \bar{\Gamma}_{ij}^-$	Input matrix update
Final update (at arrival time $t = t_k$)	
$\hat{\mathbf{z}}_{j,k} = \Phi_{kj}^{-1} \hat{\mathbf{z}}_k^-$	State estimate back-propagation
$\mathbf{P}_{j,k} = \Phi_{kj}^{-1} (\mathbf{P}_k^- + \Gamma_{kj} \mathbf{Q}_j \Gamma_{kj}^T - \Gamma_{kj} \mathbf{Q}_j \bar{\Gamma}_{kj}^{-T} - \bar{\Gamma}_{kj} \mathbf{Q}_j \Gamma_{kj}^T) \Phi_{kj}^{-T}$	Covariance back-propagation
\rightarrow Template "Kalman filter update for delayed measurements"	

4.3.5 Measurement Prediction

Alternatively, instead of propagating the state estimate back from the current time t_k to the measurement time t_j , the measurement $\tilde{\mathbf{y}}_j$ can be predicted from the measurement time t_j to the current time t_k in order to find a relation between the measurement and the newest state estimate $\hat{\mathbf{z}}_k^-$. A sketch of the derivation of the propagation and update equations of the measurement prediction method can be found in section C.3.2 in the appendix.

The computation of $\mathbf{P}_{kj}^- \mathbf{H}_j^T$ with the measurement prediction method for the case when the measurement validity time is already known is recapitulated in Template 4-16. The final update when the measurement value is available is done with the Kalman filter update step for delayed measurements according to Template 4-14.

Template 4-16: Measurement prediction method

Initialization (at measurement time $t = t_j$)	
$\mathbf{P}_{jj}^- \mathbf{H}_j^T = \mathbf{P}_j^- \mathbf{H}_j^T$	State error cross-covariance initialization
Propagation (at time $t = t_i$)	
$\mathbf{P}_{ij}^- \mathbf{H}_j^T = \Phi_{i-1} \mathbf{P}_{(i-1)j}^+ \mathbf{H}_j^T$	State error cross-covariance propagation
Interim update (other measurement at time $t = t_i$)	
$\mathbf{P}_{ij}^+ \mathbf{H}_j^T = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^- \mathbf{H}_j^T$	State error cross-covariance update
Final update (at arrival time $t = t_k$)	
$\hat{\mathbf{z}}_{j,k}^- = \hat{\mathbf{z}}_j^-$ $\mathbf{P}_{j,k}^- = \mathbf{P}_j^-$	→ Template "Kalman Filter Update for Delayed Measurements"

4.3.6 History State Update

As has been revealed in section C.3.2 in the appendix, the measurement prediction approach simply takes the old state and covariance estimates for the innovation of the measurement $\tilde{\mathbf{y}}_j$ without considering the corrections of newer measurements later than the measurement at time t_j . It has been discussed that the method is statistically not correct, predominantly when the filter is not stationary. It would be better to update the old state and covariance estimates simultaneously to the current state estimate.

It will be seen that the history state update method is especially good for the usage in an embedded system because of the distributed computational effort. In the state backward propagation method many computations have to be carried out in the instant when the measurement is available. In the history state update method the state and covariance that will be used for the innovation of the measurement $\tilde{\mathbf{y}}_j$ are continuously updated whenever new information from other sensors is available. The basic idea of the history state update method is to augment the state vector by a replica of itself in the moment when a new measurement is signaled or polled. In the ensuing waiting period, this augmented part of the state vector is not propagated but only updated by the innovation of the newer measurements. As soon as the sensed value of the measurement $\tilde{\mathbf{y}}_j$ is actually available, the

continuously updated history state and corresponding covariance can be directly used to compute the innovation of $\tilde{\mathbf{y}}_j$.

At the instant of the new measurement the current state estimate vector $\hat{\mathbf{z}}_j^-$ is temporarily augmented by a copy of itself, $\hat{\mathbf{z}}_{j,j}^- = \hat{\mathbf{z}}_j^-$, yielding a vector with twice the number of states

$$\begin{pmatrix} \hat{\mathbf{z}}_j^- \\ \hat{\mathbf{z}}_{j,j}^- \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{z}}_j^- \\ \hat{\mathbf{z}}_j^- \end{pmatrix} \quad (4.129)$$

The same is done with the state estimate error covariance matrix. The current covariance matrix \mathbf{P}_j^- is extended by a copy of itself, $\mathbf{P}_{j,j}^- = \mathbf{P}_j^-$, giving

$$\begin{pmatrix} \mathbf{P}_j^- & \mathbf{P}_{jj}^- \\ \mathbf{P}_{jj}^- & \mathbf{P}_{j,j}^- \end{pmatrix} = \begin{pmatrix} \mathbf{P}_j^- & \mathbf{P}_j^- \\ \mathbf{P}_j^- & \mathbf{P}_j^- \end{pmatrix} \quad (4.130)$$

The state estimate $\hat{\mathbf{z}}_j^-$ and the history state $\hat{\mathbf{z}}_{j,j}^-$ are equal and thus fully correlated. Therefore, the off-diagonal matrices that describe the cross-covariance between the state estimate and the history state estimate are also filled with the state estimate error covariance matrix, $\mathbf{P}_{jj}^- = \mathbf{P}_j^-$.

Henceforth, the history state vector augmentation is retained and is removed not before the actual sensor value has been processed by the filter. In the meantime, the augmented state vector and the corresponding covariance matrix are propagated and updated by other newer measurements. All information that is collected by the filter until the current time t_k is likewise used to correct the current state estimate as well as the old history state at time t_j . The history state for the innovation is thus automatically kept up-to-date. The propagation and update equations of the history state update method are derived in detail in section C.3.3 in the appendix. The history state update method is summarized in Template 4-17. In contrast to the measurement prediction method of Template 4-16, the history state $\hat{\mathbf{z}}_{j,i}$ and covariance $\mathbf{P}_{j,i}$ are additionally updated at an interim update step.

Template 4-17: History state update method

Initialization (at measurement time $t = t_j$)

$\hat{\mathbf{z}}_{j,j}^- = \hat{\mathbf{z}}_j^-$	History state initialization
$\mathbf{P}_{j,j}^- = \mathbf{P}_j^-$	History state covariance initialization
$\mathbf{P}_{jj}^- = \mathbf{P}_j^-$	State error cross-covariance initialization

Propagation (at time $t = t_i$)

$$\mathbf{P}_{jj}^- = \Phi_{i-1} \mathbf{P}_{(i-1)j}^+ \quad \text{State error cross-covariance propagation}$$

Interim update (other measurement at time $t = t_i$)

$\mathbf{P}_{jj}^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{jj}^-$	State error cross-covariance update
$= (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{jj}^- - \mathbf{P}_i^- \mathbf{H}_i^\top \mathbf{K}_i^{*\top} + \mathbf{K}_i \mathbf{S}_i \mathbf{K}_i^{*\top}$	
$\mathbf{K}_i^* = \mathbf{P}_{ji}^- \mathbf{H}_i^\top \mathbf{S}_i^{-1}$	History state Kalman gain
$\hat{\mathbf{z}}_{j,i} = \hat{\mathbf{z}}_{j,i-1} + \mathbf{K}_i^* \mathbf{s}_i$	History state update
$\mathbf{P}_{j,i} = \mathbf{P}_{j,i-1} - \mathbf{K}_i^* \mathbf{H}_i \mathbf{P}_{jj}^-$	History state covariance update
$= \mathbf{P}_{j,i-1} - \mathbf{K}_i^* \mathbf{H}_i \mathbf{P}_{jj}^- - \mathbf{P}_{ji}^- \mathbf{H}_i^\top \mathbf{K}_i^{*\top} + \mathbf{K}_i^* \mathbf{S}_i \mathbf{K}_i^{*\top}$	

Final update (at arrival time $t = t_k$)

→ Template "Kalman filter update for delayed measurements"

4.3.7 Comparison of the Three Methods

In Table 4-2, all three methods are arranged side by side to emphasize the commonalities and the differences. The common parts are black whereas the differences are in red. All methods have in common that they are initialized as soon as a new measurement is indicated to the navigation filter and that thereupon the state estimate and covariance are periodically propagated and updated whenever interim measurements are available. Finally, the state estimate and covariance are updated with the actual delayed measurement value. The propagation, interim update and final update of the state $\hat{\mathbf{z}}$ and covariance \mathbf{P} are the same in all three cases. Even the initialization, propagation and interim update of the cross-covariance \mathbf{P}_{ij} are equal for all methods. Merely in the cases of the state backward propagation and measurement prediction it is possible to calculate only the smaller matrix $\mathbf{P}_{ij} \mathbf{H}_j^T$ instead of the matrix \mathbf{P}_{ij} itself, which saves some computational effort but implies that \mathbf{H}_j is already available and does not depend on the current state estimate. The single difference between the measurement prediction and the history state update methods is that the history state estimate and the corresponding covariance are updated by interim measurements in the state history update method in contrast to the measurement prediction method. If no interim updates occur, both methods are equal. In the state backward propagation method the history state and covariance can be calculated not before the actual measurement value has been arrived. However, the required transition matrices Φ_{kj} , Γ_{kj} and $\bar{\Gamma}_{kj}$ can be initialized and already propagated and updated in the meantime between the measurement validity time t_j and the measurement delivery time t_k .

If a new measurement is notified beforehand, the required computations can be well distributed over the waiting timespan, which is advantageous for the implementation on embedded systems. If the timestamp and measurement value pair arrive without preceding notification, all computations have to be caught up in the moment of the data arrival, which causes high computational load. Therefore, if technically realizable, measurement notification together with the history state update method are recommended.

In Figure 4-31 the preferred processing sequence is illustrated. The upper timeline represents the propagation and update of the core navigation state estimate (thin arrows) and the corresponding state error covariance (thick arrows). The thin black vertical lines visualize the state propagation time steps and the thick black vertical lines indicate the covariance propagation time steps. As soon as the new measurement is indicated (yellow arrow), the state estimate vector and the state error covariance are temporarily copied to the lower timeline and the state error cross-covariance matrix is created. Since the new measurement can be taken at any arbitrary time between two subsequent propagation steps, the last state and covariance may have to be separately propagated from the last propagation time to the validity time of the new measurement. In the sequel, the temporary state, covariance and cross-covariance matrix are held until the actual measurement value is available. The latter is propagated together with the core covariance. Whenever interim measurements are available, the innovation is applied on the one hand to the core state and covariance in the upper timeline and on the other hand to the temporary history state, covariance and cross-covariance in the lower timeline. Updates are indicated by the vertical green lines and the green check marks. If interim updates are pending, the state and covariance propagation time step sizes are accordingly reduced to provide the current estimates at the validity time of the interim measurement for the innovation. In this manner one or more interim measurements can be processed in the waiting time for the delayed measurement value. Finally, when the value has been arrived, the temporary state, covariance and cross-covariance are used to form the statistically correct innovation to be applied to the core state estimate and covariance.

This can be done immediately at the arrival of the sensor data or not before the next regular covariance propagation step. After the final update the lower temporary timeline is closed again until the next delayed measurement is signalized.

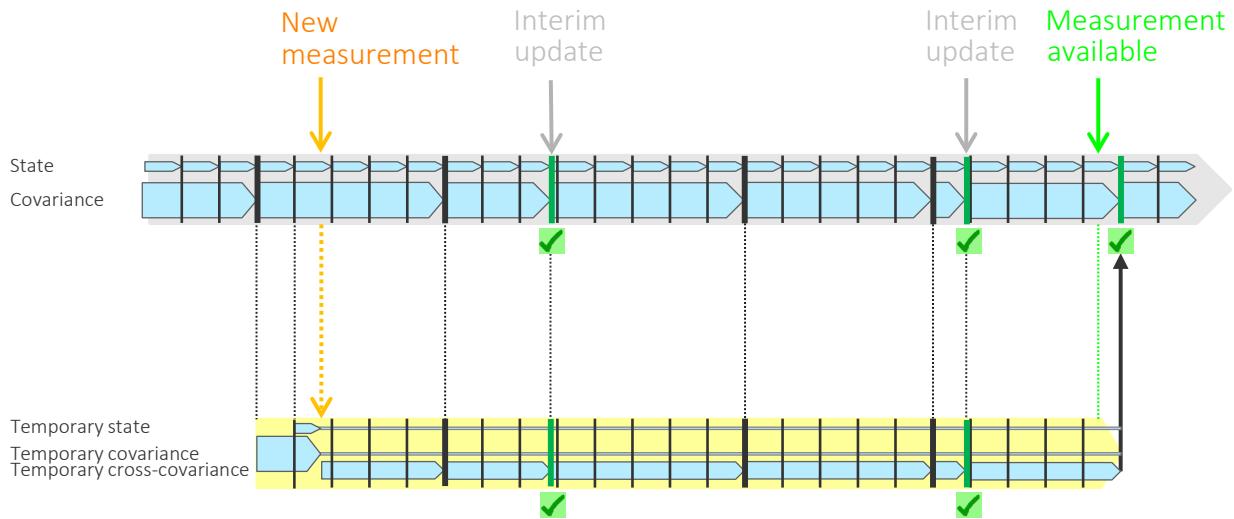


Figure 4-31: Processing sequence of a delayed measurement with additional interim updates

4.3.8 Stability Problem

A subtleness concerning the stability of the covariance propagation and update, which is inherent to all presented methods for delayed measurements, has to be addressed. Since \mathbf{P}_{jk}^- is used in the final covariance update $\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^-$, but the transposed covariance \mathbf{P}_{kj}^- is used in the according Kalman gain equation $\mathbf{K}_k = \mathbf{P}_{kj}^- \mathbf{H}_j^\top (\mathbf{R}_j + \mathbf{H}_j \mathbf{P}_{j,k}^- \mathbf{H}_j^\top)^{-1}$, the covariance propagation and update are numerically unstable, meaning that numerical errors are not damped but enhanced. This may corrupt the positive definiteness requirement of covariance matrices and lead to the collapse of the covariance solution. The reason for that will be explained later in section 5.2.7. Numerical stabilization or the use of Joseph's form are mandatory.

4.3.9 Processing more than one Delayed Measurement

In an integrated navigation system measurements of various sensors are usually processed. There may be more than one sensor that provide their sensed values only with appreciable delay. In principal, sensors can take and deliver measurements concurrently and independently of each other. The question arises how the filter can appropriately handle situations where two or more measurements are simultaneously delayed. The equations of the filter with more than one delayed measurement are summarized in Template 4-18. In the template it is assumed that the measurement of sensor 1 is taken at time t_j . At this instant the filter already waits for the delayed measurements of the sensors $\{2, \dots, M\}$. In contrast to the Template 4-17 for a single delayed measurement, the cross-covariances between the history state of sensor 1 and the history states of the other delayed sensors have to be initialized. At interim updates of uninvolved measurements, the cross-covariances between the currently installed history states $\{1, \dots, M\}$ have to be additionally updated. At the final update, when the measurement value of sensor 1 arrives, the cross-covariances between the history states of the remaining $\{2, \dots, M\}$ sensors have to be accordingly updated for use in their upcoming final updates.

Table 4-2: Comparison of the three methods for the processing of delayed measurements

	State backward propagation	Measurement prediction	History state update
Initialization (at time $t = t_j$)	$\mathbf{P}_{jj}^+ = \mathbf{P}_j^+$ $\Phi_{ii} = \mathbf{I}_n, \quad \Gamma_{ii} = \mathbf{0}, \quad \bar{\Gamma}_{ii} = \mathbf{0}$	$\hat{\mathbf{z}}_j = \hat{\mathbf{z}}_j^+, \quad \mathbf{P}_j = \mathbf{P}_j^+, \quad \mathbf{P}_{jj}^+ = \mathbf{P}_j^+$	$\hat{\mathbf{z}}_{j,j} = \hat{\mathbf{z}}_j^+, \quad \mathbf{P}_{j,j} = \mathbf{P}_j^+, \quad \mathbf{P}_{jj}^+ = \mathbf{P}_j^+$
Propagation (at time $t = t_i$)	$\hat{\mathbf{z}}_i^- = \Phi_{i-1} \hat{\mathbf{z}}_{i-1}^+$ $\mathbf{P}_i^- = \Phi_{i-1} \mathbf{P}_{i-1}^+ \Phi_{i-1}^\top + \Gamma_{i-1} \mathbf{Q}_{i-1} \Gamma_{i-1}^\top$ $\mathbf{P}_{ij}^- \mathbf{H}_j^\top = \Phi_{i-1} \mathbf{P}_{(i-1)}^+ \mathbf{H}_j^\top$ $\Phi_{ij} = \Phi_{i-1} \Phi_{(i-1)j}, \quad \Gamma_{ij} = \Phi_{i-1} \Gamma_{(i-1)j} + \Gamma_{i-1}$ $\bar{\Gamma}_{ij} = \Phi_{i-1} \bar{\Gamma}_{(i-1)j} + \Gamma_{i-1}$	$\hat{\mathbf{z}}_i^- = \Phi_{i-1} \hat{\mathbf{z}}_{i-1}^+$ $\mathbf{P}_i^- = \Phi_{i-1} \mathbf{P}_{i-1}^+ \Phi_{i-1}^\top + \Gamma_{i-1} \mathbf{Q}_{i-1} \Gamma_{i-1}^\top$ $\mathbf{P}_{ij}^- \mathbf{H}_j^\top = \Phi_{i-1} \mathbf{P}_{(i-1)j}^+ \mathbf{H}_j^\top$ $\Phi_{ij} = \Phi_{i-1} \Phi_{(i-1)j}, \quad \Gamma_{ij} = \Phi_{i-1} \Gamma_{(i-1)j} + \Gamma_{i-1}$ $\bar{\Gamma}_{ij} = \Phi_{i-1} \bar{\Gamma}_{(i-1)j} + \Gamma_{i-1}$	$\hat{\mathbf{z}}_i^- = \Phi_{i-1} \hat{\mathbf{z}}_{i-1}^+$ $\mathbf{P}_i^- = \Phi_{i-1} \mathbf{P}_{i-1}^+ \Phi_{i-1}^\top + \Gamma_{i-1} \mathbf{Q}_{i-1} \Gamma_{i-1}^\top$ $\mathbf{P}_{ij}^- = \Phi_{i-1} \mathbf{P}_{(i-1)j}^+$
Interim update (other / no measurement available at time $t = t_i$)	$\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}_i^\top (\mathbf{R}_i + \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^\top)^{-1} / \mathbf{K}_i = 0$ $\hat{\mathbf{z}}_i^+ = \hat{\mathbf{z}}_i^- + \mathbf{K}_i (\tilde{\mathbf{y}}_i - \mathbf{H}_i \hat{\mathbf{z}}_i^-)$ $\mathbf{P}_i^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_i^-$ $\mathbf{P}_{ij}^+ \mathbf{H}_j^\top = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^- \mathbf{H}_j^\top$ $\bar{\Gamma}_{ij}^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \bar{\Gamma}_{ij}^-$	$\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}_i^\top (\mathbf{R}_i + \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^\top)^{-1} / \mathbf{K}_i = 0$ $\hat{\mathbf{z}}_i^+ = \hat{\mathbf{z}}_i^- + \mathbf{K}_i (\tilde{\mathbf{y}}_i - \mathbf{H}_i \hat{\mathbf{z}}_i^-)$ $\mathbf{P}_i^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_i^-$ $\mathbf{P}_{ij}^+ \mathbf{H}_j^\top = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^- \mathbf{H}_j^\top$ $\bar{\Gamma}_{ij}^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \bar{\Gamma}_{ij}^-$	$\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}_i^\top (\mathbf{R}_i + \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^\top)^{-1} / \mathbf{K}_i = 0$ $\hat{\mathbf{z}}_i^+ = \hat{\mathbf{z}}_i^- + \mathbf{K}_i (\tilde{\mathbf{y}}_i - \mathbf{H}_i \hat{\mathbf{z}}_i^-)$ $\mathbf{P}_{ij}^+ = \mathbf{P}_{ij}^- - \mathbf{K}_i \mathbf{H}_i \mathbf{P}_{ij}^-$
Final update (measurement available at time $t = t_k$)	$\hat{\mathbf{z}}_{j,k} = \Phi_{kj}^{-1} \hat{\mathbf{z}}_k^-$ $\mathbf{P}_{j,k} = \Phi_{kj}^{-1} (\mathbf{P}_k^- + \Gamma_{kj} \mathbf{Q}_j \Gamma_{kj}^\top - \Gamma_{kj} \mathbf{Q}_j \bar{\Gamma}_{kj}^\top - \bar{\Gamma}_{kj} \mathbf{Q}_j \Gamma_{kj}^\top) \Phi_{kj}^{-\top}$	$\mathbf{K}_k = \mathbf{P}_{kj}^- \mathbf{H}_j^\top (\mathbf{R}_j + \mathbf{H}_j \mathbf{P}_j \mathbf{H}_j^\top)^{-1}$ $\hat{\mathbf{z}}_k^- = \hat{\mathbf{z}}_k^- + \mathbf{K}_k (\tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_j^-)$ $\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^-$	$\mathbf{K}_k = \mathbf{P}_{kj}^- \mathbf{H}_j^\top (\mathbf{R}_j + \mathbf{H}_j \mathbf{P}_j \mathbf{H}_j^\top)^{-1}$ $\hat{\mathbf{z}}_k^- = \hat{\mathbf{z}}_k^- + \mathbf{K}_k (\tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_j^-)$ $\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^-$

Template 4-18: History state update method with more than one delayed measurement

Initialization (at measurement time $t = t_j$)		
$\hat{\mathbf{z}}_{j,j}^- = \hat{\mathbf{z}}_j^-$		History state initialization
$\mathbf{P}_{j,j}^- = \mathbf{P}_j^-$		History state covariance initialization
$\mathbf{P}_{jj}^- = \mathbf{P}_j^-$		State error cross-covariance initialization
$\mathbf{P}_{j,j_m,j}^- = \mathbf{P}_{j_m,j}^- , \quad m = \{2, \dots, M\}$		History state cross-covariances initialization
Propagation (at time $t = t_i$)		
$\mathbf{P}_{ij_m}^- = \Phi_{i-1} \mathbf{P}_{(i-1)j_m}^+ , \quad m = \{1, \dots, M\}$		State error cross-covariances propagation
Interim update (other measurement at time $t = t_i$)		
$\mathbf{P}_{ij_m}^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij_m}^- , \quad m = \{1, \dots, M\}$		State error cross-covariance update
$\mathbf{K}_{j_m,i}^* = \mathbf{P}_{ij_m}^{-T} \mathbf{H}_i^T \mathbf{S}_i^{-1}$		History state Kalman gain
$\hat{\mathbf{z}}_{j_m,i} = \hat{\mathbf{z}}_{j_m,i-1} + \mathbf{K}_i^* \mathbf{s}_i$		History state update
$\mathbf{P}_{j_m,i} = \mathbf{P}_{j_m,i-1} - \mathbf{K}_{j_m,i}^* \mathbf{H}_i \mathbf{P}_{ij_m}^-$		History state covariance update
$\mathbf{P}_{j_m j_l, i} = \mathbf{P}_{j_m j_l, i-1} - \mathbf{K}_{j_m,i}^* \mathbf{H}_i \mathbf{P}_{ij_l}^- , \quad l = \{1, \dots, M\} \setminus \{m\}$		History state cross-covariances update
Final update (at arrival time $t = t_k$)		
→ Template "Kalman Filter Update for Delayed Measurements"		
$\mathbf{P}_{kj_m}^+ = \mathbf{P}_{kj_m}^- - \mathbf{K}_k \mathbf{H}_{j_1} \mathbf{P}_{j_l j_m, k-1} , \quad m = \{2, \dots, M\}$		State error cross-covariance update
$\mathbf{K}_{j_m,k}^* = \mathbf{P}_{j_m j_1, k-1}^{-T} \mathbf{H}_{j_1}^T \mathbf{S}_k^{-1}$		History state Kalman gain
$\hat{\mathbf{z}}_{j_m,k} = \hat{\mathbf{z}}_{j_m,k-1} + \mathbf{K}_{j_m,k}^* \mathbf{s}_k$		History state update
$\mathbf{P}_{j_m,k} = \mathbf{P}_{j_m,k-1} - \mathbf{K}_{j_m,k}^* \mathbf{H}_{j_1} \mathbf{P}_{j_l j_m, k-1}$		History state covariance update

4.3.10 Example

The Generic Lissajous scenario (section E.1.1) is used as example to demonstrate the measurement prediction and history state update methods. IMU 1 and IMU 2 (Table E-5) with 100 Hz sample rates are compared. The IMU measurements are solely affected by velocity/angular random walk noise N and turn-on biases. All other IMU errors are set to zero. A generic GPS receiver provides pseudorange measurements to six satellites in view (section E.2.2). Additionally, barometric height measurements are available. Both the pseudorange and barometric height measurements are only falsified by white Gaussian noise. The standard deviations of the measurement noise vary with time as shown in Figure 4-33. The filter performance is analyzed by comparing the predicted covariance with the covariance of the actually realized navigation state error of 150 Monte Carlo runs. The IMU measurements are not delayed. The IMU sample rate serves as base rate of the navigation system. The pseudoranges and barometric height are provided with the same rate. Three different rates will be compared, that is 0.2 Hz, 0.5 Hz and 1 Hz. The pseudorange measurements are delayed by 0.97 s and 0.15 s, respectively, whereas the barometric height measurements are on time. The measurement sequence is illustrated in Figure 4-32. The non-delayed barometric height measurement values arrive in the waiting time for the GPS measurements, shortly, that is 0.01 s, before these become available. In the first case, the less accurate IMU 1 is used, the aiding rate is 0.2 Hz and the GPS delay is 0.97 s. Initially, the measurement prediction method is used without stability enhancement. The resulting position error is shown in

Figure 4-34. It can be observed that the solution becomes unstable at second 95, emphasizing the urgent necessity of additional stabilization, which will thus be applied henceforth. The position and orientation errors with stability enhancement are plotted in Figure 4-35 and Figure 4-36. The blue curves belong to the solution using neither measurement prediction (MP) nor history state update (HSU), the red curves to the solution using MP but not HSU and the green curves to the solution of the HSU method. The continuous lines are the 3σ standard deviations as predicted by the filter whereas as the dotted lines are the actual 3σ standard deviations as result of the Monte Carlo simulation. The following can be stated: without any method, the predicted and actual variances do not match. In fact, the actual variances are larger than the predicted, which is dangerous. This is especially true in the first settling period immediately after initialization when the variances change strongly with time. If the MP method is applied, the results become more consistent but still differ primarily in the settling period. Only with the HSU method, predicted and actual variances coincide well, which means that the filter provides a statistically consistent solution. The difference between predicted and actual covariances declines if the navigation state error covariance does not change significantly due to process noise or other intermediate updates between measurement validity and arrival times. The navigation error dynamics are usually low and decoupled from the platform dynamics. If the aiding measurement variances are essentially constant during a longer period, the navigation filter will reach steady-state and the posterior covariances will become virtually constant. The distribution and exchange of uncertainty amongst the different error states, which results in more extensive changes of the covariances with time, takes place mainly in the settling period shortly after initialization. The change of the covariance between validity and arrival time is reduced if

- the IMU process noise is decreased (by using a higher grade IMU)
- the aiding measurement delay is decreased
- the aiding measurement rate is increased

Furthermore, the covariance mismatch diminishes if the variance of the delayed aiding measurement is large and its corresponding innovation low because the measurement then influences the solution only little. The previous statements shall be proven by the next simulations. In Figure 4-37 and Figure 4-38, the GPS pseudorange rate has been increased to 0.5 Hz. The covariance mismatch is still observable but turns out to be smaller. In Figure 4-39 and Figure 4-40, the GPS pseudorange rate is further raised to 1 Hz. Only the solution without MP and HSU is shown (as will also be done in the subsequent figures). Compared to the previous cases, the difference between the predicted and actual variances is further reduced. Figure 4-41 and Figure 4-42 show the results if the GPS measurement delay is only 0.15 s instead of 0.97 s. In Figure 4-43 and Figure 4-44, the aiding measurement error variance has been increased to about 5 m. Finally, IMU 1 is replaced by the more accurate IMU 2. In Figure 4-45 and Figure 4-46, it can be seen that the reduced process noise covariance leads to an increased statistical consistence even without MP and HSU.

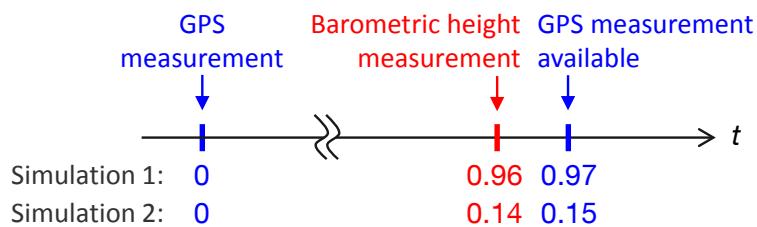


Figure 4-32: Measurement sequence

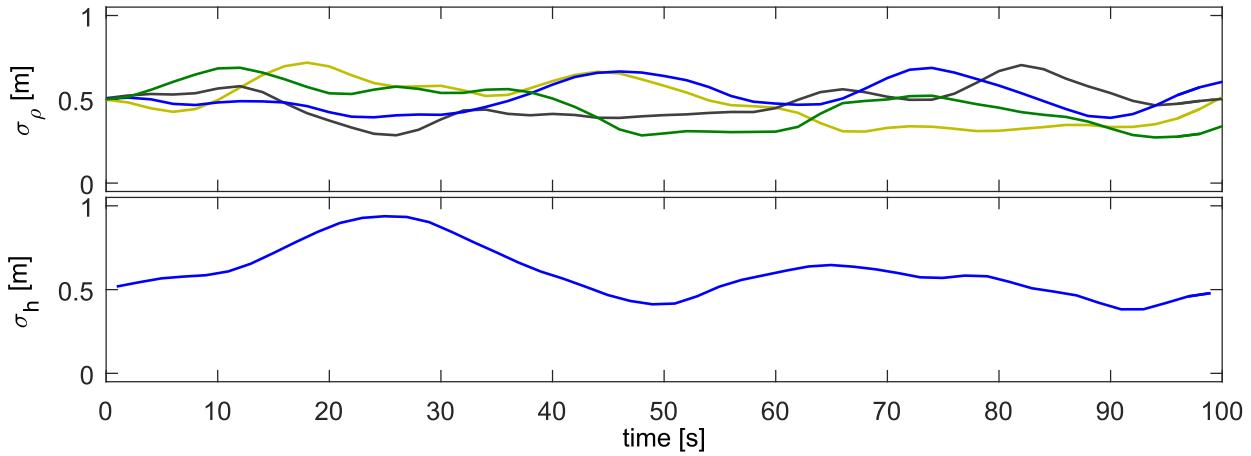


Figure 4-33: Variation of pseudorange and barometric height standard deviations with time

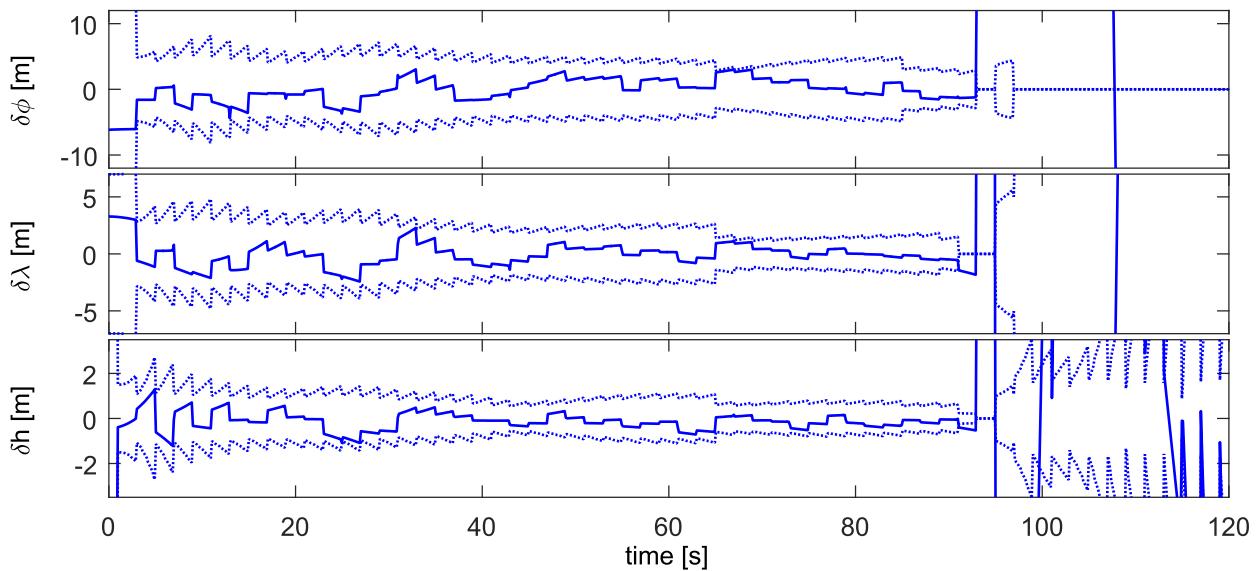


Figure 4-34: Instability of the position error without stability enhancement

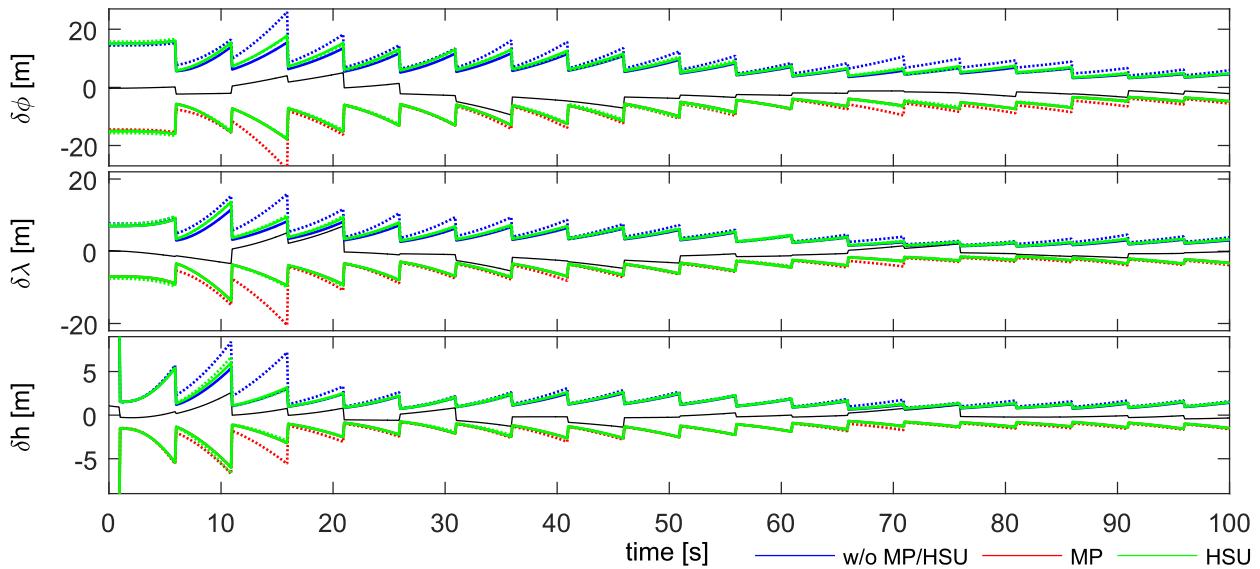


Figure 4-35: Position error (IMU 1, 0.2 Hz update rate, 0.97 s GPS delay)

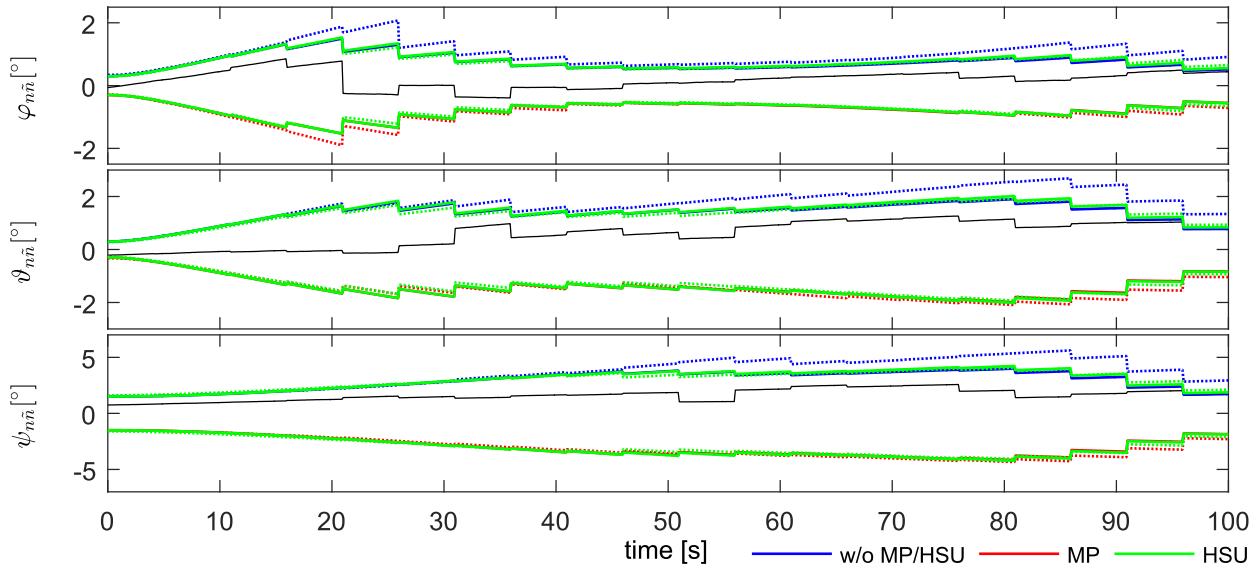


Figure 4-36: Orientation error (IMU 1, 0.2 Hz update rate, 0.97 s GPS delay)

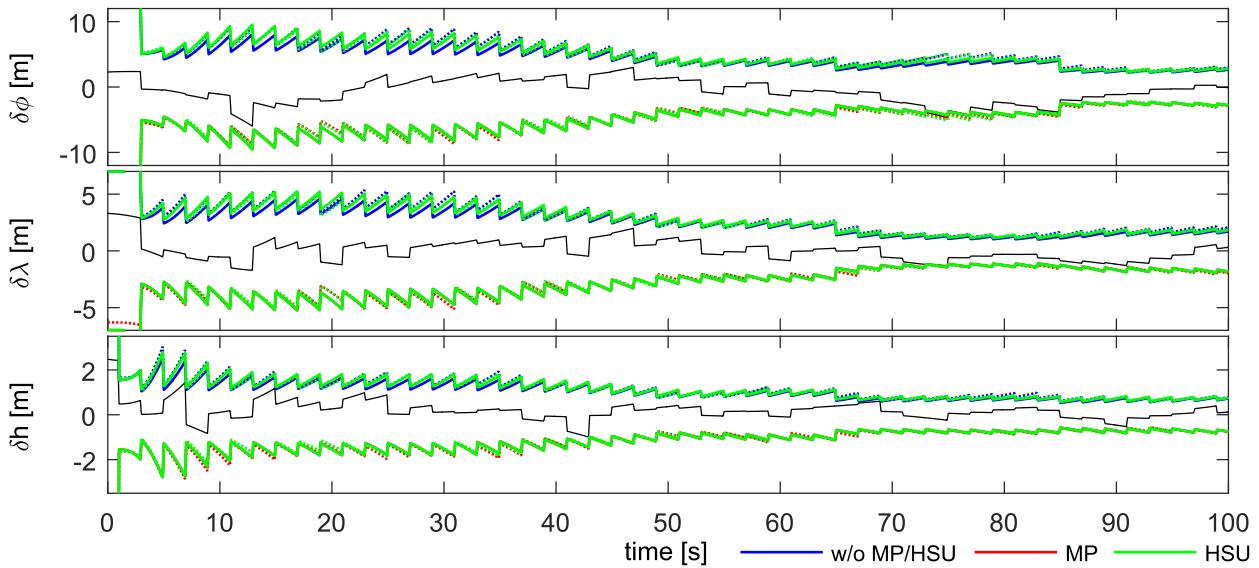


Figure 4-37: Position error (IMU 1, 0.5 Hz update rate, 0.97 s GPS delay)

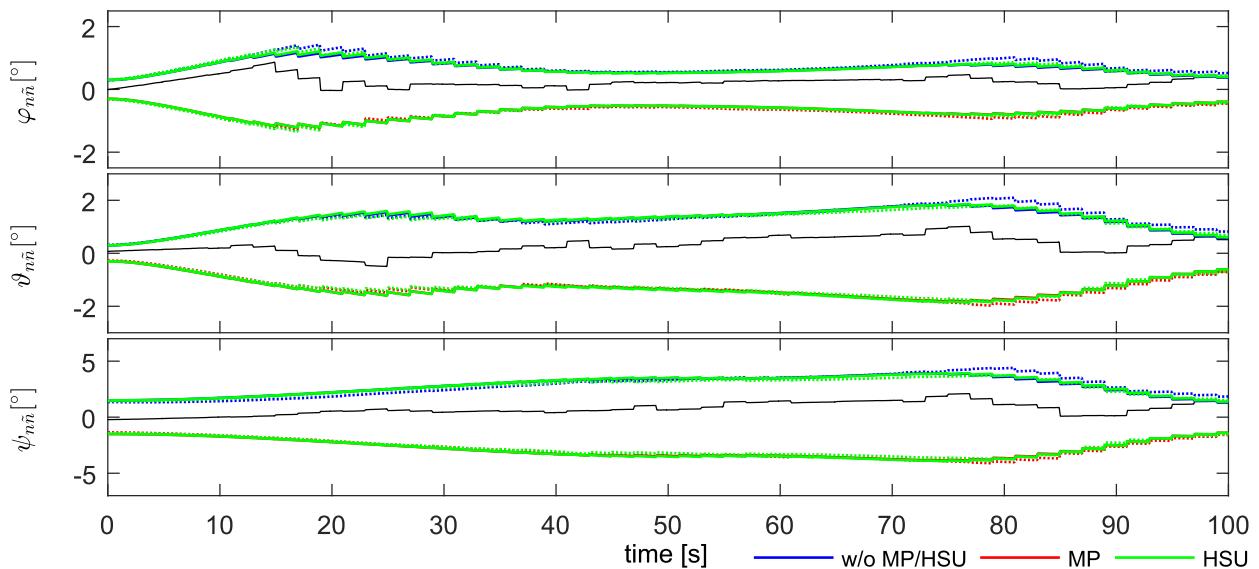


Figure 4-38: Orientation error (IMU 1, 0.5 Hz update rate, 0.97 s GPS delay)

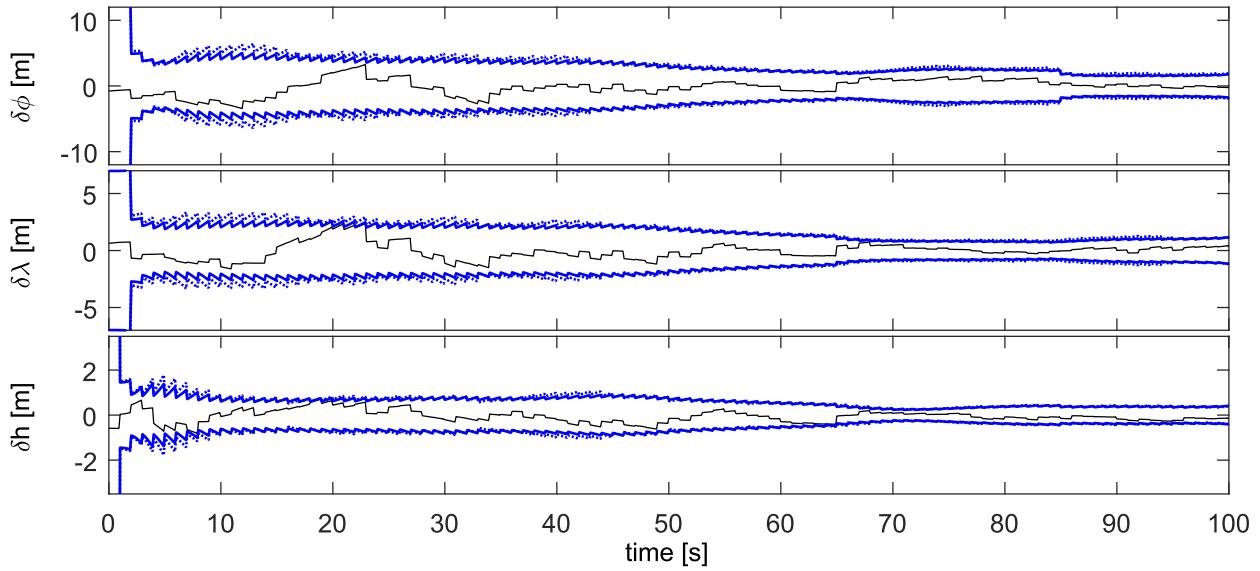


Figure 4-39: Position error (IMU 1, 1 Hz update rate, 0.97 s GPS delay)

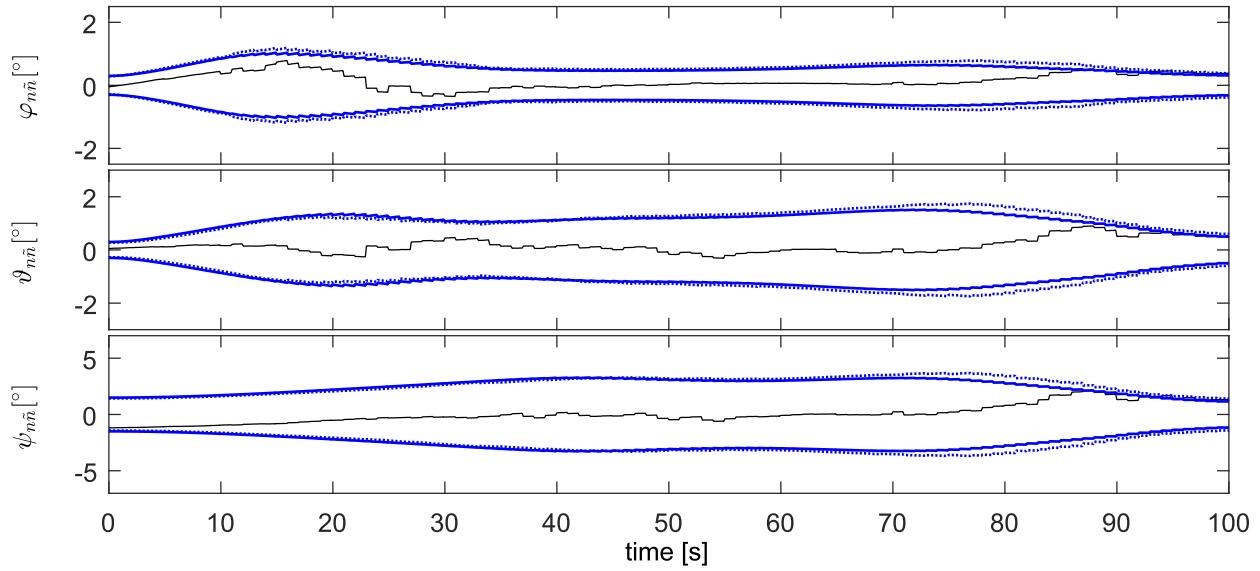


Figure 4-40: Orientation error (IMU 1, 1 Hz update rate, 0.97 s GPS delay)

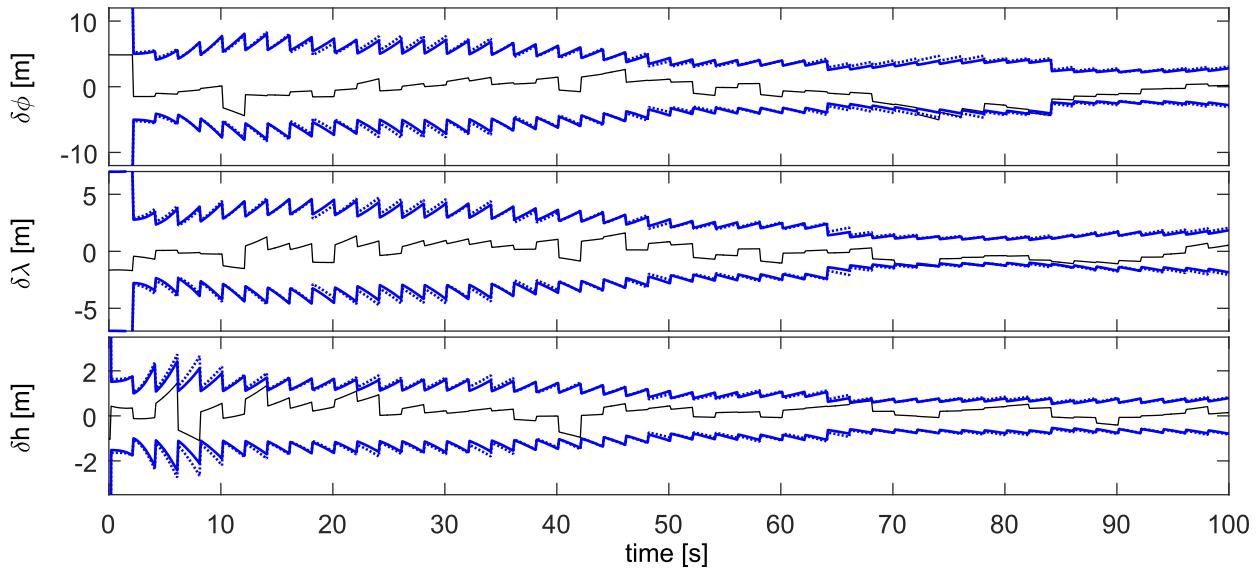


Figure 4-41: Position error (IMU 1, 0.5 Hz update rate, 0.15 s GPS delay)

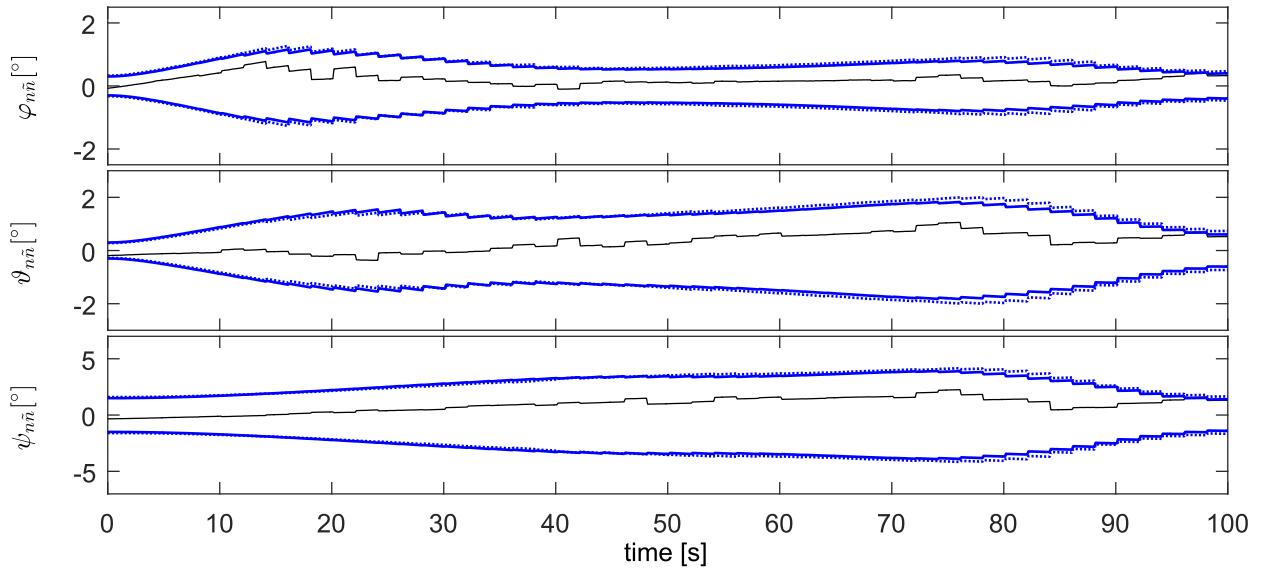


Figure 4-42: Orientation error (IMU 1, 0.5 Hz update rate, 0.15 s GPS delay)

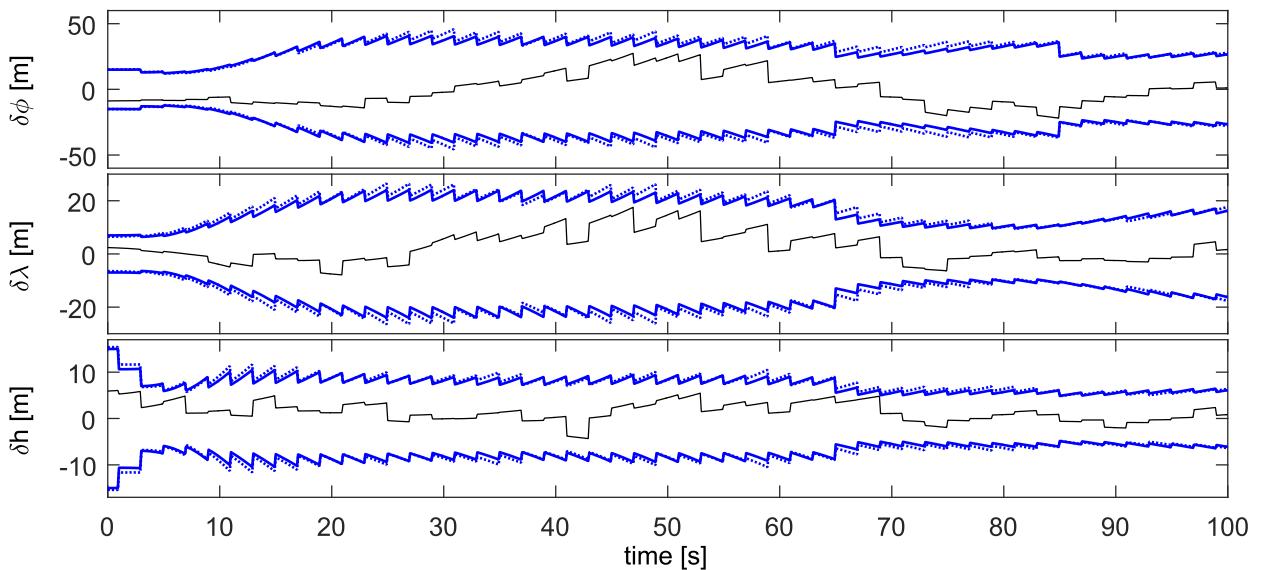


Figure 4-43: Position error (IMU 1, 0.5 Hz update rate, 0.97 s GPS delay, ≈5 m error)

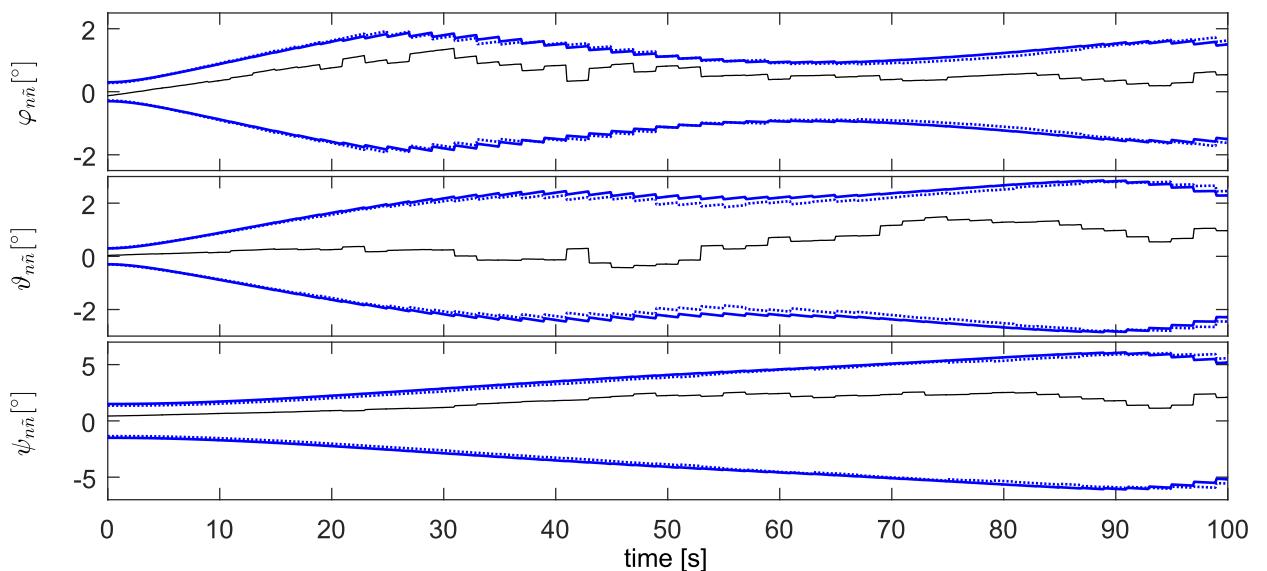


Figure 4-44: Orientation error (IMU 1, 0.5 Hz update rate, 0.97 s GPS delay, ≈5 m error)

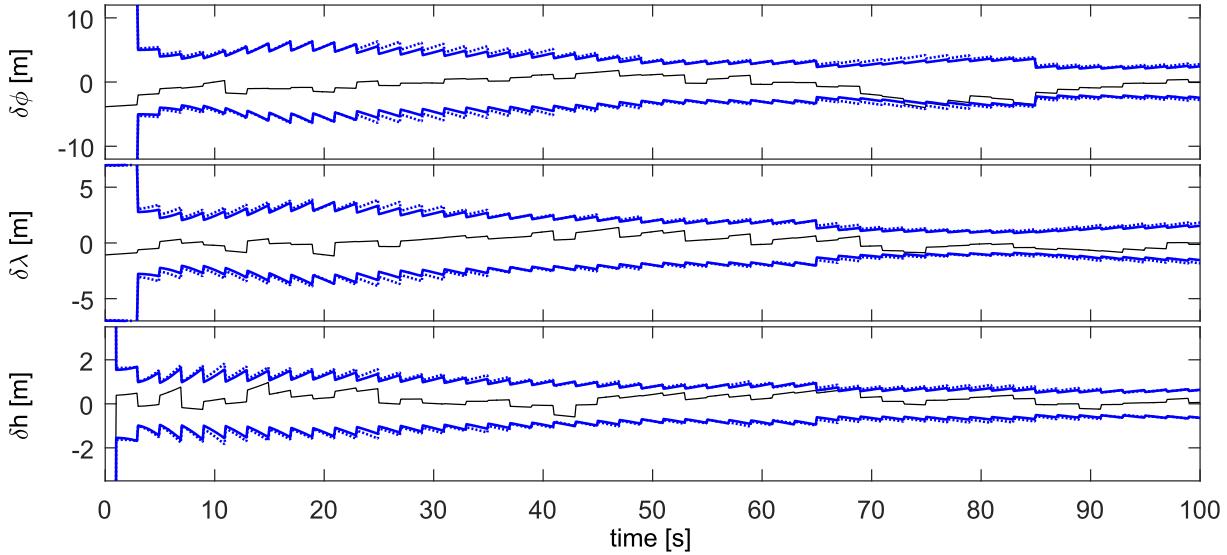


Figure 4-45: Position error (IMU 2, 0.5 Hz update rate, 0.97 s GPS delay)

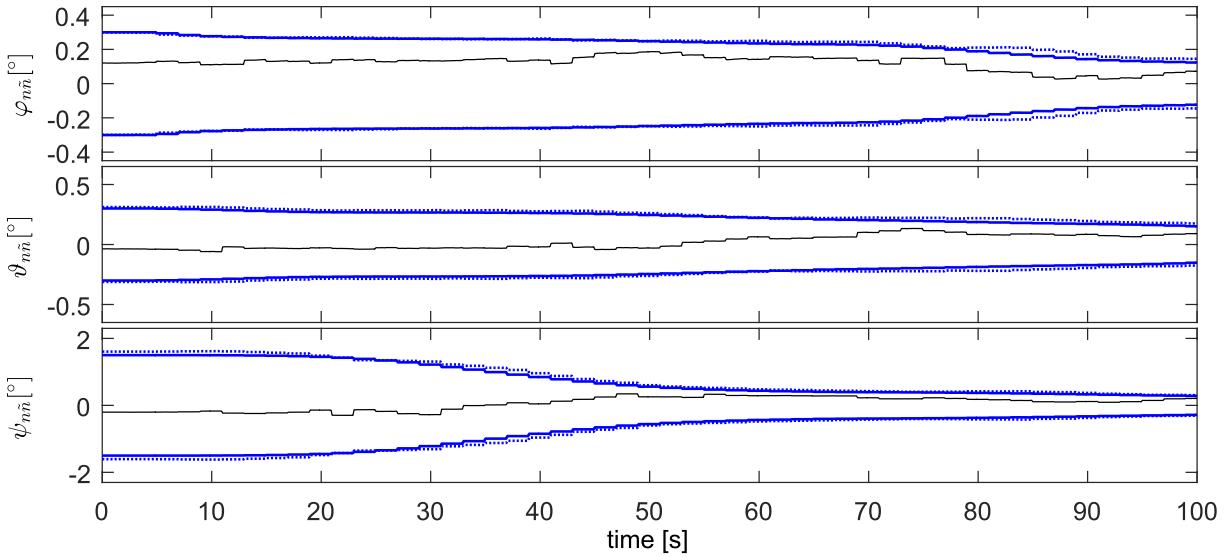


Figure 4-46: Orientation error (IMU 2, 0.5 Hz update rate, 0.97 s GPS delay)

4.3.11 Conclusion

The measurement prediction and history state update methods should be used if different aiding measurements with large delays and considerably changing variances are fused and the IMU is low grade with high measurement errors. The application of one of the methods is especially reasonable in the settling period after initialization. The history state update method outperforms the measurement prediction method. Therefore, the history state update method is favored over the measurement prediction method, particularly since the computational load is not much higher. The increased effort of the history state update method can be spared if the process noise is low (higher grade IMU), the aiding measurement delays are short and there are many different aiding measurements with preferentially high sampling rates and quasi-stationary measurement variances. If the measurement prediction or history state update methods are applied, then stability enhancement is compulsory since the methods are innately unstable. Since the measurement prediction and history state update methods can be easier implemented on embedded systems, these algorithms are preferred to the state backward propagation method.

4.4 Navigation Filter State Augmentation

4.4.1 Motivation

The CKF (2.11) presumes white and Gaussian distributed process and measurement noises. Optimality, meaning minimal traces of the state error covariance matrices, and statistical consistency are only achieved if these noise requirements are not violated.

As has been seen in the reality modeling chapter (chapter 3), the stochastic errors of measurements that are processed by an integrated navigation filter are hardly ever purely white but most often contain colored, that is time correlated components. Adequate models describing the main error characteristics have been derived in the reality modeling chapter for measurements and modeling residuals. All these models have in common that they filter white and Gaussian distributed input noise to get time correlated output noise with the desired PSD. All of them can be represented in state-space form and are thus well suited for the integration in a CKF based navigation error filter.

In this section, it is demonstrated how available information about nominal sensor errors, existing in form of dynamic stochastic models, can be incorporated into the integrated navigation filter.

4.4.2 Kalman Filter State Augmentation

The self-evident way to regard time correlated measurement errors in the filter is to augment the core navigation error model (2.6) or (2.7) for position, velocity and orientation with the identified models of these errors. The Kalman filter thus estimates the measurement error states in addition to the original navigation error states. It is for example common to expand the navigation error state vector with states for the turn-on biases and scale factor errors of the accelerometers and gyroscopes. The additionally estimated values are used to calibrate the measured accelerations and angular rates in-flight.

The sensor error components with different correlation time lengths are separately modeled and superposed to obtain the overall measurement error. The gyroscope error is for example composed of the turn-on bias, scale factor error, misalignment error, bias instability and angular random walk, amongst others. Therefore, the state-space model (2.10) is generalized for q instead of one process noise and r instead of one measurement noise

$$\begin{aligned} \mathbf{z}_k &= \Phi_{k-1} \mathbf{z}_{k-1} + \sum_{i=1}^q (\Gamma_{i,k-1} \boldsymbol{\omega}_{i,k-1}) \\ \tilde{\mathbf{y}}_k &= \mathbf{H}_k \mathbf{z}_k + \sum_{i=1}^r \mathbf{v}_{i,k} \end{aligned} \quad (4.131)$$

Each process noise $\boldsymbol{\omega}_i$ comes along with its own input matrix Γ_i to account for the fact that each accelerometer and gyroscope channel may have its own error models. Each of the q process noises is described by a distinct linear discrete-time state-space model of the form

$$\begin{aligned} \mathbf{z}_{\omega_i,k} &= \Phi_{\omega_i} \mathbf{z}_{\omega_i,k-1} + \Gamma_{\omega_i} \mathbf{n}_{\omega_i,k-1} & \mathbf{n}_{\omega_i,k} &\sim WN(0, \mathbf{Q}_{\omega_i}) \\ \boldsymbol{\omega}_{i,k} &= \mathbf{H}_{\omega_i} \mathbf{z}_{\omega_i,k} + \mathbf{D}_{\omega_i} \mathbf{n}_{\omega_i,k} & i &= 1 \dots q \end{aligned} \quad (4.132)$$

The output $\boldsymbol{\omega}_i$ is the actual measurement error which, in turn, is input of the navigation state error model (4.131). The process noise models are in general time-invariant. Furthermore, it is assumed that the input noises of the different process noise models are not correlated

$$\mathbb{E}[\mathbf{n}_{\omega_i,k} \mathbf{n}_{\omega_j,k}^\top] = \mathbf{0} \quad \forall i \neq j, \forall k \quad (4.133)$$

Likewise, each of the r measurement noises is represented by a linear discrete-time time-invariant state-space model

$$\begin{aligned} \mathbf{z}_{v_i,k} &= \Phi_{v_i} \mathbf{z}_{v_i,k-1} + \Gamma_{v_i} \mathbf{n}_{v_i,k-1} & \mathbf{n}_{v_i,k} &\sim WN(0, \mathbf{Q}_{v_i}) \\ \mathbf{v}_{i,k} &= \mathbf{H}_{v_i} \mathbf{z}_{v_i,k} + \mathbf{D}_{v_i} \mathbf{n}_{v_i,k} & i &= 1 \dots r \end{aligned} \quad (4.134)$$

The same assumptions as for the process noise models are valid

$$\mathbb{E}[\mathbf{n}_{v_i,k} \mathbf{n}_{v_j,k}^\top] = \mathbf{0} \quad \forall i \neq j, \forall k \quad (4.135)$$

Moreover, the input noises \mathbf{n}_{ω_i} of the process noise models are not correlated with the input noises \mathbf{n}_{v_i} of the measurement noise models

$$\mathbb{E}[\mathbf{n}_{v_i,k} \mathbf{n}_{\omega_j,k}^\top] = \mathbf{0} \quad \forall i, j, k, l \quad (4.136)$$

Note that the state-space models generally have a direct feed-through $\mathbf{D}_{\omega/v}$ if the underlying stochastic process is described by an AR(MA) model that has been converted either in controllable or observable canonical form. Non-correlated white process and measurement noise as required by the CKF can actually be handled as a special case of the presented state-space models by setting $\Phi_{\omega/v} = \mathbf{0}$, $\Gamma_{\omega/v} = \mathbf{0}$, $\mathbf{H}_{\omega/v} = \mathbf{0}$ and $\mathbf{D}_{\omega/v} = \mathbf{I}_{q/r}$. In contrast, fully time correlated biases are accounted for with $\Phi_{\omega/v} = \mathbf{I}_{q/r}$, $\Gamma_{\omega/v} = \mathbf{0}$, $\mathbf{H}_{\omega/v} = \mathbf{I}_{q/r}$ and $\mathbf{D}_{\omega/v} = \mathbf{0}$.

The navigation error model (4.131), the process noise models (4.132) and the measurement noise models (4.134) are put together to get the combined model

$$\begin{aligned} \underbrace{\begin{pmatrix} \mathbf{z}_k \\ \mathbf{z}_{\omega_1,k} \\ \vdots \\ \mathbf{z}_{\omega_q,k} \\ \mathbf{z}_{v_1,k} \\ \vdots \\ \mathbf{z}_{v_r,k} \end{pmatrix}}_{=: \mathbf{z}_{a,k}} &= \underbrace{\begin{pmatrix} \Phi_{k-1} & \Gamma_{1,k-1} \mathbf{H}_{\omega_1} & \cdots & \Gamma_{q,k-1} \mathbf{H}_{\omega_q} & 0 \\ \Phi_{\omega_1} & 0 & & & 0 \\ 0 & \ddots & \ddots & & 0 \\ 0 & 0 & \Phi_{\omega_q} & & 0 \\ \hline 0 & 0 & & \Phi_{v_1} & 0 \\ 0 & 0 & & 0 & \ddots & \Phi_{v_r} \end{pmatrix}}_{=: \Phi_{a,k-1}} \cdot \begin{pmatrix} \mathbf{z}_{k-1} \\ \mathbf{z}_{\omega_1,k-1} \\ \vdots \\ \mathbf{z}_{\omega_q,k-1} \\ \mathbf{z}_{v_1,k-1} \\ \vdots \\ \mathbf{z}_{v_r,k-1} \end{pmatrix} \\ &+ \underbrace{\begin{pmatrix} \Gamma_{1,k-1} \mathbf{D}_{\omega_1} & \cdots & \Gamma_{q,k-1} \mathbf{D}_{\omega_q} & 0 \\ \Gamma_{\omega_1} & 0 & & 0 \\ 0 & \ddots & \Gamma_{\omega_q} & 0 \\ 0 & 0 & \Gamma_{v_1} & 0 \\ \hline 0 & 0 & 0 & \ddots & \Gamma_{v_r} \end{pmatrix}}_{=: \Gamma_{a,k-1}} \cdot \underbrace{\begin{pmatrix} \mathbf{n}_{\omega_1,k-1} \\ \vdots \\ \mathbf{n}_{\omega_q,k-1} \\ \mathbf{n}_{v_1,k-1} \\ \vdots \\ \mathbf{n}_{v_r,k-1} \end{pmatrix}}_{=: \mathbf{n}_{a,k-1}} \\ \tilde{\mathbf{y}}_k &= \underbrace{\left(\mathbf{H}_k \mid \mathbf{0} \mid \mathbf{H}_{v_1} \cdots \mathbf{H}_{v_r} \right)}_{=: \mathbf{H}_{a,k}} \cdot \mathbf{z}_{a,k} + \underbrace{\left(\mathbf{0} \mid \mathbf{D}_{v_1} \cdots \mathbf{D}_{v_r} \right)}_{=: \mathbf{D}_a} \cdot \mathbf{n}_{a,k} \end{aligned} \quad (4.137)$$

with the augmented system matrix Φ_a , input matrix Γ_a , observation matrix \mathbf{H}_a and feed-through matrix \mathbf{D}_a . The augmented system model, which serves as basis for the CKF (2.11), is

$$\begin{aligned}\mathbf{z}_{a,k} &= \Phi_{a,k-1} \mathbf{z}_{a,k-1} + \Gamma_{a,k-1} \mathbf{n}_{a,k-1} \\ \tilde{\mathbf{y}}_k &= \mathbf{H}_{a,k} \mathbf{z}_{a,k} + \mathbf{D}_a \mathbf{n}_{a,k}\end{aligned}\quad (4.138)$$

This is actually a special form of a state-space model for a Kalman filter: the process noise as well as the measurement noise originate from the same source at first glance. In the respective literature the Kalman filter is derived for independent process noise and measurement noise sources that are not correlated with each other. Therefore, it is worth addressing this special form of the Kalman filter briefly. First, let us again have a look at the Kalman filter update equations. Since the prior state estimate $\hat{\mathbf{z}}_{a,k}^-$ immediately before the update is not correlated with the noise $\mathbf{n}_{a,k}$, which means $E[\delta\mathbf{z}_{a,k}^- \mathbf{n}_{a,k}^T] = 0$ (indeed the prior state estimate $\hat{\mathbf{z}}_{a,k}^-$ is correlated with the noise $\mathbf{n}_{a,k-1}$ of the earlier time step t_{k-1}), the state and covariance update are the same as in the CKF, now with the measurement noise $\mathbf{v}_k = \mathbf{D}_a \mathbf{n}_{a,k}$ with covariance $\mathbf{R}_a = \mathbf{D}_a \mathbf{Q}_a \mathbf{D}_a^T$ with $\mathbf{Q}_a = E[\mathbf{n}_{a,k} \mathbf{n}_{a,k}^T]$.

Second, an eye is cast over the propagation equations. Here, the situation turns out to be other than in the CKF. In order to see the difference, the prior state error covariance $\mathbf{P}_{a,k}^-$ is derived again. For that purpose, the prior state estimate error $\delta\mathbf{z}_{a,k}^- = \mathbf{z}_{a,k}^- - \hat{\mathbf{z}}_{a,k}^-$ is calculated. True state propagation and estimated state propagation, respectively, are

$$\begin{aligned}\mathbf{z}_{a,k} &= \Phi_{a,k-1} \mathbf{z}_{a,k-1} + \Gamma_{a,k-1} \mathbf{n}_{a,k-1} \\ \hat{\mathbf{z}}_{a,k}^- &= \Phi_{a,k-1} \hat{\mathbf{z}}_{a,k-1}^+\end{aligned}\quad (4.139)$$

The state error $\delta\mathbf{z}_{a,k}^-$ is found by subtracting both propagation equations

$$\delta\mathbf{z}_{a,k}^- = \Phi_{a,k-1} \delta\mathbf{z}_{a,k-1}^+ + \Gamma_{a,k-1} \mathbf{n}_{a,k-1} \quad (4.140)$$

The prior covariance matrix $\mathbf{P}_{a,k}^-$ is then

$$\begin{aligned}\mathbf{P}_{a,k}^- &= E\left[\left(\hat{\mathbf{z}}_{a,k}^- - E[\hat{\mathbf{z}}_{a,k}^-]\right)\left(\hat{\mathbf{z}}_{a,k}^- - E[\hat{\mathbf{z}}_{a,k}^-]\right)^T\right] = E[\delta\mathbf{z}_{a,k}^- \delta\mathbf{z}_{a,k}^{+T}] \\ &= \Phi_{a,k-1} \mathbf{P}_{a,k-1}^+ \Phi_{a,k-1}^T + \Gamma_{a,k-1} \mathbf{Q}_a \Gamma_{a,k-1}^T \\ &\quad + \Phi_{a,k-1} E[\delta\mathbf{z}_{a,k-1}^+ \mathbf{n}_{a,k-1}^T] \Gamma_{a,k-1}^T + \Gamma_{a,k-1} E[\mathbf{n}_{a,k-1} \delta\mathbf{z}_{a,k-1}^{+T}] \Phi_{a,k-1}^T\end{aligned}\quad (4.141)$$

Compared to the CKF covariance propagation, the cross-covariance matrix $E[\delta\mathbf{z}_{a,k-1}^+ \mathbf{n}_{a,k-1}^T]$ is not zero. In order to determine this cross-covariance matrix, the estimated state after the update $\hat{\mathbf{z}}_{a,k-1}^+$ is required

$$\hat{\mathbf{z}}_{a,k-1}^+ = \hat{\mathbf{z}}_{a,k-1}^- + \mathbf{K}_{k-1} (\tilde{\mathbf{y}}_k - \mathbf{H}_{a,k-1} \hat{\mathbf{z}}_{a,k-1}^-) \quad (4.142)$$

Inserting the observation equation of (4.138) yields further

$$\hat{\mathbf{z}}_{a,k-1}^+ = \hat{\mathbf{z}}_{a,k-1}^- + \mathbf{K}_{k-1} (\mathbf{H}_{a,k-1} \mathbf{z}_{a,k-1} + \mathbf{D}_a \mathbf{n}_{a,k-1} - \mathbf{H}_{a,k-1} \hat{\mathbf{z}}_{a,k-1}^-) \quad (4.143)$$

The posterior state error $\delta\mathbf{z}_{a,k-1}^+ = \mathbf{z}_{a,k-1} - \hat{\mathbf{z}}_{a,k-1}^+$ is consequently given with

$$\delta\mathbf{z}_{a,k-1}^+ = (\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}_{a,k-1}) \delta\mathbf{z}_{a,k-1}^- - \mathbf{K}_{k-1} \mathbf{D}_a \mathbf{n}_{a,k-1} \quad (4.144)$$

The wanted cross-covariance thus becomes

$$E[\delta\mathbf{z}_{a,k-1}^+ \mathbf{n}_{a,k-1}^T] = E[((\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}_{a,k-1}) \delta\mathbf{z}_{a,k-1}^- - \mathbf{K}_{k-1} \mathbf{D}_a \mathbf{n}_{a,k-1}) \mathbf{n}_{a,k-1}^T] = -\mathbf{K}_{k-1} \mathbf{D}_a \mathbf{Q}_a \quad (4.145)$$

since the prior state estimate error $\delta \mathbf{z}_{a,k-1}^-$ is not correlated with the noise $\mathbf{n}_{a,k-1}$, $E[\delta \mathbf{z}_{a,k-1}^- \mathbf{n}_{a,k-1}^T] = 0$. The covariance after the propagation (4.141) is finally

$$\begin{aligned}\mathbf{P}_{a,k}^- &= \Phi_{a,k-1} \mathbf{P}_{a,k-1}^+ \Phi_{a,k-1}^T + \Gamma_{a,k-1} \mathbf{Q}_a \Gamma_{a,k-1}^T \\ &\quad - \Phi_{a,k-1} \mathbf{K}_{k-1} \mathbf{D}_a \mathbf{Q}_a \Gamma_{a,k-1}^T - \Gamma_{a,k-1} \mathbf{Q}_a \mathbf{D}_a^T \mathbf{K}_{k-1}^T \Phi_{a,k-1}^T\end{aligned}\quad (4.146)$$

In comparison with the CKF, we see that two additional terms appear in the covariance propagation equation. Note that the extended covariance propagation equation (4.146) applies only in the first propagation step immediately after the last update. If interim propagation steps without preceding update are executed, for example in a longer period between two subsequent aiding measurement samples, the regular covariance propagation equation of the CKF is valid. It is advisable to calculate the term $\Phi_{a,k} \mathbf{K}_k \mathbf{D}_a \mathbf{Q}_a \Gamma_{a,k}^T$ to get a better understanding which measurements are concerned

$$\Phi_{a,k} \mathbf{K}_k \mathbf{D}_a \mathbf{Q}_a \Gamma_{a,k}^T = \Phi_{a,k} \mathbf{K}_k \left(\mathbf{0} \mid \mathbf{0} \mid \mathbf{D}_{v_1} \mathbf{Q}_{v_1} \Gamma_{v_1}^T \cdots \mathbf{D}_{v_r} \mathbf{Q}_{v_r} \Gamma_{v_r}^T \right) \quad (4.147)$$

This means that the extended covariance propagation equation (4.146) has only to be applied if the input and feed-through matrices Γ_{v_i} and \mathbf{D}_{v_i} of the measurement noise models that have been involved in the preceding update are both not zero. In the special case of a constant bias the extended covariance propagation is thus not required. The same holds for pure white measurement noise. The covariance propagation equations are recapitulated

$$\begin{aligned}\mathbf{P}_{a,k}^- &= \Phi_{a,k-1} \mathbf{P}_{a,k-1}^+ \Phi_{a,k-1}^T + \Gamma_{a,k-1} \mathbf{Q}_a \Gamma_{a,k-1}^T && \text{Covariance propagation} \\ &\quad - \Phi_{a,k-1} \mathbf{K}_{k-1} \mathbf{D}_a \mathbf{Q}_a \Gamma_{a,k-1}^T - \Gamma_{a,k-1} \mathbf{Q}_a \mathbf{D}_a^T \mathbf{K}_{k-1}^T \Phi_{a,k-1}^T && \text{(with preceding update)} \\ \mathbf{P}_{a,k}^- &= \Phi_{a,k-1} \mathbf{P}_{a,k-1}^+ \Phi_{a,k-1}^T + \Gamma_{a,k-1} \mathbf{Q}_a \Gamma_{a,k-1}^T && \text{Covariance propagation} \\ &&& \text{(without preceding update)}\end{aligned}\quad (4.148)$$

Template 4-19 summarizes the equations of the modified Kalman filter that bases upon the state-space model (4.138).

In section C.4 in the appendix it is exemplarily shown how to augment the navigation state vector by states for the estimation of IMU turn-on biases and scale factor errors.

Template 4-19: Augmented state Kalman filter**Augmented State Model**

$$\Phi_{a,k} = \begin{pmatrix} \Phi_k & \Gamma_{1,k} H_{\omega_1} & \cdots & \Gamma_{q,k} H_{\omega_q} & 0 \\ 0 & \Phi_{\omega_1} & & 0 & 0 \\ 0 & & \ddots & & 0 \\ 0 & 0 & & \Phi_{\omega_q} & 0 \\ 0 & & & 0 & \Phi_{v_1} & 0 \\ 0 & 0 & & 0 & 0 & \ddots \\ 0 & & & 0 & 0 & \Phi_{v_r} \end{pmatrix}, \quad \Gamma_{a,k} = \begin{pmatrix} \Gamma_{1,k} D_{\omega_1} & \cdots & \Gamma_{q,k} D_{\omega_q} & 0 \\ \Gamma_{\omega_1} & & 0 & 0 \\ 0 & \ddots & & 0 \\ 0 & 0 & & \Gamma_{\omega_q} \\ 0 & & & 0 \\ 0 & & & 0 & \Gamma_{v_1} & 0 \\ 0 & & & 0 & 0 & \ddots \\ 0 & & & 0 & 0 & \Gamma_{v_r} \end{pmatrix}$$

$$H_{a,k} = (H_k \mid 0 \mid H_{v_1} \cdots H_{v_r}), \quad D_a = (0 \mid D_{v_1} \cdots D_{v_r})$$

$$Q_a = \begin{pmatrix} Q_{\omega_1} & 0 & & \\ \vdots & 0 & & \\ 0 & Q_{\omega_q} & & \\ & & \ddots & 0 \\ 0 & & 0 & Q_{v_1} \\ & & & \ddots \\ 0 & & 0 & Q_{v_r} \end{pmatrix}$$

Initial Values

$$\hat{z}_{a,0} = \begin{pmatrix} \hat{z}_0 \\ \hat{z}_{\omega_1,0} \\ \vdots \\ \hat{z}_{\omega_q,0} \\ \hat{z}_{v_1,0} \\ \vdots \\ \hat{z}_{v_r,0} \end{pmatrix}, \quad P_{a,0} = \begin{pmatrix} P_{zz,0} & 0 & & & 0 \\ 0 & P_{\omega_1\omega_1,0} & 0 & & 0 \\ 0 & & \ddots & & 0 \\ 0 & 0 & & P_{\omega_q\omega_q,0} & \\ & & & & P_{v_1v_1,0} \\ 0 & 0 & & 0 & \ddots \\ 0 & & & 0 & P_{v_rv_r,0} \end{pmatrix}$$

Propagation

$$\hat{z}_{a,k}^- = \Phi_{a,k-1} \hat{z}_{a,k-1}^+ \quad \text{State propagation}$$

$$P_{a,k}^- = \Phi_{a,k-1} P_{a,k-1}^+ \Phi_{a,k-1}^T + \Gamma_{a,k-1} Q_a \Gamma_{a,k-1}^T - \Phi_{a,k-1} K_{k-1} D_a Q_a \Gamma_{a,k-1}^T - \Gamma_{a,k-1} Q_a D_a^T K_{k-1}^T \Phi_{a,k-1}^T \quad \text{Covariance propagation (with preceding update)}$$

$$P_{a,k}^- = \Phi_{a,k-1} P_{a,k-1}^+ \Phi_{a,k-1}^T + \Gamma_{a,k-1} Q_a \Gamma_{a,k-1}^T \quad \text{Covariance propagation (without preceding update)}$$

Update

$$R_a = D_a Q_a D_a^T \quad \text{Measurement covariance}$$

$$s_k = \tilde{y}_k - H_{a,k} \hat{z}_{a,k}^- \quad \text{Innovation}$$

$$S_k = H_{a,k} P_{a,k}^- H_{a,k}^T + R_a \quad \text{Innovation covariance}$$

$$K_k = P_{a,k}^- H_{a,k}^T S_k^{-1} \quad \text{Kalman gain}$$

$$\hat{z}_{a,k}^+ = \hat{z}_{a,k}^- + K_k s_k \quad \text{State estimate update}$$

$$P_{a,k}^+ = (I_{n_a} - K_k H_{a,k}) P_{a,k}^- \quad \text{Covariance update}$$

$$= (I_{n_a} - K_k H_{a,k}) P_{a,k}^- (I_{n_a} - K_k H_{a,k})^T + K_k R_a K_k^T$$

4.5 Schmidt-Kalman Filter for Noise-Like Errors

4.5.1 Motivation

Constant or bias-like measurement errors with long correlation time lengths can be well estimated by the navigation filter using state augmentation as in the previous section, provided that they are observable. The turn-on biases and scale factor errors of the accelerometers and gyroscopes or the clock bias and drift of the GNSS receiver are typical errors to be estimated by the navigation filter. On the other hand, noise-like stochastic errors with short correlation times are difficult to observe by the navigation filter. Think for example of the accelerometer and gyroscope bias instabilities or of the small bias-like misalignments, which can hardly be observed by the filter. Due to the lack of observability, in many cases it does not make sense to augment the navigation error model by models for these errors in order to let them be determined by the filter. This would primarily inflate the dimension of the filter and rapidly increase the computational effort but would bring only little additional benefit.

In this case, a good way is to distinguish between states that shall be estimated and states that shall not be estimated but whose influence on the states to be estimated shall be considered at least. The former are designated as the *estimated* states. The latter are the so called *considered* states. Only their statistical influence on the estimated states is considered but their expected values are not estimated. In this section, the Schmidt-Kalman filter is introduced as versatile tool for problems with a mixture of states to be estimated and states to be considered and is adapted to the navigation filter context.

4.5.2 Schmidt-Kalman Filter

The Kalman gain matrix of Template 4-19 can be split into three parts

$$\mathbf{K}_k = \begin{pmatrix} \mathbf{K}_{z,k} \\ \hline \mathbf{K}_{\omega_1,k} \\ \vdots \\ \mathbf{K}_{\omega_q,k} \\ \hline \mathbf{K}_{v_1,k} \\ \vdots \\ \mathbf{K}_{v_r,k} \end{pmatrix} \quad (4.149)$$

$\mathbf{K}_{z,k}$ relates the innovation of the measurement $\tilde{\mathbf{y}}_k$ to the navigation state error, $\mathbf{K}_{\omega_1,k} \dots \mathbf{K}_{\omega_q,k}$ are the correction factors of the process noise model states and $\mathbf{K}_{v_1,k} \dots \mathbf{K}_{v_r,k}$ the correction factors of the measurement noise model states. The Kalman gain matrices of the process and measurement noise model states that shall only be considered are set to zero. The Kalman gains of the navigation state and the process and measurement noise model states that shall be estimated remain. The innovation of the aiding measurement $\tilde{\mathbf{y}}_k$ is thus only distributed among the estimated states but not among the considered states. It is important to notice that the second form of the covariance update equation in (2.11), Joseph's form, has to be compulsorily used instead of the short form $\mathbf{P}_{a,k}^+ = (\mathbf{I}_{n_a} - \mathbf{K}_k \mathbf{H}_{a,k}) \mathbf{P}_{a,k}^-$ when the Kalman gain matrix entries of the considered states are set to zero. The reason for that is that the assumptions made during the conversion from Joseph's form to the short form are not valid in the case with mixed estimated and considered states. The following equalities

$$\begin{aligned}
\mathbf{P}_{a,k}^+ &= (\mathbf{I}_{n_a} - \mathbf{K}_k \mathbf{H}_{a,k}) \mathbf{P}_{a,k}^- (\mathbf{I}_{n_a} - \mathbf{K}_k \mathbf{H}_{a,k})^\top + \mathbf{K}_k \mathbf{R}_a \mathbf{K}_k^\top \\
&= \mathbf{P}_{a,k}^- - \mathbf{K}_k \mathbf{H}_{a,k} \mathbf{P}_{a,k}^- - \mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top \mathbf{K}_k^\top + \mathbf{K}_k (\mathbf{R}_a + \mathbf{H}_{a,k} \mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top) \mathbf{K}_k^\top \\
&= (\mathbf{I}_{n_a} - \mathbf{K}_k \mathbf{H}_{a,k}) \mathbf{P}_{a,k}^- - \mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top \mathbf{S}_k^{-1} \mathbf{S}_k \mathbf{K}_k^\top + \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top
\end{aligned} \tag{4.150}$$

hold for both cases whereas the further simplification with $\mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top \mathbf{S}_k^{-1} = \mathbf{K}_k$

$$\begin{aligned}
\mathbf{P}_{a,k}^+ &= (\mathbf{I}_{n_a} - \mathbf{K}_k \mathbf{H}_{a,k}) \mathbf{P}_{a,k}^- - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top + \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top \\
&= (\mathbf{I}_{n_a} - \mathbf{K}_k \mathbf{H}_{a,k}) \mathbf{P}_{a,k}^-
\end{aligned} \tag{4.151}$$

is only possible in the case of the nominal Kalman filter without considered states. In the filter with estimated and considered states

$$-\mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top \mathbf{S}_k^{-1} \mathbf{S}_k \mathbf{K}_k^\top + \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top \neq \mathbf{0} \tag{4.152}$$

because $\mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top \mathbf{S}_k^{-1} \neq \mathbf{K}_k$. The matrix $\mathbf{P}_{a,k}^- \mathbf{H}_{a,k}^\top \mathbf{S}_k^{-1}$ can actually have non-zero entries in the rows belonging to the considered states.

If the integrated navigation filter uses Joseph's form of the covariance update equation innately, it can be easily switched from state estimation to state consideration and back. This might be advantageous in the early phase shortly after initialization when the filter settles and the state estimates are adjusted considerably. Some of the nuisance states could be treated as considered states in the beginning such that the filter modifies only the remaining states. When the main adjustment after initialization has accomplished, the initially considered states can be changed to estimated states.

The augmented state covariance matrix can obviously become quite large. Hundred or more states are not unusual when models for the accelerometer, gyroscope, GNSS and barometer measurements and other sensors are incorporated into the integrated navigation filter. Although the performance of processors increases steadily, the computational effort may be problematic on smaller embedded systems. The covariance propagation involves most of the operations of the filter. The number of multiplications depends on the number of states to the third power.

The numerical effort can be reduced if the covariance matrix is partitioned in the core navigation error block, process noise blocks and measurement noise blocks and if the propagation, Kalman gain and covariance update equations are expanded accordingly. Parts of the update equations that contain the zeroed Kalman gains of the considered states fall away and do not have to be calculated. Furthermore, due to the symmetry of the covariance matrix, only the upper or lower triangular of the covariance matrix has to be calculated. This adapted form of the Kalman filter is known as Schmidt-Kalman filter [13] [108]. The Schmidt-Kalman filter approach was originally presented in [109]. A good introduction of the usage of a Schmidt-Kalman filter for integrated navigation can be found in [27] or [110]. The covariance matrix of the augmented state vector $\hat{\mathbf{z}}_{a,k}^-$

$$\mathbf{P}_{a,k}^- = \left(\begin{array}{c|cc|cc} \mathbf{P}_{zz,k}^- & \mathbf{P}_{z\omega_1,k}^- & \dots & \mathbf{P}_{z\omega_q,k}^- & \mathbf{P}_{zv_1,k}^- & \dots & \mathbf{P}_{zv_r,k}^- \\ \hline \mathbf{P}_{\omega_1 z,k}^- & \mathbf{P}_{\omega_1 \omega_1,k}^- & \dots & \mathbf{P}_{\omega_1 \omega_q,k}^- & \mathbf{P}_{\omega_1 v_1,k}^- & \dots & \mathbf{P}_{\omega_1 v_r,k}^- \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \hline \mathbf{P}_{\omega_q z,k}^- & \mathbf{P}_{\omega_q \omega_1,k}^- & \dots & \mathbf{P}_{\omega_q \omega_q,k}^- & \mathbf{P}_{\omega_q v_1,k}^- & \dots & \mathbf{P}_{\omega_q v_r,k}^- \\ \hline \mathbf{P}_{v_1 z,k}^- & \mathbf{P}_{v_1 \omega_1,k}^- & \dots & \mathbf{P}_{v_1 \omega_q,k}^- & \mathbf{P}_{v_1 v_1,k}^- & \dots & \mathbf{P}_{v_1 v_r,k}^- \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \hline \mathbf{P}_{v_r z,k}^- & \mathbf{P}_{v_r \omega_1,k}^- & \dots & \mathbf{P}_{v_r \omega_q,k}^- & \mathbf{P}_{v_r v_1,k}^- & \dots & \mathbf{P}_{v_r v_r,k}^- \end{array} \right) \tag{4.153}$$

contains the navigation error state covariance $\mathbf{P}_{zz,k}^-$, the covariances and cross-covariances of the process noises $\mathbf{P}_{\omega_s \omega_j, k}^-$ and measurement noises $\mathbf{P}_{v_t v_j, k}^-$ and the cross-covariances between the navigation error states and the process noise states $\mathbf{P}_{z \omega_s, k}^-$ and measurement noise states $\mathbf{P}_{z v_t, k}^-$, respectively. Expanding the short form of the covariance update equation in (2.11) yields

$$\begin{aligned}\mathbf{P}_{zz,k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{zz,k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^- \\ \mathbf{P}_{z \omega_s, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{z \omega_s, k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i \omega_s, k}^- \\ \mathbf{P}_{z v_t, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{z v_t, k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i v_t, k}^- \\ \mathbf{P}_{\omega_s \omega_j, k}^+ &= \mathbf{P}_{\omega_s \omega_j, k}^- - \mathbf{K}_{\omega_s, k} \mathbf{H}_k \mathbf{P}_{z \omega_j, k}^- - \mathbf{K}_{\omega_s, k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i \omega_j, k}^- \\ \mathbf{P}_{v_t v_j, k}^+ &= \mathbf{P}_{v_t v_j, k}^- - \mathbf{K}_{v_t, k} \mathbf{H}_k \mathbf{P}_{z v_j, k}^- - \mathbf{K}_{v_t, k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i v_j, k}^- \\ \mathbf{P}_{\omega_s v_j, k}^+ &= \mathbf{P}_{\omega_s v_j, k}^- - \mathbf{K}_{\omega_s, k} \mathbf{H}_k \mathbf{P}_{z v_j, k}^- - \mathbf{K}_{\omega_s, k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i v_j, k}^- \quad s = 1 \dots q, \quad t = 1 \dots r\end{aligned}\tag{4.154}$$

The Kalman gains of the considered process and measurement noise states $\mathbf{K}_{\omega_s, k}$ and $\mathbf{K}_{v_t, k}$ are set to zero. The expanded covariance update equations (4.154) simplify to

$$\begin{aligned}\mathbf{P}_{zz,k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{zz,k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^- \\ \mathbf{P}_{z \omega_s, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{z \omega_s, k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i \omega_s, k}^- \\ \mathbf{P}_{z v_t, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{z v_t, k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i v_t, k}^- \\ \mathbf{P}_{\omega_s \omega_j, k}^+ &= \mathbf{P}_{\omega_s \omega_j, k}^- \\ \mathbf{P}_{v_t v_j, k}^+ &= \mathbf{P}_{v_t v_j, k}^- \\ \mathbf{P}_{\omega_s v_j, k}^+ &= \mathbf{P}_{\omega_s v_j, k}^- \quad s = 1 \dots q, \quad t = 1 \dots r\end{aligned}\tag{4.155}$$

Expanding the covariance propagation equations without preceding update in (4.148) yields

$$\begin{aligned}\mathbf{P}_{zz,k}^- &= \Phi_{k-1} \mathbf{P}_{zz,k-1}^+ \Phi_{k-1}^\top + \Gamma_{k-1} \sum_{i=1}^q \left(\mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i \omega_i, k-1}^+ \mathbf{H}_{\omega_i}^\top + \mathbf{D}_{\omega_i} \mathbf{Q}_{\omega_i} \mathbf{D}_{\omega_i}^\top \right) \Gamma_{k-1}^\top \\ &\quad + \Gamma_{k-1} \sum_{i=1}^q \left(\mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i z, k-1}^+ \right) \Phi_{k-1}^\top + \Phi_{k-1} \sum_{i=1}^q \left(\mathbf{P}_{z \omega_i, k-1}^+ \mathbf{H}_{\omega_i}^\top \right) \Gamma_{k-1}^\top \\ \mathbf{P}_{z \omega_s, k}^- &= \Phi_{k-1} \mathbf{P}_{z \omega_s, k-1}^+ \Phi_{\omega_s}^\top + \Gamma_{k-1} \left(\mathbf{H}_{\omega_s} \mathbf{P}_{\omega_s \omega_s, k-1}^+ \Phi_{\omega_s}^\top + \mathbf{D}_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^\top \right) \\ \mathbf{P}_{z v_t, k}^- &= \Phi_{k-1} \mathbf{P}_{z v_t, k-1}^+ \Phi_{v_t}^\top \\ \mathbf{P}_{\omega_s \omega_j, k}^- &= \Phi_{\omega_s} \mathbf{P}_{\omega_s \omega_j, k-1}^+ \Phi_{\omega_j}^\top + \delta_{sj} \Gamma_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^\top \\ \mathbf{P}_{v_t v_j, k}^- &= \Phi_{v_t} \mathbf{P}_{v_t v_j, k-1}^+ \Phi_{v_j}^\top + \delta_{tj} \Gamma_{v_t} \mathbf{Q}_{v_t} \Gamma_{v_t}^\top \\ \mathbf{P}_{\omega_s v_j, k}^- &= \Phi_{\omega_s} \mathbf{P}_{\omega_s v_j, k-1}^+ \Phi_{v_j}^\top \quad s = 1 \dots q, \quad t = 1 \dots r\end{aligned}\tag{4.156}$$

The states of the separate process and measurement noise models are not correlated among each other in the beginning, that is

$$\mathbf{P}_{\omega_s \omega_j, 0} = 0, \quad \mathbf{P}_{v_t v_j, 0} = 0, \quad \mathbf{P}_{\omega_s v_j, 0} = 0, \quad s \neq j, t \neq j\tag{4.157}$$

Since these cross-covariances are subsequently not altered by the covariance update (4.155) and no uncertainty is added at the covariance propagation step (4.156), they always remain zero. The covariances $\mathbf{P}_{\omega_s \omega_s, k}^-$ and $\mathbf{P}_{v_t v_t, k}^-$ are in general unequal zero in the beginning. Additionally, uncertainty is added in the covariance propagation step. However, the covariances are not modified by the filter update. Hence, the distinction between a priori and a posteriori matrices becomes superfluous

$$\begin{aligned}\mathbf{P}_{\omega_s \omega_s, k} &= \Phi_{\omega_s} \mathbf{P}_{\omega_s \omega_s, k-1} \Phi_{\omega_s}^T + \Gamma_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^T \\ \mathbf{P}_{v_t v_t, k} &= \Phi_{v_t} \mathbf{P}_{v_t v_t, k-1} \Phi_{v_t}^T + \Gamma_{v_t} \mathbf{Q}_{v_t} \Gamma_{v_t}^T\end{aligned}\quad (4.158)$$

The simplified covariance update equations (4.155) are consequently

$$\begin{aligned}\mathbf{P}_{zz, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{zz, k}^- - \mathbf{K}_{z,k} \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^- \\ \mathbf{P}_{z\omega_s, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{z\omega_s, k}^- \\ \mathbf{P}_{zv_t, k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{zv_t, k}^- - \mathbf{K}_{z,k} \mathbf{H}_{v_t} \mathbf{P}_{v_t v_t, k}^- \quad s = 1 \dots q, \quad t = 1 \dots r\end{aligned}\quad (4.159)$$

The covariance propagation equations (4.156) reduce to

$$\begin{aligned}\mathbf{P}_{zz, k}^- &= \Phi_{k-1} \mathbf{P}_{zz, k-1}^+ \Phi_{k-1}^T + \Gamma_{k-1} \sum_{i=1}^q \left(\mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i \omega_i, k-1} \mathbf{H}_{\omega_i}^T + \mathbf{D}_{\omega_i} \mathbf{Q}_{\omega_i} \mathbf{D}_{\omega_i}^T \right) \Gamma_{k-1}^T \\ &\quad + \Gamma_{k-1} \sum_{i=1}^q \left(\mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i z, k-1}^+ \right) \Phi_{k-1}^T + \Phi_{k-1} \sum_{i=1}^q \left(\mathbf{P}_{z\omega_i, k-1}^+ \mathbf{H}_{\omega_i}^T \right) \Gamma_{k-1}^T \\ \mathbf{P}_{z\omega_s, k}^- &= \Phi_{k-1} \mathbf{P}_{z\omega_s, k-1}^+ \Phi_{\omega_s}^T + \Gamma_{k-1} \left(\mathbf{H}_{\omega_s} \mathbf{P}_{\omega_s \omega_s, k-1} \Phi_{\omega_s}^T + \mathbf{D}_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^T \right) \\ \mathbf{P}_{zv_t, k}^- &= \Phi_{k-1} \mathbf{P}_{zv_t, k-1}^+ \Phi_{v_t}^T \\ \mathbf{P}_{\omega_s \omega_s, k}^- &= \Phi_{\omega_s} \mathbf{P}_{\omega_s \omega_s, k-1} \Phi_{\omega_s}^T + \Gamma_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^T \\ \mathbf{P}_{v_t v_t, k}^- &= \Phi_{v_t} \mathbf{P}_{v_t v_t, k-1} \Phi_{v_t}^T + \Gamma_{v_t} \mathbf{Q}_{v_t} \Gamma_{v_t}^T \quad s = 1 \dots q, \quad t = 1 \dots r\end{aligned}\quad (4.160)$$

Next, the covariance propagation equation with preceding update in (4.148) is addressed. For that, the additional term that has been elaborated in (4.147) is picked up again. Inserting the split Kalman gain (4.149) into (4.147) and setting the Kalman gains of the considered states zero yields

$$\Phi_{a,k} \mathbf{K}_k \mathbf{D}_a \mathbf{Q}_a \Gamma_{a,k}^T = \left(\begin{array}{c|c|c|c|c} 0 & 0 & \Phi_k \mathbf{K}_{z,k} \mathbf{D}_{v_1} \mathbf{Q}_{v_1} \Gamma_{v_1}^T & \dots & \Phi_k \mathbf{K}_{z,k} \mathbf{D}_{v_r} \mathbf{Q}_{v_r} \Gamma_{v_r}^T \\ \hline 0 & 0 & & & 0 \\ \hline 0 & 0 & & & 0 \end{array} \right) \quad (4.161)$$

We can see that solely the cross-covariances $\mathbf{P}_{zv_t, k}^-$ are affected. Therefore, if an update has occurred immediately before the upcoming propagation step, the corresponding cross-covariance propagation equation for $\mathbf{P}_{zv_t, k}^-$ in (4.160) has to be replaced by

$$\mathbf{P}_{zv_t, k}^- = \Phi_{k-1} \mathbf{P}_{zv_t, k-1}^+ \Phi_{v_t}^T - \Phi_{k-1} \mathbf{K}_{z,k-1} \mathbf{D}_{v_t} \mathbf{Q}_{v_t} \Gamma_{v_t}^T \quad (4.162)$$

Expanding the innovation covariance equation in (2.11) gives

$$\begin{aligned}\mathbf{S}_k &= \mathbf{H}_k \mathbf{P}_{zz, k}^- \mathbf{H}_k^T + \sum_{i=1}^r \left(\mathbf{H}_{v_i} \mathbf{P}_{v_i v_i, k} \mathbf{H}_{v_i}^T + \mathbf{D}_{v_i} \mathbf{Q}_{v_i} \mathbf{D}_{v_i}^T \right) \\ &\quad + \sum_{i=1}^r \left(\mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^- \right) \mathbf{H}_k^T + \mathbf{H}_k \sum_{i=1}^r \left(\mathbf{P}_{zv_i, k}^- \mathbf{H}_{v_i}^T \right)\end{aligned}\quad (4.163)$$

The measurement covariance matrix is abbreviated as

$$\mathbf{R}_k = \sum_{i=1}^r \left(\mathbf{H}_{v_i} \mathbf{P}_{v_i v_i, k} \mathbf{H}_{v_i}^T + \mathbf{D}_{v_i} \mathbf{Q}_{v_i} \mathbf{D}_{v_i}^T \right) \quad (4.164)$$

The equation for the Kalman gain of the estimated states becomes

$$\mathbf{K}_{z,k} = \left(\mathbf{P}_{zz, k}^- \mathbf{H}_k^T + \sum_{i=1}^r \left(\mathbf{P}_{zv_i, k}^- \mathbf{H}_{v_i}^T \right) \right) \mathbf{S}_k^{-1} \quad (4.165)$$

Finally, Joseph's form of the covariance update equation (4.159) is given for the sake of completeness. It is analogously found by inserting the matrices (4.137) in Joseph's form of the covariance update equation in (2.11), setting the Kalman gains of the considered process and measurement error models

zero and then expanding the equation. Merely the update equation of the navigation error covariance changes, all other covariance update equations are the same as in the short form

$$\begin{aligned}\mathbf{P}_{zz,k}^+ &= (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \mathbf{P}_{zz,k}^- (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k)^T + \mathbf{K}_{z,k} \mathbf{R}_k \mathbf{K}_{z,k}^T \\ &\quad - \mathbf{K}_{z,k} \sum_{i=1}^r (\mathbf{H}_{\nu_i} \mathbf{P}_{\nu_i z,k}^-) (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k)^T - (\mathbf{I}_n - \mathbf{K}_{z,k} \mathbf{H}_k) \sum_{i=1}^r (\mathbf{P}_{z\nu_i,k}^- \mathbf{H}_{\nu_i}^T) \mathbf{K}_{z,k}^T\end{aligned}\quad (4.166)$$

The integrated navigation filter usually processes more than one aiding measurement, for example position, velocity, GNSS pseudorange and range rate, barometric altitude, magnetic field or orientation measurements. Separate error models describe the time correlated noises of the different measurements. Only those error models that are related to the currently processed measurement are to be used in the equations for the measurement covariance \mathbf{R}_k , innovation covariance \mathbf{S}_k , Kalman gain \mathbf{K}_k and navigation error state covariance $\mathbf{P}_{zz,k}^+$. The measurement error models that are not involved in the current update are treated like the process noise cross-covariances $\mathbf{P}_{z\omega_s,k}^-$, that is only their cross-covariance with the navigation error state $\mathbf{P}_{z\nu_t,k}^-$ must be updated according to

$$\mathbf{P}_{z\nu_t,k}^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{z\nu_t,k}^- \quad (4.167)$$

The corresponding presuppositions are summarized in Template 4-20, the covariance propagation equations in Template 4-21 and the covariance update equations in Template 4-22.

Further information

In the appendix in section C.5.1 it is shown how the Schmidt-Kalman filter for time-correlated measurements corresponds to a Kalman filter for state cross-correlated measurements. Moreover, Template C-2 in section C.5.2 in the appendix gives the equations of the Schmidt-Kalman filter for out-of-sequence measurements.

Template 4-20: Models and presuppositions

Models	Process noise $s = 1 \dots q$	Measurement noise $t = 1 \dots r$
State model		
$\mathbf{z}_{k+1} = \Phi_k \mathbf{z}_k + \sum_{i=1}^q (\boldsymbol{\Gamma}_{i,k} \boldsymbol{\omega}_{i,k})$	$\mathbf{z}_{\omega_s,k+1} = \Phi_{\omega_s} \mathbf{z}_{\omega_s,k} + \boldsymbol{\Gamma}_{\omega_s} \boldsymbol{\eta}_{\omega_s,k}$	$\mathbf{z}_{\nu_t,k+1} = \Phi_{\nu_t} \mathbf{z}_{\nu_t,k} + \boldsymbol{\Gamma}_{\nu_t} \boldsymbol{\eta}_{\nu_t,k}$
$\tilde{\mathbf{y}}_k = \mathbf{H}_k \mathbf{z}_k + \sum_{i=1}^r \mathbf{v}_{i,k}$	$\boldsymbol{\omega}_{s,k} = \mathbf{H}_{\omega_s} \mathbf{z}_{\omega_s,k} + \mathbf{D}_{\omega_s} \boldsymbol{\eta}_{\omega_s,k}$	$\mathbf{v}_{t,k} = \mathbf{H}_{\nu_t} \mathbf{z}_{\nu_t,k} + \mathbf{D}_{\nu_t} \boldsymbol{\eta}_{\nu_t,k}$
where	$\boldsymbol{\eta}_{\omega_s,k} \sim WN(0, \mathbf{Q}_{\omega_s})$	$\boldsymbol{\eta}_{\nu_t,k} \sim WN(0, \mathbf{Q}_{\nu_t})$
$E[\boldsymbol{\eta}_{\omega_s,k} \boldsymbol{\eta}_{\omega_s,l}^T] = 0 \quad \forall s, j, k, l$	$E[\boldsymbol{\eta}_{\omega_s,k} \boldsymbol{\eta}_{\omega_s,l}^T] = 0 \quad \forall k \neq l$	$E[\boldsymbol{\eta}_{\nu_t,k} \boldsymbol{\eta}_{\nu_t,l}^T] = 0 \quad \forall k \neq l$
	$E[\boldsymbol{\eta}_{\omega_s,k} \boldsymbol{\eta}_{\omega_j,k}^T] = 0 \quad \forall j \neq s, k$	$E[\boldsymbol{\eta}_{\nu_t,k} \boldsymbol{\eta}_{\nu_j,k}^T] = 0 \quad \forall j \neq t, k$

Template 4-21: Propagation equations**Navigation state propagation**

$$\begin{aligned}\hat{\mathbf{z}}_k^- &= \Phi_{k-1} \hat{\mathbf{z}}_{k-1}^+ && \text{State estimate propagation} \\ \mathbf{Q}_k &= \sum_{i=1}^q \Gamma_{i,k-1} \left(\mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i \omega_i, k-1} \mathbf{H}_{\omega_i}^T + \mathbf{D}_{\omega_i} \mathbf{Q}_{\omega_i} \mathbf{D}_{\omega_i}^T \right) \Gamma_{i,k-1}^T && \text{Effective input covariance} \\ \mathbf{P}_{zz,k}^- &= \Phi_{k-1} \mathbf{P}_{zz,k-1}^+ \Phi_{k-1}^T + \mathbf{Q}_k && \text{Covariance propagation} \\ &\quad + \sum_{i=1}^q \left(\Gamma_{i,k-1} \mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i z, k-1}^+ \right) \Phi_{k-1}^T + \Phi_{k-1} \sum_{i=1}^q \left(\mathbf{P}_{z \omega_i, k-1}^+ \mathbf{H}_{\omega_i}^T \Gamma_{i,k-1}^T \right)\end{aligned}$$

Process noise propagation

$$\begin{aligned}\mathbf{P}_{z \omega_s, k}^- &= \Phi_{k-1} \mathbf{P}_{z \omega_s, k-1}^+ \Phi_{\omega_s}^T + \Gamma_{s,k-1} \left(\mathbf{H}_{\omega_s} \mathbf{P}_{\omega_s \omega_s, k-1} \Phi_{\omega_s}^T + \mathbf{D}_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^T \right) && \text{Process noise cross-covariance propagation} \\ \mathbf{P}_{\omega_s \omega_s, k}^- &= \Phi_{\omega_s} \mathbf{P}_{\omega_s \omega_s, k-1} \Phi_{\omega_s}^T + \Gamma_{\omega_s} \mathbf{Q}_{\omega_s} \Gamma_{\omega_s}^T && \text{Process noise covariance propagation}\end{aligned}$$

Measurement noise propagation

$$\begin{aligned}\mathbf{P}_{z v_t, k}^- &= \Phi_{k-1} \mathbf{P}_{z v_t, k-1}^+ \Phi_{v_t}^T - \Phi_{k-1} \mathbf{K}_{k-1} \mathbf{D}_{v_t} \mathbf{Q}_{v_t} \Gamma_{v_t}^T && \text{Measurement noise cross-covariance propagation (with preceding update)} \\ \mathbf{P}_{z v_t, k}^- &= \Phi_{k-1} \mathbf{P}_{z v_t, k-1}^+ \Phi_{v_t}^T && \text{Measurement noise cross-covariance propagation (without preceding update)} \\ \mathbf{P}_{v_t v_t, k}^- &= \Phi_{v_t} \mathbf{P}_{v_t v_t, k-1} \Phi_{v_t}^T + \Gamma_{v_t} \mathbf{Q}_{v_t} \Gamma_{v_t}^T && \text{Measurement noise covariance propagation}\end{aligned}$$

Template 4-22: Update equations**Navigation state update**

$$\begin{aligned}\mathbf{R}_k &= \sum_{i=1}^r \left(\mathbf{H}_{v_i} \mathbf{P}_{v_i v_i, k} \mathbf{H}_{v_i}^T + \mathbf{D}_{v_i} \mathbf{Q}_{v_i} \mathbf{D}_{v_i}^T \right) && \text{Measurement covariance} \\ \mathbf{s}_k &= \tilde{\mathbf{y}}_k - \mathbf{H}_k \hat{\mathbf{z}}_k^- && \text{Innovation} \\ \mathbf{S}_k &= \mathbf{H}_k \mathbf{P}_{zz,k}^- \mathbf{H}_k^T + \mathbf{R}_k + \sum_{i=1}^r \left(\mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^- \right) \mathbf{H}_k^T + \mathbf{H}_k \sum_{i=1}^r \mathbf{P}_{z v_i, k}^- \mathbf{H}_{v_i}^T && \text{Innovation covariance} \\ \mathbf{K}_k &= \left(\mathbf{P}_{zz,k}^- \mathbf{H}_k^T + \sum_{i=1}^r \mathbf{P}_{z v_i, k}^- \mathbf{H}_{v_i}^T \right) \mathbf{S}_k^{-1} && \text{Kalman gain} \\ \hat{\mathbf{z}}_k^+ &= \hat{\mathbf{z}}_k^- + \mathbf{K}_k \mathbf{s}_k && \text{Navigation state update} \\ \mathbf{P}_{zz,k}^+ &= (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{zz,k}^- - \mathbf{K}_k \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^- && \text{Navigation covariance update} \\ &= (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{zz,k}^- (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T && \text{(Joseph's form)} \\ &\quad - \mathbf{K}_k \sum_{i=1}^r (\mathbf{H}_{v_i} \mathbf{P}_{v_i z, k}^-) (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k)^T - (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \sum_{i=1}^r (\mathbf{P}_{z v_i, k}^- \mathbf{H}_{v_i}^T) \mathbf{K}_k^T\end{aligned}$$

Only the state and measurement noise cross-covariances that are related to the measurement are to be used in the sums!

Process noise update

$$\mathbf{P}_{z \omega_s, k}^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{z \omega_s, k}^- \quad \text{Process noise cross - covariance update}$$

Measurement noise update

$$\mathbf{P}_{z v_t, k}^+ = \begin{cases} (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{z v_t, k}^- - \mathbf{K}_k \mathbf{H}_{v_t} \mathbf{P}_{v_t v_t, k} & \text{Measurement noise cross - covariance update related to the measurement} \\ (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{z v_t, k}^- & \text{all others} \end{cases}$$

4.6 Integration of Reality Models into Navigation Filter

In this section, the reality models for the IMU errors, gravity model error, random vibration, satellite clock error, satellite orbit error, ionosphere error, receiver clock error and receiver noise, which have been derived in chapter 3, are integrated into the navigation error filter one after another. Bias-like errors are integrated according to section 4.4 and noise-like errors that shall only be statistically considered are integrated according to section 4.5. For each error, it is analyzed how the error affects the integrated navigation solution error. Recommendations are given which error components shall be regarded in the navigation error filter and which model order is necessary to represent the main characteristic of the error on the one hand but to keep the additional computational effort as low as possible.

4.6.1 Inertial Measurement Unit Errors

Turn-on biases, scale factor errors and misalignment errors can be straightforwardly integrated into the navigation error filter since they are modeled as constants. The turn-on biases will be usually estimated. The misalignments are generally only considered. The scale factor errors are estimated or considered depending on the magnitudes of the measurement signals. If for example the carrier platform is excited by high monotonous angular rates like in a spinning missile, it is recommended to estimate the scale factor errors, too.

The focus here is on the bias instability as noise-like error and misalignments as bias-like errors, which shall be considered but not estimated by the navigation filter. Their influence on the integrated navigation solution shall be clarified by means of an example.

The General Aviation Aircraft scenario (section E.1.2) is chosen for the analysis. IMU 1 and IMU 2 (Table E-5) with 100 Hz sample rate are compared. The IMU measurements are solely affected by velocity/angular random walk noise N and bias instability noise B or misalignment errors, respectively. All other IMU errors are set to zero. A generic GPS receiver provides 1 Hz pseudorange measurements to six satellites in view, which are only falsified by white Gaussian noise with 2 m standard deviation (section E.2.2). The filter performance is analyzed by comparing the predicted covariance with the covariance of the actually realized navigation state error of 100 Monte Carlo runs. The effects of the bias instability error and misalignments are separately analyzed. First, the IMU measurements are only corrupted by velocity/angular random walk noise N and bias instability noise B . The bias instability noise is generated by a 4th order model (3.27). Second, the IMU measurements are affected by velocity/angular random walk noise N and misalignment errors. Contrary to the IMU specifications, the misalignment angles are randomly drawn with unitary standard deviation of 0.5 mrad.

Neglecting the bias instability

In the first case, lower grade IMU 1 is used. Figure 4-47 shows the position and orientation errors of the navigation solution if the bias instability of the accelerometers and gyroscopes is neglected and only the white noise is properly accounted for in the process noise covariance matrix. The blue dotted lines represent the 3σ standard deviations as predicted by the filter. The grey dotted curves are the true 3σ standard deviations provided by the Monte Carlo simulation. As expected, the actual

deviations are much larger than the predicted ones. This situation is dangerous since valid aiding measurements could be rejected by the integrity monitor.

In a further simulation run, IMU 1 is replaced by the higher grade IMU 2. The bias instability input noise spectral density B of the gyroscopes is about one fiftieth of IMU 1. As can be seen in Figure 4-48, the influence of the bias instability is much smaller than with IMU 1. In fact, the influence of the bias instability depends on the ratio of white noise spectral density over bias instability input noise spectral density N/B . It can be concluded that the bias instability has only to be considered if this ratio exceeds a certain limit. For IMU 1, the ratio N/B is about 3.1 for the gyroscopes and 0.7 for the accelerometers. For IMU 2, the ratio N/B is about 42 for the gyroscopes and 5 for the accelerometers.

High order bias instability model

Next, six 4th order bias instability models for each of the three accelerometer and gyroscope axes are incorporated into the integrated navigation filter. The models in the navigation error filter are principally the same models that have been used for the synthesis of the measurements in the simulation. The predicted and actual covariances should perfectly match now. This is in fact confirmed by Figure 4-49. Merely the actual standard deviation of the heading error ψ_{nn} deviates slightly. This is probably due to the too low number of Monte Carlo runs. It is interesting to observe that the obtained standard deviations are smaller than the actually realized standard deviations in the first case where the bias instability model has been neglected. The predicted standard deviations are slightly larger than the predicted standard deviations of the first case.

Low order bias instability model

In order to noticeably lower the computational load, it is analyzed if a 1st order model with only two states can already represent the main portion of the bias instability noise effectually. The results are plotted in Figure 4-50 for IMU 1 and in Figure 4-51 for IMU 2. It seems that the results with the 1st order models do not deviate much from those of the 4th order models. For IMU 2, the 1st order model should be actually adequate.

Neglecting the time correlation of the bias instability

The numerical effort could be dramatically lowered if the time correlation is neglected and the bias instability is approximated by white noise. Then, the corresponding covariance can be computed offline with the 4th order model and only the process noise covariance matrix has to be adequately adjusted. However, as demonstrated by the position and orientation errors in Figure 4-52, the true standard deviations exceed the predicted ones. This is especially true for the position error. It can be concluded that it is not advisable to neglect the time correlation of the bias instability noise.

Neglecting misalignment cross-correlations

After having analyzed the bias instability, the influence of misalignment errors is considered next. IMU 2 is used for this analysis. As with the bias instability, the time correlation of the misalignments is initially neglected by the navigation filter. Only the covariance of the misalignments is considered. The resulting position and orientation errors are illustrated in Figure 4-54. It can be seen that the actual navigation error standard deviations are clearly larger than the predicted ones.

Consideration of misalignment cross-correlations

Finally, the time correlation of the misalignments is additionally considered by the integrated navigation filter. In Figure 4-53, the corresponding navigation errors are shown. It is observable that the actual and predicted statistics coincide well.

Conclusion

In terms of the bias instability, it has been revealed that low order bias instability models are already adequate to represent this noise. If 1st order bias instability models are applied, only six additional cross-covariances of dimension $n \times 2$ have to be additionally considered by the filter, where n is the dimension of the navigation state vector to be estimated. The bias instability covariance can be assumed to be constant and does not have to be propagated by the filter.

The consideration of the bias instability by the navigation error filter is only necessary if the ratio of the white noise density to bias instability input noise density N/B is low. This is most often the case with low grade IMU and less often with high grade IMU.

The bias-like misalignment errors, however, should be always considered by the integrated navigation filter. Their integration into the navigation error filter is straightforward. Only six additional cross-covariances of dimension $n \times 1$ have to be propagated by the filter. The misorientation of the complete gyroscope triad with respect to the reference frame should be considered in the same manner. It has to be kept in mind that the navigation system is not operated as an end in itself. The orientation angles are required by the flight guidance and control system and other important subsystems. Therefore, the orientation of the IMU reference axes and the corresponding uncertainty is of particular importance. The correct statistical consideration of internal axis misalignments and IMU installation angle errors plays an important role especially for other navigation sensors that require the relative orientation with respect to the IMU axes like cameras of image aided navigation systems.

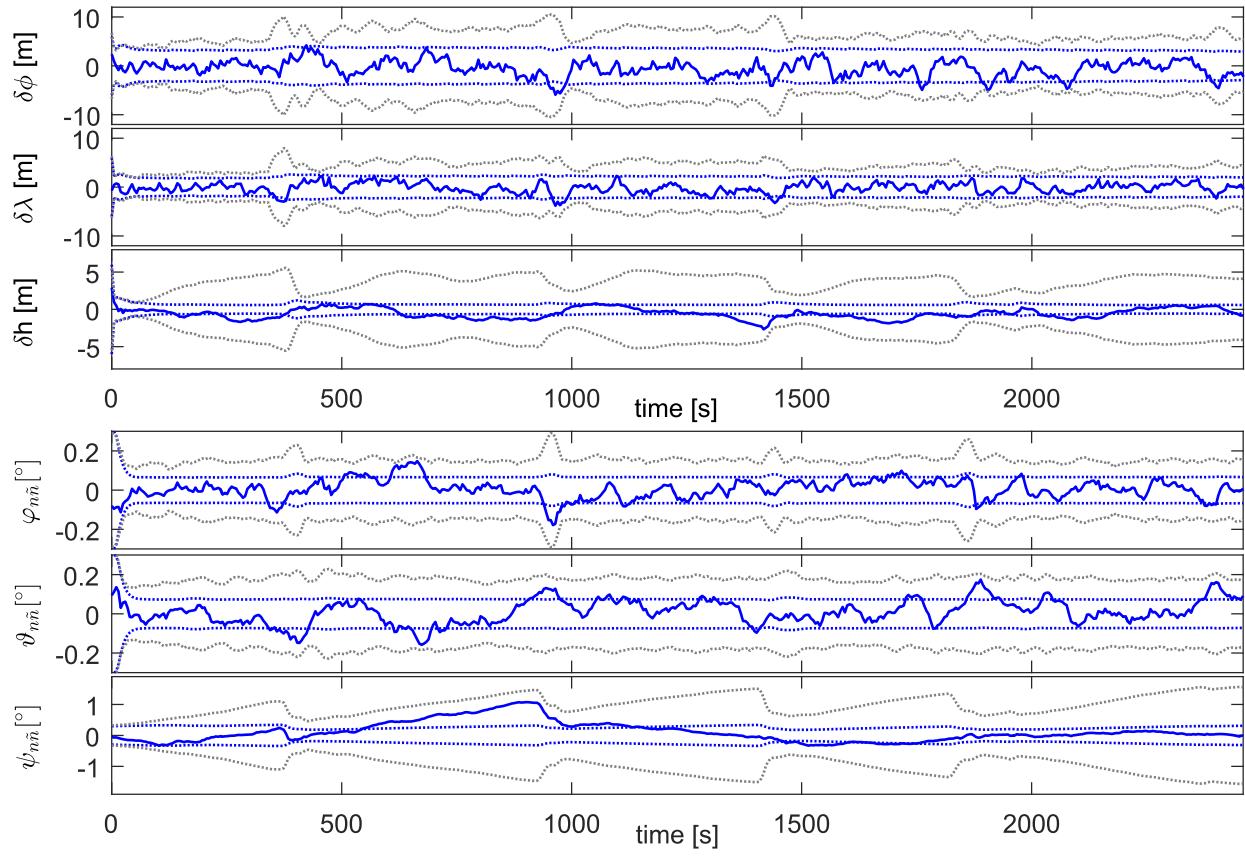


Figure 4-47: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if bias instability is not considered (IMU 1)

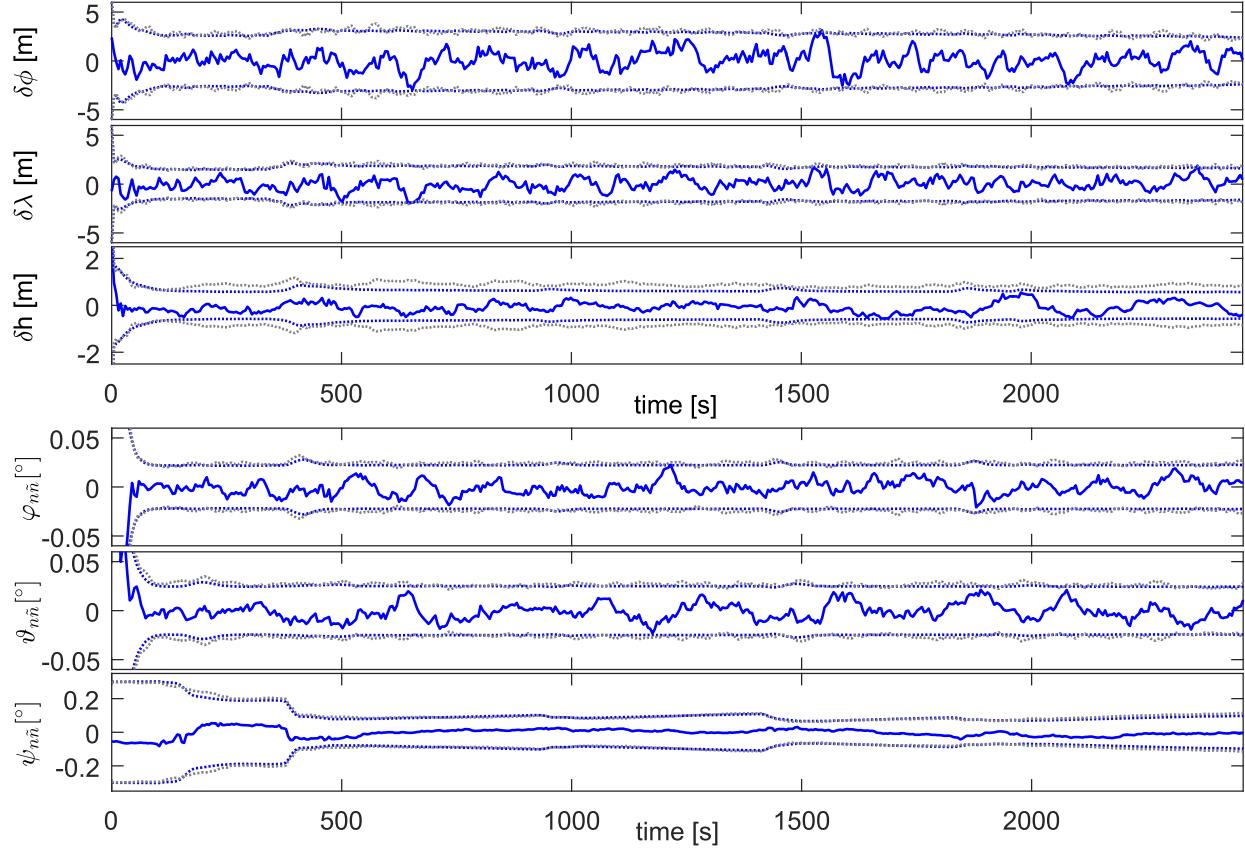


Figure 4-48: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if bias instability is not considered (IMU 2)

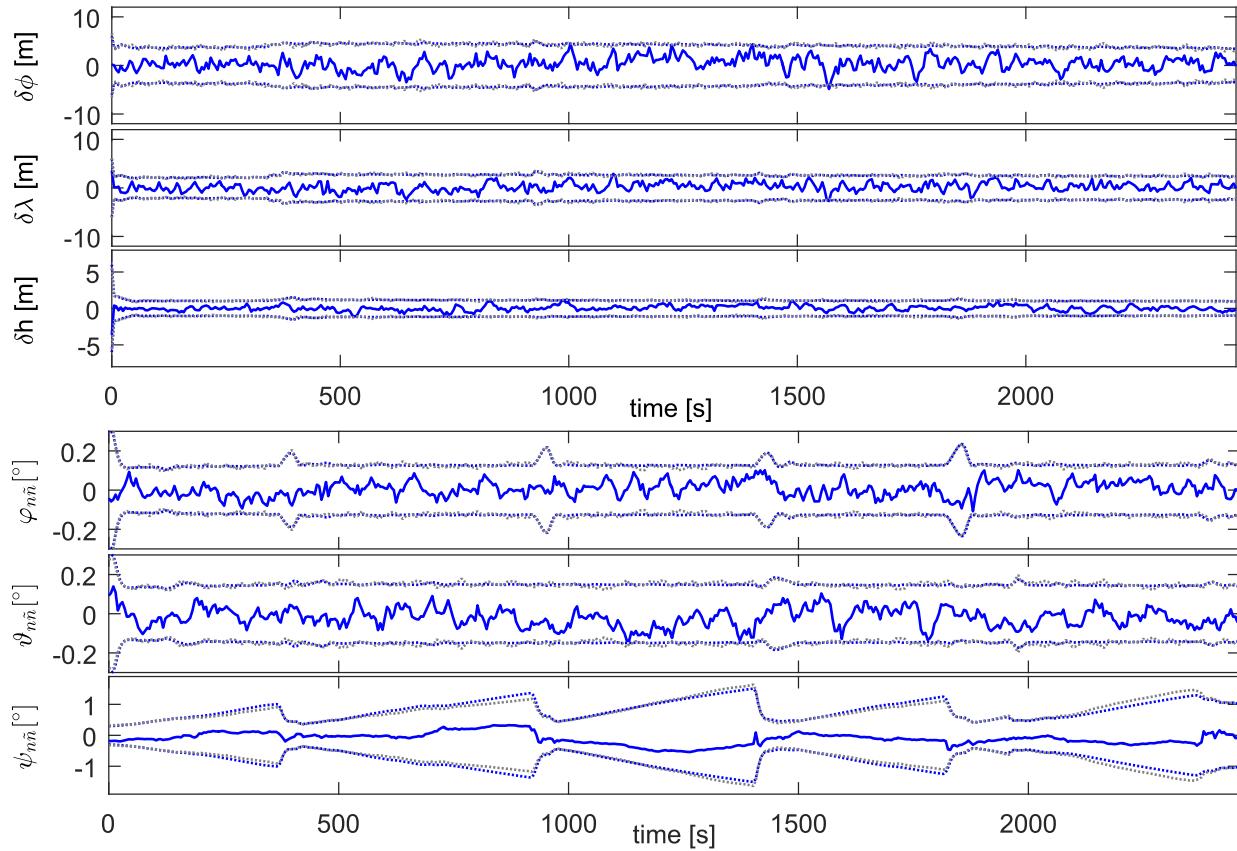


Figure 4-49: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if 4th order bias instability model is used (IMU 1)

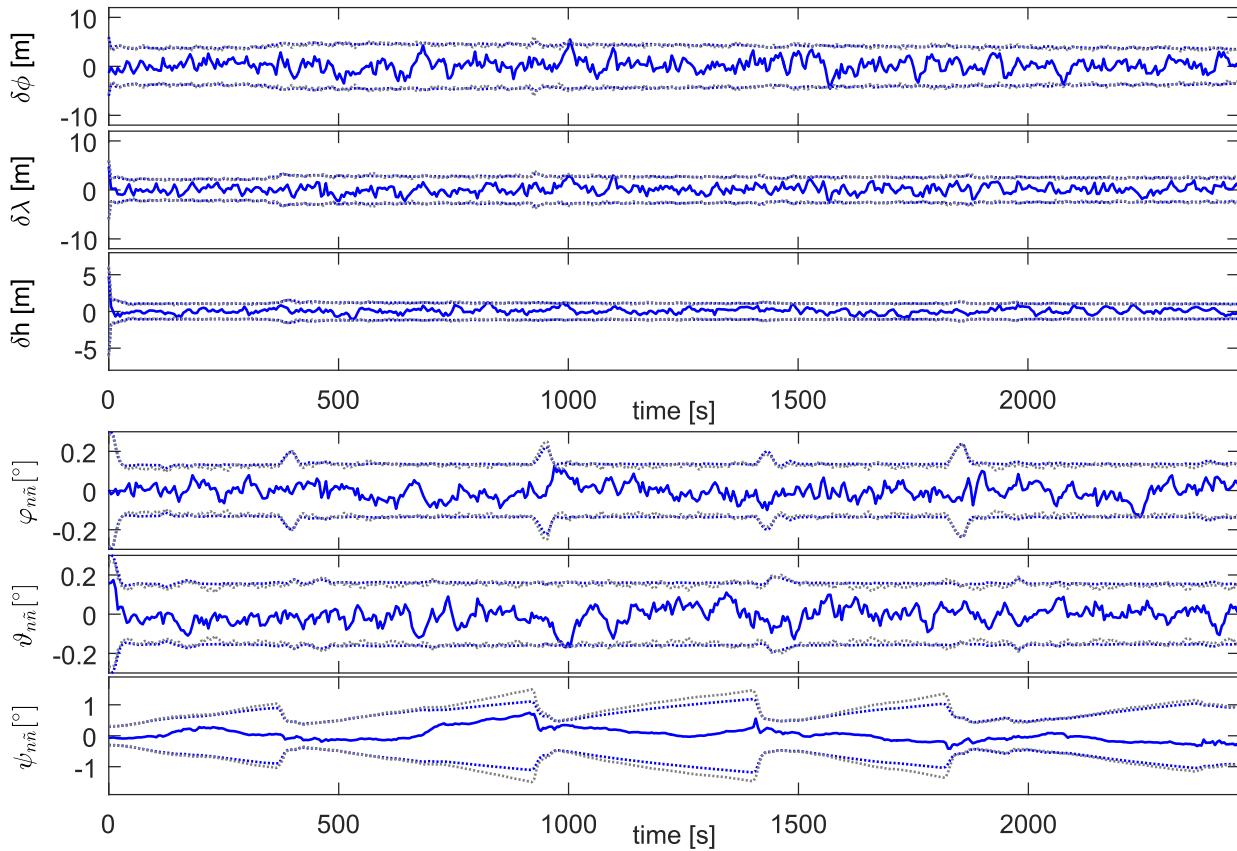


Figure 4-50: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if 1st order bias instability model is used (IMU 1)

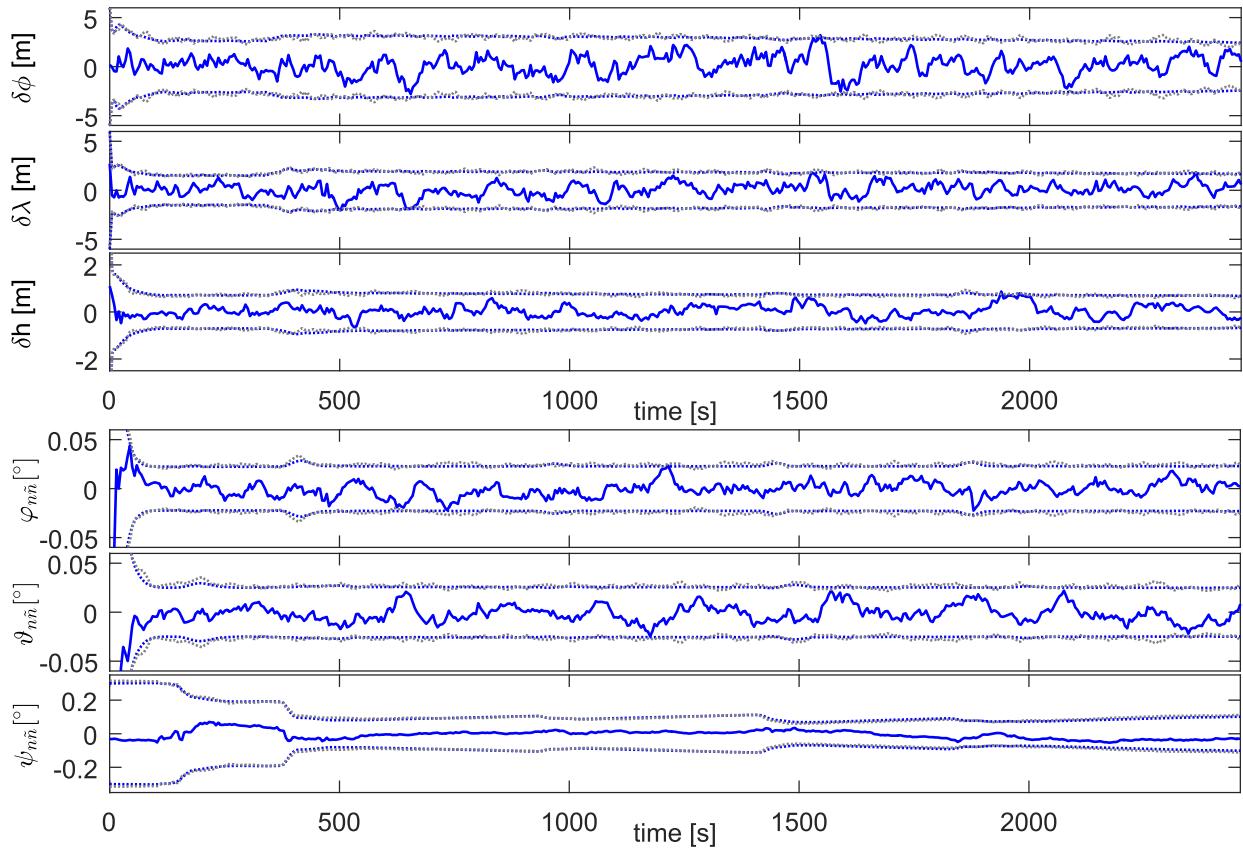


Figure 4-51: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if 1st order bias instability model is used (IMU 2)

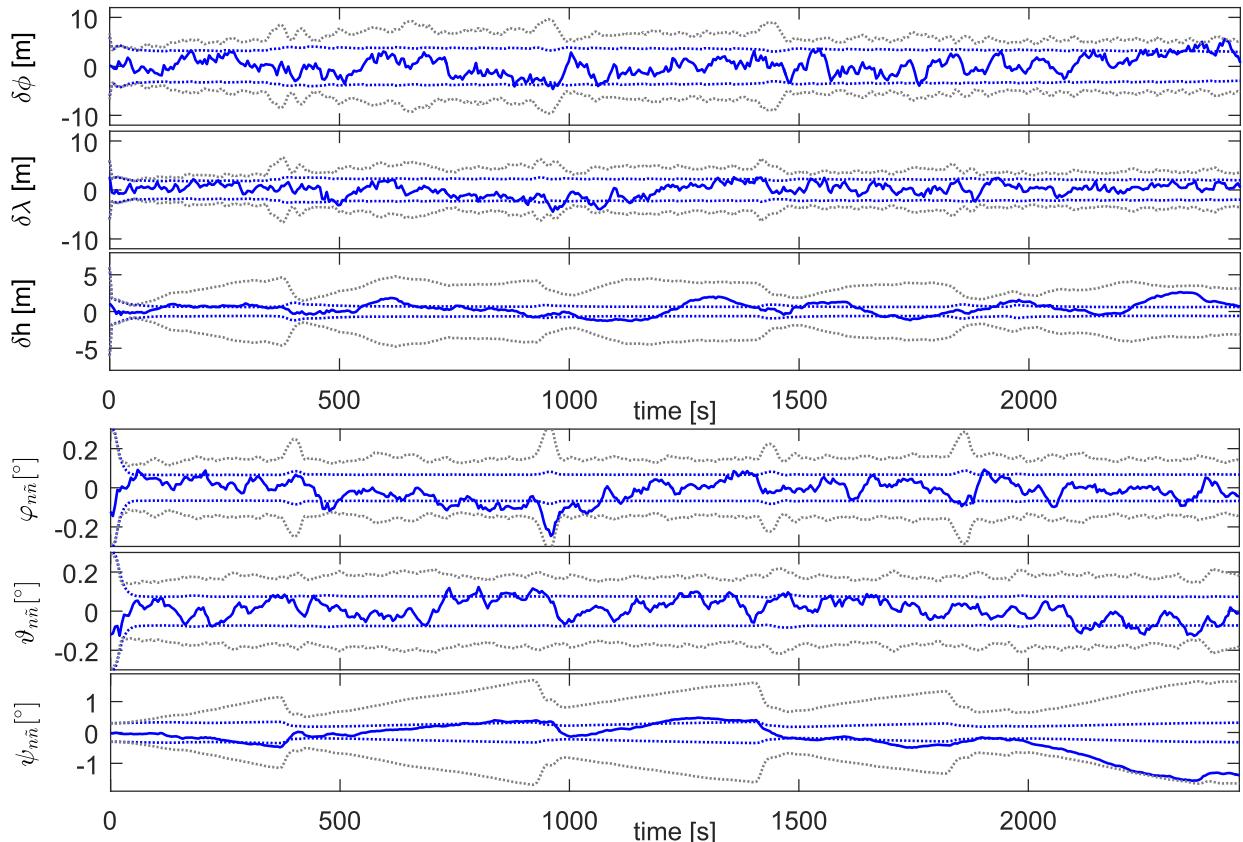


Figure 4-52: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if 4th order bias instability model is used, but time correlation is neglected (IMU 1)

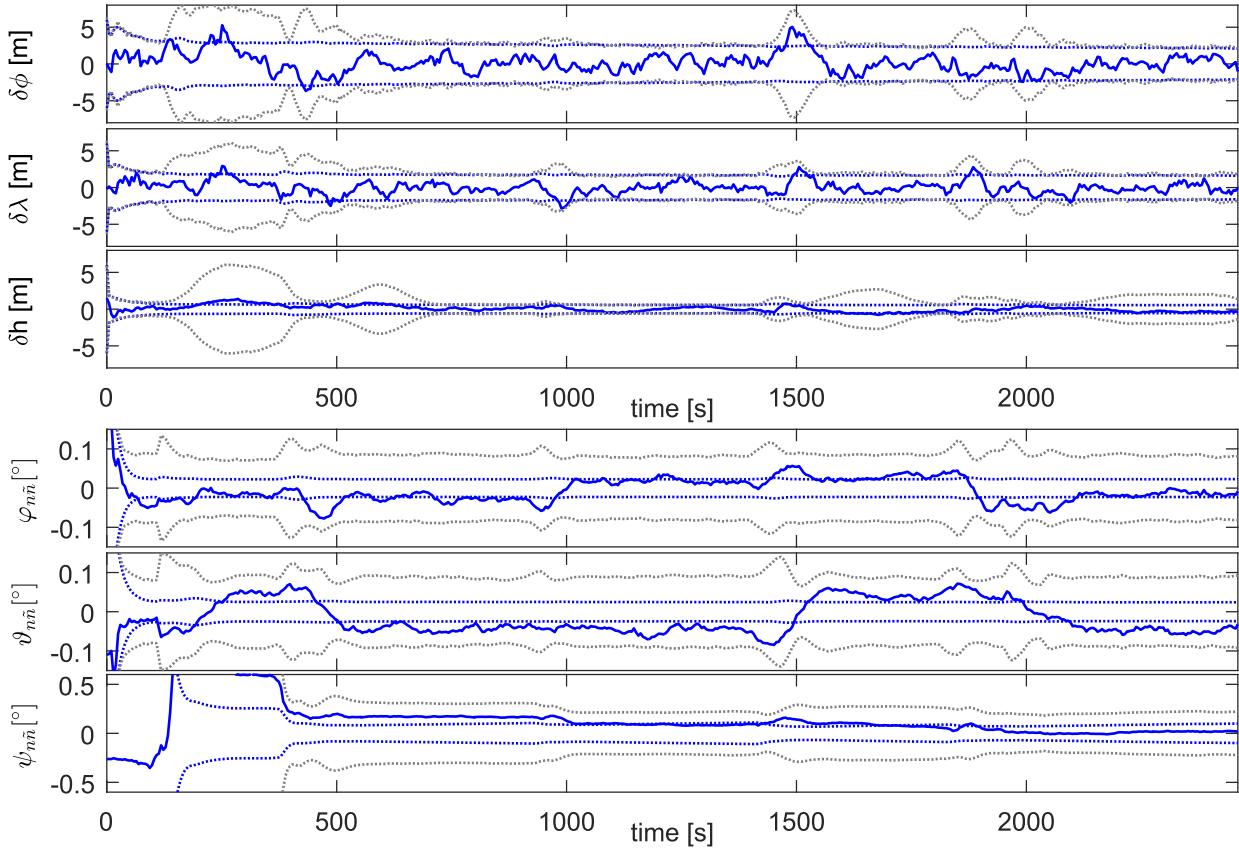


Figure 4-53: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if misalignments are not considered (IMU 2)

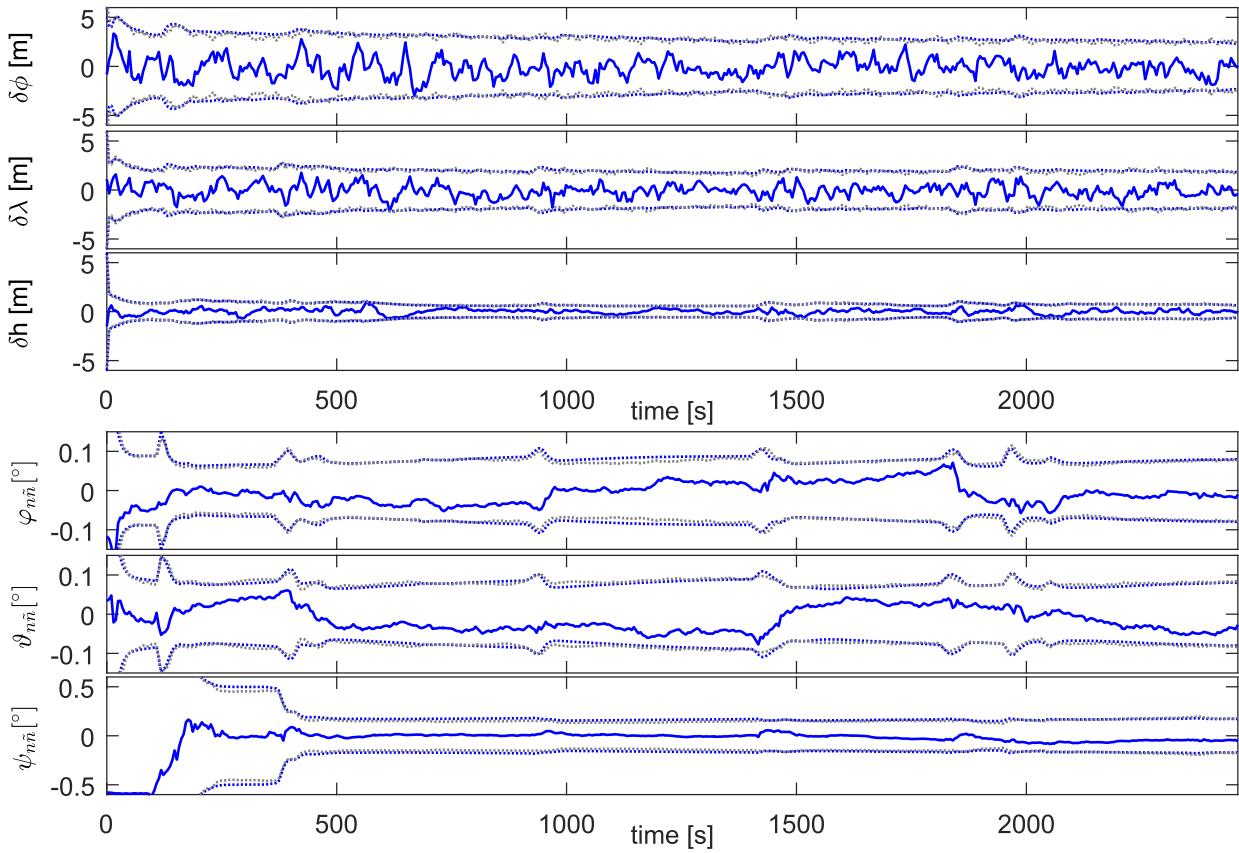


Figure 4-54: Position and orientation error and 3σ boundaries as predicted by filter (blue dotted lines) and from Monte Carlo simulation (grey dotted lines) if misalignments are considered (IMU 2)

4.6.2 Gravity Model Error

The effectiveness of the application of a stochastic gravity error model on the accuracy of the inertial navigation solution shall be demonstrated by means of an example. For that, a flight across the Alps in south-west direction starting from Salzburg is analyzed. The flight is carried out straight and leveled with constant height and velocity $\mathbf{v}_n^T = (-75, -150, 0) \text{ m/s}$. The ground track and the gravity error of the ellipsoidal gravity model with respect to the EGM2008 model in down direction $\delta\gamma_n$ are shown in Figure 4-55. The navigation error covariance is propagated once with and once without considering the gravity error.

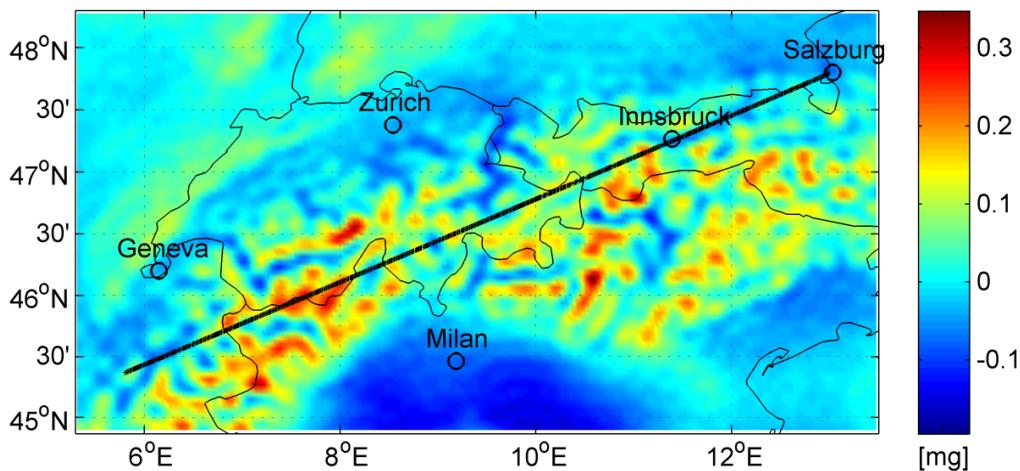


Figure 4-55: Ground track of the analyzed flight trajectory across the Alps from Salzburg in south-west direction. The color indicates the normal gravity error w.r.t. EGM2008

The inertial navigation solution is computed using Somigliana's gravity formula. Since the focus is on the influence of the gravity model error, inertial measurements are freely integrated without external aiding (even the barometer aiding of the unstable height channel is omitted) and it is assumed that the accelerometers and gyroscopes are solely affected by white noise. Furthermore, the initial state is perfectly known with zero covariance. The inertial sensors are navigation grade with velocity/angular random walk power spectral densities $8 \mu\text{g}/\sqrt{\text{Hz}}$ and $0.0025^\circ/\sqrt{\text{h}}$, respectively.

The stochastic gravity error model parameters that have been estimated for the Alps region as had been done before for the U.S. territories are

$$\sigma_T = 15.6167 \text{ m}^2/\text{s}^2, \quad \beta = 3.14 \cdot 10^{-5} \text{ m}^{-1} \quad (4.168)$$

Note that the standard deviation of the disturbing gravity potential σ_T is almost three times higher than in the Midwestern and Southern U.S. territories. The corresponding correlation length is $\frac{2.146}{\beta} = 68.3 \text{ km}$ and is thus, as expected, shorter than in the Midwestern and Southern U.S. territories. This is due to the higher gravity variations in mountainous regions.

In the case in which the gravity error is considered, the navigation error filter uses the propagation step of the Schmidt-Kalman filter for colored process noise (Template 4-21). Figure 4-56 shows the position, velocity and orientation errors of ten Monte Carlo runs and the propagated standard deviations with and without consideration of the gravity error. Note that the gravity error is the same in all Monte Carlo runs and only the IMU measurement noise is truly random. The error trends of the Monte Carlo runs are therefore similar. It can be observed that the height error is affected more

severely than the horizontal position errors. This is due to the instability of the height channel. The magnitudes of the horizontal position errors after one hour are as expected for the navigation grade class. The gravity error influences the translational as well as rotational navigation states. The predicted covariance that considers the gravity error seems to match the actual (unknown) covariance better than the predicted covariance without gravity error consideration. The predicted covariance that does not consider the gravity error seems to be too optimistic. The height channel prediction with gravity error consideration might be slightly too conservative.

Figure 4-57 shows the true gravity errors in north, east and down directions along the flight path at height $h=0$ and a realization of the applied stochastic gravity error model. It can be seen that the magnitude as well as the time correlation coincide qualitatively well. Therefore, the stochastic model with the chosen parameters represents the effect of the gravity model error quite well.

Conclusion

The combination of a simple normal gravity model like Somigliana with a stochastic model for the uncompensated residual error makes sense if the navigation system shall be able to operate longer time spans without external aiding measurements. This is especially interesting for navigation grade IMU that are innately capable of bridging longer unaided flight periods due to their low measurement errors. In these periods, the residual gravity error can have a considerable influence on the accuracy of the inertial navigation solution. Tables of standard deviations and correlation length parameters for different regions of the world or for the intended area of operation can be stored on the navigation system. Alternatively to the Somigliana gravity model and stochastic model for the uncompensated gravity error, the EGM2008 model can be used with sufficiently high degree and order. It is an advantage of the EGM model that gravity values can be directly calculated for all heights without the need of any height continuation. The effort to compute gravity values in real-time might however be too high for embedded systems. Alternatively, pre-processed tables of gravity model errors can be filed in the storage of the navigation system. Then, the current gravity value has only to be linearly interpolated from the table during run-time. Table values can also be computed for different heights, depending on the expected envelope of the application.

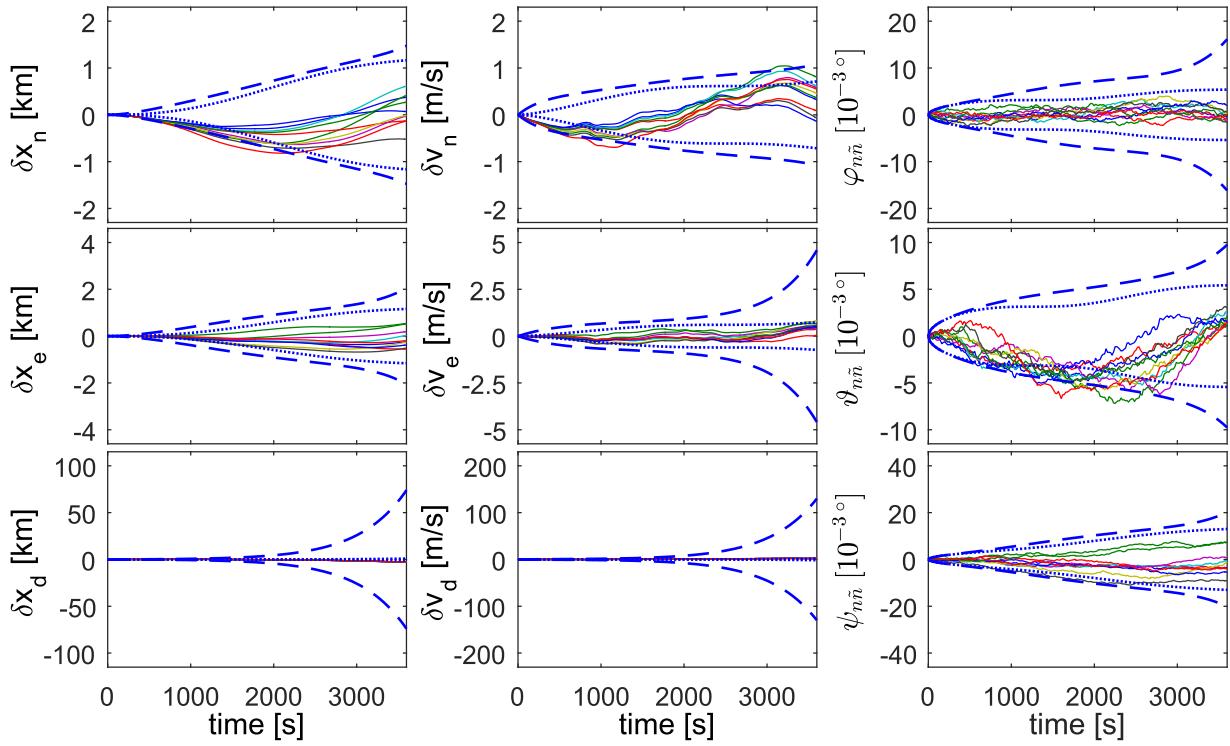


Figure 4-56: Inertial navigation errors and predicted 3σ standard deviations w/o gravity error model (dotted) and with gravity error model (dashed) at height $h = 0$

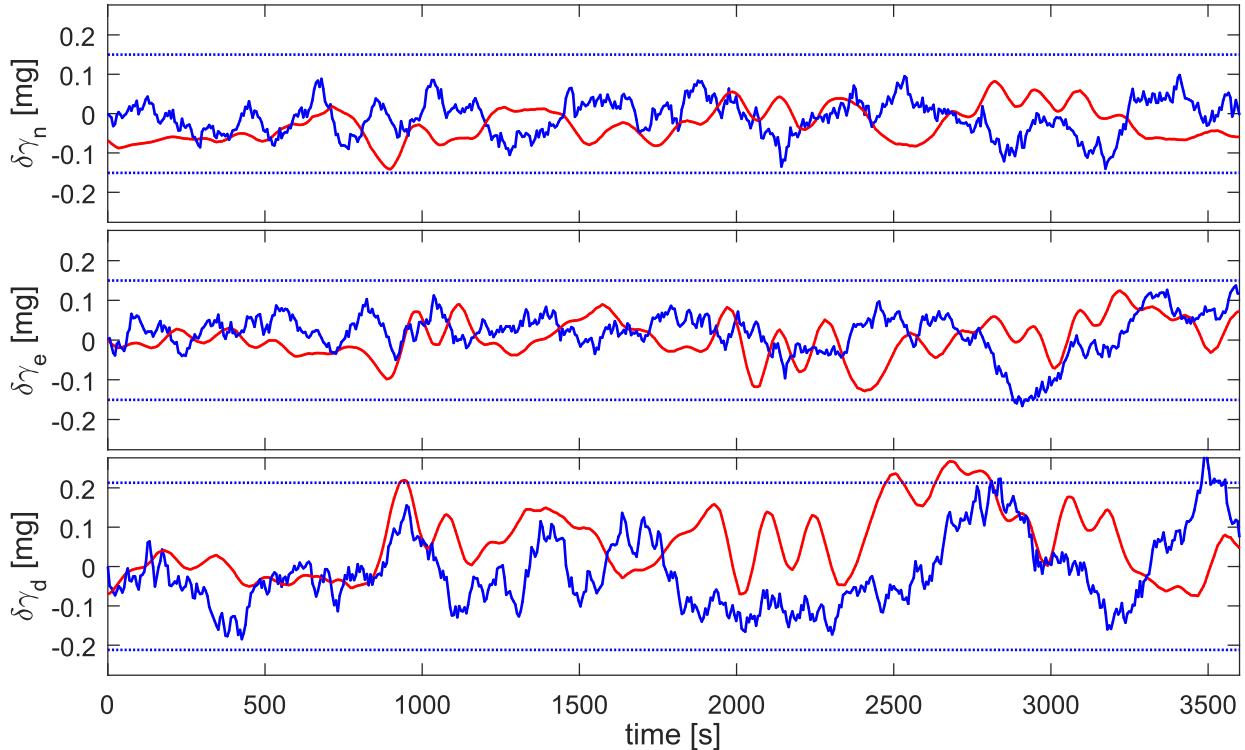


Figure 4-57: True gravity errors (red) and stochastically modeled gravity errors (blue) at height $h = 0$

4.6.3 Random Vibration

The General Aviation Aircraft scenario (section E.1.2) is used as example to demonstrate the effect of random vibration on the navigation state estimation error. IMU 3 (Table E-5) with 200 Hz sample rate is selected. The IMU measurements are solely affected by velocity/angular random walk noise N and additional vibrational noise. All other IMU errors are set to zero. A generic GPS receiver provides 1 Hz pseudorange measurements to six satellites in view, which are only falsified by white Gaussian noise with 2 m standard deviation (section E.2.2). The filter performance is analyzed by comparing the predicted covariance with the covariance of the actually realized navigation state error of 100 Monte Carlo runs.

Generation of colored noise with specified PSD

The red curve in Figure 4-59 represents the desired normalized PSD of typical vibrational noise. It has a steep increasing slope at 40 Hz, a constant level of 45 dB/Hz between 40 and 70 Hz and drops off with 21 dB/Hz per decade for frequencies higher than 70 Hz. The accelerometer vibrational noise is obtained by scaling the normalized PSD with the accelerometer white noise PSD, the gyroscope noise accordingly by using the gyroscope white noise PSD for scaling. The generation of colored noise with the specified PSD S begins with the discretization of S in the range between 0 Hz and the Nyquist frequency $f = 1/(2\Delta t)$ with $N/2$ samples, yielding the samples S_k at frequencies $f_k = k/(N\Delta t)$, $k = 1 \dots N/2$. N is the number of required samples in time domain and Δt is the sample time of the IMU. Then, $N/2$ complex-valued noise samples Z_k have to be created in the frequency domain with amplitude $\sqrt{S_k/\Delta t}$

$$Z_k = \sqrt{\frac{S_k}{\Delta t}} e^{2\pi i \eta_k} \quad (4.169)$$

where η_k is white uniformly distributed noise, that is $\eta_k \sim WU(0, 1)$. Next, the array of complex noise samples is extended to satisfy symmetry according to

$$Z_k = \{0, Z_1, \dots, Z_k, Z_k^*, \dots, Z_1^*\} \quad (4.170)$$

Therein, Z_k^* is the conjugated complex of Z_k . The searched noise in time domain is finally obtained by transforming the extended complex noise (4.170) from frequency to time domain by inverse DFT and scaling with \sqrt{N}

$$\omega_{vib,k} = \sqrt{N} \mathcal{F}^{-1}(Z_k) \quad (4.171)$$

Figure 4-58 shows realizations of the vibrational noise on the accelerometer measurements and on the gyroscope measurements. The oscillating character of the generated vibrational noise is well observable. This is due to the quite narrow frequency band (40 to 70 Hz). The realized noise is more or less negatively correlated noise. The corresponding normalized PSD of the realizations is illustrated by the blue curve in Figure 4-59. Generally, the effect of negatively correlated noise on the navigation state estimation error is smaller than of positively correlated noise since subsequent samples are averaged by the integration.

First, the vibrational noise is not considered by the navigation filter. The process noise covariance matrix considers only the white sensor noise of the accelerometers and gyroscopes. The effect of the random vibration is thus neglected. Second, the vibrational noise is considered in the navigation filter

with magnitude and time correlation. For that, the IMU signals are high-pass filtered with a fifth order Butterworth filter. The coefficients of ARMA(1,0), ARMA(3,0) and ARMA(7,0) models are online estimated with Burg's method. The estimated vibration noise models are then integrated into the navigation filter as described in section 4.5. The window size is 400 samples. Third, only the current covariance of the estimated ARMA models is taken, but the vibrational noise is assumed to be white and the time correlation is neglected. Fourth and lastly, the vibrational noise is simply encountered by increased constant process noise without online model estimation.

Figure 4-60 shows the position error and Figure 4-61 the orientation error for the first case when the vibrations are not considered by the navigation filter. Besides the results of an arbitrarily selected run, the true 3σ standard deviations of all Monte Carlo runs are plotted as grey dotted lines and the 3σ standard deviations that are predicted by the navigation filter as blue dotted lines. It can be observed that the actual navigation state errors are much larger than pretended by the predicted statistics of the filter. This situation is dangerous since the predicted error covariance is too optimistic. The filter runs the risk to reject aiding measurements due to wrong innovation statistics. In Figure 4-64 and Figure 4-65 the position and orientation errors are shown for the second case with online estimated vibrational noise models. The higher the order of the model, the better is the coincidence of predicted and true error statistics. The seventh order model actually predicts the true error statistics quite well and produces a statistically consistent solution. It is a positive fact that the true error bounds are smaller than the predicted error bounds in all cases. In Figure 4-64 and Figure 4-65 the position and orientation errors of the third case, in which the time correlation is neglected, are illustrated. The result of the first order model is similar to that of the previous case with consideration of the time correlation. The difference to the previous case appears more clearly with the third and seventh order models. The results are less consistent than with consideration of the time correlation. However, all results are again conservative with respect to the predicted and true error bounds. In Figure 4-66 and Figure 4-67 the position and orientation errors of the fourth case are plotted. The increased constant process noise covariance is obtained by scaling the velocity/angular random walk noise of the IMU once with factor 10 and once with factor 1000. Factor 10 seems to be too small since the actual 3σ standard deviations are larger than the predicted ones. Factor 1000 is in contrast too large since the predicted 3σ standard deviations are noticeably larger than the actual ones.

Conclusion

The analyzed random vibration primarily exhibits negatively correlated noise. This is because vibration consists in fact of superposed sinusoidal motion with different frequencies and amplitudes. The negative correlation effect comes from the insufficient sampling of the highly frequent oscillations. Owing to the negative correlation character of vibrational noise the identification time span of Burg's method can be quite short, for example one covariance propagation time step.

It turned out that the identification of the vibrational noise works fine and yields accurate and statistically consistent results if the model order is chosen sufficiently large. It is advantageous that the filter provides conservative results even if low order noise models are used, meaning that the predicted error statistics are larger than the actual ones. It is an interesting method for highly accurate navigation systems where the accuracy of the navigation solution is the decisive design criterion. The method implies higher computational load.

If no vibration identification is applied but only the process noise covariance is increased accordingly, satisfactory results can be obtained if the process noise covariance matrix is not too small. The process

noise tuning can be accomplished by means of simulations that also represent vibrations adequately and by analyzing recorded flight data with the IMU installed on the platform at the intended location. This procedure is laborious and especially difficult if the vibrations change during operation or if the navigation system is applied as black box on different platforms with different dynamics and vibrational noise. As a rule of thumb the process noise covariance should be tuned rather too large than optimal to obtain a robust navigation solution. The predicted error standard deviations then over-bound the actual ones, accepting that accuracy is given away in favor of robustness.

If a suitable constant process noise covariance matrix can be found, it is favored over the vibrational noise estimation method due to the lower complexity and computational effort. Moreover, if robustness for example in safety critical applications is the main design criterion, the constant covariance matrix with over-bounding is also preferred. Otherwise, if the vibrational noise is estimated online, the system designer does not have to bother with the tuning of the process noise covariance since it is automatically adapted by the algorithm. Furthermore, the time correlation is considered correctly.

A final remark: the identified noise models also represent parts of the IMU measurement errors. Sensor errors with short correlation lengths like the acceleration/angular rate white noise hence do not have to be considered separately. Only those errors whose correlation lengths are longer than the chosen identification time span of Burg's method, like biases or in-run bias instabilities, have still to be accounted for with extra models.

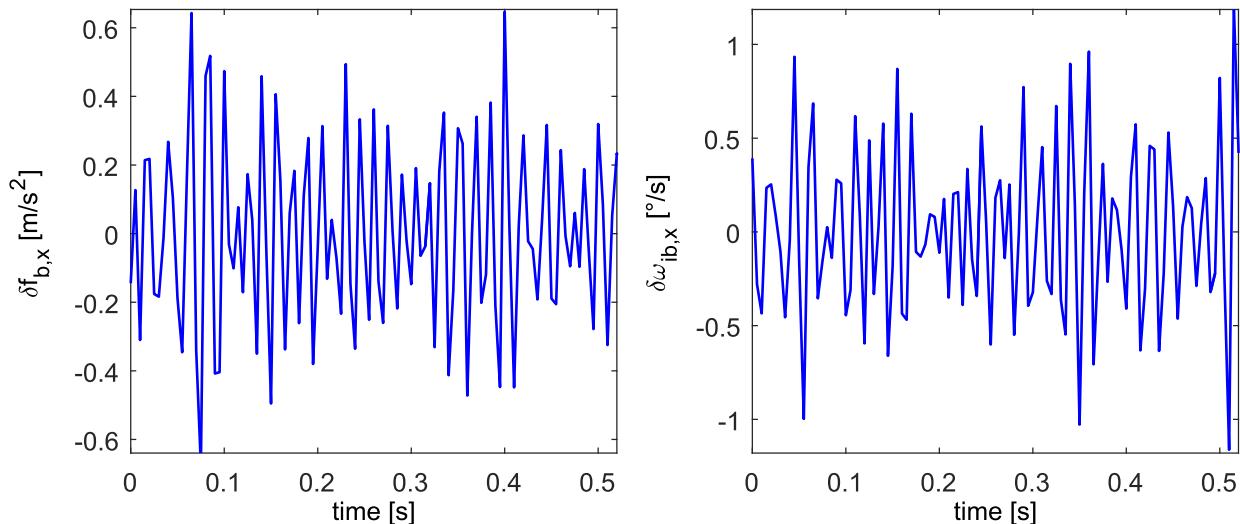
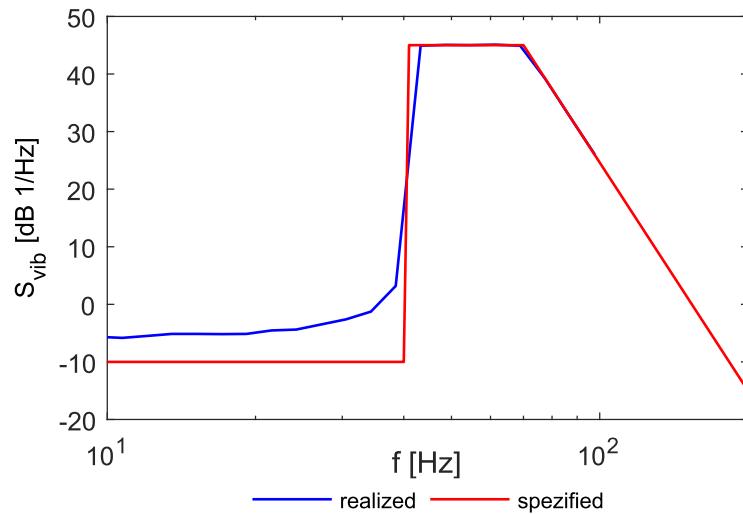
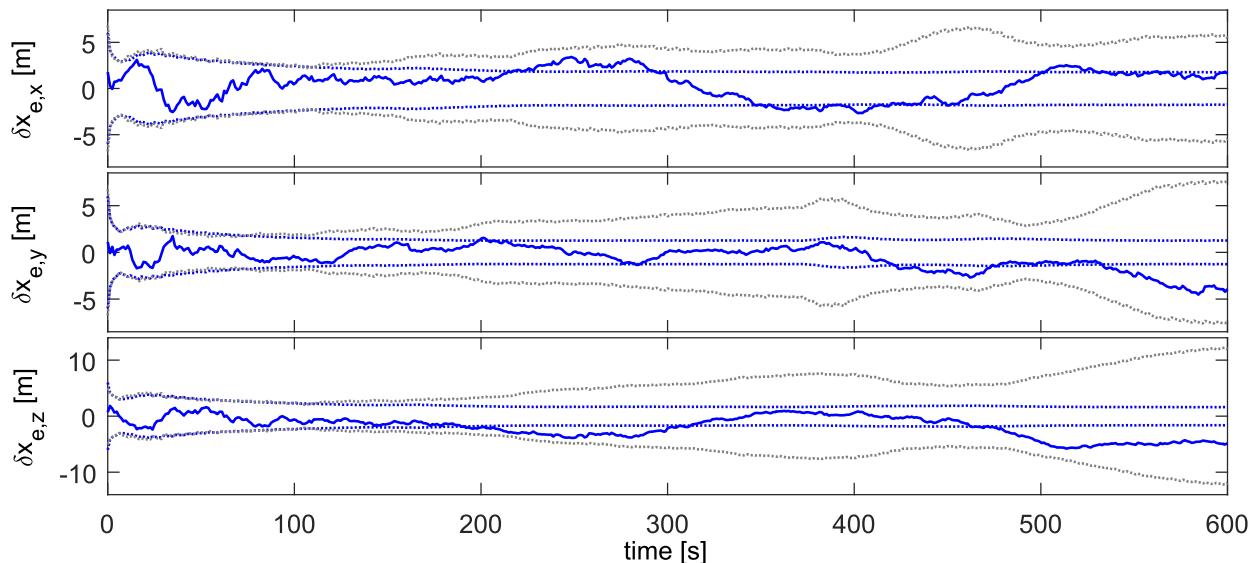
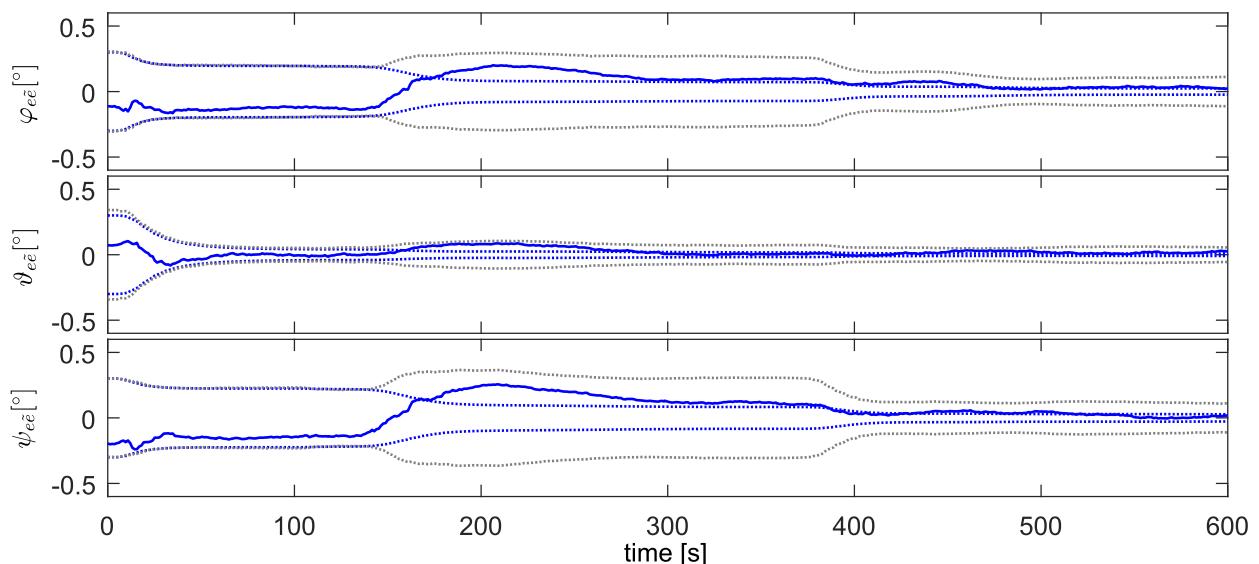


Figure 4-58: Vibrational noise on x-axis acceleration and angular rate

**Figure 4-59: Vibrational noise on accelerations****Figure 4-60: Position error without vibration model****Figure 4-61: Orientation error without vibration model**

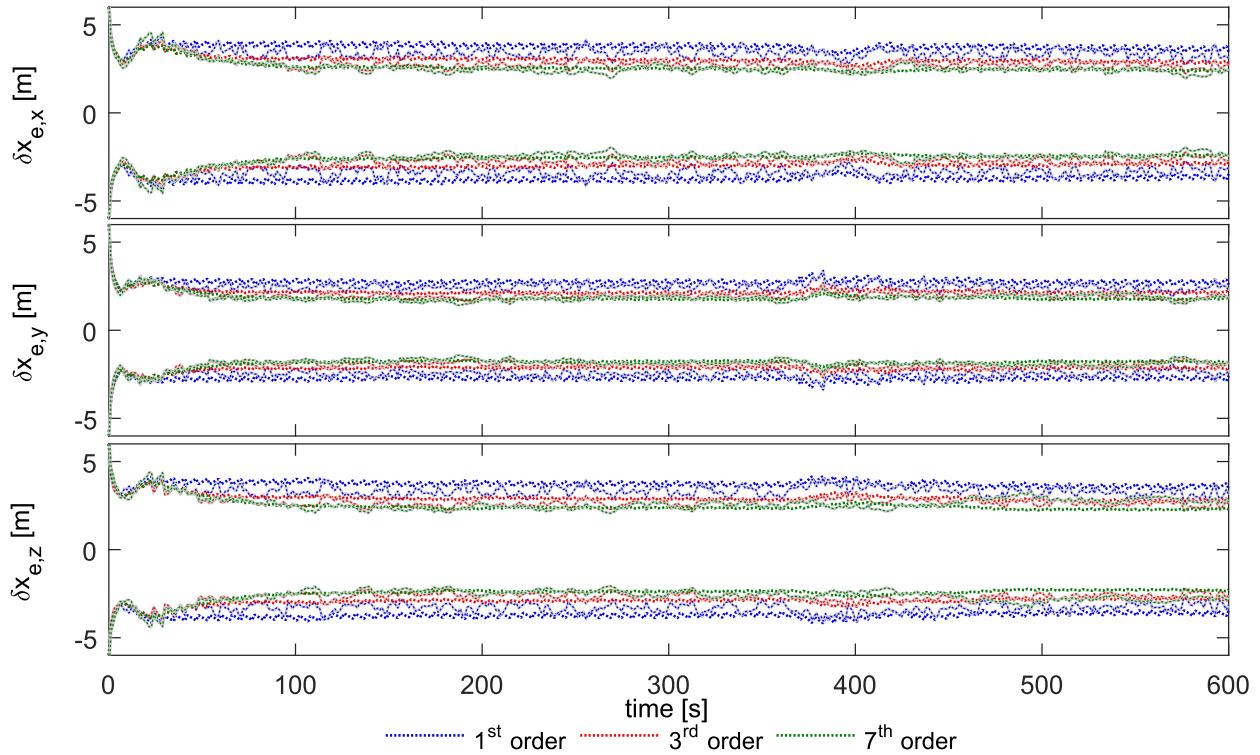


Figure 4-62: 3σ standard deviations of the position error with vibration models of different order. The colored dotted lines represent the predicted standard deviations, the grey and colored dotted lines the actual standard deviations of the Monte Carlo simulation

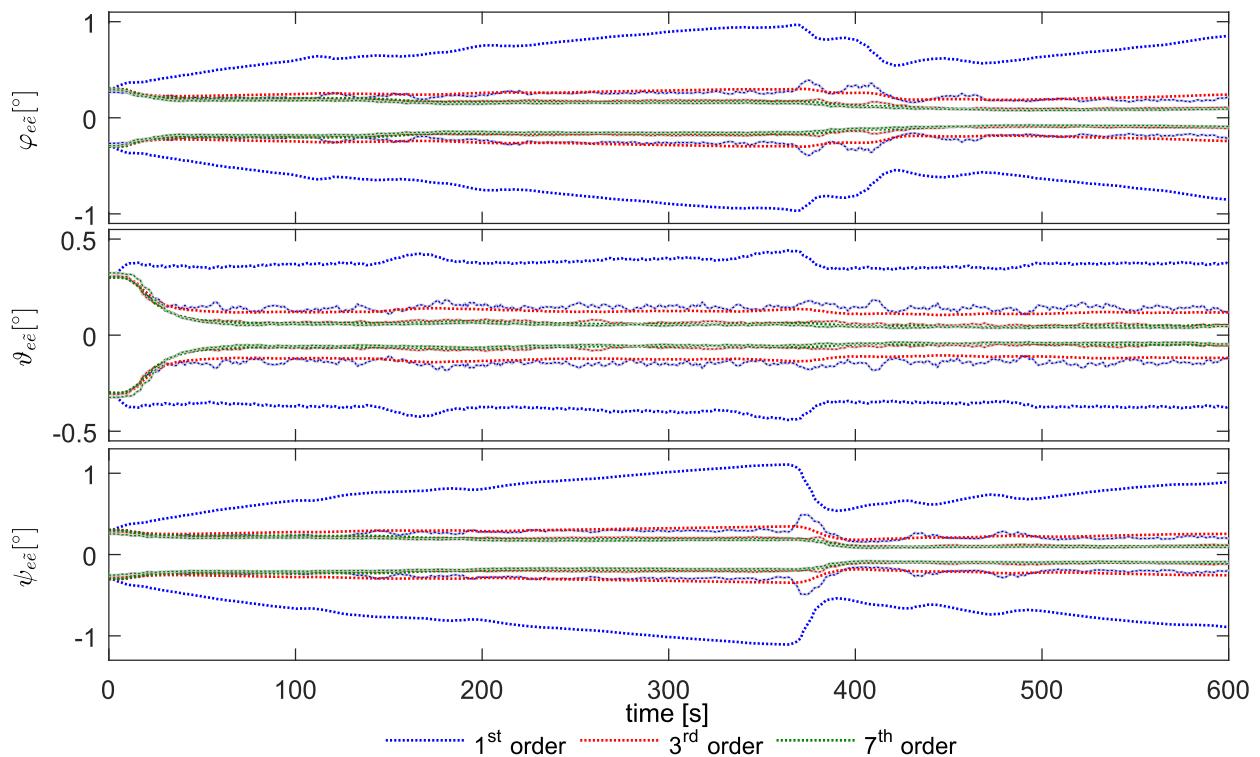


Figure 4-63: 3σ standard deviations of the orientation error with vibration models of different order. The colored dotted lines represent the predicted standard deviations, the grey and colored dotted lines the actual standard deviations of the Monte Carlo simulation

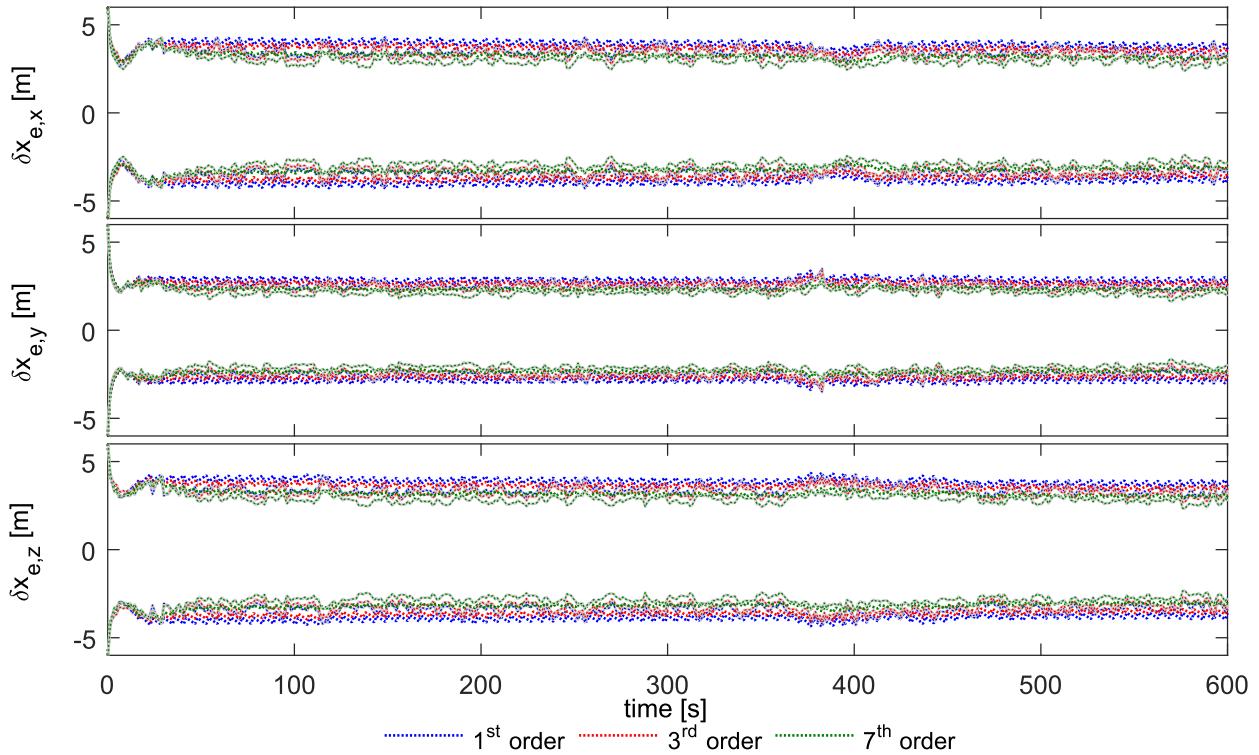


Figure 4-64: 3σ standard deviations of the position error with vibration models of different order, only covariance. The colored dotted lines represent the predicted standard deviations, the grey and colored dotted lines the actual standard deviations of the Monte Carlo simulation

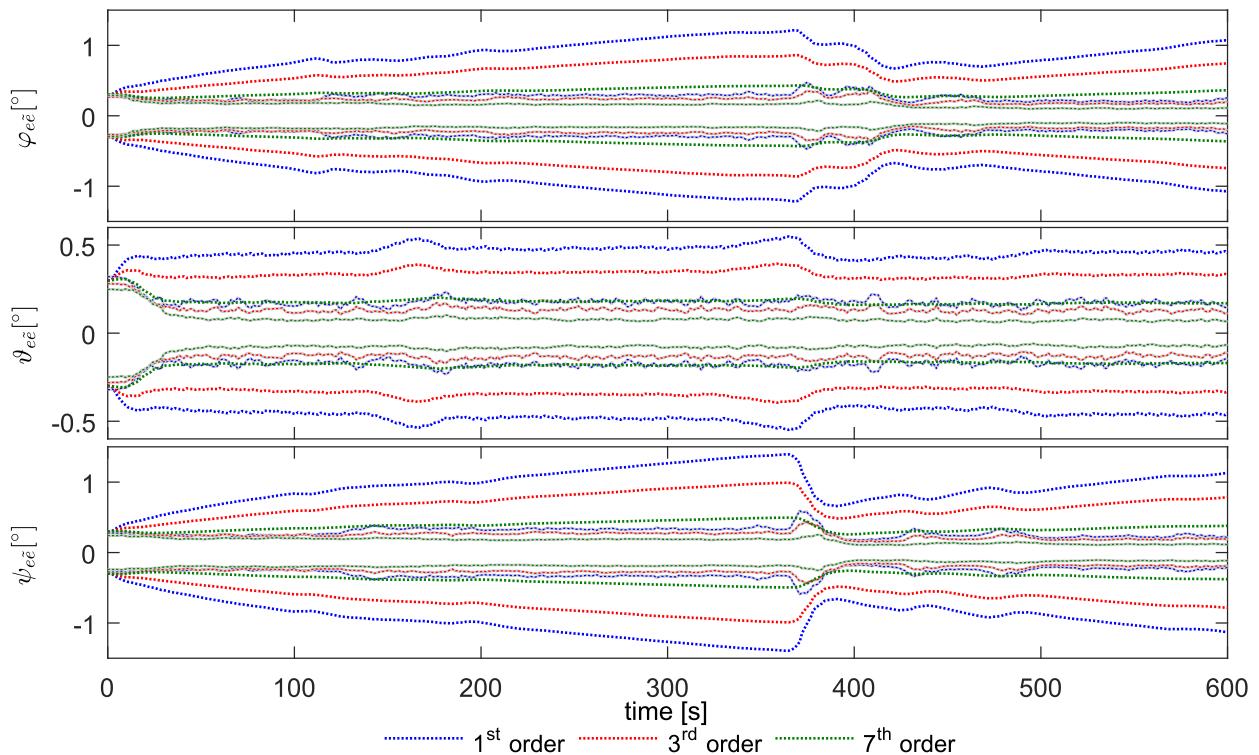


Figure 4-65: 3σ standard deviations of the orientation error with vibration models of different order, only covariance. The colored dotted lines represent the predicted standard deviations, the grey and colored dotted lines the actual standard deviations of the Monte Carlo simulation

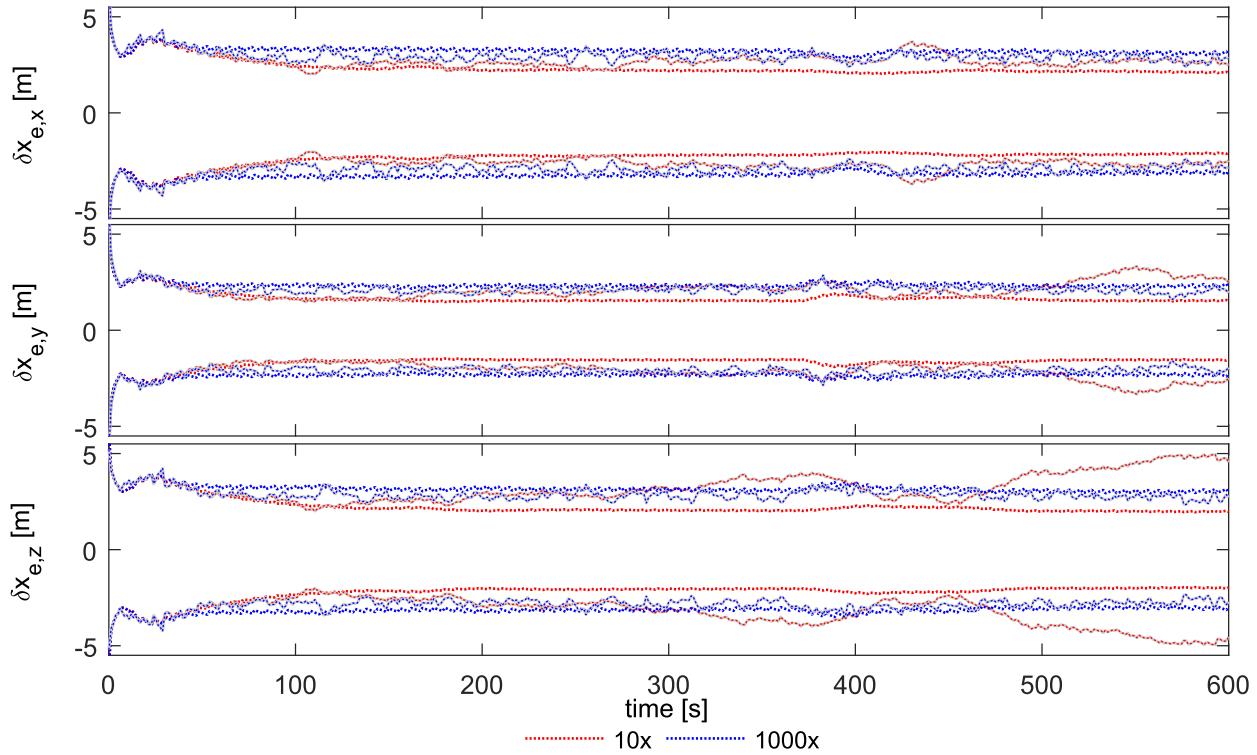


Figure 4-66: 3σ standard deviations of the position error without vibration model but over-bounded process noise. The colored dotted lines represent the predicted standard deviations, the grey and colored dotted lines the actual standard deviations of the Monte Carlo simulation

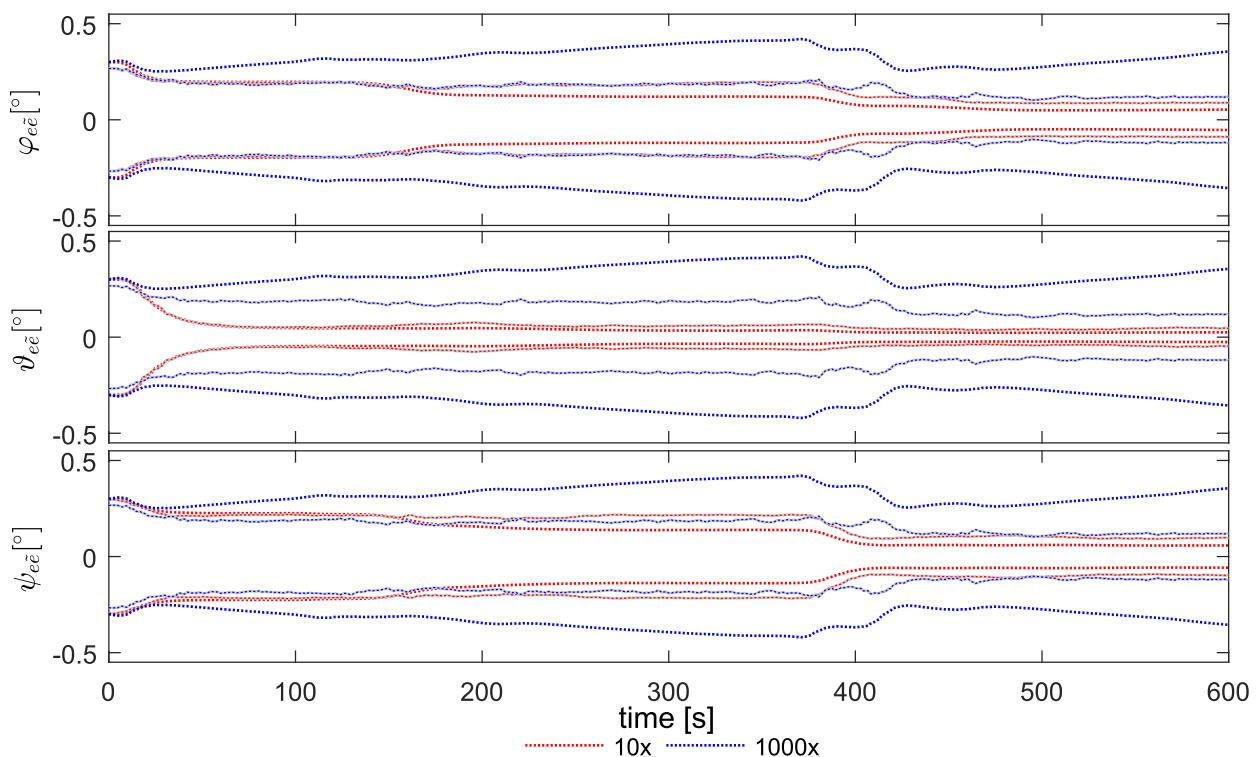


Figure 4-67: 3σ standard deviations of the orientation error without vibration model but over-bounded process noise. The colored dotted lines represent the predicted standard deviations, the grey and colored dotted lines the actual standard deviations of the Monte Carlo simulation

4.6.4 Satellite Clock Error

The dataset cutover model (3.111) and the noise model (3.114) are integrated into the navigation filter as considered states. The General Aviation Aircraft scenario (section E.1.2) is used as example to demonstrate the influence of the satellite clock error on the integrated navigation solution. IMU 3 (Table E-5) with 100 Hz sample rate is selected. The IMU measurements are solely affected by velocity/angular random walk noise N . All other IMU errors are set to zero. A generic GPS receiver provides 5 Hz pseudorange measurements. Start time is at 7pm, June 22, 2012 (section E.2.2). Besides the satellite clock error, the GPS raw measurements are only affected by white Gaussian noise with standard deviation $\sigma = 5 \text{ m}$ in order not to be distracted from other error influences.

At sensor data generation the true satellite clock error is computed from precise ephemerides. In the navigation filter the satellite clock error correction is computed from the correction coefficients of the broadcast navigation message. The satellite clock errors at June 22 are shown in Figure 4-68, Figure 4-69 and Figure 4-70 for the IIA, IIR-A/B and IIR-M satellites that are in view at 7pm. The large error growth of PRN 30 over the course of the day is noticeable. The dataset cutovers occur every two hours and are indicated by the grey dotted lines. The upload cutovers are marked by the colored dotted lines. The TOE of the last upload cutovers are also listed in Table 4-3.

Neglect of the residual satellite clock error

The navigation system uses the six satellites with the highest elevation angles. These are PRN 16, PRN 21, PRN 30, PRN 6, PRN 18 and PRN 29. PRN 29 is exchanged by PRN 3 after about 410 seconds. In the first simulation the residual satellite clock error is not considered. The corresponding position error is plotted in Figure 4-71. As can be easily seen, the predicted position error statistics and the true error do not match. An innovation based integrity monitor would have detected the mismatch and would probably have refused the pseudorange measurements.

Consideration of the residual satellite clock error by derived models

If, however, the satellite clock errors are considered in the navigation error filter, the predicted position error variances are increased and the actual errors lie within the expected 3σ bounds, as shown in Figure 4-72. The remarkable increase of the standard deviation at about 410 seconds after start is due to the exchange of a IIR-M satellite with lower residual satellite clock error by an older IIA satellite with larger residual clock error.

Consideration of the residual satellite clock error by derived models, neglecting time correlation

In the next simulation, the instantaneous variances of the residual satellite clock errors as predicted by the derived models are considered by the navigation filter but not the time correlation. This means that the cross-covariances between the navigation error states and the satellite clock error model states are not accounted for. The position error is plotted in Figure 4-73. Since the residual satellite clock error has predominantly bias-like character, the white noise assumption is not adequate. Therefore, the predicted standard deviations of the position error components are still too optimistic. Only the height error lies now within the predicted 3σ bound.

Consideration of the residual satellite clock error by over-bounded white noise

Alternatively, the residual satellite clock errors are not considered but encountered by over-bounded white noise on the pseudoranges. The resulting position error is shown in Figure 4-74 for an assumed standard deviation of 10 m. The actual position error lies now indeed within the predicted 3σ bounds, but performance is given away. This can be confirmed if the predicted standard deviations of the position error components at the end of the trajectory are compared. In the case when the errors are considered by models the position error standard deviations are 2.3 m in northern direction, 1.3 m in eastern direction and 0.65 m in height direction. In the case when the errors are over-bounded by white noise the corresponding position error standard deviations are 4 m, 3.6 m and 0.85 m. Moreover, the designer has to decide for a proper tuning. If models are used, the error is automatically considered without the need for tuning.

If measurements of more than six satellites are available, the effect of the satellite clock errors might be smaller because the clock errors of the single satellites are positive or negative and may hence compensate themselves. If only four satellites are used and these satellites are accidentally the older IIA satellites (because these satellites currently have, for example, the highest elevation), the effect of the satellite clock errors on the estimated position error is appreciably larger.

Table 4-3: TOE of last upload cutovers at 06/22/2012, 7pm, of satellites in view

PRN	GPS second	hh:mm:ss	PRN	GPS second	hh:mm:ss	PRN	GPS second	hh:mm:ss
3	467984	9:59:44	13	453584	5:59:44	5	n/a	n/a
6	431984	23:59:44	16	489584	15:59:44	7	496784	5:59:44
8	431984	23:59:44	18	439184	1:59:44	29	439184	1:59:44
26	496784	5:59:44	19	424784	21:59:44			
30	489584	15:59:44	21	431984	23:59:44			

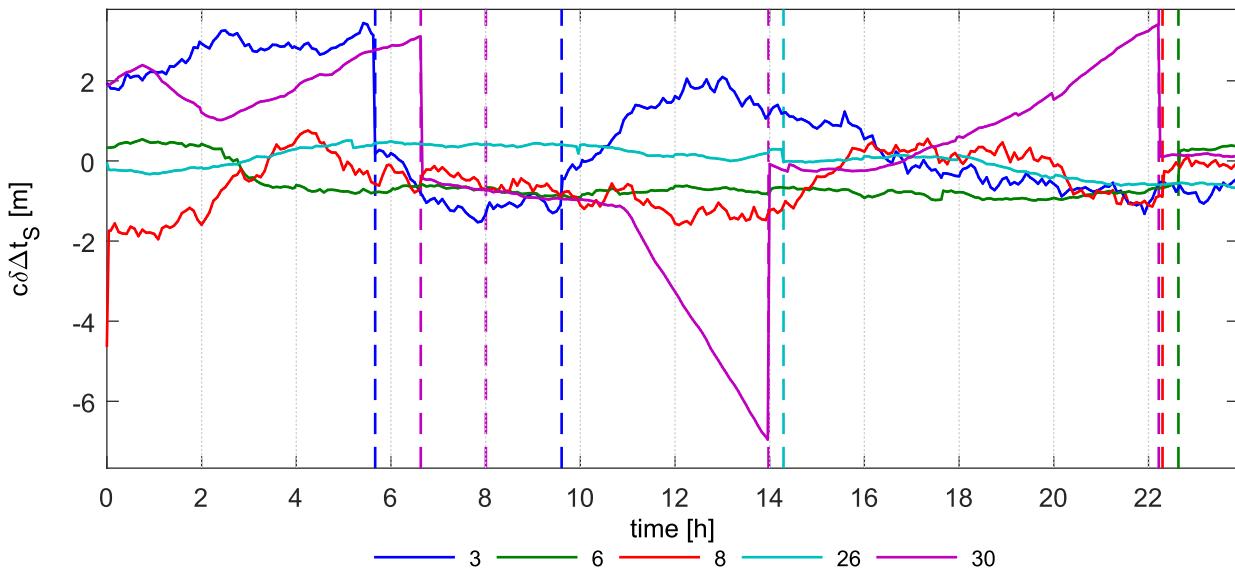


Figure 4-68: Satellite clock errors of IIA satellites at 06/22/2012. The dotted lines mark upload cutovers

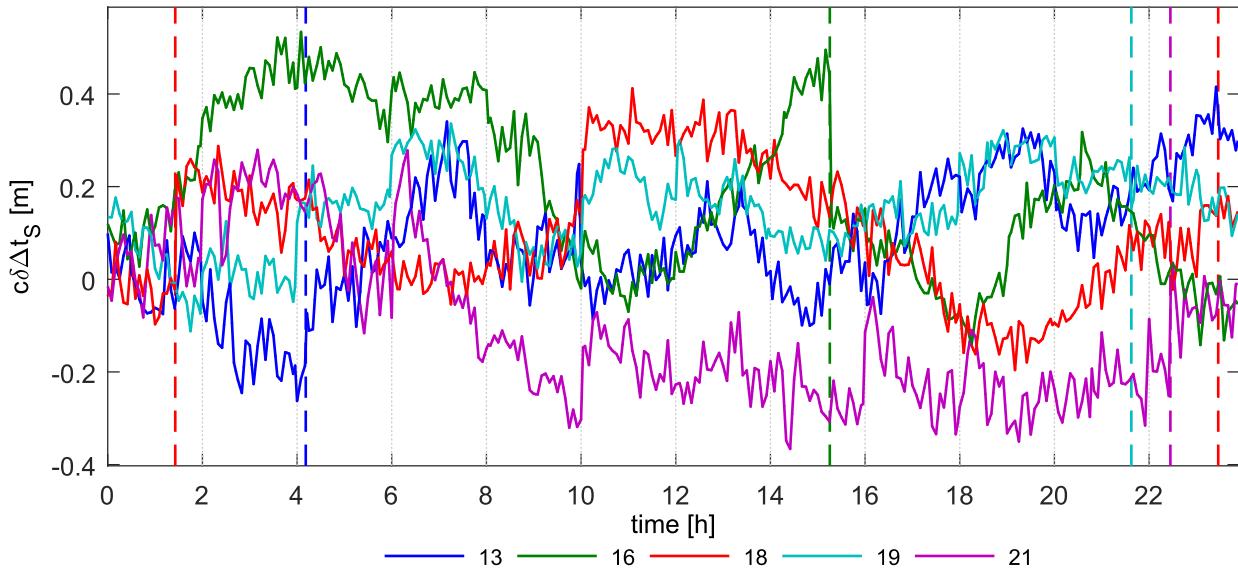


Figure 4-69: Satellite clock error of IIR-A and IIR-B satellites at 06/22/2012

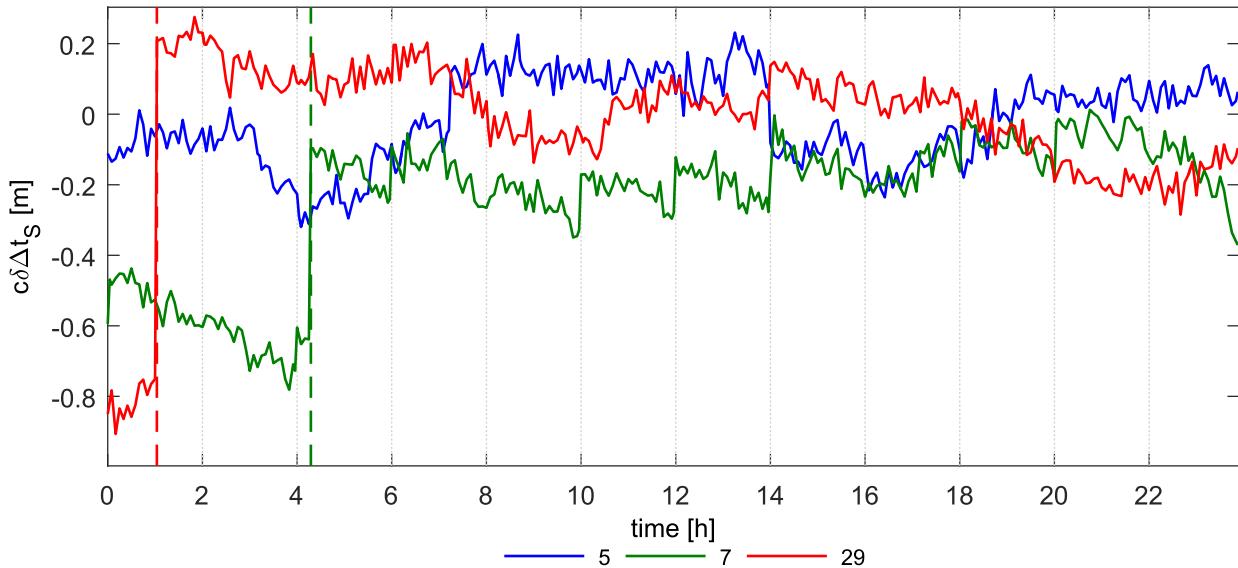
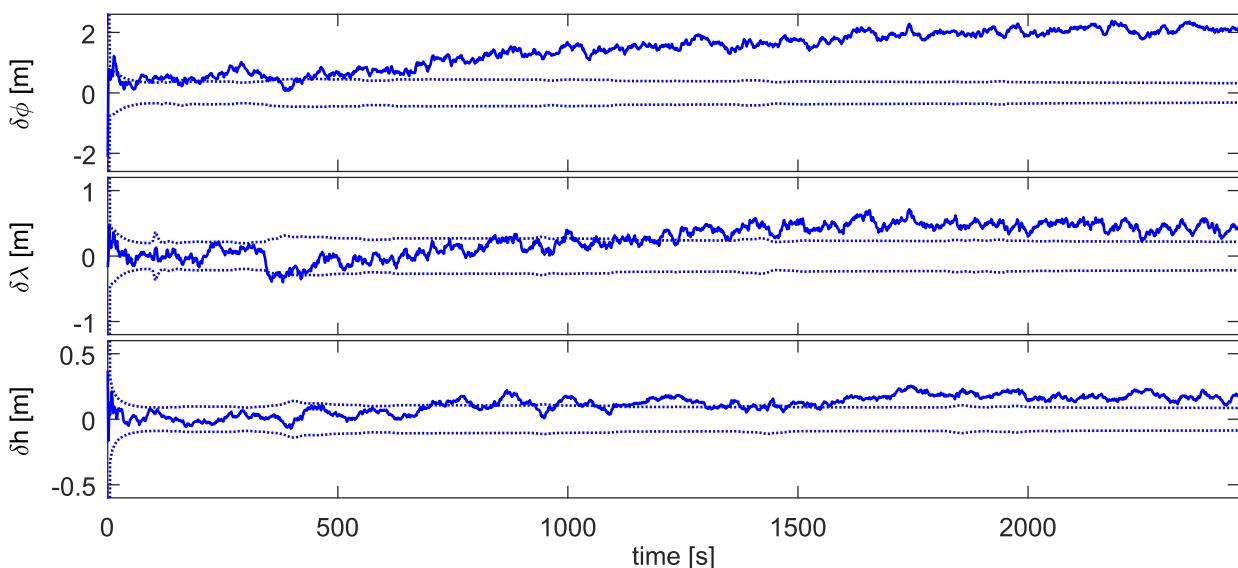


Figure 4-70: Satellite clock error of IIR-M satellites at 06/22/2012

Figure 4-71: Position error with predicted 3σ boundaries (dotted) if the clock error is not considered

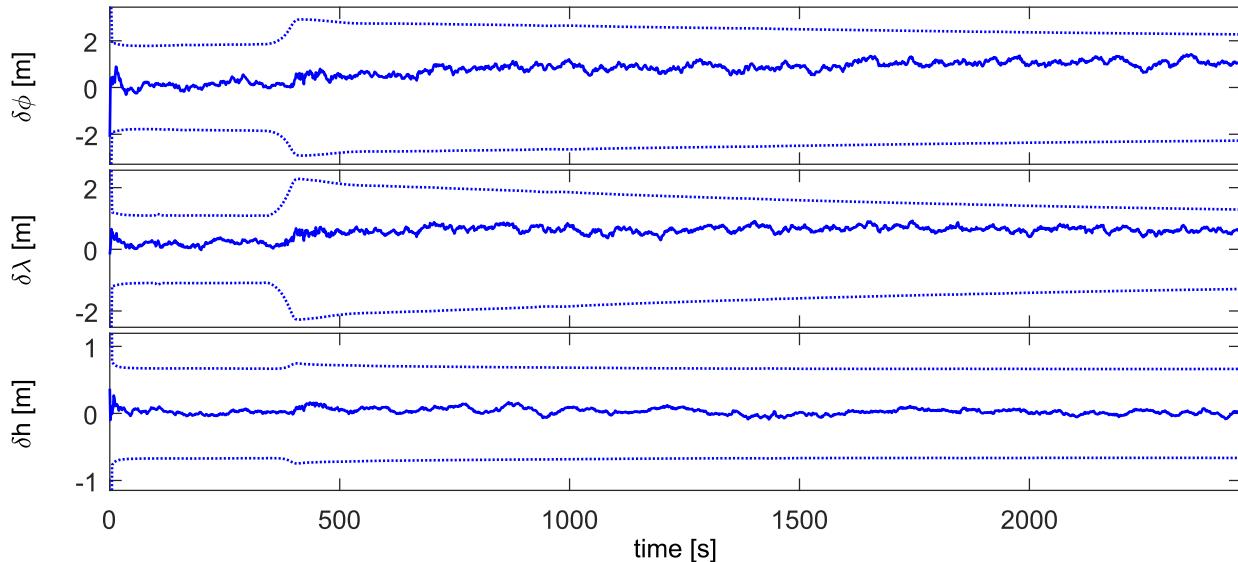


Figure 4-72: Position error with predicted 3σ boundaries (dotted) if the clock error is considered

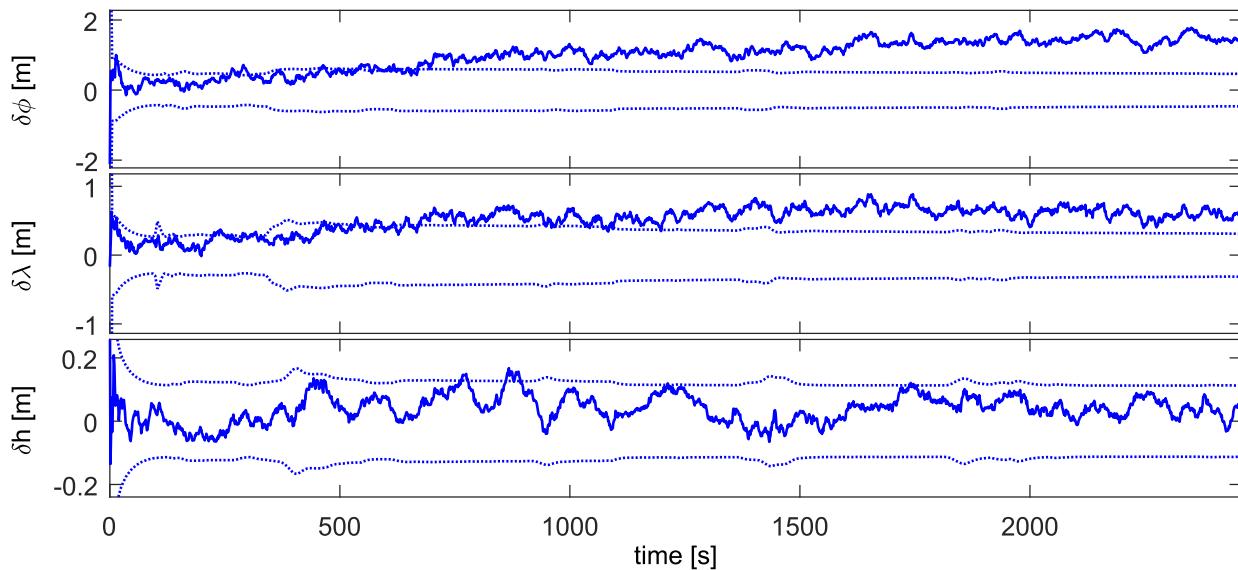


Figure 4-73: Position error with predicted 3σ boundaries (dotted) if only the clock error variance is considered, but the time correlation is neglected

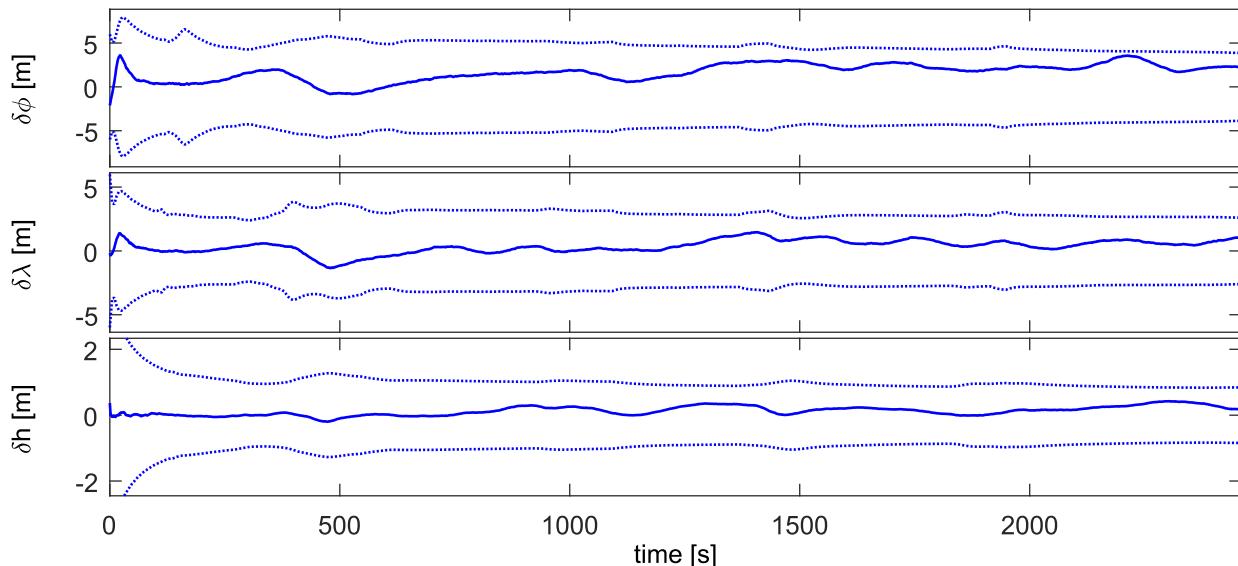


Figure 4-74: Position error with predicted 3σ boundaries (dotted) if noise variance is over-bounded

4.6.5 Satellite Position Error

The example of the previous section is picked up again to exemplify the effect of the satellite orbit errors on the integrated navigation solution. The same conditions as for the satellite clock error example hold. This time, the take-off is at 18:40 (GPS time). PRNs 3, 6, 18 and 21 are used for the solution. The satellite orbit errors in along-track, cross-track and radial directions at June 22 are shown in Figure 4-75 for the IIA and in Figure 4-76 for the IIR-A/B and IIR-M satellites that are in view at 7pm. The dataset cutovers occur every two hours and are indicated by the grey dotted lines. The upload cutovers are marked by the colored dotted lines. In Figure 4-77, the orbit error components and the predicted 3σ standard deviations of the four used satellites are plotted. Cutovers take place at second 1200. It can be seen that the variances of the radial error components are minimum at cutover. The modeled variances seem to fit quite well to the actually realized orbit errors.

Besides the satellite orbit errors, the GPS raw measurements are only affected by white noise in order not to be distracted from other error influences. The true satellite positions are computed from precise ephemerides at sensor data generation. In the navigation filter the satellite positions are computed from the broadcast ephemerides of the navigation message.

Neglect of the residual satellite position error

In Figure 4-78, the position error of the integrated navigation solution is plotted for the case when the satellite orbit errors are not considered by the navigation error filter. The predicted 3σ standard deviations are the red curves. It can be observed that the influence of the satellite orbit errors on the navigation state errors is smaller than of the satellite clock errors. In this example, mainly the lateral position errors are affected by the satellite orbit errors. Especially the north direction error violates the predicted statistics.

Consideration of the residual satellite position error by derived models

If the satellite orbit errors are, however, considered by the navigation error filter, as illustrated in Figure 4-78 by the green 3σ boundaries, the situation is improved since the predicted variances of the position error components are slightly increased.

Generally, it can be stated that the effect of the satellite position error model is comparatively small. Orbit errors most often compensate one another. The high modeling effort and additional computational load in contrast to the performance improvement is likely not justified in the most applications.

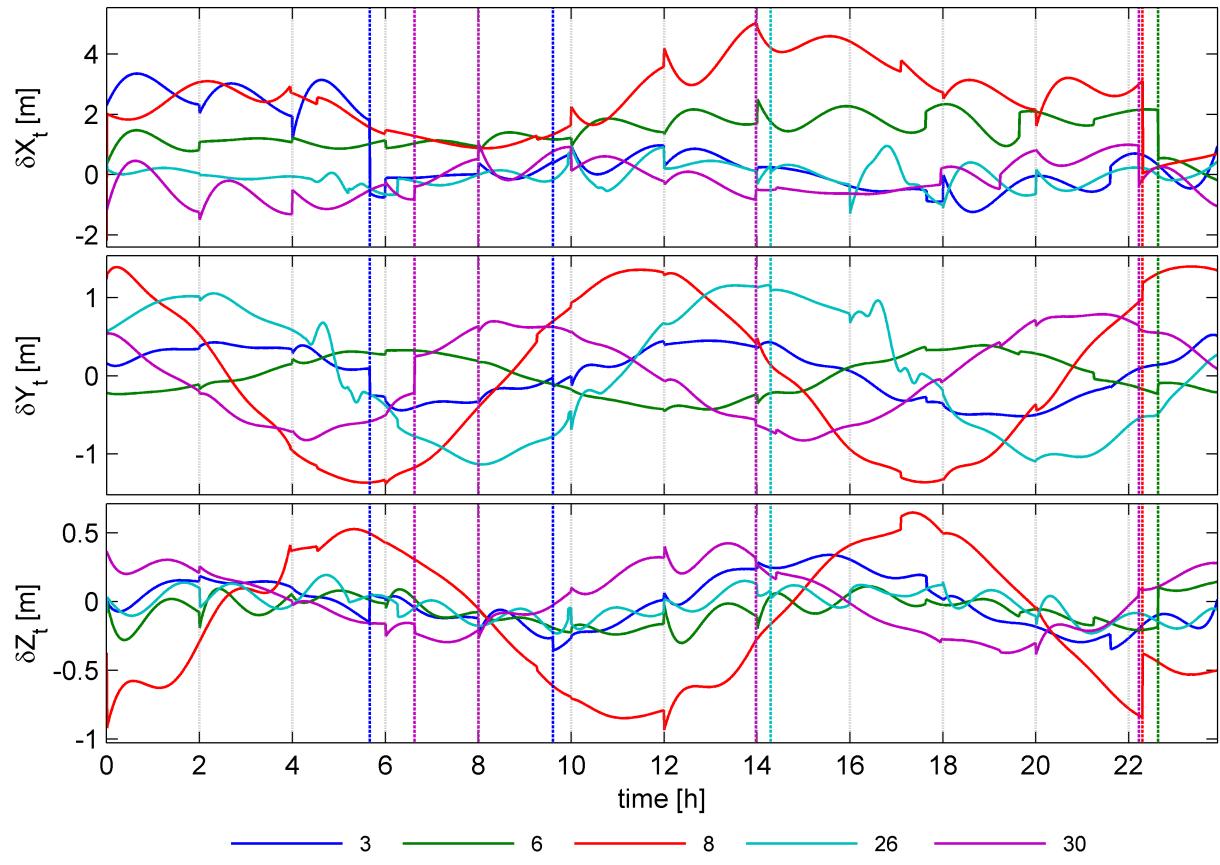


Figure 4-75: Along-track, cross-track and radial orbit errors of IIA satellites

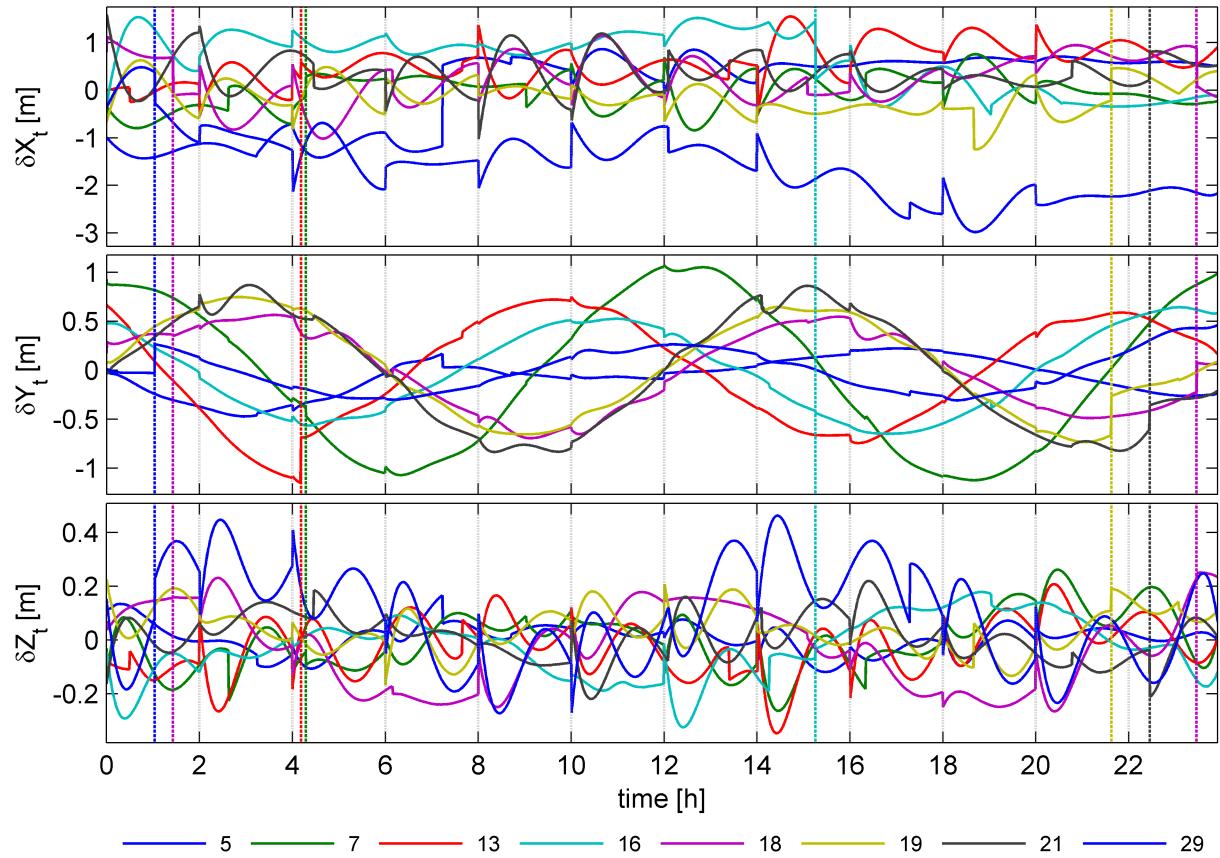


Figure 4-76: Along-track, cross-track and radial orbit errors of IIR-A, IIR-B and IIR-M satellites

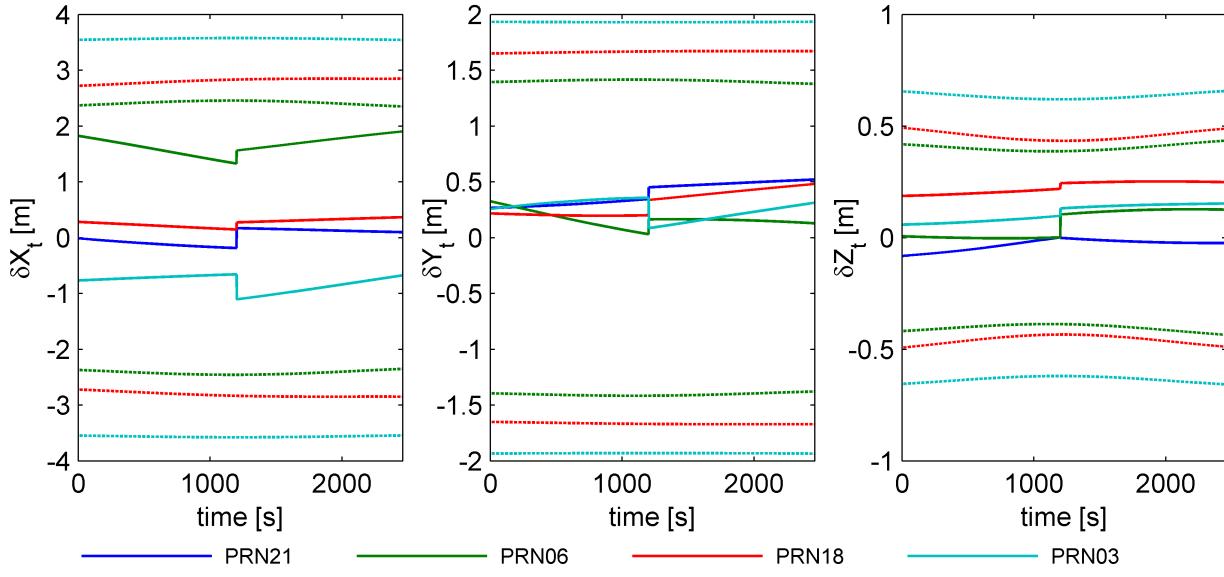


Figure 4-77: Along-track, cross-track and radial orbit errors of used satellites

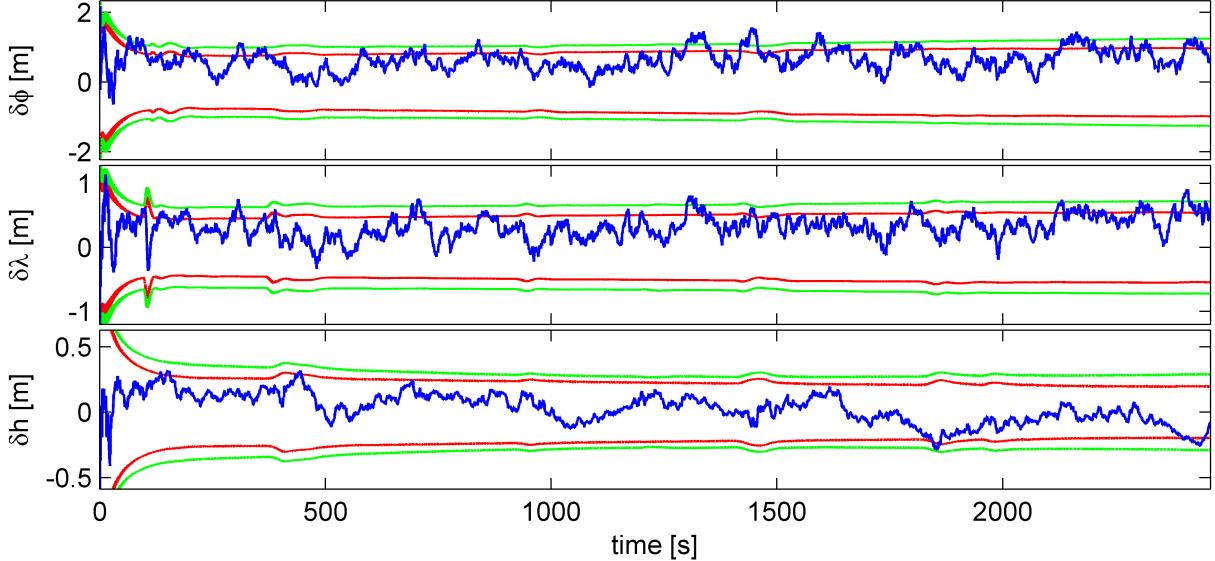


Figure 4-78: Position error, orbit error not considered (red), orbit error considered (green)

4.6.6 Ionosphere Error

The estimation of the sunspot number is demonstrated by means of a simulation example. For that, the solution of a stationary integrated navigation system at N47° E11° at 0 m height is analyzed. The system is initialized at June 22, 2012 at 7pm. IMU 3 (Table E-5) is chosen as inertial sensor. The IMU sample rate is 100 Hz and the GPS sample rate is 1 Hz. The IMU error model considers solely velocity/angular random walk noise and turn-on biases. The receiver clock model bases upon the specification of the Rakon IT225B oscillator in section 3.5.5. The eight satellites with the largest elevation angles are used in the navigation error filter (compare with the sky plot in Figure E-12). The initial state errors are randomly drawn according to the given initial covariance \mathbf{P}_0 .

The ionospheric delay $\Delta\rho_{iono}$ is the only error on the pseudorange measurements besides the receiver clock error $c\Delta t_R$ and additional white measurement noise with a standard deviation of 0.75 m. The

ionospheric delay of the precise TEC map serves as reference and is used in the measurement synthesis in the simulation. The realized clock error is shown in Figure 4-79.

First, the ionospheric delay is compensated with the Klobuchar model. The navigation state errors are shown in Figure 4-80 together with the predicted 3σ bounds. The estimated receiver clock bias error $c\delta\Delta t_R$ and drift error $c\delta\dot{\Delta t}_R$ are plotted in Figure 4-81. It can be observed that the realizations of the latitude error $\delta\phi$ and height error δh are apparently larger than the 3σ bounds, which indicates that the non-compensated ionospheric residual error $\delta\rho_{iono}$ mainly affects the position solution. The receiver clock bias is also biased just as the position error. This situation is dangerous because the integrity monitor might reject the GPS measurements due to the too optimistic covariance estimate. Second, the ionospheric delay is compensated with the NeQuick2 model. The sunspot number R_{12} is estimated by the navigation filter. The navigation state errors and the receiver clock errors are given in Figure 4-82 and Figure 4-83, respectively. The initial sunspot number value $R_{12}(t_0)$ has been set to 90 and its initial standard deviation has been chosen as 30. Figure 4-84 shows the error of the sunspot number δR_{12} with respect to the initially assumed number. It can be seen that the actual navigation state errors and the receiver clock error now fit much better to the predicted statistics. Furthermore, the actual position errors are smaller than in the previous case with Klobuchar model compensation. The sunspot number settles slowly with time. The correlation coefficients ρ between the height error δh , receiver clock bias error $c\delta\Delta t_R$ and sunspot number error δR_{12} are illustrated in Figure 4-85. As expected, the height error and the receiver clock bias error are positively correlated whereas the height error and the sunspot number error as well as the receiver clock bias error and the sunspot number error are negatively correlated. The filter can slowly resolve the correlation between the three states as the satellite constellation and the ionosphere change steadily with time. The filter cannot distinct between the common component of the initial height error projected onto the line-of-sight directions and the initial receiver clock bias. This effect is especially dominant if the elevation angles of the currently used satellite constellation do not differ much. The same is valid for the initial height error and sunspot number error and the receiver clock error and the sunspot number error since the ionospheric errors on the single line-of-sight ranges are strongly correlated with each other. Corresponding to the decrease of the correlation, the height error and the receiver clock bias error settle more slowly as in the previous case due to the additional correlated state. In fact, the smallest of the initial variances of the three errors sets the level to which the other two, more uncertain errors will settle in the initial period. The initial variances of all three errors have been chosen well-balanced in this example.

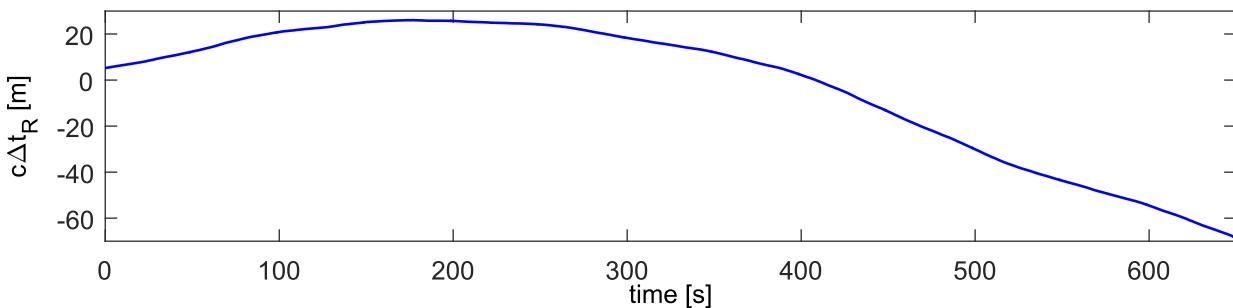


Figure 4-79: Realization of the receiver clock error

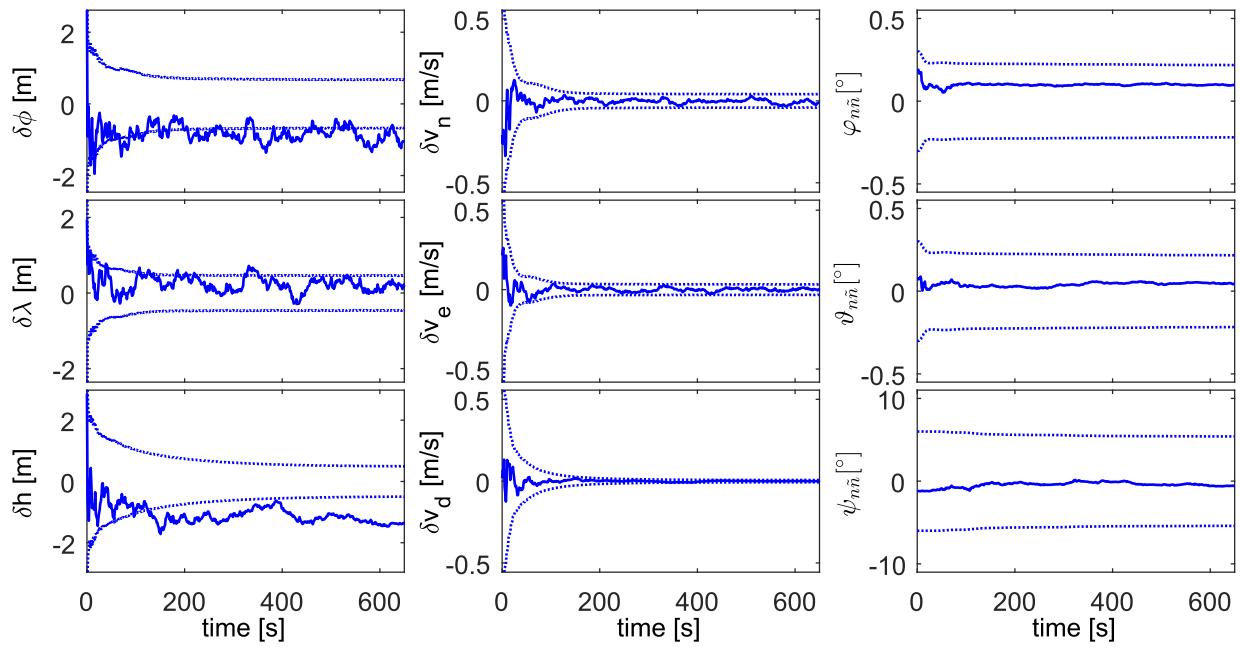


Figure 4-80: Navigation state estimation error with Klobuchar model ionospheric delay compensation

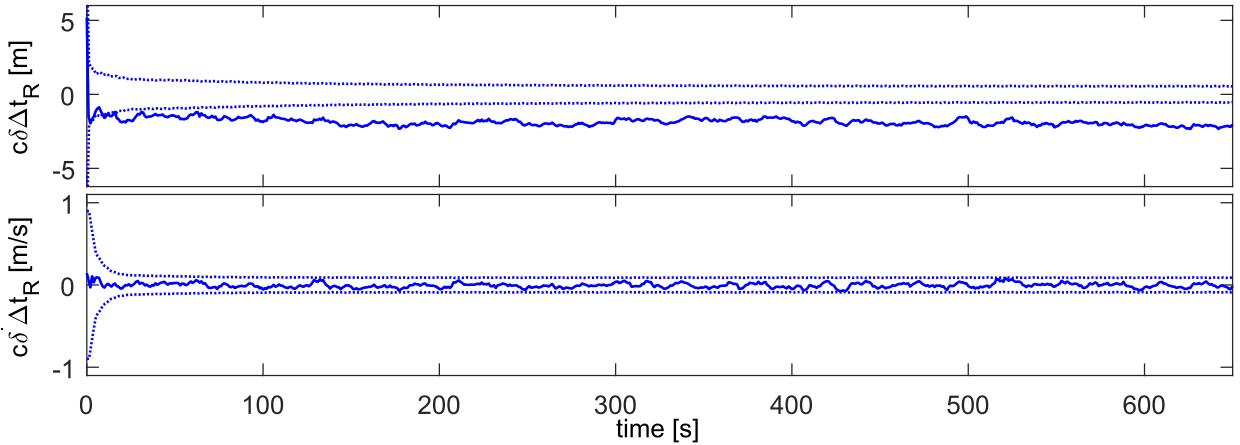


Figure 4-81: Receiver clock estimation error with Klobuchar model ionospheric delay compensation

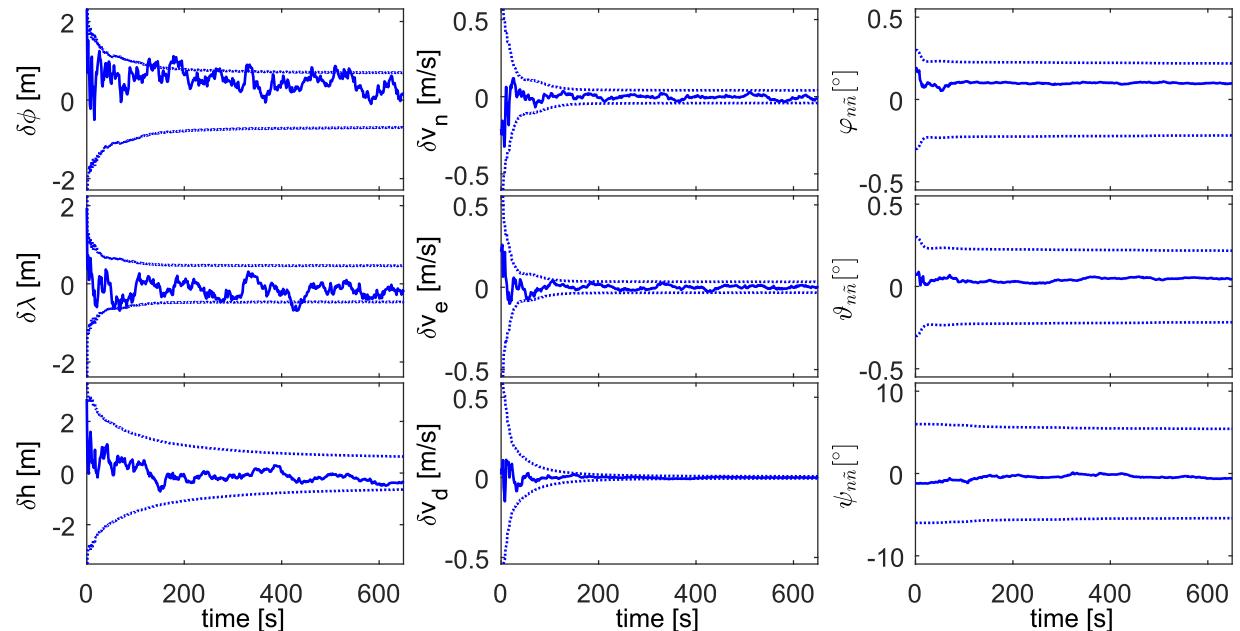


Figure 4-82: Navigation state error with NeQuick2 model and estimation of the sunspot number

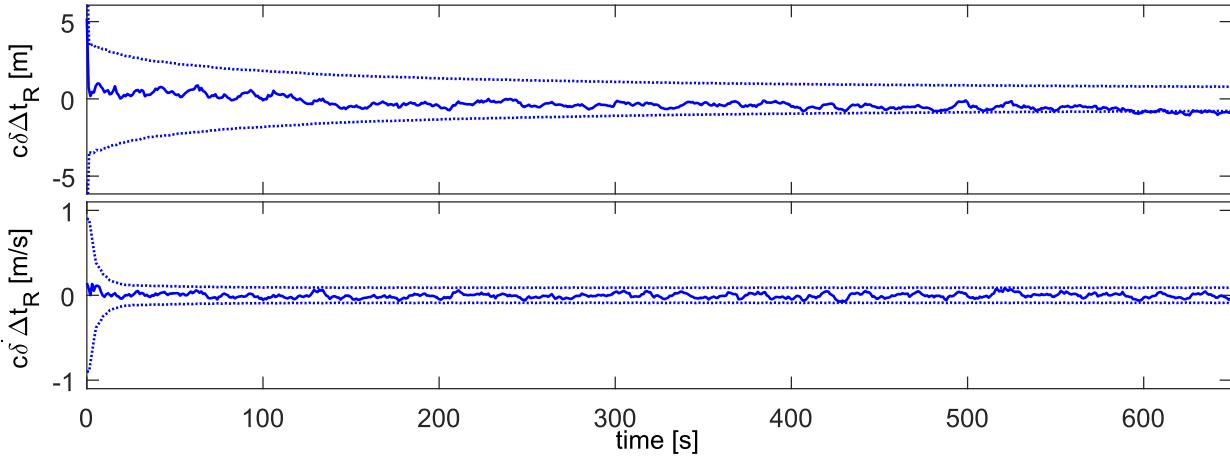


Figure 4-83: Receiver clock error with NeQuick2 model and estimation of the sunspot number

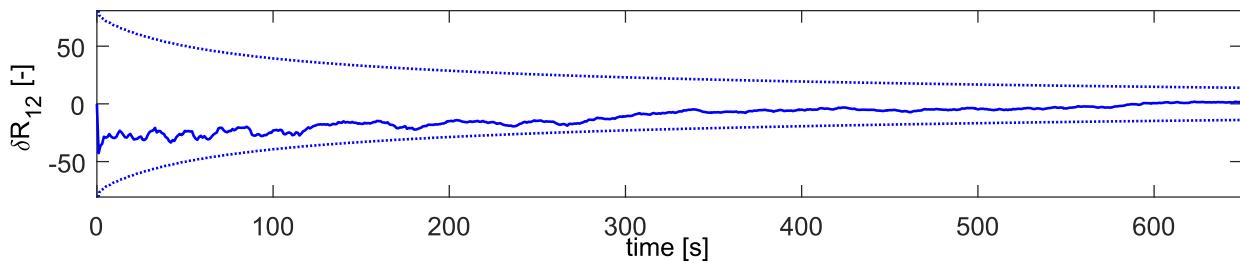


Figure 4-84: Sunspot number

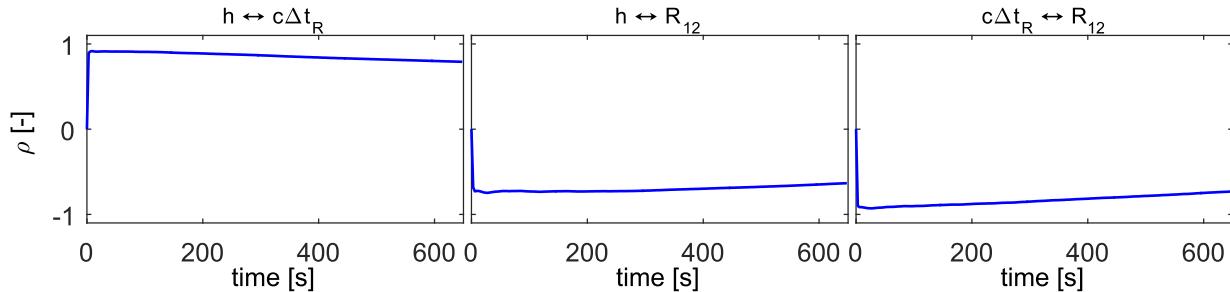


Figure 4-85: Correlation between height, receiver clock error and sunspot number

4.6.7 Receiver Clock Error

The navigation state error model is augmented by the two-state model (3.181) for the receiver clock bias and drift. The random walk frequency, flicker frequency and white frequency noise with bias-like effect on the receiver clock bias are considered as colored process noises using (3.28). The noise-like influences of the flicker and white phase noise are handled as receiver noise, which will be discussed in the next section. Furthermore, it is analyzed if it is sufficient to simply add adequately tuned white process noise to the two receiver clock error states instead of noise models. The analysis is done by means of the first 180 seconds of the General Aviation Aircraft scenario of section E.1.2, using IMU 3 (Table E-5) with 100 Hz sample rate. Take-off date is June 22, 2012 at 7pm. The navigation solution is aided by 1 Hz pseudorange measurements, which are exclusively affected by white Gaussian noise with 2 m standard deviation. It is assumed that the receiver works with an ON-169A oscillator with the model parameters given in Table 3-35. Besides the random walk frequency, white frequency and flicker frequency noise, a constant receiver clock drift with 5 m/s standard deviation is added to the receiver clock error. The flight is segmented into three subsequent periods. In the first period between second 0 and second 60 six satellites are used. In the second period between second 60 and second

120 only three satellites are tracked and in the third period between second 120 and second 180, again five satellites are available to aid the integrated navigation solution. The minimum elevation mask is set to five degree. Satellites with lowest elevation angles are selected in this exemplary simulation. The corresponding sky plot can be found in Figure E-12. The flicker frequency noise is synthesized with a fourth order pink noise model. In the filter itself, only a first order model is applied. The predicted standard deviations are compared with the true error statistics from 200 Monte Carlo runs.

The 3σ position error boundaries are shown in Figure 4-28. The receiver clock bias and drift errors are illustrated in Figure 4-87. The predicted standard deviations are represented by the black dotted lines and the actual standard deviations from the Monte Carlo runs by the grey dotted lines. There is a good coincidence between predicted and true error statistics. In the periods with six and five satellites the position error growth is limited whereas it is unbounded in the interim period with only three satellites. That is as expected since at least four satellites are required to compute a unique PVT solution. In Figure 4-87, the height error of Figure 4-28 is additionally plotted in red. It can be clearly seen that both errors are negatively correlated, which means that the filter cannot uniquely distinguish between height error and receiver clock error. Since the height is an important and safety critical input for flight guidance, the height error should be as small as possible. Therefore, it is important to accurately model the receiver clock error in order to support the filter in distinguishing between both errors. As will be seen, the minimum (and statistically correct) height error is only possible if the receiver clock error is correctly considered by the filter.

Next, the random walk frequency, flicker frequency and white frequency noise are not depicted by the models but replaced by simple substitute white noise on both receiver error states, represented by a diagonal process noise covariance matrix, as is often encountered in integrated navigation filter designs. Here, three different process noise covariances

$$\mathbf{Q}_1 = \text{diag}\left(1^2 \frac{\text{m}^2}{\text{s}^2}, 1^2 \frac{\text{m}^2}{\text{s}^4}\right), \mathbf{Q}_2 = \text{diag}\left(0.1^2 \frac{\text{m}^2}{\text{s}^2}, 0.1^2 \frac{\text{m}^2}{\text{s}^4}\right), \mathbf{Q}_3 = \text{diag}\left(0.01^2 \frac{\text{m}^2}{\text{s}^2}, 0.01^2 \frac{\text{m}^2}{\text{s}^4}\right) \quad (4.172)$$

are analyzed. The position error standard deviations are illustrated in Figure 4-88 and the corresponding receiver clock error standard deviations in Figure 4-89. The solid lines are the predicted and the dotted lines the actual 3σ standard deviations of the Monte Carlo simulation. The black dotted lines represent the reference standard deviations of the first case with noise models. It is observable that mainly the height error and less the translational position errors are affected. With the process noise covariances \mathbf{Q}_1 and \mathbf{Q}_2 the height error is larger than in the first case. With the covariance \mathbf{Q}_3 the predicted height error is smaller. There is a good coincidence between predicted and true standard deviations in the case of the large clock error process noise covariance. However, the uncertainty of the receiver clock bias and thus of the height error is much larger than with the correct receiver clock error model. In the second case, the predicted and true standard deviations match less, but at least the predicted standard deviation is larger than the true one, which is conservative and less problematic. In the case of the smallest clock error process noise covariance, the predicted standard deviation is smaller than with correct clock error model and moreover the predicted standard deviation is smaller than the true one, which is dangerous. In summary, the larger the noise, the better is the coincidence and the true noise standard deviation is smaller or equal than the predicted. In contrast, if the noise is chosen too small, the predicted error is smaller than the actual. This means that a trade-off between robustness on the one hand and accuracy primarily of the height error on the other hand has to be done. It can be stated that it is not easy to correctly tune the process noise covariance and it is therefore recommended to model the receiver clock errors.

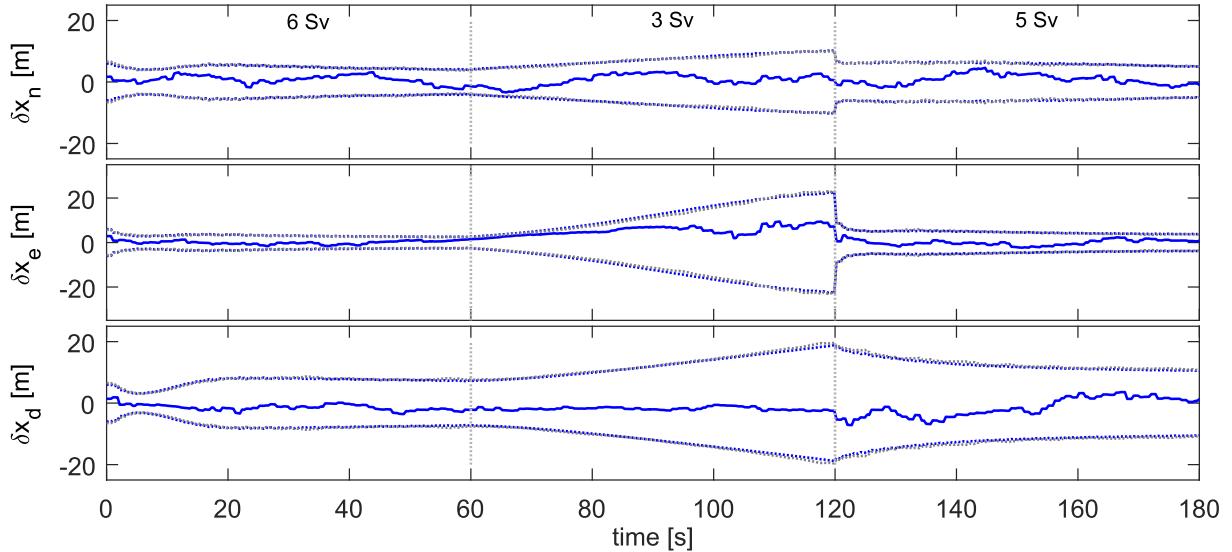


Figure 4-86: Position error and 3σ boundaries as predicted by filter (blue dotted) and from Monte Carlo simulation (grey dotted) if the receiver clock noise is considered

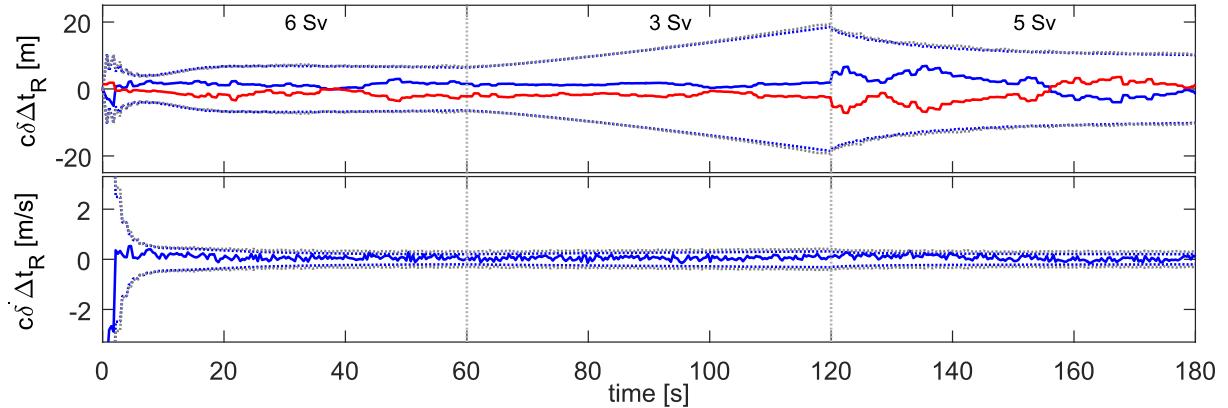


Figure 4-87: Receiver clock error and 3σ boundaries as predicted by filter (blue dotted) and from Monte Carlo simulation (grey dotted) if the receiver clock noise is considered. The red line is the height error

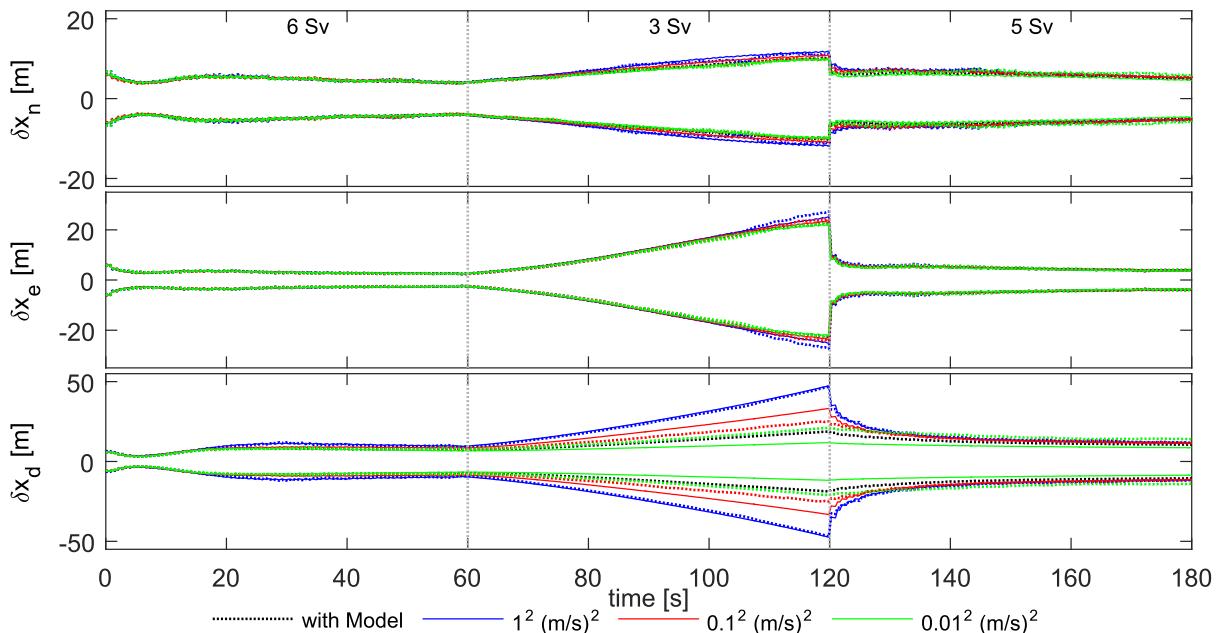


Figure 4-88: Position error and 3σ boundaries as predicted by filter (solid) and from Monte Carlo simulation (dotted) if substitute white receiver clock error process noise is used

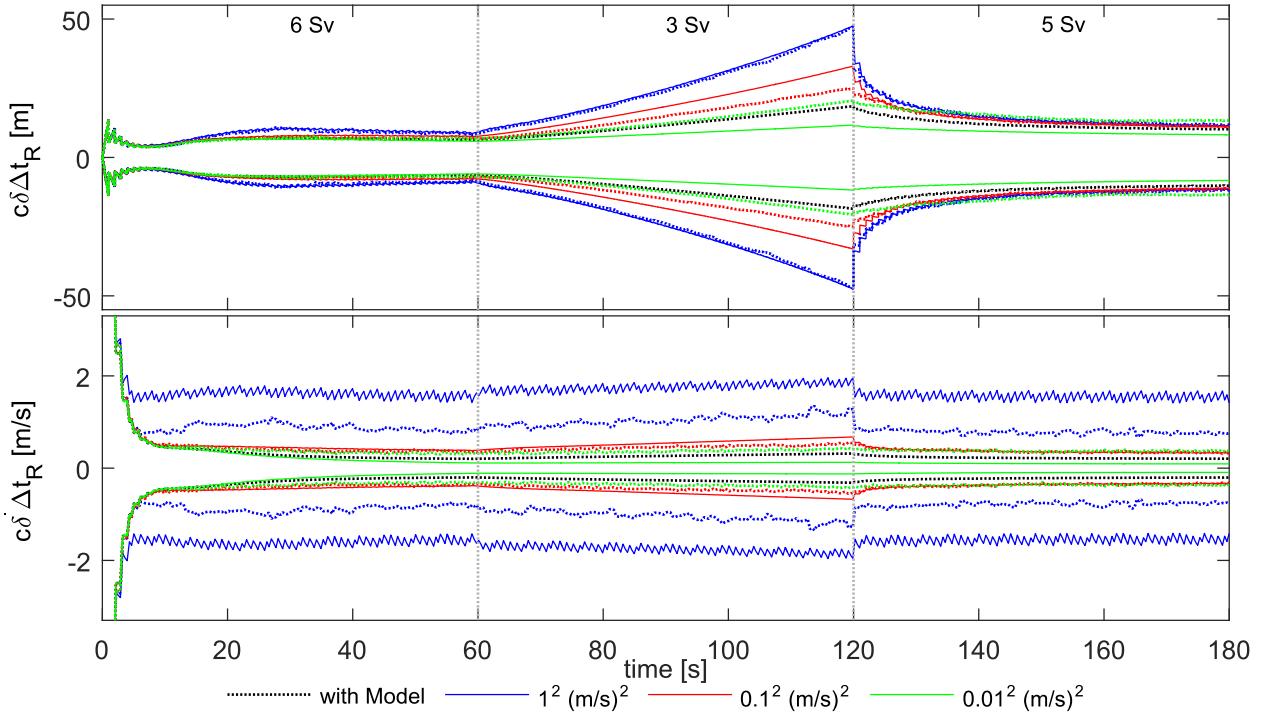


Figure 4-89: Receiver clock error and 3σ boundaries as predicted by filter (solid) and from Monte Carlo simulation (dotted) if substitute white receiver clock error process noise is used

4.6.8 Receiver Noise

The first 600 seconds of the Generic Aviation Aircraft scenario of the previous section are used to demonstrate how the receiver noise is considered in the integrated navigation filter. The sample time of the tracking loop simulation is $T = 2 \text{ ms}$, which is just the integration and dumping timespan of the correlators. The bandwidth of the PLL is initially set to $B_P = 8 \text{ Hz}$, the bandwidth of the frequency assistance to $B_F = 4 \text{ Hz}$ and the bandwidth of the DLL to $B_C = 1 \text{ Hz}$.

Figure 4-90 exemplarily shows the carrier-to-noise ratio of satellite PRN 16 that is assumed for the simulation of the satellite signal. The simulated prompt in-phase and quadrature samples \tilde{I}_P and \tilde{Q}_P are shown in a plane in Figure 4-91 and dependent on the time together with the early and late in-phase and quadrature samples \tilde{I}_E , \tilde{I}_L , \tilde{Q}_E and \tilde{Q}_L in Figure 4-92. It can be seen that the mean values of the quadrature samples are zero whereas those of the in-phase samples are unequal zero. The prompt in-phase correlator output values \tilde{I}_P are larger than the early and late in-phase correlator output values \tilde{I}_E and \tilde{I}_L , which are equal in the mean. Figure 4-93 shows the outputs of the phase, frequency and code discriminators \tilde{d}_ϕ , \tilde{d}_f and \tilde{d}_c and the corresponding 3σ standard deviations. The code discriminator \tilde{d}_c is scaled with the actual low-pass filtered signal power $N = \tilde{I}_P^2 + \tilde{Q}_P^2$. The cut-off time constant of the low-pass filter is chosen as $T_c = 0.25 \text{ s}$. In Figure 4-94 the noise of the carrier, range rate and pseudorange measurements and the corresponding 3σ standard deviations are illustrated. It can be observed that the noise standard deviations clearly depend on the current carrier-to-noise ratio. Figure 4-95 shows the autocorrelation of the simulated carrier phase, range rate and pseudorange measurement noises. The correlation length of the carrier phase is less than 0.1 s. The autocorrelation of the range rate noise is virtually zero. The pseudorange measurement noise, however, features noticeable correlation with time with the chosen filter parameters. If the output sample rate of the GPS receiver is chosen as 5 Hz, the pseudorange measurements are correlated about 40%. If the GPS receiver rate is 1 Hz, the measurements are not correlated. In this case, the

correlation of the receiver noise does not have to be considered in the integrated navigation filter, which is favorable due to the lower implementation effort and computational load. The resulting position error, velocity error and orientation error, respectively, of the integrated navigation solution are illustrated in Figure 4-96, Figure 4-97 and Figure 4-98.

In a second simulation run, a constant power scaling $N = 2^{22}$ is used in the code discriminator d_r . The code discriminator output, range measurement noise and the corresponding autocorrelation are plotted in Figure 4-99. It is interesting that the effect of the carrier-to-noise ratio on the code discriminator is now contrary to the previous case. This is because the actual signal power at low carrier-to-noise ratios is lower than the constant reference power that is used for the scaling. Thus, the code discriminator values become larger with decreasing carrier-to-noise ratios compared with the variable signal power scaling. The standard deviation of the pseudorange measurement noise now seems to be independent of the carrier-to-noise ratio. Moreover, the 3σ bounds are lower than in the previous case with variable signal power scaling. The autocorrelation of the range measurement noise, however, is increased. When the carrier-to-noise ratio decreases, the correlation increases.

In a third run, the bandwidths of the tracking loops are increased. The bandwidth of the PLL part is set to $B_P = 32 \text{ Hz}$, the bandwidth of the FLL part to $B_F = 16 \text{ Hz}$ and the bandwidth of the DLL to $B_C = 2 \text{ Hz}$. The measurement noises are shown in Figure 4-100. Compared to Figure 4-94, the noise level is increased now. The correlation of the pseudorange noise is slightly reduced.

A final remark: as already stated, it cannot be separated between the thermal noise on the signal and the oscillator noise. The influence of the oscillator noise on the tracking loops, however, is much smaller than that of the thermal noise and does not have to be considered in the tracking loops. The estimated C/N₀ ratio actually contains the oscillator noise.

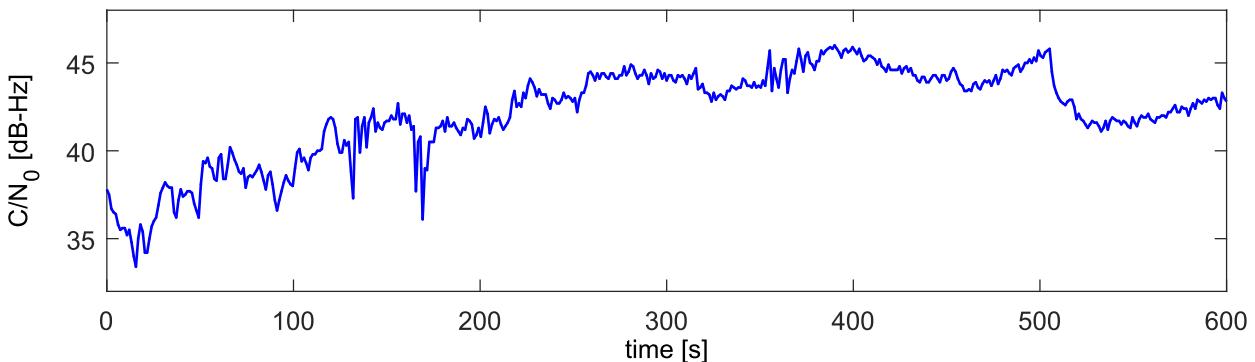


Figure 4-90: Assumed carrier-to-noise ratio of PRN 16

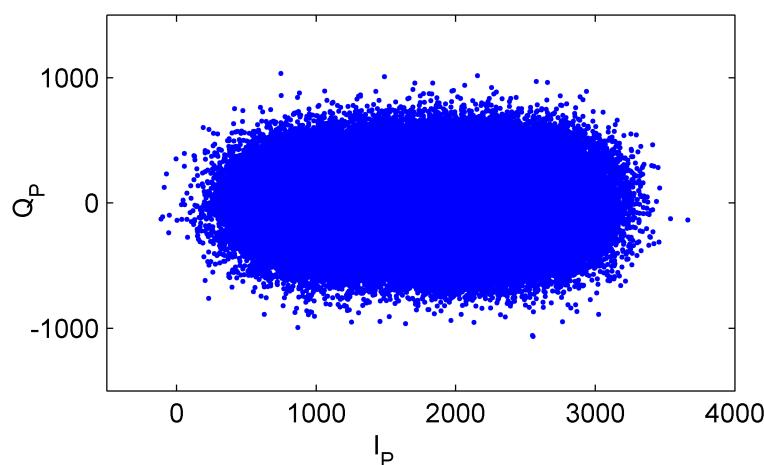


Figure 4-91: Prompt in-phase and quadrature phase samples of PRN 16

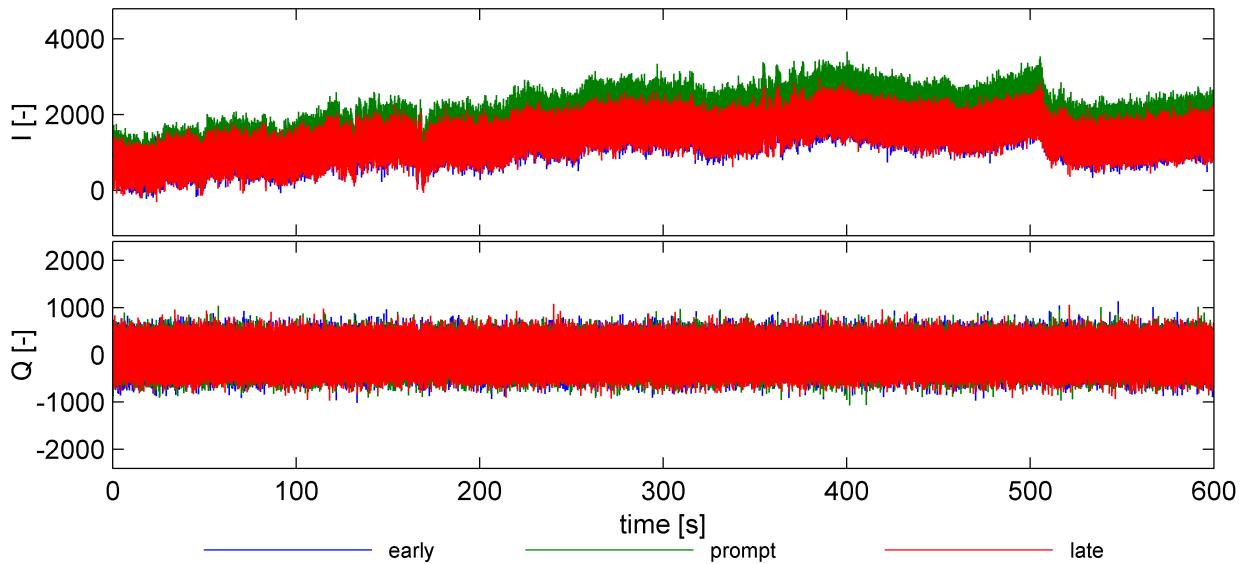


Figure 4-92: In-phase and quadrature phase signals of PRN 16

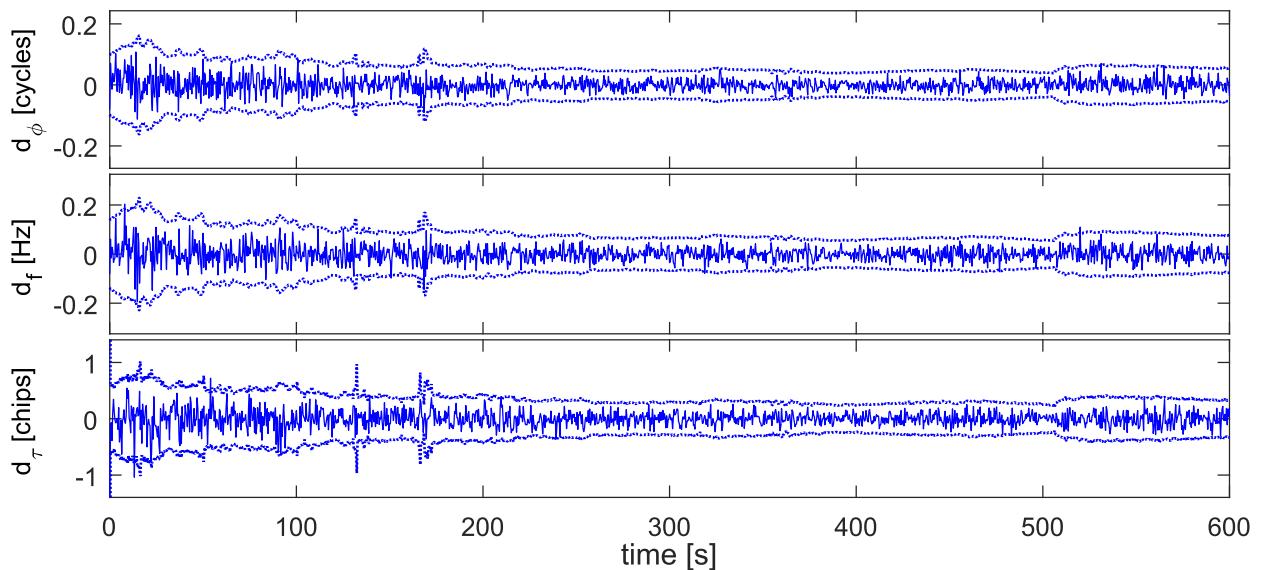


Figure 4-93: Phase, frequency and code discriminator outputs and 3σ boundaries (dotted) of PRN 16

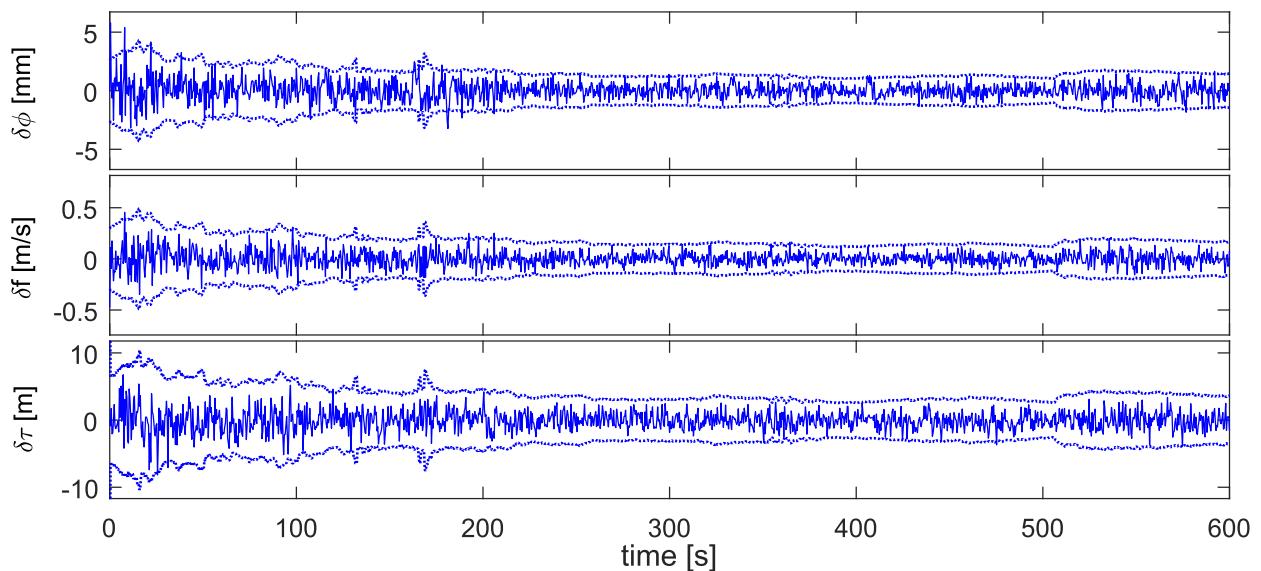
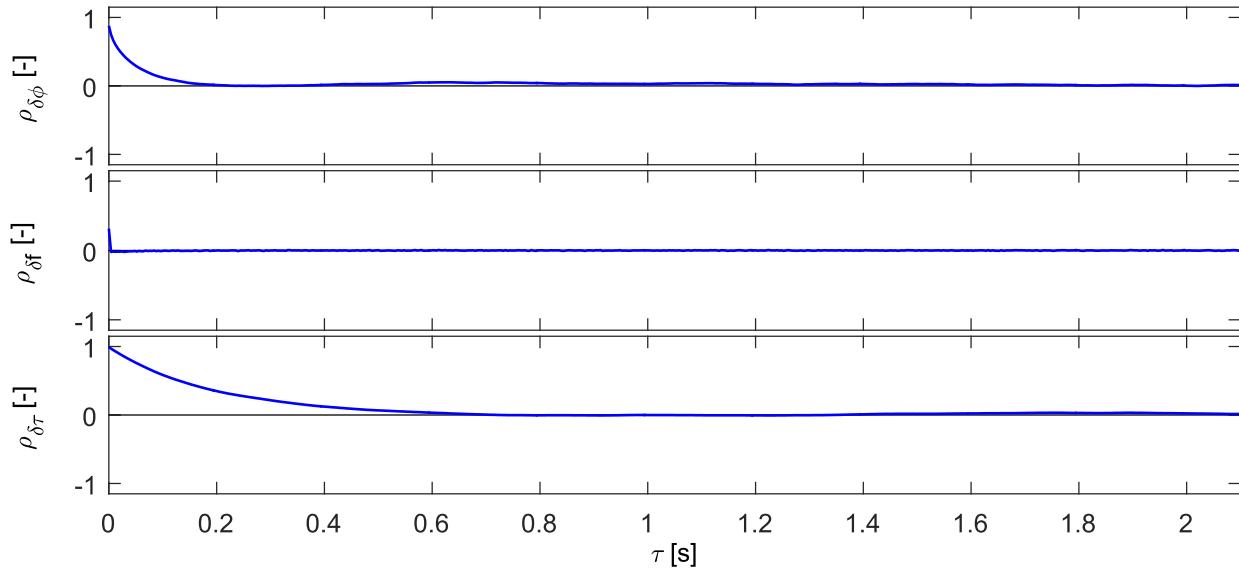
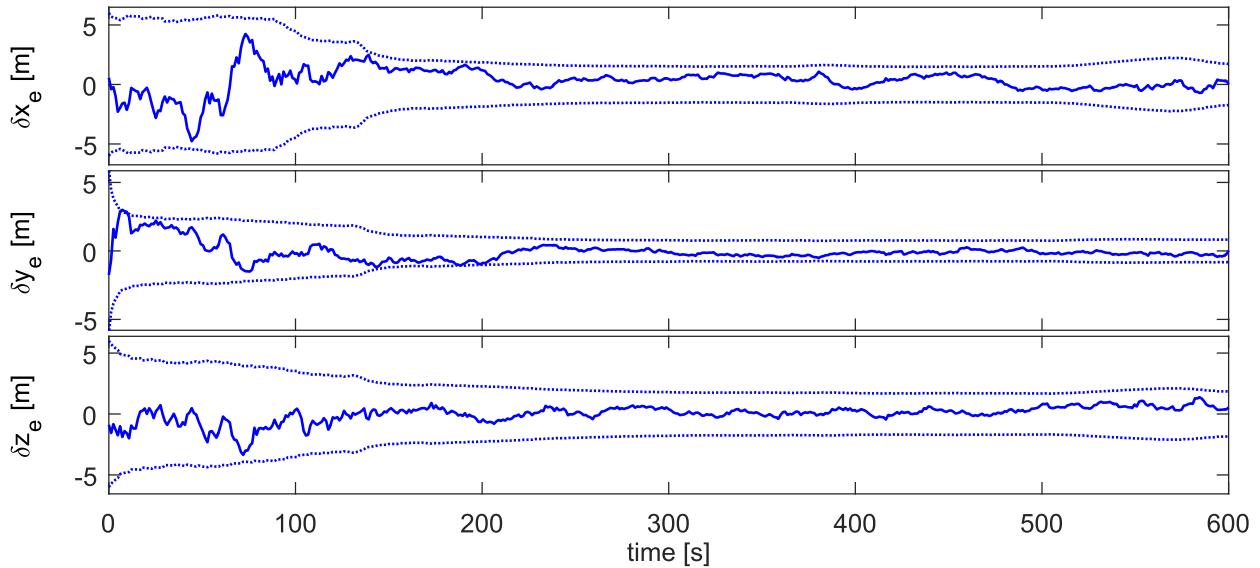
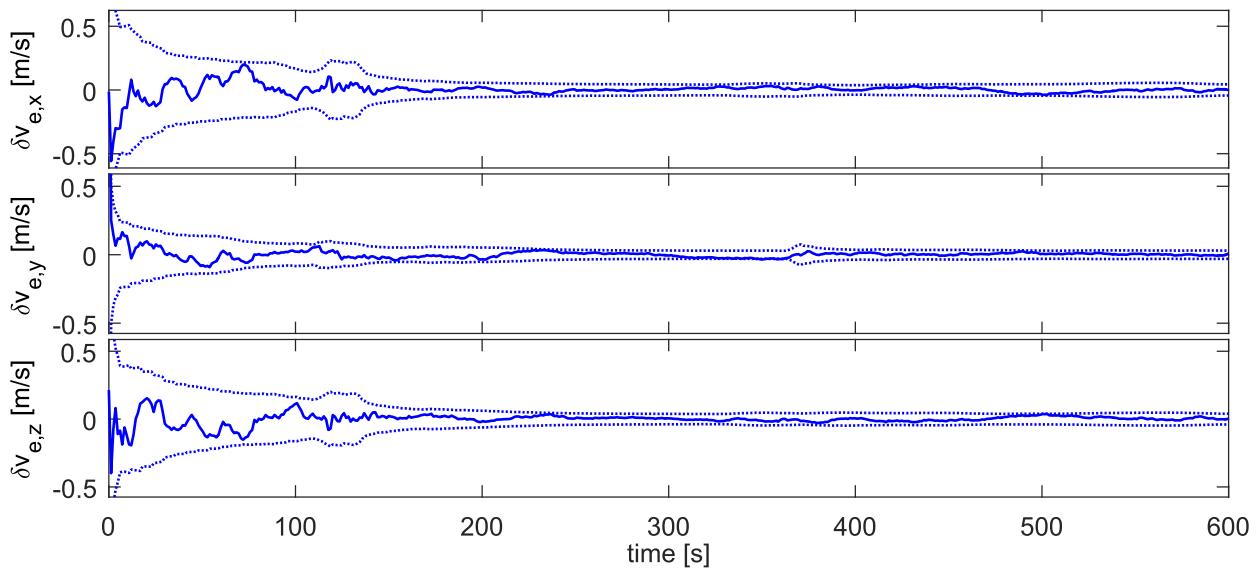


Figure 4-94: Carrier phase, rate and range measurement noise and 3σ boundaries (dotted) of PRN 16

**Figure 4-95: Autocorrelation of carrier phase, rate and range measurement noise of PRN 16****Figure 4-96: Position error and 3σ boundaries as predicted by filter (dotted lines)****Figure 4-97: Velocity error and 3σ boundaries as predicted by filter (dotted lines)**

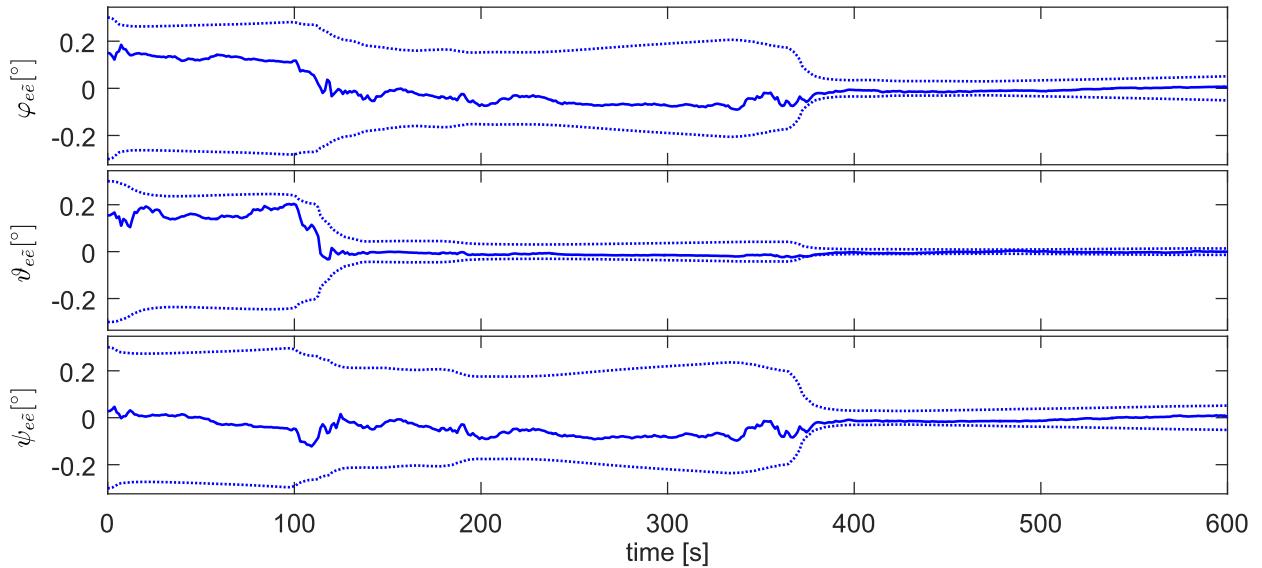


Figure 4-98: Orientation error and 3σ boundaries as predicted by filter (dotted lines)

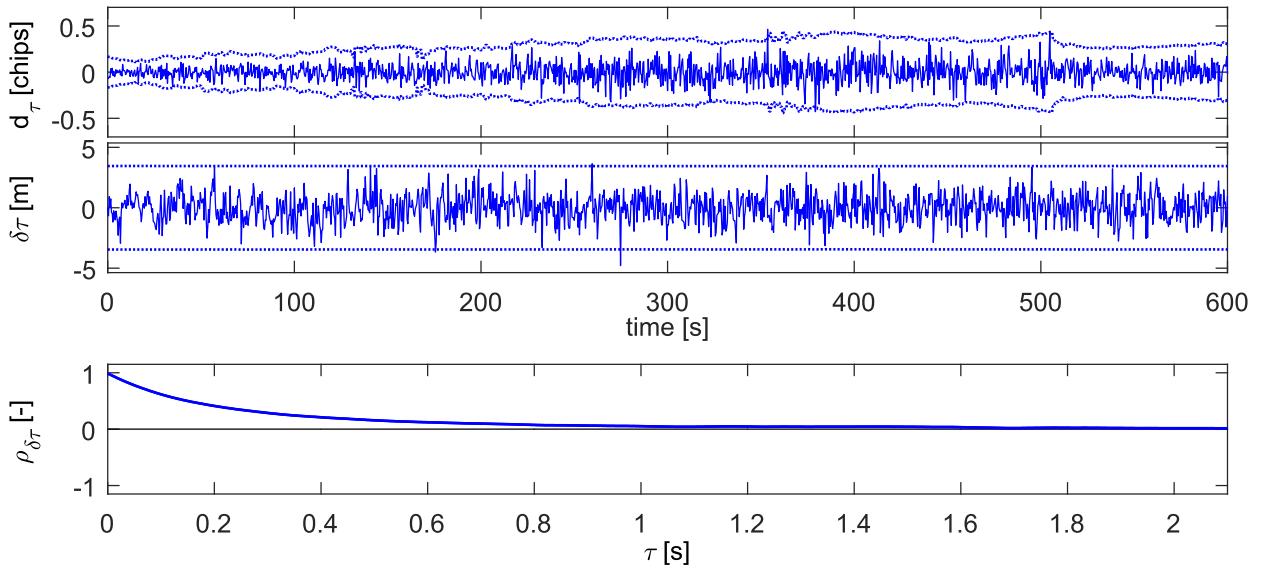


Figure 4-99: Code discriminator, noise and autocorrelation of PRN 16 with constant power scaling

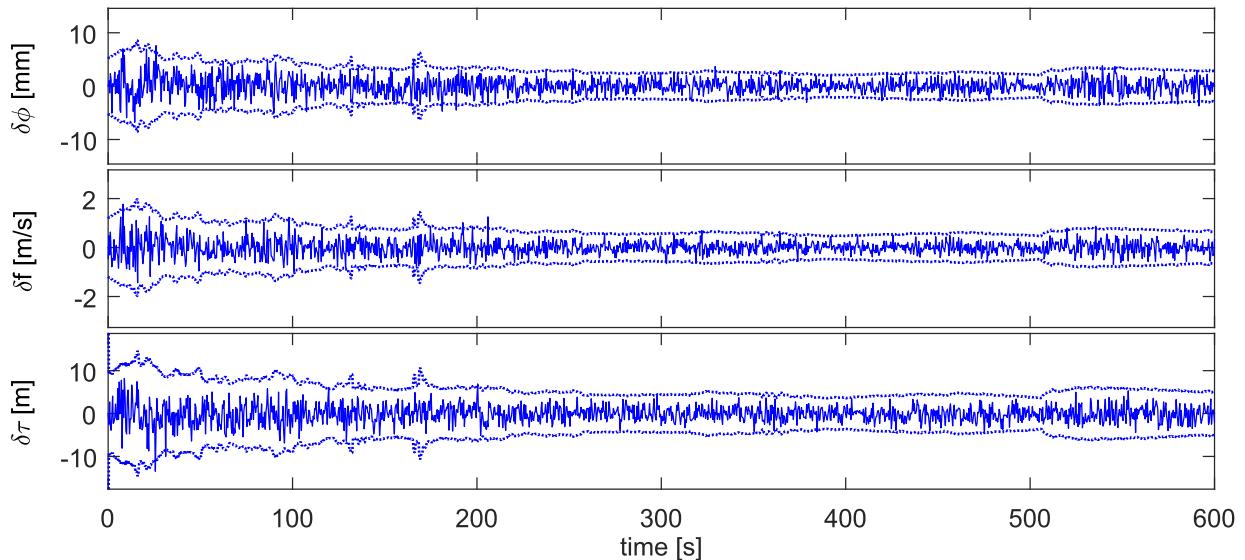


Figure 4-100: Measurement noises and 3σ boundaries (dotted) of PRN 16 with increased bandwidths

5 Navigation Filter

Performance Prediction and Stability

5.1 Steady-State Kalman Filter

5.1.1 Motivation

It can often be observed that the navigation error filter achieves steady-state rapidly after the initial settling phase. In quasi-stationary flight periods with low dynamics, for example straight and leveled, and with aiding measurement noise covariances that vary only slightly, the settled navigation error covariance does also not change much.

It would be nice to be able to appraise the steady-state covariance at least of the nine core navigation states (position, velocity and orientation) in advance without large computational effort, providing that the presumed process noise and aiding measurement noise covariance matrices are more or less constant, that is time-independent, in the foreseeable future.

The method that will be elaborated in this section can for example be applied in mission planning tools where many scenarios under different conditions are tested in parallel and the user quickly gets an evaluation of the expected steady-state navigation error statistics. It could also be used in the navigation system itself in order to continuously monitor the filter performance by comparing the actually estimated covariance of the filter with the predicted one. The navigation system can thus quickly react on suspicious deviations.

In [111], the closed-form solution of the simpler tracking filter problem, which makes use of an exponentially correlated acceleration model as described in [112], is derived. In [113], an analytical steady-state solution of the navigation filter is searched. The found solution is not purely analytic but requires numerically evaluated correction factors.

5.1.2 Preliminaries

Mathematically expressed, the steady-state solution \mathbf{P}_∞ of the CKF with covariance propagation, Kalman gain and covariance update equations (2.11) is searched. In addition, the following assumptions are made:

- The process noise covariance matrix \mathbf{Q}_k is constant, that is time-independent
- The measurement noise covariance matrix \mathbf{R}_k is constant, that is time-independent
- The system matrices Φ , Γ and \mathbf{H} are constant, that is time-independent
- The aiding measurement sample time and the covariance propagation time Δt are equal, meaning that there is an update step after each propagation step.

If the Kalman filter equations (2.11) are combined in a single equation, we get

$$\mathbf{P}_k^- = \Phi \mathbf{P}_{k-1}^- \Phi^T - \Phi \mathbf{P}_{k-1}^- \mathbf{H}^T (\mathbf{R}_k + \mathbf{H} \mathbf{P}_{k-1}^- \mathbf{H}^T)^{-1} \mathbf{H} \mathbf{P}_{k-1}^- \Phi^T + \Gamma \mathbf{Q}_k \Gamma^T \quad (5.1)$$

The searched steady-state solution \mathbf{P}_∞ is obtained by setting $\mathbf{P}_k^- = \mathbf{P}_{k-1}^-$

$$\mathbf{P}_\infty = \Phi \mathbf{P}_\infty \Phi^T - \Phi \mathbf{P}_\infty \mathbf{H}^T (\mathbf{R}_k + \mathbf{H} \mathbf{P}_\infty \mathbf{H}^T)^{-1} \mathbf{H} \mathbf{P}_\infty \Phi^T + \Gamma \mathbf{Q}_k \Gamma^T \quad (5.2)$$

which is a discrete-time algebraic Riccati equation (DARE). The Kalman filter is actually an iterative solver for the DARE. Starting from the given initial covariance matrix $\mathbf{P}_0 \in \mathbb{R}^{(n,n)}$ at time $t = t_0$, the covariance is iteratively adapted until it finally converges to the steady-state covariance \mathbf{P}_∞ . Because of its complicated structure, the analytical solution of the DARE is difficult.

If one is only interested in the steady-state solution, as we are, the discrete-time difference equation (5.1) can alternatively be converted to continuous-time by reducing the discrete-time step Δt to the differential time step dt , which yields the well-known continuous-time Riccati differential equation

$$\dot{\mathbf{P}}(t) = -\mathbf{P}(t) \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}(t) + \mathbf{F} \mathbf{P}(t) + \mathbf{P}(t) \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \quad (5.3)$$

with the continuous-time process noise covariance matrix \mathbf{Q} and measurement noise covariance matrix \mathbf{R} related to the discrete time correspondents by

$$\mathbf{Q} = \Delta t \mathbf{Q}_k, \quad \mathbf{R} = \Delta t \mathbf{R}_k \quad (5.4)$$

[13]. Its steady-state solution is the continuous-time algebraic Riccati equation (CARE)

$$-\mathbf{P}_\infty \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}_\infty + \mathbf{F} \mathbf{P}_\infty + \mathbf{P}_\infty \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T = 0 \quad (5.5)$$

Now, the powerful and comprehensive theory of continuous-time Riccati differential equations can be used to find the steady-state solution \mathbf{P}_∞ . This way has been chosen in the following.

5.1.3 Alternative Representation of the Riccati Equation

As is shown in [114], the quadratic Riccati equation (5.3) can alternatively be represented by a linear homogeneous differential equation system if the covariance matrix $\mathbf{P}(t)$ is decomposed into the numerator matrix $\mathbf{N}(t)$ and denominator matrix $\mathbf{D}(t)$

$$\begin{pmatrix} \dot{\mathbf{N}}(t) \\ \dot{\mathbf{D}}(t) \end{pmatrix} = \underbrace{\begin{pmatrix} \mathbf{F} & \mathbf{G} \mathbf{Q} \mathbf{G}^T \\ \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} & -\mathbf{F}^T \end{pmatrix}}_{=: \Psi} \begin{pmatrix} \mathbf{N}(t) \\ \mathbf{D}(t) \end{pmatrix}, \quad \begin{pmatrix} \mathbf{N}_0 \\ \mathbf{D}_0 \end{pmatrix} = \begin{pmatrix} \mathbf{P}_0 \\ \mathbf{I}_n \end{pmatrix} \quad (5.6)$$

$$\mathbf{P}(t) = \mathbf{N}(t) \mathbf{D}^{-1}(t)$$

The coefficient matrix $\Psi \in \mathbb{R}^{(2n,2n)}$ is a Hamiltonian matrix. The properties of Hamiltonian matrices can be found in [114]. One property is that all eigenvalues of Ψ are symmetric to the real as well as imaginary axes of the complex plane. The solution of the linear homogeneous representation of the Riccati equation (5.6) is

$$\begin{pmatrix} \mathbf{N}(t) \\ \mathbf{D}(t) \end{pmatrix} = e^{\Psi(t-t_0)} \begin{pmatrix} \mathbf{P}_0 \\ \mathbf{I}_n \end{pmatrix} \quad (5.7)$$

$$\mathbf{P}(t) = \mathbf{N}(t) \mathbf{D}^{-1}(t)$$

As will be seen in the next section, the steady-state solution of the Riccati equation (5.5) can be easily calculated with the alternative representation (5.6).

5.1.4 Solution of the Matrix Exponential

The question arises how to calculate the matrix exponential $e^{\Psi(t-t_0)}$ in (5.7). We can find a multitude of solution approaches in the literature, for example [115]. Most of them comprise numerical solution schemes.

The ODE methods make use of the fact that the matrix exponential is the solution of a matrix differential equation if the unit matrix is chosen as initial value

$$\begin{aligned}\dot{\mathbf{X}}(t) &= \Psi \mathbf{X}(t) & \mathbf{X}_0 &= \mathbf{I}_{2n} \\ \mathbf{X}(t) &= e^{\Psi(t-t_0)}\end{aligned}\tag{5.8}$$

The differential equation is integrated with a numerical scheme like 4th order Runge-Kutta. One obtains steady-state if the solution does not change with time anymore. This method does not provide any advantages compared to solving the original problem, the continuous-time Riccati equation (5.3), with a numerical integrator.

The eigenstructure decomposition method decomposes the matrix exponential $e^{\Psi(t-t_0)}$ in its eigenvector matrix and eigenvalue exponential matrix as

$$e^{\Psi(t-t_0)} = \mathbf{V} e^{\Lambda(t-t_0)} \mathbf{V}^{-1} = \mathbf{V} \text{diag}\left(e^{\lambda_1(t-t_0)}, \dots, e^{\lambda_{2n}(t-t_0)}\right) \mathbf{V}^{-1}\tag{5.9}$$

where $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_{2n})$ are the eigenvalues and $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_{2n})$ the eigenvectors of Ψ . The eigenvalues and eigenvectors of the Hamiltonian matrix Ψ may be computed analytically or numerically. However, the eigenvector matrix \mathbf{V} may be (close to) singular. Then, the inverse \mathbf{V}^{-1} and consequently the matrix exponential $e^{\Psi(t-t_0)}$ cannot be calculated or only with considerable numerical errors. Ψ is called defective if Ψ does not have a set of linearly independent eigenvectors. This is the case if there are eigenvalues with geometric multiplicity lower than algebraic multiplicity. The method is interesting if the eigenvalues and eigenvectors can be calculated analytically.

5.1.5 Steady-State Solution of the Riccati Equation

In [114] criteria for the existence of unique and stabilizing solutions of the CARE (5.5) are given. In our case we have a dual Hermitian CARE. With $\mathbf{B}\mathbf{B}^* = -\mathbf{G}\mathbf{Q}\mathbf{G}^T$ and $\mathbf{C}^*\mathbf{C} = -\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}$ Lemma 2.4.1 in [114] says

- If (\mathbf{F}, \mathbf{B}) is stabilizable and (\mathbf{C}, \mathbf{F}) is detectable, then
 - (i) [...] Ψ does not have eigenvalues on the imaginary axis.
 - (ii) $\mathbf{P}_{zz,\infty}$ exists, is stabilizing (i.e., $\mathbf{P}_{zz,\infty} = \mathbf{P}_{zz,s}$) and positive semi-definite.
 - (iv) $\mathbf{P}_{zz,\infty}$ is real (symmetric) if Ψ is real.

According to the Corollary 2.4.3 in [114] the CARE has a (unique) stabilizing solution $\mathbf{P}_{zz,s}$ if and only if Ψ has no purely imaginary eigenvalues.

In the following, the eigenstructure decomposition method (5.9) is used to compute the matrix exponential $e^{\Psi(t-t_0)}$ of (5.7). The Hamiltonian matrix Ψ as defined in (5.6) is first of all decomposed into its eigenmodes

$$\Psi = \begin{pmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{pmatrix} \begin{pmatrix} \Lambda & 0 \\ 0 & -\Lambda \end{pmatrix} \begin{pmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{pmatrix}^{-1}\tag{5.11}$$

Therein, the eigenvalue matrix

$$\mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_n \end{pmatrix} \quad (5.12)$$

contains all the unstable eigenvalues $\lambda_1, \dots, \lambda_n$, $\operatorname{Re}(\lambda_i) > 0$. Consequently, $-\mathbf{\Lambda}$ contains all the stable eigenvalues, which are the unstable ones reflected in the imaginary axis. The eigenvector matrix

$$\begin{pmatrix} \mathbf{V}_{11} \\ \mathbf{V}_{21} \end{pmatrix} = (\mathbf{v}_{\lambda_1}, \mathbf{v}_{\lambda_2}, \dots, \mathbf{v}_{\lambda_n}) \quad (5.13)$$

consists of the eigenvectors that belong to the unstable eigenvalues $\lambda_1, \dots, \lambda_n$. The eigenvector matrix

$$\begin{pmatrix} \mathbf{V}_{12} \\ \mathbf{V}_{22} \end{pmatrix} = (\mathbf{v}_{-\lambda_1}, \mathbf{v}_{-\lambda_2}, \dots, \mathbf{v}_{-\lambda_n}) \quad (5.14)$$

is composed of the eigenvectors that are related to the stable eigenvalues $-\lambda_1, \dots, -\lambda_n$. Note that since we are searching a unique stabilizing solution, the Hamiltonian matrix $\mathbf{\Psi}$ does not have purely imaginary eigenvalues. Inserting the eigenmode decomposition (5.11) into the continuous-time solution of the linear homogeneous differential equation (5.7) yields

$$\begin{aligned} \begin{pmatrix} \mathbf{N}(t) \\ \mathbf{D}(t) \end{pmatrix} &= \begin{pmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{pmatrix} \begin{pmatrix} e^{\mathbf{\Lambda}(t-t_0)} & 0 \\ 0 & e^{-\mathbf{\Lambda}(t-t_0)} \end{pmatrix} \begin{pmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{P}_0 \\ \mathbf{I}_n \end{pmatrix} \\ \mathbf{P}(t) &= \mathbf{N}(t)\mathbf{D}^{-1}(t) \end{aligned} \quad (5.15)$$

where $e^{\mathbf{V}\mathbf{\Lambda}(t-t_0)\mathbf{V}^{-1}} = \mathbf{V}e^{\mathbf{\Lambda}(t-t_0)}\mathbf{V}^{-1}$ has been used. Explicitly calculating $\mathbf{N}(t)$ and $\mathbf{D}(t)$ gives

$$\begin{aligned} \mathbf{N}(t) &= (\mathbf{V}_{11} e^{\mathbf{\Lambda}(t-t_0)} - \mathbf{V}_{12} e^{-\mathbf{\Lambda}(t-t_0)} \mathbf{V}_{22}^{-1} \mathbf{V}_{21})(\mathbf{V}_{11} - \mathbf{V}_{12} \mathbf{V}_{22}^{-1} \mathbf{V}_{21})^{-1} \mathbf{P}_{zz,0} \\ &\quad + (\mathbf{V}_{12} e^{-\mathbf{\Lambda}(t-t_0)} - \mathbf{V}_{11} e^{\mathbf{\Lambda}(t-t_0)} \mathbf{V}_{11}^{-1} \mathbf{V}_{12})(\mathbf{V}_{22} - \mathbf{V}_{21} \mathbf{V}_{11}^{-1} \mathbf{V}_{12})^{-1} \\ \mathbf{D}(t) &= (\mathbf{V}_{21} e^{\mathbf{\Lambda}(t-t_0)} - \mathbf{V}_{22} e^{-\mathbf{\Lambda}(t-t_0)} \mathbf{V}_{22}^{-1} \mathbf{V}_{21})(\mathbf{V}_{11} - \mathbf{V}_{12} \mathbf{V}_{22}^{-1} \mathbf{V}_{21})^{-1} \mathbf{P}_{zz,0} \\ &\quad + (\mathbf{V}_{22} e^{-\mathbf{\Lambda}(t-t_0)} - \mathbf{V}_{21} e^{\mathbf{\Lambda}(t-t_0)} \mathbf{V}_{11}^{-1} \mathbf{V}_{12})(\mathbf{V}_{22} - \mathbf{V}_{21} \mathbf{V}_{11}^{-1} \mathbf{V}_{12})^{-1} \end{aligned} \quad (5.16)$$

For the approximation $t \rightarrow \infty$ the matrix exponential $e^{-\mathbf{\Lambda}(t-t_0)}$ vanishes and $\mathbf{N}(t)$ and $\mathbf{D}(t)$ converge to

$$\begin{aligned} \mathbf{N}(t \rightarrow \infty) &= \mathbf{V}_{11} e^{\mathbf{\Lambda}(t-t_0)} \mathbf{A} \\ \mathbf{D}(t \rightarrow \infty) &= \mathbf{V}_{21} e^{\mathbf{\Lambda}(t-t_0)} \mathbf{A} \end{aligned} \quad (5.17)$$

with the definition

$$\mathbf{A} := (\mathbf{V}_{11} - \mathbf{V}_{12} \mathbf{V}_{22}^{-1} \mathbf{V}_{21})^{-1} \mathbf{P}_{zz,0} - \mathbf{V}_{11}^{-1} \mathbf{V}_{12} (\mathbf{V}_{22} - \mathbf{V}_{21} \mathbf{V}_{11}^{-1} \mathbf{V}_{12})^{-1} \quad (5.18)$$

The steady-state solution \mathbf{P}_∞ does not depend on the time t and is

$$\begin{aligned} \mathbf{P}_\infty &= \mathbf{P}(t \rightarrow \infty) = \mathbf{N}(t \rightarrow \infty)\mathbf{D}^{-1}(t \rightarrow \infty) \\ &= \mathbf{V}_{11} e^{\mathbf{\Lambda}(t-t_0)} \mathbf{A} \mathbf{A}^{-1} e^{-\mathbf{\Lambda}(t-t_0)} \mathbf{V}_{21}^{-1} \\ &= \mathbf{V}_{11} \mathbf{V}_{21}^{-1} \end{aligned} \quad (5.19)$$

The steady-state solution \mathbf{P}_∞ is unique, it does not depend on the initial values of \mathbf{N} and \mathbf{D} . The steady-state solution exists if the eigenvector submatrix \mathbf{V}_{21} is invertible, that is $\det(\mathbf{V}_{21}) \neq 0$.

5.1.6 Application to Navigation Error Filter

The general solution (5.19) is now applied to the navigation error filter. The state vector contains the nine core navigation error states, which are the position, velocity and orientation errors. An analytical solution will be derived. This is charming because the approach is not iterative, the computing time is predictable and thus likewise suitable for offline and online performance predictions of the navigation error filter. Starting point is the linearized navigation error ODE in n -frame (2.7). The steady-state solution is only little influenced by the matrices \mathbf{F}_{11} , \mathbf{F}_{21} , \mathbf{F}_{22} , \mathbf{F}_{31} , \mathbf{F}_{32} and \mathbf{F}_{33} and thus these matrices will be set to zero in the following. Furthermore, the following two assumptions are made:

- The measurements of the accelerometer and gyroscope triads are uncorrelated and have the same variance in all directions

$$\mathbf{Q}_f = \sigma_f^2 \mathbf{I}_3, \quad \mathbf{Q}_\omega = \sigma_\omega^2 \mathbf{I}_3 \quad (5.20)$$

yielding a diagonal process noise matrix

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_f & 0 \\ 0 & \mathbf{Q}_\omega \end{pmatrix} \quad (5.21)$$

- The aiding measurement $\tilde{\mathbf{y}}$ is a position measurement with Cartesian error equation

$$\delta \mathbf{y} = \underbrace{(\mathbf{D} \quad 0 \quad 0)}_{\mathbf{H}} \delta \mathbf{z}_n \quad (5.22)$$

with diagonal covariance matrix \mathbf{R}_{pos} and isotropic variance

$$\mathbf{R}_{pos} = \sigma_{pos}^2 \mathbf{I}_3 \quad (5.23)$$

It is a well-known fact that the heading angle error $\psi_{n\bar{n}}$ is not observable in straight and leveled flight. Generally spoken for arbitrary flight orientations, the angle error about the current acceleration direction $\mathbf{f}_b/\|\mathbf{f}_b\|$ is not observable [12]. The Hamiltonian matrix $\mathbf{\Psi}$ consequently has two eigenvalues $\lambda = 0$, i.e. two eigenvalues on the imaginary axis. Therefore, according to (5.10), a unique solution cannot be found for the nine state problem. A way out is to introduce an auxiliary frame (called o -frame) whose z -axis is just aligned to the current acceleration measurement. Then, the unobservable angle can be clearly separated from the remaining navigation states, which are the position, velocity and the two attitude angles φ_{ob} and ϑ_{ob} . In the end, the found solution in o -frame has to be transformed back to n -frame. The transformation of the state vector $\delta \mathbf{z}_n$ in n -frame to o -frame is accomplished by means of the transformation matrix \mathbf{T}_{on}

$$\begin{pmatrix} \delta \mathbf{x}_o \\ \delta \mathbf{v}_o \\ \varphi_{oo} \\ \vartheta_{oo} \\ \psi_{oo} \end{pmatrix} = \underbrace{\begin{pmatrix} \mathbf{R}_{on} \mathbf{D} & 0 & 0 \\ 0 & \mathbf{R}_{on} & 0 \\ 0 & 0 & \mathbf{R}_{on} \end{pmatrix}}_{\mathbf{T}_{on}} \begin{pmatrix} \delta \phi \\ \delta \lambda \\ \delta h \\ \delta \mathbf{v}_n \\ \varphi_{n\bar{n}} \\ \vartheta_{n\bar{n}} \\ \psi_{n\bar{n}} \end{pmatrix} \quad (5.24)$$

The latitudinal, longitudinal and height errors are mapped to Cartesian errors in the o -frame. \mathbf{R}_{on} is the transformation matrix between the n - and the auxiliary o -frame. It is composed of the o -frame basis vectors \mathbf{e}_x , \mathbf{e}_y and \mathbf{e}_z specified in n -frame

$$\mathbf{R}_{on} = (\mathbf{e}_{x,n} \quad \mathbf{e}_{y,n} \quad \mathbf{e}_{z,n})^T \quad (5.25)$$

The third basis vector points in the opposite direction of the current acceleration

$$\mathbf{e}_{z,n} = -\mathbf{R}_{nb} \mathbf{f}_b / \|\mathbf{f}_b\| \quad (5.26)$$

The first basis vector shall be orthogonal to the third one

$$\mathbf{e}_{x,n} = \frac{(\mathbf{R}_3(\frac{\pi}{2}) \mathbf{R}_2(\frac{\pi}{2}) \mathbf{R}_1(\frac{\pi}{2}) \mathbf{e}_{z,n}) \times \mathbf{e}_{z,n}}{\|(\mathbf{R}_3(\frac{\pi}{2}) \mathbf{R}_2(\frac{\pi}{2}) \mathbf{R}_1(\frac{\pi}{2}) \mathbf{e}_{z,n}) \times \mathbf{e}_{z,n}\|} \quad (5.27)$$

The vectors $\mathbf{R}_3(\frac{\pi}{2}) \mathbf{R}_2(\frac{\pi}{2}) \mathbf{R}_1(\frac{\pi}{2}) \mathbf{e}_{z,n}$ and $\mathbf{e}_{z,n}$ span a plane and the unit vector $\mathbf{e}_{x,n}$ is normal to that plane. The three rotations by $\pi/2$ ensure that the two vectors are never co-linear. The second basis vector completes the orthonormal right hand system

$$\mathbf{e}_{y,n} = \frac{\mathbf{e}_{z,n} \times \mathbf{e}_{x,n}}{\|\mathbf{e}_{z,n} \times \mathbf{e}_{x,n}\|} \quad (5.28)$$

The navigation error ODE transformed to \mathbf{o} -frame with omitted angle error ψ_{oo} is

$$\begin{pmatrix} \dot{\delta}\mathbf{x}_o \\ \dot{\delta}\mathbf{v}_o \\ \dot{\phi}_{oo} \\ \dot{\theta}_{oo} \end{pmatrix} \doteq \begin{pmatrix} 0 & \mathbf{I}_3 & 0 \\ 0 & 0 & -\text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \delta\mathbf{x}_o \\ \delta\mathbf{v}_o \\ \phi_{oo} \\ \theta_{oo} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ \mathbf{R}_{ob} & 0 \\ 0 & \mathbf{R}_{ob,2 \times 3} \end{pmatrix} \begin{pmatrix} \delta\mathbf{f}_b \\ \delta\boldsymbol{\omega}_{ib} \end{pmatrix} \quad (5.29)$$

Therein,

$$\text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} = \begin{pmatrix} 0 & -\tilde{f}_{\tilde{o},z} \\ \tilde{f}_{\tilde{o},z} & 0 \\ 0 & 0 \end{pmatrix} \quad (5.30)$$

and

$$\mathbf{R}_{ob,2 \times 3} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \mathbf{R}_{ob} \quad (5.31)$$

The Hamiltonian matrix $\boldsymbol{\Psi}$ of (5.6) is

$$\boldsymbol{\Psi} = \begin{pmatrix} 0 & \mathbf{I}_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} & 0 & \mathbf{Q}_f & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{Q}_\omega \\ \mathbf{R}_{pos}^{-1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\mathbf{I}_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{veck}^{-1}(\tilde{\mathbf{f}}_o)^T_{3 \times 2} & 0 \end{pmatrix} \quad (5.32)$$

Note that now $\mathbf{Q}_\omega = \sigma_\omega^2 \mathbf{I}_2$. Next, the eigenvalues and eigenvectors of $\boldsymbol{\Psi}$ are analytically calculated in section D.1 in the appendix. The eigenvalues λ_{1-16} are according to (D.14) and (D.16)

$$\begin{aligned} \lambda_{1/2} &= \pm \sqrt{A + B} & A &= \sqrt[3]{-q + \sqrt{q^2 + p^3}} \\ \lambda_{3/4} &= \pm \sqrt{A \left(-\frac{1}{2} + \frac{\sqrt{3}}{2}i \right) + B \left(-\frac{1}{2} - \frac{\sqrt{3}}{2}i \right)} & B &= \sqrt[3]{-q - \sqrt{q^2 + p^3}} \end{aligned} \quad (5.33)$$

$$\begin{aligned}\lambda_{5/6} &= \pm \sqrt{A\left(-\frac{1}{2} - \frac{\sqrt{3}}{2}i\right) + B\left(-\frac{1}{2} + \frac{\sqrt{3}}{2}i\right)} \\ \lambda_{13/14} &= \pm \sqrt{+\frac{\sigma_f}{\sigma_{pos}}i}, \quad \lambda_{15/16} = \pm \sqrt{-\frac{\sigma_f}{\sigma_{pos}}i}\end{aligned}$$

$$p = \frac{1}{3} \frac{\sigma_f^2}{\sigma_{pos}^2}, \quad q = -\frac{1}{2} \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2$$

Using the eigenvectors (D.20), the steady-state solution is according to (5.19) $\mathbf{P}_{zz,\infty} = \mathbf{V}_{11} \mathbf{V}_{21}^{-1}$ with the numerator matrix

$$\mathbf{V}_{11} = \left(\begin{array}{ccc|ccc|cc} \lambda_1 A_1 & \lambda_3 A_3 & \lambda_5 A_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_7 A_7 & \lambda_9 A_9 & \lambda_{11} A_{11} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_{13}^3 \sigma_f^2 & \lambda_{15}^3 \sigma_f^2 \\ \hline \lambda_1^2 A_1 & \lambda_3^2 A_3 & \lambda_5^2 A_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_7^2 A_7 & \lambda_9^2 A_9 & \lambda_{11}^2 A_{11} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_{13}^4 \sigma_f^2 & \lambda_{15}^4 \sigma_f^2 \\ \hline 0 & 0 & 0 & -B_7 & -B_9 & -B_{11} & 0 & 0 \\ B_1 & B_3 & B_5 & 0 & 0 & 0 & 0 & 0 \end{array} \right) \quad (5.34)$$

and the denominator matrix

$$\mathbf{V}_{21} = \left(\begin{array}{ccc|ccc|cc} A_1 & A_3 & A_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & A_7 & A_9 & A_{11} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_{13}^2 \frac{\sigma_f^2}{\sigma_{pos}^2} & \lambda_{15}^2 \frac{\sigma_f^2}{\sigma_{pos}^2} \\ \hline \lambda_1^5 & \lambda_3^5 & \lambda_5^5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_7^5 & \lambda_9^5 & \lambda_{11}^5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_{13}^5 & \lambda_{15}^5 \\ \hline 0 & 0 & 0 & -\lambda_7^4 \|\tilde{\mathbf{f}}_b\| & -\lambda_9^4 \|\tilde{\mathbf{f}}_b\| & -\lambda_{11}^4 \|\tilde{\mathbf{f}}_b\| & 0 & 0 \\ \lambda_1^4 \|\tilde{\mathbf{f}}_b\| & \lambda_3^4 \|\tilde{\mathbf{f}}_b\| & \lambda_5^4 \|\tilde{\mathbf{f}}_b\| & 0 & 0 & 0 & 0 & 0 \end{array} \right) \quad (5.35)$$

where

$$A_i := \lambda_i^2 \sigma_f^2 - \sigma_\omega^2 \|\tilde{\mathbf{f}}_b\|^2, \quad B_i := \lambda_i^3 \sigma_\omega^2 \|\tilde{\mathbf{f}}_b\| \quad (5.36)$$

The searched steady-state solution is thus finally

$$\mathbf{P}_\infty = \left(\begin{array}{ccc|ccc|cc} P_{\infty,11} & & P_{\infty,14} & & & P_{\infty,18} & & \\ & P_{\infty,22} & & P_{\infty,25} & & P_{\infty,27} & & \\ & & P_{\infty,33} & & & & & \\ \hline P_{\infty,14} & & P_{\infty,44} & & & P_{\infty,48} & & \\ & P_{\infty,25} & & P_{\infty,55} & & P_{\infty,57} & & \\ & & P_{\infty,36} & & P_{\infty,66} & & & \\ \hline P_{\infty,27} & & P_{\infty,48} & & & P_{\infty,77} & & \\ P_{\infty,18} & & & & & & P_{\infty,88} & \end{array} \right) \quad (5.37)$$

$$P_{\infty,11} = \frac{\lambda_1 A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - \lambda_3 A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + \lambda_5 A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)}{A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)}$$

$$\begin{aligned}
P_{\infty,22} &= \frac{-\lambda_7 A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + \lambda_9 A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - \lambda_{11} A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)}{-A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)} \\
P_{\infty,33} &= \sigma_{pos}^2 \frac{\lambda_{13} \lambda_{15}^2 - \lambda_{15} \lambda_{13}^3}{\lambda_{15}^3 - \lambda_{13}^3} \\
P_{\infty,44} &= -\frac{\lambda_1^2 A_1 (A_3 \lambda_5^4 - A_5 \lambda_3^4) - \lambda_3^2 A_3 (A_1 \lambda_5^4 + A_5 \lambda_1^4) + \lambda_5^2 A_5 (A_1 \lambda_3^4 - A_3 \lambda_1^4)}{A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)} \\
P_{\infty,55} &= -\frac{\lambda_7^2 A_7 (A_9 \lambda_{11}^4 - A_{11} \lambda_9^4) - \lambda_9^2 A_9 (A_7 \lambda_{11}^4 + A_{11} \lambda_7^4) + \lambda_{11}^2 A_{11} (A_7 \lambda_9^4 - A_9 \lambda_7^4)}{A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) - A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) + A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)} \\
P_{\infty,66} &= -\sigma_f^2 \frac{\lambda_{13}^2 - \lambda_{15}^2}{\lambda_{15}^3 - \lambda_{13}^3} \\
P_{\infty,77} &= \frac{1}{\|\tilde{\mathbf{f}}_b\|} \frac{-B_7 (A_9 \lambda_{11}^5 - A_{11} \lambda_9^5) + B_9 (A_7 \lambda_{11}^5 + A_{11} \lambda_7^5) - B_{11} (A_7 \lambda_9^5 - A_9 \lambda_7^5)}{-A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)} \\
P_{\infty,88} &= \frac{1}{\|\tilde{\mathbf{f}}_b\|} \frac{B_1 (A_3 \lambda_5^5 - A_5 \lambda_3^5) - B_2 (A_1 \lambda_5^5 + A_5 \lambda_1^5) + B_3 (A_1 \lambda_3^5 - A_3 \lambda_1^5)}{A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)} \\
P_{\infty,44} &= \frac{\lambda_1 A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - \lambda_3 A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + \lambda_5 A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)}{A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)} \\
P_{\infty,25} &= \frac{-\lambda_7 A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + \lambda_9 A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - \lambda_{11} A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)}{-A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)} \\
P_{\infty,36} &= \sigma_{pos}^2 \frac{\lambda_{13} \lambda_{15}^3 - \lambda_{15} \lambda_{13}^2}{\lambda_{15}^3 - \lambda_{13}^3} \\
P_{\infty,18} &= \frac{1}{\|\tilde{\mathbf{f}}_b\|} \frac{\lambda_1 A_1 (A_3 \lambda_5^5 - A_5 \lambda_3^5) - \lambda_3 A_3 (A_1 \lambda_5^5 + A_5 \lambda_1^5) + \lambda_5 A_5 (A_1 \lambda_3^5 - A_3 \lambda_1^5)}{A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - A_3 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + A_5 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)} \\
P_{\infty,27} &= \frac{1}{\|\tilde{\mathbf{f}}_b\|} \frac{\lambda_7 A_7 (A_9 \lambda_{11}^5 - A_{11} \lambda_9^5) - \lambda_9 A_9 (A_7 \lambda_{11}^5 + A_{11} \lambda_7^5) + \lambda_{11} A_{11} (A_7 \lambda_9^5 - A_9 \lambda_7^5)}{-A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)} \\
P_{\infty,48} &= \frac{1}{\|\tilde{\mathbf{f}}_b\|} \frac{\lambda_1^2 A_1 (A_3 \lambda_5^5 - A_5 \lambda_3^5) - \lambda_3^2 A_3 (A_1 \lambda_5^5 + A_5 \lambda_1^5) + \lambda_5^2 A_5 (A_1 \lambda_3^5 - A_3 \lambda_1^5)}{A_1 (\lambda_3^5 \lambda_5^4 - \lambda_5^5 \lambda_3^4) - A_2 (\lambda_1^5 \lambda_5^4 + \lambda_5^5 \lambda_1^4) + A_3 (\lambda_1^5 \lambda_3^4 - \lambda_3^5 \lambda_1^4)} \\
P_{\infty,57} &= \frac{1}{\|\tilde{\mathbf{f}}_b\|} \frac{\lambda_7^2 A_7 (A_9 \lambda_{11}^5 - A_{11} \lambda_9^5) - \lambda_9^2 A_9 (A_7 \lambda_{11}^5 + A_{11} \lambda_7^5) + \lambda_{11}^2 A_{11} (A_7 \lambda_9^5 - A_9 \lambda_7^5)}{-A_7 (\lambda_9^5 \lambda_{11}^4 - \lambda_{11}^5 \lambda_9^4) + A_9 (\lambda_7^5 \lambda_{11}^4 + \lambda_{11}^5 \lambda_7^4) - A_{11} (\lambda_7^5 \lambda_9^4 - \lambda_9^5 \lambda_7^4)}
\end{aligned}$$

5.1.7 Example

The correctness of the found solution is verified by means of a simulation example. The calculated steady-state solution is compared with the predicted covariance of the CKF. For that, a non-accelerated straight and leveled flight with 100 m/s in northern and eastern directions is considered. IMU 2 (Table E-5) provides measurements with 100 Hz sample rate. The measurements are only affected by velocity/angular random walk noise N . The inertial navigation solution is aided by 1 Hz position measurements with isotropic standard deviation of 2 m. The position error is shown in Figure 5-1, the velocity error in Figure 5-2 and the orientation error in Figure 5-3. The blue dotted lines are

the 3σ boundaries predicted by the CKF. The red dotted lines are the calculated steady-state solutions. It can be observed that there is a good coincidence between the CKF solution and the calculated steady-state solution at the end of the simulation.

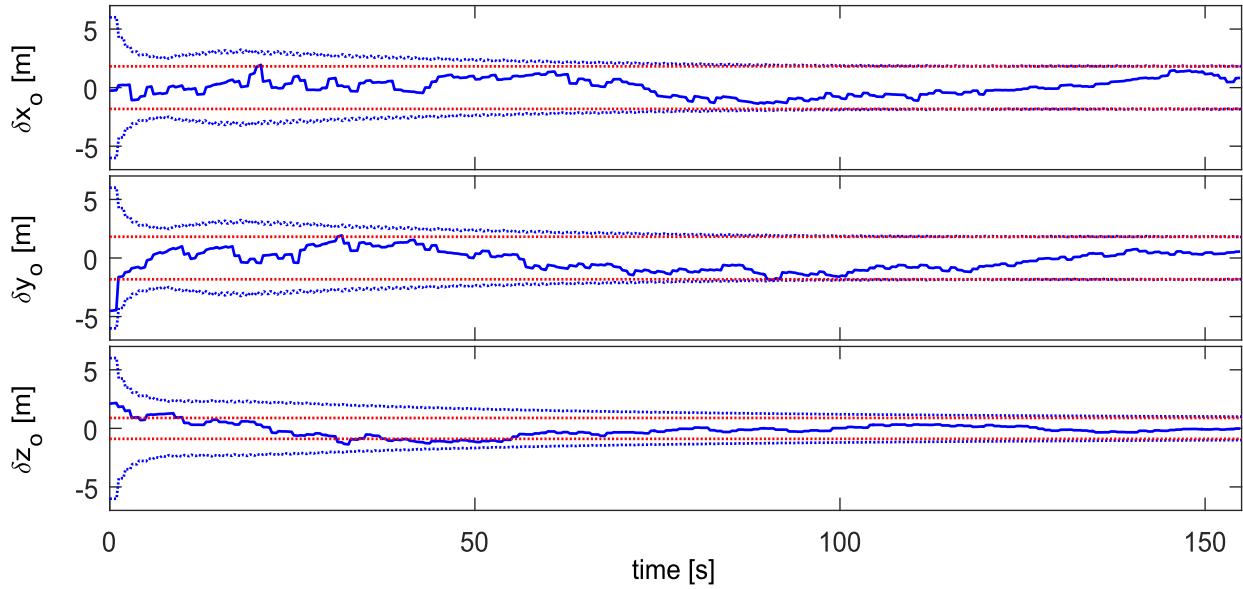


Figure 5-1: Position error and 3σ boundaries (dotted, blue: filter prediction, red: steady-state)

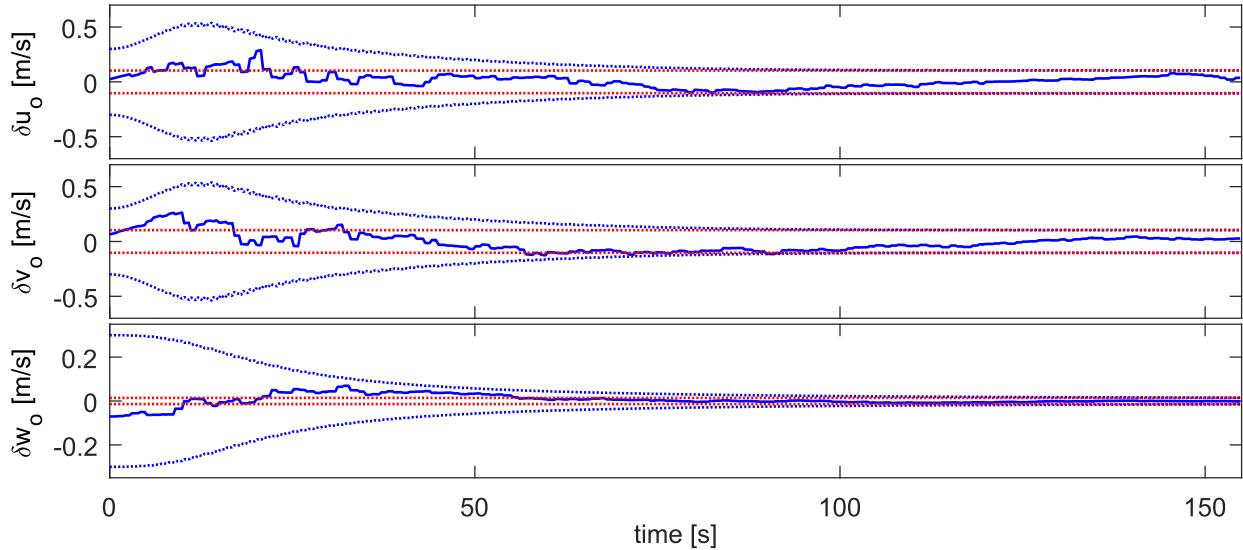


Figure 5-2: Velocity error and 3σ boundaries (dotted, blue: filter prediction, red: steady-state)

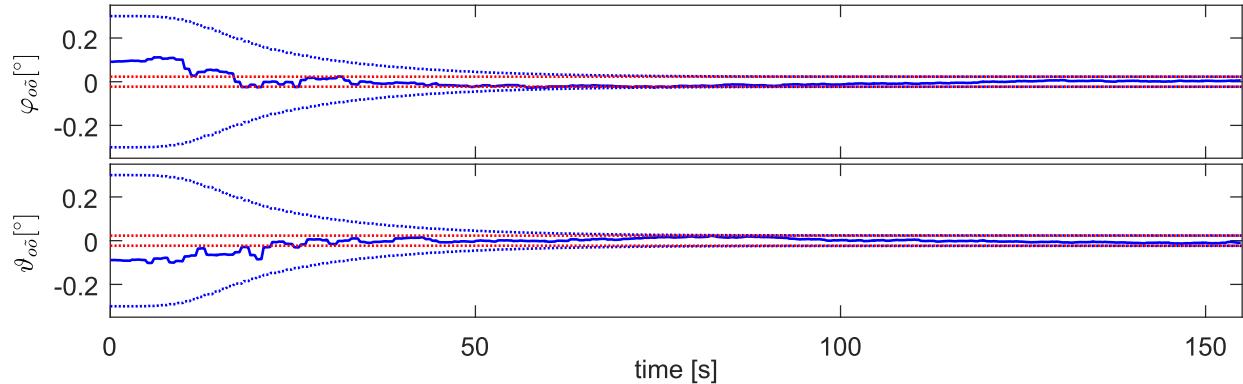


Figure 5-3: Orientation error and 3σ boundaries (dotted, blue: filter prediction, red: steady-state)

5.2 Kalman Filter Stability

5.2.1 Motivation

During the numerous simulations that have been conducted in the context of this thesis cases were observed from time to time in which the Kalman filter solution immediately became unstable. The computed state covariance matrix quickly diverged although the initial covariance matrix \mathbf{P}_0 , the process noise covariance matrix \mathbf{Q} and the measurement noise covariance matrix \mathbf{R} were well posed, symmetric and positive definite. When looking in detail into the concerned algorithms, the reason of the instability became clear. In the covariance update equation the transposed covariance matrix \mathbf{P}^T was used instead of the covariance matrix \mathbf{P} , $\mathbf{P}_k^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^{-T}$. When programming the algorithm, it seemed to be irrelevant if the covariance matrix \mathbf{P} itself or its transpose is inserted because, in theory, the covariance matrix \mathbf{P} is symmetric, that is $\mathbf{P} = \mathbf{P}^T$. However, it turned out that numerical effects like rounding errors led to a gradual desymmetrization of the covariance matrix and subsequently to the destabilization of the filter in the case of the transposed covariance matrix, in contrast to the normal, non-transposed case where the destabilization could not be observed. The first conclusion of the observation was that it is irremissible to use a stability enhancement method to mitigate the negative destabilizing effect of numerical errors. A simple but effective method is to occasionally add the covariance and its transpose, $\mathbf{P}_k^+ = \frac{1}{2}(\mathbf{P}_k^+ + \mathbf{P}_k^{+T})$. Alternatively, but computationally more expansive, Joseph's form of the covariance update equation can be applied, which inherently induces symmetry without further measures. At second glance, the interesting question arose how this effect can be explained mathematically. In the following, an attempt is made to give an answer to this question. The answer is especially important when delayed measurements are processed with the measurement prediction or history state update methods. There, a similar effect that stems from the same reason occurs in the derived equations nominally. In this case, symmetry enhancement is compulsory. In literature, many contributions can be found that deal with the stability analysis of Kalman filter, for example with regard to process noise uncertainties or numerical approximations as discussed in [116] or [117], but not with this special issue.

5.2.2 Discrete-Time Kalman Filter Variants to be Analyzed

Four different Kalman filter variants, defined by

$$\begin{aligned} \text{Propagation: } & \mathbf{P}_k^- = \Phi \mathbf{P}_{k-1}^+ \Phi^T + \Gamma \mathbf{Q}_k \Gamma^T \\ \text{Update: } & \mathbf{S}_k = \mathbf{R}_k + \mathbf{H} \mathbf{P}_k^- \mathbf{H}^T \\ \text{Variant(1)} & \quad \mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T \mathbf{S}_k^{-1}, \quad \mathbf{P}_k^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^- \\ \text{Variant(2)} & \quad \mathbf{K}_k = \mathbf{P}_k^{-T} \mathbf{H}^T \mathbf{S}_k^{-1}, \quad \mathbf{P}_k^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^- \\ \text{Variant(3)} & \quad \mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T \mathbf{S}_k^{-1}, \quad \mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H} \mathbf{P}_k^{-T} \\ \text{Variant(4)} & \quad \mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T \mathbf{S}_k^{-1}, \quad \mathbf{P}_k^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^{-T} \end{aligned} \tag{5.38}$$

shall be analyzed in terms of their stability. The state covariance propagation and the calculation of the innovation covariance \mathbf{S}_k are the same for all four variants. The four variants vary in the calculation of the Kalman gain and the covariance update. The first variant is the CKF (2.11) and serves as reference. In the second variant, the a priori covariance matrix \mathbf{P}_k^- in the Kalman gain

equation is transposed. In the third variant, the a priori covariance matrix \mathbf{P}_k^- is transposed in the covariance update equation. Finally, in the fourth variant, the a priori covariance matrices in the Kalman gain as well as in the covariance update equation are transposed. Other variants are possible and have been analyzed but go beyond the scope of this thesis. The destabilization of the filter has been observed in the variants (2) – (4). Variant (3) is relevant for the delayed measurement processing. In general, it can be assumed that the discrete-time difference equation (5.1) and its continuous-time differential correspondent (5.3) have similar dynamic behavior if the time step size of the discrete-time difference equation Δt is sufficiently small. The stability analysis should thus be possible in discrete- as well as continuous-time domain. Statements concerning the stability that are made for discrete-time filters are likewise valid for continuous-time filters and vice versa. It turned out to be advantageous to examine the stability in continuous-time due to the comprehensive theory of Riccati, Sylvester and Lyapunov differential equations. Therefore, the four variants are converted from discrete-time to continuous-time next.

5.2.3 Conversion to Continuous-Time Differential Equations

The continuous-time version of variant (1) is the conventional Riccati equation (5.3). The conversion of variant (2) is carried out in the same way as in the case of the CKF. The covariance update equation is inserted into the propagation equation. With the discretized system and input matrices $\Phi = \mathbf{I}_n + \mathbf{F}\Delta t + O(\Delta t^2)$, $\Gamma = \mathbf{G}\Delta t + O(\Delta t^2)$ and $\mathbf{Q}_k = \mathbf{Q}/\Delta t$ one gets

$$\begin{aligned} \mathbf{P}_k^- = & (\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}) \mathbf{P}_{k-1}^- \\ & + [\mathbf{F}(\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}) \mathbf{P}_{k-1}^- + (\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}) \mathbf{P}_{k-1}^- \mathbf{F}^\top] \Delta t + \mathbf{G} \mathbf{Q} \mathbf{G}^\top \Delta t + O(\Delta t^2) \end{aligned} \quad (5.39)$$

The difference quotient is formed

$$\frac{\mathbf{P}_k^- - \mathbf{P}_{k-1}^-}{\Delta t} = -\frac{\mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^-}{\Delta t} + \mathbf{F}(\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}) \mathbf{P}_{k-1}^- + (\mathbf{I}_n - \mathbf{K}_{k-1} \mathbf{H}) \mathbf{P}_{k-1}^- \mathbf{F}^\top + \mathbf{G} \mathbf{Q} \mathbf{G}^\top + O(\Delta t) \quad (5.40)$$

With $\mathbf{R}_k = \mathbf{R}/\Delta t$ the Kalman gain becomes

$$\mathbf{K}_{k-1} = \mathbf{P}_{k-1}^{-\top} \mathbf{H}^\top (\mathbf{R} + \mathbf{H} \mathbf{P}_{k-1}^- \mathbf{H}^\top \Delta t)^{-1} \Delta t \quad (5.41)$$

Inserting the Kalman gain \mathbf{K}_{k-1} into the difference quotient (5.40) and letting the time step size Δt converge against zero yields the differential quotient

$$\lim_{\Delta t \rightarrow 0} \frac{\mathbf{P}_k^- - \mathbf{P}_{k-1}^-}{\Delta t} = -\mathbf{P}_{k-1}^{-\top} \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \mathbf{P}_{k-1}^- + \mathbf{F} \mathbf{P}_{k-1}^- + \mathbf{P}_{k-1}^- \mathbf{F}^\top + \mathbf{G} \mathbf{Q} \mathbf{G}^\top \quad (5.42)$$

which is a Riccati like differential equation but with transposed covariance in the quadratic term

$$\dot{\mathbf{P}}(t) = -\mathbf{P}^\top(t) \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \mathbf{P}(t) + \mathbf{F} \mathbf{P}(t) + \mathbf{P}(t) \mathbf{F}^\top + \mathbf{G} \mathbf{Q} \mathbf{G}^\top \quad (5.43)$$

In variant (3) the combined covariance propagation and update equation is

$$\begin{aligned} \mathbf{P}_k^- = & (\mathbf{P}_{k-1}^- - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-\top}) \\ & + [\mathbf{F}(\mathbf{P}_{k-1}^- - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-\top}) + (\mathbf{P}_{k-1}^- - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-\top}) \mathbf{F}^\top] \Delta t + \mathbf{G} \mathbf{Q} \mathbf{G}^\top \Delta t + O(\Delta t^2) \end{aligned} \quad (5.44)$$

Analogous to the previous case, the difference quotient is formed

$$\begin{aligned} \frac{\mathbf{P}_k^- - \mathbf{P}_{k-1}^-}{\Delta t} = & -\frac{\mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-\top}}{\Delta t} \\ & + \mathbf{F}(\mathbf{P}_{k-1}^- - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-\top}) + (\mathbf{P}_{k-1}^- - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-\top}) \mathbf{F}^\top + \mathbf{G} \mathbf{Q} \mathbf{G}^\top + O(\Delta t) \end{aligned} \quad (5.45)$$

With the conventional Kalman gain the differential quotient

$$\lim_{\Delta t \rightarrow 0} \frac{\mathbf{P}_k^- - \mathbf{P}_{k-1}^-}{\Delta t} = -\mathbf{P}_{k-1}^- \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}_{k-1}^{-T} + \mathbf{F} \mathbf{P}_{k-1}^- + \mathbf{P}_{k-1}^- \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \quad (5.46)$$

leads to the Riccati like differential equation

$$\dot{\mathbf{P}}(t) = -\mathbf{P}(t) \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}(t)^T + \mathbf{F} \mathbf{P}(t) + \mathbf{P}(t) \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \quad (5.47)$$

In this case the rear covariance in the quadratic term is transposed compared to the conventional Riccati equation.

The conversion of variant (4) is a little bit more complicated because two subsequent time steps are involved. In accordance with (5.39) the covariance propagation between time t_{k-1} and t_k can be expressed as

$$\begin{aligned} \mathbf{P}_k^- = & (\mathbf{P}_{k-1}^{-T} - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-T}) \\ & + [\mathbf{F}(\mathbf{P}_{k-1}^{-T} - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-T}) + (\mathbf{P}_{k-1}^{-T} - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-1}^{-T}) \mathbf{F}^T] \Delta t + \mathbf{G} \mathbf{Q} \mathbf{G}^T \Delta t + O(\Delta t^2) \end{aligned} \quad (5.48)$$

Equally, the covariance propagation between t_{k-2} and t_{k-1} is

$$\begin{aligned} \mathbf{P}_{k-1}^{-T} = & (\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) \\ & + [\mathbf{F}(\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) + (\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) \mathbf{F}^T] \Delta t + \mathbf{G} \mathbf{Q} \mathbf{G}^T \Delta t + O(\Delta t^2) \end{aligned} \quad (5.49)$$

Inserting (5.49) in (5.48) and forming the difference quotient yields

$$\begin{aligned} \frac{\mathbf{P}_k^- - \mathbf{P}_{k-2}^-}{2 \Delta t} = & \frac{-\mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T}{2 \Delta t} \\ & + \frac{1}{2} [\mathbf{F}(\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) + (\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) \mathbf{F}^T] + \frac{1}{2} \mathbf{G} \mathbf{Q} \mathbf{G}^T \\ & + \frac{-\mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-2}^- + \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T}{2 \Delta t} \\ & - \frac{1}{2} \mathbf{K}_{k-1} \mathbf{H} [\mathbf{F}(\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) + (\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) \mathbf{F}^T] \\ & - \frac{1}{2} \mathbf{K}_{k-1} \mathbf{H} \mathbf{G} \mathbf{Q} \mathbf{G}^T \\ & + \frac{1}{2} \mathbf{F}(\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-2}^- + \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) \\ & + \frac{1}{2} (\mathbf{P}_{k-2}^- - \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T - \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-2}^- + \mathbf{K}_{k-1} \mathbf{H} \mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{K}_{k-2}^T) \mathbf{F}^T \\ & + \frac{1}{2} \mathbf{G} \mathbf{Q} \mathbf{G}^T + O(\Delta t) \end{aligned} \quad (5.50)$$

With the Kalman gains

$$\mathbf{K}_{k-1} = \mathbf{P}_{k-1}^- \mathbf{H}^T (\mathbf{R} + \mathbf{H} \mathbf{P}_{k-1}^- \mathbf{H}^T \Delta t)^{-1} \Delta t, \quad \mathbf{K}_{k-2} = \mathbf{P}_{k-2}^- \mathbf{H}^T (\mathbf{R} + \mathbf{H} \mathbf{P}_{k-2}^- \mathbf{H}^T \Delta t)^{-1} \Delta t \quad (5.51)$$

the differential quotient becomes

$$\begin{aligned} \lim_{\Delta t \rightarrow 0} \frac{\mathbf{P}_k^- - \mathbf{P}_{k-2}^-}{2 \Delta t} = & -\frac{1}{2} (\mathbf{P}_{k-2}^- \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}_{k-2}^{-T} + \mathbf{P}_{k-2}^{-T} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}_{k-2}^-) \\ & + \mathbf{F} \mathbf{P}_{k-2}^- + \mathbf{P}_{k-2}^- \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \end{aligned} \quad (5.52)$$

which is a Riccati like differential equation with two superposed quadratic terms

$$\dot{\mathbf{P}}(t) = -\frac{1}{2} (\mathbf{P}(t) \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}(t)^T + \mathbf{P}^T(t) \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}(t)) + \mathbf{F} \mathbf{P}(t) + \mathbf{P}(t) \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \quad (5.53)$$

The continuous-time differential equations are summarized for the four different variants

$$\begin{aligned}
 \text{Variant (1)} \quad \dot{\tilde{\mathbf{P}}} &= -\mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P} + \mathbf{F} \mathbf{P} + \mathbf{P} \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \\
 \text{Variant (2)} \quad \dot{\tilde{\mathbf{P}}} &= -\mathbf{P}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P} + \mathbf{F} \mathbf{P} + \mathbf{P} \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \\
 \text{Variant (3)} \quad \dot{\tilde{\mathbf{P}}} &= -\mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}^T + \mathbf{F} \mathbf{P} + \mathbf{P} \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T \\
 \text{Variant (4)} \quad \dot{\tilde{\mathbf{P}}} &= -\frac{1}{2} \mathbf{P}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P} - \frac{1}{2} \mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \mathbf{P}^T + \mathbf{F} \mathbf{P} + \mathbf{P} \mathbf{F}^T + \mathbf{G} \mathbf{Q} \mathbf{G}^T
 \end{aligned} \tag{5.54}$$

The time argument (t) is omitted for convenience.

5.2.4 Covariance Error Differential Equations

In this section the covariance error differential equations are derived for the four different variants in order to predict the dynamic behavior of errors (for example rounding errors due to limited number representation). The true, that is error-free state, process noise and inverse measurement noise covariance matrices are split into erroneous and perturbation matrices

$$\begin{aligned}
 \mathbf{P} &= \tilde{\mathbf{P}} + \delta\mathbf{P} \\
 \mathbf{Q} &= \tilde{\mathbf{Q}} + \delta\mathbf{Q} \\
 \mathbf{R}^{-1} &= \tilde{\mathbf{R}}^{-1} + \delta\mathbf{R}^{-1}
 \end{aligned} \tag{5.55}$$

Inserting these split matrices into the covariance differential equation (5.3) yields

$$\begin{aligned}
 \dot{\tilde{\mathbf{P}}} + \delta\dot{\mathbf{P}} &= -(\tilde{\mathbf{P}} + \delta\mathbf{P}) \mathbf{H}^T (\tilde{\mathbf{R}}^{-1} + \delta\mathbf{R}^{-1}) \mathbf{H} (\tilde{\mathbf{P}} + \delta\mathbf{P}) \\
 &\quad + \mathbf{F}(\tilde{\mathbf{P}} + \delta\mathbf{P}) + (\tilde{\mathbf{P}} + \delta\mathbf{P}) \mathbf{F}^T + \mathbf{G}(\tilde{\mathbf{Q}} + \delta\mathbf{Q}) \mathbf{G}^T
 \end{aligned} \tag{5.56}$$

The erroneous matrices $\tilde{\mathbf{P}}$, $\tilde{\mathbf{Q}}$ and $\tilde{\mathbf{R}}^{-1}$ fulfill the Riccati equation exactly

$$\dot{\tilde{\mathbf{P}}} = -\tilde{\mathbf{P}} \mathbf{H}^T \tilde{\mathbf{R}}^{-1} \mathbf{H} \tilde{\mathbf{P}} + \mathbf{F} \tilde{\mathbf{P}} + \mathbf{P} \mathbf{F}^T + \mathbf{G} \tilde{\mathbf{Q}} \mathbf{G}^T \tag{5.57}$$

The differential equation for the perturbation covariance matrix becomes

$$\begin{aligned}
 \text{Variant (1)} \quad \delta\dot{\mathbf{P}} &= -\tilde{\mathbf{P}} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P} - \delta\mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}} - \delta\mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P} \\
 &\quad + \mathbf{F} \delta\mathbf{P} + \delta\mathbf{P} \mathbf{F}^T - \tilde{\mathbf{P}} \mathbf{H}^T \delta\mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}} + \mathbf{G} \delta\mathbf{Q} \mathbf{G}^T
 \end{aligned} \tag{5.58}$$

The same is repeated with variants (2) – (4)

$$\begin{aligned}
 \text{Variant (2)} \quad \delta\dot{\mathbf{P}} &= -\delta\mathbf{P}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P} + (\mathbf{F} - \tilde{\mathbf{P}}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}) \delta\mathbf{P} + \delta\mathbf{P}^T (\mathbf{F}^T - \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}) \\
 &\quad - \tilde{\mathbf{P}}^T \mathbf{H}^T \delta\mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}} + \mathbf{G} \delta\mathbf{Q} \mathbf{G}^T
 \end{aligned} \tag{5.59}$$

$$\begin{aligned}
 \text{Variant (3)} \quad \delta\dot{\mathbf{P}} &= -\delta\mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P}^T + (\mathbf{F} - \tilde{\mathbf{P}} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}) \delta\mathbf{P}^T + \delta\mathbf{P} (\mathbf{F}^T - \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}^T) \\
 &\quad - \tilde{\mathbf{P}} \mathbf{H}^T \delta\mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}^T + \mathbf{G} \delta\mathbf{Q} \mathbf{G}^T
 \end{aligned}$$

$$\begin{aligned}
 \text{Variant (4)} \quad \delta\dot{\mathbf{P}} &= -\frac{1}{2} (\delta\mathbf{P} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P}^T + \delta\mathbf{P}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P}) \\
 &\quad + (\mathbf{F} - \frac{1}{2} \tilde{\mathbf{P}}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}) \delta\mathbf{P} + \delta\mathbf{P} (\mathbf{F}^T - \frac{1}{2} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}^T) \\
 &\quad - \frac{1}{2} \tilde{\mathbf{P}} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \delta\mathbf{P}^T - \frac{1}{2} \delta\mathbf{P}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}} \\
 &\quad - \frac{1}{2} (\tilde{\mathbf{P}} \mathbf{H}^T \delta\mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}^T + \tilde{\mathbf{P}}^T \mathbf{H}^T \delta\mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}) + \mathbf{G} \delta\mathbf{Q} \mathbf{G}^T
 \end{aligned}$$

Next, the covariance $\tilde{\mathbf{P}}$ as well as the covariance error $\delta\mathbf{P}$ are split into symmetric and asymmetric that is skew-symmetric parts

$$\tilde{\mathbf{P}} = \tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_a, \quad \delta\mathbf{P} = \delta\mathbf{P}_s + \delta\mathbf{P}_a \tag{5.60}$$

Since the error-free covariance $\mathbf{P} = \tilde{\mathbf{P}} + \delta\mathbf{P} = \tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_a + \delta\mathbf{P}_s + \delta\mathbf{P}_a$ is symmetric, the following relation between $\tilde{\mathbf{P}}_a$ and $\delta\mathbf{P}_a$ has to hold

$$\tilde{\mathbf{P}}_a = -\delta\mathbf{P}_a \quad (5.61)$$

The inverse measurement noise covariance error $\delta\mathbf{R}^{-1}$ and the process noise covariance error $\delta\mathbf{Q}$ are also parted into symmetric and asymmetric matrices

$$\delta\mathbf{R}^{-1} = \delta\mathbf{R}_s^{-1} + \delta\mathbf{R}_a^{-1}, \quad \delta\mathbf{Q} = \delta\mathbf{Q}_s + \delta\mathbf{Q}_a \quad (5.62)$$

Inserting these split matrices into the covariance error differential equation of variant (1) yields

$$\begin{aligned} \delta\dot{\mathbf{P}}_s + \delta\dot{\mathbf{P}}_a &= -(\delta\mathbf{P}_s + \delta\mathbf{P}_a)\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}(\delta\mathbf{P}_s + \delta\mathbf{P}_a) - (\tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_a)\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}(\delta\mathbf{P}_s + \delta\mathbf{P}_a) \\ &\quad - (\delta\mathbf{P}_s + \delta\mathbf{P}_a)\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}(\tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_a) + \mathbf{F}(\delta\mathbf{P}_s + \delta\mathbf{P}_a) + (\delta\mathbf{P}_s + \delta\mathbf{P}_a)\mathbf{F}^T \\ &\quad - (\tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_a)\mathbf{H}^T(\delta\mathbf{R}_s^{-1} + \delta\mathbf{R}_a^{-1})\mathbf{H}(\tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_a) + \mathbf{G}\delta\mathbf{Q}_s\mathbf{G}^T + \mathbf{G}\delta\mathbf{Q}_a\mathbf{G}^T \end{aligned} \quad (5.63)$$

After using relation (5.61), cancelling double terms and reordering, the differential equation can be separated into a differential equation for the symmetric covariance error $\delta\mathbf{P}_s$ and a differential equation for the asymmetric error $\delta\mathbf{P}_a$. The symmetric differential equation, which contains only symmetric terms, is

$$\begin{aligned} \delta\dot{\mathbf{P}}_s &= -\delta\mathbf{P}_s\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\delta\mathbf{P}_s - \tilde{\mathbf{P}}_s\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\delta\mathbf{P}_s - \delta\mathbf{P}_s\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}_s + \mathbf{F}\delta\mathbf{P}_s + \delta\mathbf{P}_s\mathbf{F}^T + \delta\mathbf{P}_a\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\delta\mathbf{P}_a \\ &\quad - \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_s^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\delta\mathbf{P}_a - \delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_s^{-1}\mathbf{H}\delta\mathbf{P}_a + \delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \mathbf{G}\delta\mathbf{Q}_s\mathbf{G}^T \end{aligned} \quad (5.64)$$

The asymmetric differential equation is

$$\begin{aligned} \delta\dot{\mathbf{P}}_a &= -\tilde{\mathbf{P}}_s\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\delta\mathbf{P}_a - \delta\mathbf{P}_a\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \mathbf{F}\delta\mathbf{P}_a + \delta\mathbf{P}_a\mathbf{F}^T \\ &\quad + \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_s^{-1}\mathbf{H}\delta\mathbf{P}_a - \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_s^{-1}\mathbf{H}\tilde{\mathbf{P}}_s \\ &\quad - \delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\delta\mathbf{P}_a + \mathbf{G}\delta\mathbf{Q}_a\mathbf{G}^T \end{aligned} \quad (5.65)$$

The segmentation of the covariance error differential equation into symmetric and asymmetric equations is also done for the variants (2) – (4) in the same manner. The symmetric covariance error does not influence the stability of the Kalman filter. The asymmetric covariance error, however, may lead to a destabilization of the filter. The instability threat is particularly high if the asymmetric error differential equation is unstable or critically stable. The attention is therefore concentrated on the asymmetric error differential equations, which are recapitulated for the four variants

$$\begin{aligned} \text{Variant (1)} \quad \delta\dot{\mathbf{P}}_a &= -\delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\delta\mathbf{P}_a + (\mathbf{F} - \tilde{\mathbf{P}}_s\mathbf{H}^T\tilde{\mathbf{R}}_s^{-1}\mathbf{H})\delta\mathbf{P}_a + \delta\mathbf{P}_a(\mathbf{F}^T - \mathbf{H}^T\tilde{\mathbf{R}}_s^{-1}\mathbf{H}\tilde{\mathbf{P}}_s) \\ &\quad - \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \mathbf{G}\delta\mathbf{Q}_a\mathbf{G}^T \\ \text{Variant (2)} \quad \delta\dot{\mathbf{P}}_a &= +\delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\delta\mathbf{P}_a + \mathbf{F}\delta\mathbf{P}_a + \delta\mathbf{P}_a\mathbf{F}^T - \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \mathbf{G}\delta\mathbf{Q}_a\mathbf{G}^T \\ \text{Variant (3)} \quad \delta\dot{\mathbf{P}}_a &= +\delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\delta\mathbf{P}_a + \mathbf{F}\delta\mathbf{P}_a + \delta\mathbf{P}_a\mathbf{F}^T - \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \mathbf{G}\delta\mathbf{Q}_a\mathbf{G}^T \\ \text{Variant (4)} \quad \delta\dot{\mathbf{P}}_a &= +\delta\mathbf{P}_a\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\delta\mathbf{P}_a + \mathbf{F}\delta\mathbf{P}_a + \delta\mathbf{P}_a\mathbf{F}^T - \tilde{\mathbf{P}}_s\mathbf{H}^T\delta\mathbf{R}_a^{-1}\mathbf{H}\tilde{\mathbf{P}}_s + \mathbf{G}\delta\mathbf{Q}_a\mathbf{G}^T \end{aligned} \quad (5.66)$$

It can be easily observed that the variants (2) – (4) have the same asymmetric error dynamics. Only the asymmetric error dynamics of the nominal case (1) differs.

5.2.5 Stability Analysis

In this section it is analyzed if the asymmetric covariance error differential equations (5.66) are stable or not. In case of stability asymmetries coming for example from numerical rounding are continuously damped. The solution of the covariance differential equation is hence stable. Otherwise, the asymmetric error will continuously grow and finally destabilize the filter solution. A glance on the asymmetric covariance error differential equations (5.66) clarifies that the differential equations are generally of Riccati type.

Since the measurement covariance matrix \mathbf{R} and the process noise covariance matrix \mathbf{Q} are often given, or can be symmetrized before using in the algorithm, the asymmetrical errors $\delta\mathbf{R}_a^{-1}$ and $\delta\mathbf{Q}_a$ can be assumed to be zero

$$\begin{array}{ll} \text{Variant(1)} & \dot{\delta\mathbf{P}}_a = (\mathbf{F} - \tilde{\mathbf{P}}_s \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}) \delta\mathbf{P}_a + \delta\mathbf{P}_a (\mathbf{F}^T - \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \tilde{\mathbf{P}}_s) \\ \text{Variants (2), (3), (4)} & \dot{\delta\mathbf{P}}_a = \mathbf{F} \delta\mathbf{P}_a + \delta\mathbf{P}_a \mathbf{F}^T \end{array} \quad (5.67)$$

The simplified asymmetric covariance error differential equations (5.67) are then homogeneous Lyapunov differential equations

$$\dot{\delta\mathbf{P}}_a(t) = \mathbf{M} \delta\mathbf{P}_a(t) + \delta\mathbf{P}_a(t) \mathbf{M}^T, \quad \delta\mathbf{P}_{a,0} = \delta\mathbf{P}_a(t_0) \quad (5.68)$$

with $\mathbf{M} = \mathbf{F} - \tilde{\mathbf{P}}_s \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$ in variant (1) and $\mathbf{M} = \mathbf{F}$ in variants (2), (3) and (4). The unique solution of the homogeneous Lyapunov differential equation (5.68) is given with

$$\delta\mathbf{P}_a(t) = e^{\mathbf{M}(t-t_0)} \delta\mathbf{P}_{a,0} (e^{\mathbf{M}(t-t_0)})^T \quad (5.69)$$

It can be easily seen that an initial asymmetric covariance error $\delta\mathbf{P}_{a,0}$ decays if the matrix \mathbf{M} is stable. \mathbf{M} is stable if the real parts of all eigenvalues of \mathbf{M} are negative, $\text{Re}(\lambda_i(\mathbf{M})) < 0$. The asymmetric error does not grow at least if \mathbf{M} is neutrally stable, that is the real parts of the eigenvalues of the matrix \mathbf{M} are negative or zero, $\text{Re}(\lambda_i(\mathbf{M})) \leq 0$. However, if there are one or more eigenvalues with positive real part, $\text{Re}(\lambda_i(\mathbf{M})) > 0$, the initial asymmetric covariance error $\delta\mathbf{P}_{a,0}$ will continuously increase with time.

If the asymmetric error is stable, the solutions of the differential equations (5.54) are stable and vice versa, if the asymmetric error is unstable, the differential equations (5.54) are unstable. The same can be stated for the corresponding discrete-time difference equations (5.38). (5.69) describes the principal dynamic behavior of asymmetric covariance errors. It is not only valid for the initial error $\delta\mathbf{P}_{a,0}$ at time t_0 but for all asymmetries $\delta\mathbf{P}_a(t)$ that occur during the runtime of the filter. (5.69) illustrates if and how fast numerical asymmetries due to rounding are damped. Consequently, the eigenvalues of the matrix \mathbf{M} have to be analyzed in order to make a statement about the stability of the filter equations. Let this be exemplified. Assume that the navigation error filter is set up in e -frame. The navigation error ODE is given in (2.6). If the filter receives a position aiding measurement with covariance \mathbf{R}_{pos} , the matrix $\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$ becomes

$$\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} = \begin{pmatrix} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{pmatrix} \quad (5.70)$$

where $\mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1}$ takes the relation between the discrete- and continuous-time measurement covariance matrices (5.4) into account. Δt_{pos} is the sample time of the position measurement. It is

assumed that the three directions of the position measurement are independent and isotropic and thus the measurement covariance matrix is diagonal, $\mathbf{R}_{pos} = \sigma_{pos}^2 \mathbf{I}_3$. The symmetric navigation error state covariance matrix $\tilde{\mathbf{P}}_s$ is block-partitioned as follows

$$\tilde{\mathbf{P}}_s = \begin{pmatrix} \tilde{\mathbf{P}}_{11,s} & \tilde{\mathbf{P}}_{12,s} & \tilde{\mathbf{P}}_{13,s} \\ \tilde{\mathbf{P}}_{21,s} & \tilde{\mathbf{P}}_{22,s} & \tilde{\mathbf{P}}_{23,s} \\ \tilde{\mathbf{P}}_{31,s} & \tilde{\mathbf{P}}_{32,s} & \tilde{\mathbf{P}}_{33,s} \end{pmatrix} \quad (5.71)$$

The matrix \mathbf{M} of variant (1) is hence

$$\mathbf{M} = \mathbf{F} - \tilde{\mathbf{P}}_s \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} = \left(\begin{array}{c|c|c} -\tilde{\mathbf{P}}_{11,s} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & \mathbf{I}_3 & \mathbf{0}_3 \\ \hline \mathbf{\Gamma}_e - \tilde{\mathbf{P}}_{21,s} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & \mathbf{0}_3 & -\text{veck}^{-1}(\tilde{\mathbf{f}}_{\dot{e}}) \\ \hline -\tilde{\mathbf{P}}_{31,s} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & \mathbf{0}_3 & -\mathbf{\Omega}_{ie} \end{array} \right) \quad (5.72)$$

and of variants (2), (3) and (4)

$$\mathbf{M} = \mathbf{F} = \left(\begin{array}{c|c|c} \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \hline \mathbf{\Gamma}_e & \mathbf{0}_3 & -\text{veck}^{-1}(\tilde{\mathbf{f}}_{\dot{e}}) \\ \hline \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{\Omega}_{ie} \end{array} \right) \quad (5.73)$$

The eigenvalues λ_i of both matrices are obtained by solving the characteristic equation

$$\det(\lambda \mathbf{I}_9 - \mathbf{M}) = 0 \quad (5.74)$$

The characteristic equation of variant (1) is given with

$$\det \left(\begin{array}{c|c|c} \lambda \mathbf{I}_3 + \tilde{\mathbf{P}}_{11,s} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & -\mathbf{I}_3 & \mathbf{0}_3 \\ \hline -\mathbf{\Gamma}_e + \tilde{\mathbf{P}}_{21,s} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & \lambda \mathbf{I}_3 & \text{veck}^{-1}(\tilde{\mathbf{f}}_{\dot{e}}) \\ \hline \tilde{\mathbf{P}}_{31,s} \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} & \mathbf{0}_3 & \lambda \mathbf{I}_3 + \mathbf{\Omega}_{ie} \end{array} \right) = 0 \quad (5.75)$$

which becomes after some conversions

$$\det(\lambda \mathbf{I}_3 + \mathbf{\Omega}_{ie}) \cdot \det \left(\lambda^2 \mathbf{I}_3 - \mathbf{\Gamma}_e + \left(\lambda \tilde{\mathbf{P}}_{11,s} + \tilde{\mathbf{P}}_{21,s} - \text{veck}^{-1}(\tilde{\mathbf{f}}_{\dot{e}}) (\lambda \mathbf{I}_3 + \mathbf{\Omega}_{ie})^{-1} \tilde{\mathbf{P}}_{31,s} \right) \mathbf{R}_{pos}^{-1} \Delta t_{pos}^{-1} \right) = 0 \quad (5.76)$$

If the determinants would be calculated, one would obtain a general 6th order polynomial for the eigenvalues of the second determinant, which cannot be solved analytically. Therefore, the eigenvalues have to be found by means of simulation. A statement about the stability is thus postponed to the numerical evaluation in the following. The situation turns out to be much simpler for the variants (2), (3) and (4). Here, the characteristic equation is given with

$$\det \left(\begin{array}{c|c|c} \lambda \mathbf{I}_3 & -\mathbf{I}_3 & \mathbf{0}_3 \\ \hline -\mathbf{\Gamma}_e & \lambda \mathbf{I}_3 & \text{veck}^{-1}(\tilde{\mathbf{f}}_{\dot{e}}) \\ \hline \mathbf{0}_3 & \mathbf{0}_3 & \lambda \mathbf{I}_3 + \mathbf{\Omega}_{ie} \end{array} \right) = \lambda^3 \det(\lambda \mathbf{I}_3 + \mathbf{\Omega}_{ie}) \det(\lambda \mathbf{I}_3 - \lambda^{-1} \mathbf{\Gamma}_e) = 0 \quad (5.77)$$

Solving the characteristic equation yields the eigenvalues of the variants (2), (3) and (4)

$$\lambda = \{-\omega_s i, -\omega_s i, +\omega_s i, +\omega_s i, -\sqrt{2}\omega_s, +\sqrt{2}\omega_s, +\omega_{ie} i, -\omega_{ie} i, 0\} \quad (5.78)$$

where $\omega_s = \sqrt{GM/r^3}$ is the Schuler frequency with the gravity constant GM and the norm of the position vector $r = \|\mathbf{x}_e\|$. It can be observed that seven eigenvalues lie directly on the imaginary axis,

one on the negative real axis and one on the positive real axis. This means that there are one stable, seven asymptotically stable and one unstable eigenvalue. The unstable eigenvalue arises from the unstable height channel. Due to the majority of asymptotically stable and unstable eigenvalues, it becomes obvious that asymmetric covariance errors once captured in the covariance matrix are not damped but are even amplified, which quickly leads to the observed destabilization of the filter.

The found results shall be substantiated by numerical values. Assume the following situation:

- The navigation system is stationary at the position $\phi = 0, \lambda = 0, h = 0$.
- The state error covariance \mathbf{P} is directly initialized with the steady-state solution \mathbf{P}_s .
- The sample rate of the accelerometer and gyroscope measurements is 100 Hz. The measurements are affected by purely white noise with densities of $0.25 \text{ mg}/\sqrt{\text{Hz}}$ (1σ) and $0.3^\circ/\sqrt{\text{h}}$ (1σ), respectively.
- The system is aided by a three dimensional position measurement. The sample rate is 100 Hz. The measurement error is white noise with 3 m standard deviation (1σ). The measurement error covariance matrix is diagonal and isotropic, $\mathbf{R}_{\text{pos}} = \sigma_{\text{pos}}^2 \mathbf{I}_3$.

At first, the eigenvalues of (5.72) and (5.73) are computed. Then, the solutions of the discrete-time covariance Kalman filter equations (5.38), the corresponding continuous-time covariance Riccati equations (5.54) and the continuous-time asymmetric covariance error Riccati equations (5.67) are analyzed and compared. For that, the initial steady-state covariance matrix is superposed with small symmetric and asymmetric error covariances

$$\mathbf{P}_0 = \mathbf{P}_s - \delta\mathbf{P}_s - \delta\mathbf{P}_a \quad (5.79)$$

The initial covariance error is taken as

$$\delta\mathbf{P}_0 = 0.1 \mathbf{L} \quad (5.80)$$

where \mathbf{L} is the lower triangular matrix of \mathbf{P}_s , with the main diagonal also set to zero. The symmetric covariance error matrix is derived as

$$\delta\mathbf{P}_s = \frac{1}{2}(\delta\mathbf{P}_0 + \delta\mathbf{P}_0^T) \quad (5.81)$$

and the asymmetric covariance error as

$$\delta\mathbf{P}_a = \frac{1}{2}(\delta\mathbf{P}_0 - \delta\mathbf{P}_0^T) \quad (5.82)$$

The differential matrix equations are integrated with 8th order Runge-Kutta integration scheme. In the following, variant (1) and variant (3), which is representative for the two other variants (2) and (4), are analyzed. The eigenvalues of variant (1) at the beginning just after initialization are computed as

$$\begin{aligned} \lambda_1 &= -0.1373 & , \lambda_2 &= -0.1373 \\ \lambda_3 &= -0.0750 + 0.1227i & , \lambda_4 &= -0.0750 - 0.1227i \\ \lambda_5 &= -0.0750 + 0.1227i & , \lambda_6 &= -0.0750 - 0.1227i \\ \lambda_7 &= -0.0639 + 0.0639i & , \lambda_8 &= -0.0639 + 0.0639i & , \lambda_9 &= 0 \end{aligned} \quad (5.83)$$

The eigenvalues of variant (3) are

$$\begin{aligned} \lambda_1 &= -0.0018 & , \lambda_2 &= +0.0018 \\ \lambda_3 &= +0.0012i & , \lambda_4 &= -0.0012i \\ \lambda_5 &= +0.0012i & , \lambda_6 &= -0.0012i \\ \lambda_7 &= +0.0001i & , \lambda_8 &= +0.0001i & , \lambda_9 &= 0 \end{aligned} \quad (5.84)$$

The eigenvalues of variant (1) and variant (3) are illustrated in Figure 5-4. It can be seen that all eigenvalues of variant (1) except the zero eigenvalue lie in the left side of the complex plane. The asymmetric error dynamics are stable. The eigenvalues of variant (3), except one are, however, unstable or asymptotically stable, as predicted in (5.78). The asymmetric error dynamics are unstable. In Figure 5-5 the asymmetric error of variant (1) and in Figure 5-6 the asymmetric error of variant (3) are plotted for the first hundred seconds. The blue curves illustrate the asymmetric error of the solution of the continuous-time differential equations (5.54). The red curves represent the asymmetric error of the solution of the discrete-time Kalman filter equations (5.38). The green curves are the solutions of the asymmetric error differential equations (5.66). In Figure 5-5 it can be seen that there is a good coincidence of the blue, red and green curves. This means on the one hand that the initial idea to analyze the continuous-time differential equations instead of the discrete-time Kalman filter equations is justified and on the other hand that the derived differential equation for the asymmetric covariance error represents the actual asymmetric covariance error of the Kalman filter solution well. It can be observed that all initial asymmetries are reduced to values near zero in the initial settlement phase. The solution of the nominal filter is stable. In Figure 5-6 the green curves lie exactly on the blue curves. This means that the continuous-time differential equation and the asymmetric error differential equation yield same results. The discrete-time solution and the continuous-time solution slightly deviate from each other. However, the trends of all curves are equal. It can be seen that the asymmetric error increases and is not damped as in the case of variant (1). The filter is unstable.

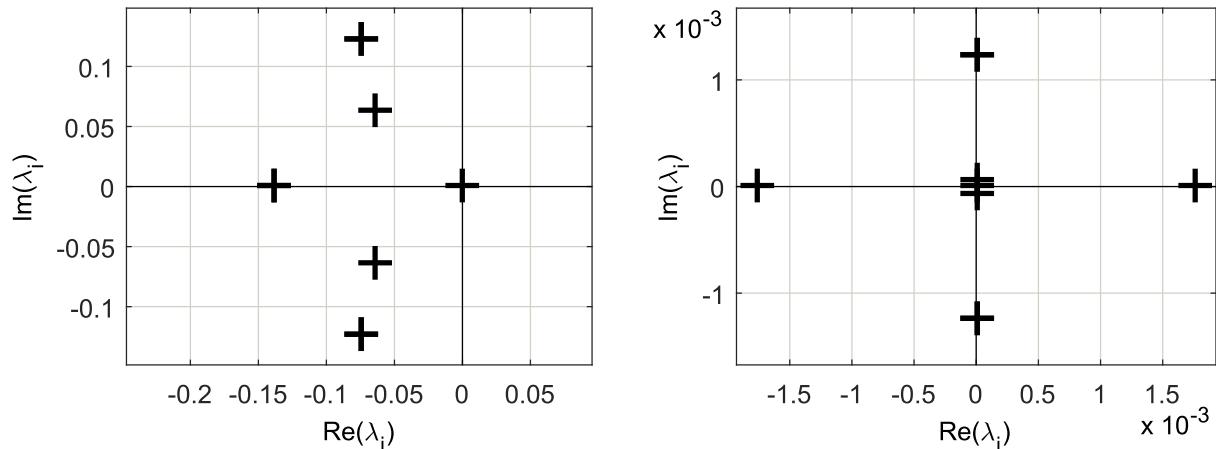


Figure 5-4: Eigenvalues of variant (1) (left) and variant (3) (right)

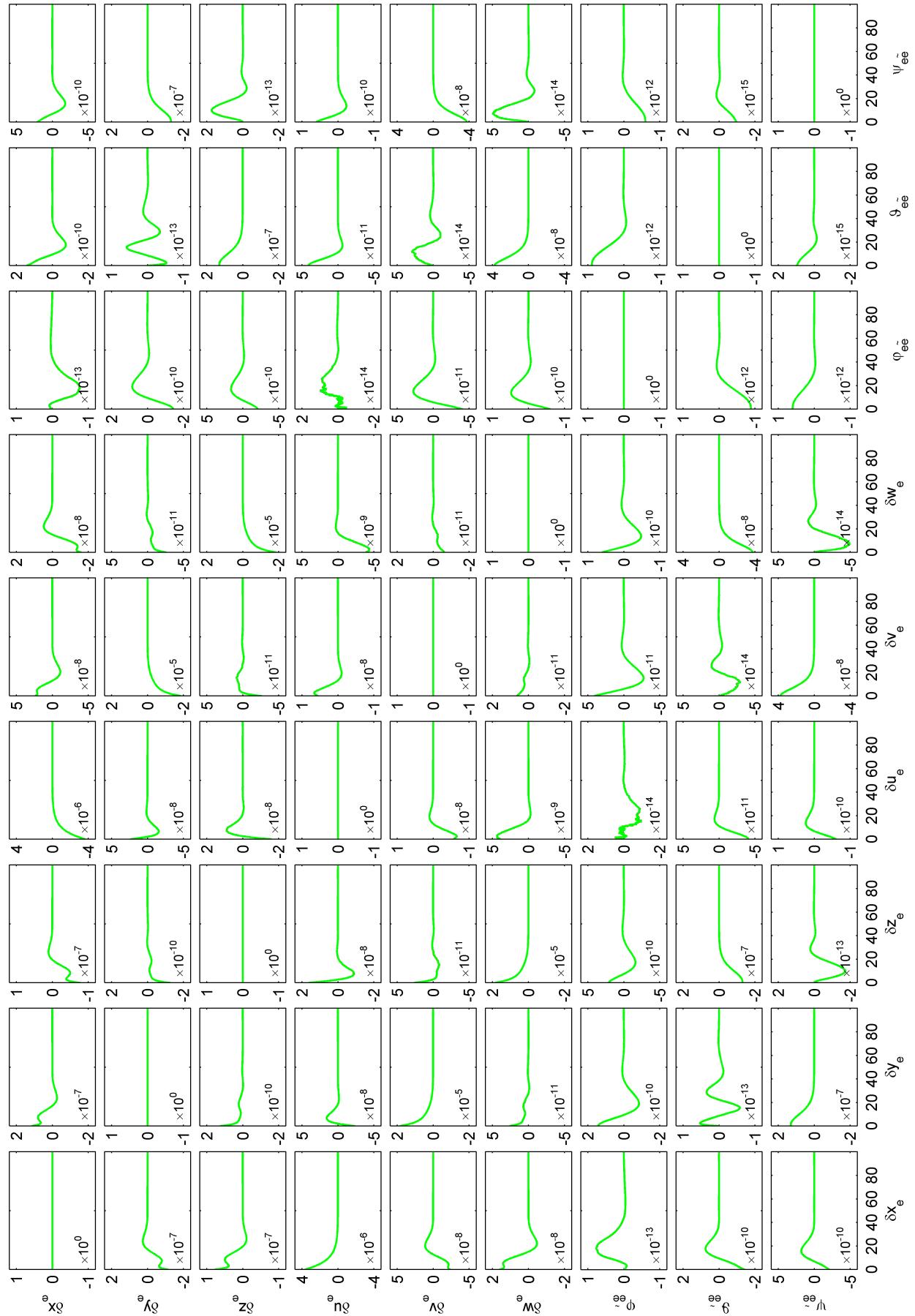


Figure 5-5: Asymmetric covariance error of variant (1)

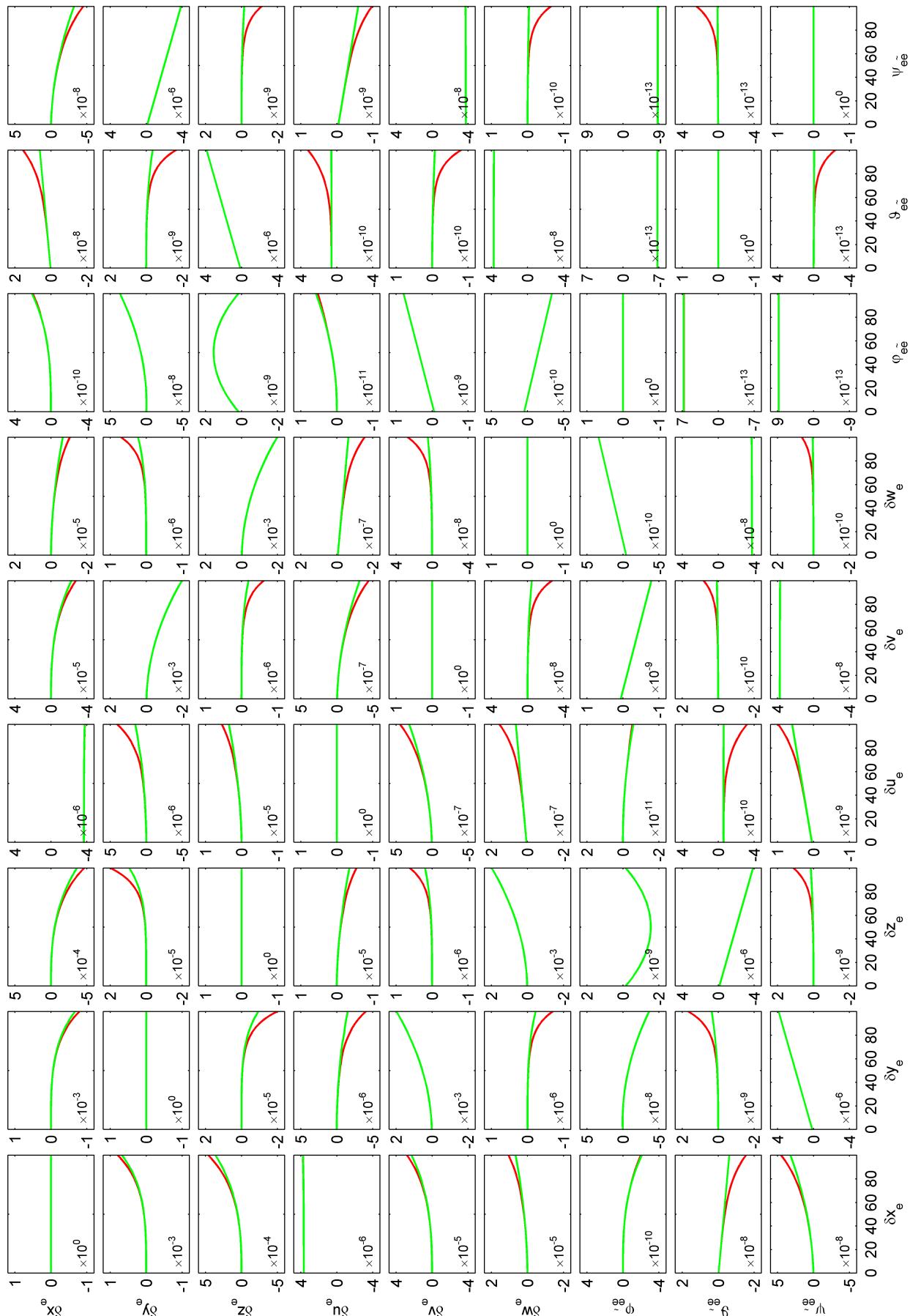


Figure 5-6: Asymmetric covariance error of variant (3)

5.2.6 Stability Enhancement

Measures to enhance the filter stability are given in section D.2 in the appendix.

5.2.7 Out-of-Sequence Measurement Stability Problem

A subtleness concerning the stability of the covariance propagation and update that is inherent to all presented methods for delayed measurements has to be addressed. For the final update step, when the actual measurement value has been arrived, the cross-covariance matrix \mathbf{P}_{kj}^- is required in the equations for the Kalman gain and the update of the state error covariance matrix, as derived in (4.125) and (4.127),

$$\begin{aligned}\mathbf{K}_k &= \mathbf{P}_{kj}^- \mathbf{H}_j^\top (\mathbf{R}_j + \mathbf{H}_j \mathbf{P}_{j,k-1} \mathbf{H}_j^\top)^{-1} \\ \mathbf{P}_k^+ &= \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^-\end{aligned}\quad (5.85)$$

Let us consider a simple example to illustrate the stability problem: assume that a navigation filter processes only aiding measurements of a single sensor, for example a GNSS receiver. The measurement is taken at time t_{k-1} and delivered to the navigation filter at time t_k . There are no further interim updates of other sensors. The situation is depicted in Figure 5-7. The covariance at the instant of the measurement is \mathbf{P}_{k-1} . The covariance propagation from time t_{k-1} to the current time t_k is accomplished by a single propagation step

$$\begin{aligned}\mathbf{P}_k^- &= \Phi_{k-1} \mathbf{P}_{k-1} \Phi_{k-1}^\top + \Gamma_{k-1} \mathbf{Q}_{k-1} \Gamma_{k-1}^\top \\ \mathbf{P}_{k(k-1)}^- &= \Phi_{k-1} \mathbf{P}_{k-1}\end{aligned}\quad (5.86)$$

The Kalman gain and the covariance update at time t_k are

$$\begin{aligned}\mathbf{K}_k &= \mathbf{P}_{k(k-1)}^- \mathbf{H}_{k-1}^\top (\mathbf{R}_{k-1} + \mathbf{H}_{k-1} \mathbf{P}_{k-1} \mathbf{H}_{k-1}^\top)^{-1} \\ \mathbf{P}_k^+ &= \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_{k-1} \mathbf{P}_{(k-1)k}^-\end{aligned}\quad (5.87)$$

Following the procedure in section 5.2.3, the covariance update equation is converted from discrete-time to continuous-time in order to make a statement about the stability of the filter for delayed measurements. The combined covariance propagation and update equation is

$$\mathbf{P}_k^+ = \Phi_{k-1} \mathbf{P}_{k-1} \Phi_{k-1}^\top + \Gamma_{k-1} \mathbf{Q}_{k-1} \Gamma_{k-1}^\top - \mathbf{K}_k \mathbf{H}_{k-1} \mathbf{P}_{k-1}^\top \Phi_{k-1}^\top \quad (5.88)$$

With the discretized system and input matrices $\Phi = \mathbf{I}_n + \mathbf{F} \Delta t + O(\Delta t^2)$, $\Gamma = \mathbf{G} \Delta t + O(\Delta t^2)$ and $\mathbf{Q}_k = \mathbf{Q}/\Delta t$ one gets

$$\begin{aligned}\mathbf{P}_k^+ &= \mathbf{P}_{k-1} - \mathbf{K}_k \mathbf{H}_{k-1} \mathbf{P}_{k-1}^\top - \mathbf{K}_k \mathbf{H}_{k-1} \mathbf{P}_{k-1}^\top \mathbf{F}_{k-1}^\top \Delta t \\ &\quad + \mathbf{F}_{k-1} \mathbf{P}_{k-1} \Delta t + \mathbf{P}_{k-1} \mathbf{F}_{k-1}^\top \Delta t + \mathbf{G}_{k-1} \mathbf{Q} \mathbf{G}_{k-1}^\top \Delta t + O(\Delta t^2)\end{aligned}\quad (5.89)$$

The difference quotient is formed

$$\frac{\mathbf{P}_k^+ - \mathbf{P}_{k-1}}{\Delta t} = \frac{-\mathbf{K}_k \mathbf{H}_{k-1} \mathbf{P}_{k-1}^\top}{\Delta t} - \mathbf{K}_k \mathbf{H}_{k-1} \mathbf{P}_{k-1}^\top \mathbf{F}_{k-1}^\top + \mathbf{F}_{k-1} \mathbf{P}_{k-1} + \mathbf{P}_{k-1} \mathbf{F}_{k-1}^\top + \mathbf{G}_{k-1} \mathbf{Q} \mathbf{G}_{k-1}^\top + O(\Delta t) \quad (5.90)$$

With $\mathbf{R}_k = \mathbf{R}/\Delta t$ and the series expansion of the system matrix the Kalman gain becomes

$$\begin{aligned}\mathbf{K}_k &= \Phi_{k-1} \mathbf{P}_{k-1} \mathbf{H}_{k-1}^\top (\mathbf{R} + \mathbf{H}_{k-1} \mathbf{P}_{k-1} \mathbf{H}_{k-1}^\top \Delta t)^{-1} \Delta t \\ &= \mathbf{P}_{k-1} \mathbf{H}_{k-1}^\top (\mathbf{R} + \mathbf{H}_{k-1} \mathbf{P}_{k-1} \mathbf{H}_{k-1}^\top \Delta t)^{-1} \Delta t + O(\Delta t^2)\end{aligned}\quad (5.91)$$

Inserting the Kalman gain into the difference quotient and reducing the time step to infinitesimal size yields the differential quotient

$$\lim_{\Delta t \rightarrow 0} \frac{\mathbf{P}_k^+ - \mathbf{P}_{k-1}}{\Delta t} = \frac{-\mathbf{P}_{k-1} \mathbf{H}_{k-1}^\top \mathbf{R}^{-1} \mathbf{H}_{k-1} \mathbf{P}_{k-1}^\top}{\Delta t} + \mathbf{F}_{k-1} \mathbf{P}_{k-1} + \mathbf{P}_{k-1} \mathbf{F}_{k-1}^\top + \mathbf{G}_{k-1} \mathbf{Q} \mathbf{G}_{k-1}^\top \quad (5.92)$$

which is a Riccati like differential equation but with transposed covariance in the quadratic term

$$\dot{\mathbf{P}}(t) = -\mathbf{P}(t) \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \mathbf{P}(t) + \mathbf{F} \mathbf{P}(t) + \mathbf{P}(t) \mathbf{F}^\top + \mathbf{G} \mathbf{Q} \mathbf{G}^\top \quad (5.93)$$

This continuous-time covariance propagation equation exactly corresponds to variant (3). In the stability analysis it turned out that variant (3) is unstable. Therefore, it is obligatory to use symmetry conserving Joseph's form for the final covariance update or another stabilizing measure like the numerical symmetrization of the covariance matrix from time to time. Otherwise, the filter runs the risk to diverge quickly.

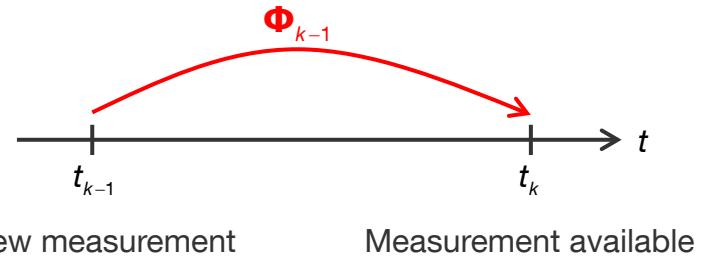


Figure 5-7: Single delayed measurement without interim update

5.2.8 Example

The General Aviation Aircraft scenario (section E.1.2) is used as example to demonstrate the destabilization effect on the navigation state estimation error. IMU 2 (Table E-5) with 100 Hz sample rate is selected. The IMU measurements are solely affected by velocity/angular random walk noise N and turn-on biases. All other IMU errors are set to zero. A generic GPS receiver provides 1 Hz pseudorange measurements to six satellites in view, which are only falsified by white Gaussian noise with 2 m standard deviation.

The position errors of variants (2), (3) and (4) are shown in Figure 5-8, Figure 5-9 and Figure 5-10. It is easily observable that variant (2) becomes unstable at second 310, variant(3) at second 322 and variant (4) at second 985.

5.2.9 Conclusion

The covariance propagation und update becomes unstable if the positive definiteness of the state error covariance is violated. This can happen if numerical errors are not adequately counteracted. It has been shown that if the transposed covariance matrix is used in the covariance update equations instead of the covariance matrix itself, which should make no difference from the mathematical point of view due to symmetry, the Kalman filter may get unstable. If measurement prediction or history state update methods are applied, the covariance propagation and update equations are innately unstable (variant(3)). It can be concluded that stabilization is compulsory in any case, even if the nominal variant (1) is implemented.

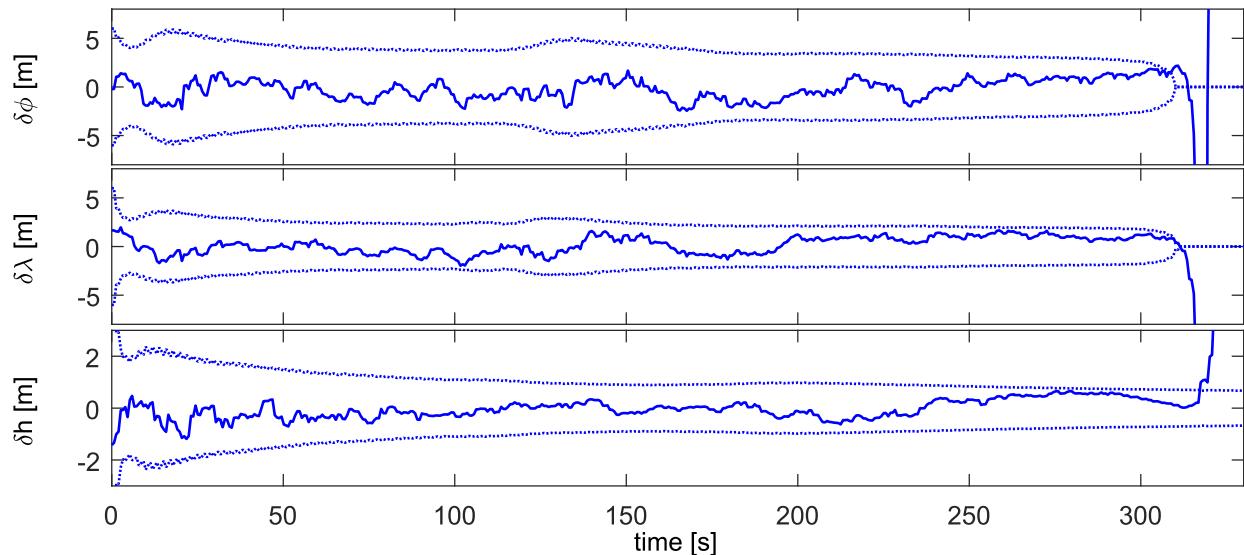


Figure 5-8: Position error and predicted 3σ boundaries with variant (2)

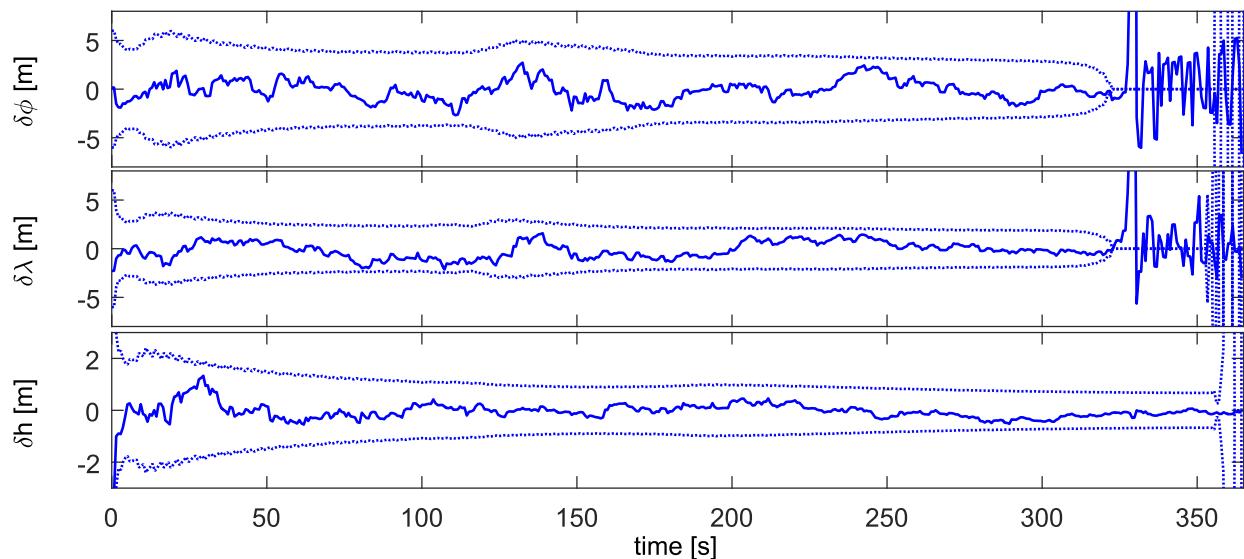


Figure 5-9: Position error and predicted 3σ boundaries (dotted lines) with variant (3)

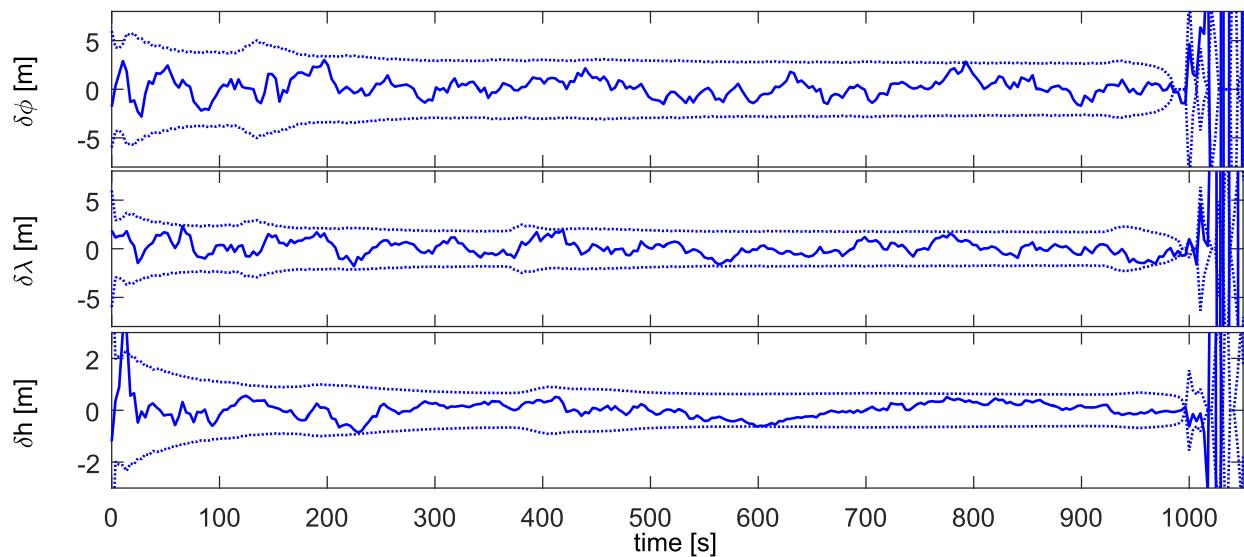


Figure 5-10: Position error and predicted 3σ boundaries (dotted lines) with variant (4)

6 Summary

6.1 Detailed Summary of the Findings

Reality modeling

In order to increase the statistical consistency of the navigation solution and to simplify the tuning of the navigation filter, which is often understood as more or less heuristic process requiring some expert experience, physically motivated models that represent the magnitudes and temporal correlations of the sensor measurement and modeling errors correctly have been derived. ARMA models revealed to be adequate for all navigation errors throughout the thesis and provide a versatile way to describe the errors in a common context. The models are suitable for measurement synthesis in the simulation and can likewise be integrated into the navigation filter in a streamlined version. It has been identified which errors primarily affect the navigation solution and should be accounted for by the navigation filter. Furthermore, it has been analyzed if the character of the single error components is bias-like or rather noise-like and whether it is worthwhile to be estimated by the navigation filter or rather only to be considered in a statistical sense. It has been discussed what is the minimum number of additionally required states to depict the main share of the error but to minimize the extra numerical effort. The following errors have been regarded:

- IMU errors

The derived accelerometer and gyroscope error models represent turn-on biases, scale factor errors, bias and scale factor temperature sensitivities, inter-axis misalignments, velocity/angular random walk, bias instabilities, quantization noise and acceleration/angular rate random walk. The gyroscope error model additionally accounts for the linear acceleration cross-coupling and triad misorientations. The turn-on bias and scale factor errors are constant with time and can thus be ideally estimated by the navigation filter and used for in-flight calibration. They are easily integrated into the filter as additional scalar states for each axis. The constant misalignment errors, acceleration cross-coupling on the angular rates and misorientation angles of the gyroscope triad are generally not estimated but have to be considered in any case as supplemental scalar states due to their gross influence on the navigation state error. If temperature sensors are available, the constant temperature sensitivities can also be included as considered states in the same manner. Bias instabilities only have to be considered if the grade of the IMU is low. A first order model for each axis is already sufficient to cover the flicker noise character. For measurement synthesis purposes for simulation it is reasonable to use bias instability models up to fourth order. It has been demonstrated how the bias instability and velocity/angular random walk parameters are estimated from Allan deviations of recorded measurements of a stationary IMU. The influence of all other colored noise components like quantization noise is comparatively small and does not have to be considered in general. In future, the error models should be expanded by non-linearities including vibration rectification.

- Gravity model error

By means of an exemplary flight across the Alps, it has been shown that the gravity compensation error is in the range of a tenth of a milli-g with correlation lengths of some tens of kilometers if a simple normal gravity model like Somigliana is used. This residual gravity error results in larger position deviations if the navigation system runs open loop without position aiding for longer times. Besides the usage of a higher order EGM2008 model the system designer can apply a stochastic model approach. The latter has been pursued by fitting a gravity disturbance model from literature to the framework of the integrated navigation system. Model parameters have been exemplarily estimated for the Alps region from the EGM2008 model with degree and order 2159. It has been revealed that the prediction of the navigation solution accuracy can be improved if the five additional states of the stochastic gravity disturbance model are considered in the navigation filter. However, this is only necessary if high accuracy is required and longer GNSS outage periods are expected, or if the navigation system is purely inertial without any GNSS aiding. Since the spatial correlation length of the stochastic process depends on the roughness of the terrain, different model parameters for the regions where the system shall be operated have to be estimated and stored. The model takes the diminishing influence of the gravity residual error with height into account.

- Random vibration

A convenient method for applications with strongly and unpredictably changing vibrational noise regimes like for example missiles with different sub-/supersonic flight phases with/without propulsion and with/without atmospheric disturbances has been presented. It has been described how the random vibration model is adapted online according to the current noise floor. The vibrational noise is represented by separate AR models per translational/rotational axis whose coefficients are estimated online by means of Burg's method from an adjustable sliding window of half a second length, for example. Random vibrational noise is characterized by short correlation lengths and negative correlation. It has been revealed that low-order AR models (between third and seventh order) already yield satisfying results and that it is advisable to consider the time correlation of the noise in the navigation filter. The method has been verified by means of exemplary simulations using typical noise power spectra given in MIL-STD-810G. Moreover, it is advantageous that other process noise errors with short correlation times like the velocity random walk noise of the accelerometer are also automatically covered and hence do not have to be separately modeled. The online adaption of the model is not required if the vibrational regime is well known and does not change noticeably during operation. In this case, the model parameters can be tuned offline by means of records from flight experiments.

- Satellite clock ephemeris error

The comparison of 2012's broadcast and precise GPS clock errors has illustrated that the residual clock error is bias-like with correlation times in the range of hours and magnitudes between 0.5 and 4 meters and increases between subsequent upload cutovers, which occur roughly once per day. Smaller changes are experienced at dataset cutovers, taking place every two hours. These bias-like pseudorange errors cause constant position offsets, especially if only few satellites are used or in view. Unsteady second order MIMO models have been proposed to describe the evolution of the correction coefficient errors δa_{f_0} and δa_{f_1} between upload cutovers. The satellite clock error depends on the satellite block type (IIA, IIR-A, IIR-B, IIR-M, IIF) and the activated clock (RB, CS). Another model has been derived for the short-term noise-like part of the residual clock error,

which is in the range of a decimeter. Model parameters have been estimated from the GPS clock errors in 2012. Simulation results have shown that the predicted position error covariance fits much better to the actual position error if the residual clock error models are considered in the navigation filter. Unfortunately, the current assignment of satellite and clock type to PRN is not contained in the broadcast navigation message and has to be uploaded into the navigation system by the user.

- Satellite position ephemeris error

The ephemeris orbit error is smaller than the satellite clock error. The radial error component, which primarily influences the pseudorange error, is in the range of a few decimeters. The error components feature twelve and two hour period oscillations. A model using the same structure as the second harmonic perturbations of GPS has been derived and the model parameters have been estimated form the orbit errors of the year 2012. In most cases, the additional expenses of the model are however disproportionate to the gain of accuracy.

- Ionospheric delay

After eliminating the slant effect, it has been observed that the ionospheric delay is bias-like with comparably long correlation lengths of half an hour or longer. Since the ionospheric intersection points are located far apart, the vertical ionospheric delays are different for each satellite in view but nevertheless strongly correlated. The ionospheric delays of the Klobuchar model with broadcast parameters, IGS TEC maps and the NeQuick2 model have been compared. Taken the TEC maps as reference, it could be seen that the NeQuick2 model yields more realistic results than the Klobuchar model. The NeQuick2 model is particularly suited for space launch applications like sounding rockets because it considers the exact signal path through the ionosphere and does not replace the ionosphere by a thin substitute shell. Since the NeQuick2 model depends only on one scalar input parameter, that is the mean sunspot number R_{12} , a very slowly changing value, the model is perfectly suited for the integration into the navigation filter. The unknown input parameter can be modeled as constant and estimated by the integrated navigation filter. In the simulation the estimated sunspot number slowly converged against a steady-state value and the predicted navigation error covariance coincided with the covariance of the actually realized errors. Since the residual ionospheric delay as well as the receiver clock bias primarily have an effect in height direction, both errors and the height error are strongly correlated amongst each other in the beginning. This correlation decreases only slowly with time due to the change of the satellite constellation. Galileo uses the NeQuickG model and broadcasts parameters within the navigation message for the calculation of the latitude dependent model input parameter. In this case, the ionospheric delay correction can be done completely outside of the navigation filter.

- Tropospheric delay

It is common practice to use a combination of zenith delay model like Saastamoinen's and mapping function like Niell's for the slant effect to compensate the hydrostatic and wet parts of the tropospheric delay. The models remove the largest share of the seasonally varying zenith delay. The residual tropospheric error in the range of one decimeter and correlation time of about half an hour can be represented by a stochastic model. A discrete-time state-space model has been derived from the well-known structure functions of small and large scale turbulent fluctuations mainly in clouds, basically using Kolmogorov's turbulence theory. Because of the small magnitude, the modeling effort is only recommended for applications requiring highest accuracy. If the conventional troposphere models are applied, the residual tropospheric error can be neglected in most applications.

- Receiver clock bias and drift

It has been seen that especially the receiver clock bias and the height error as well as the receiver clock drift and the downward directed velocity error are negatively correlated. This means that only the linear combination of these errors can be observed by the navigation filter and the navigation filter struggles to clearly distinguish between these errors. This in turn implies that if the receiver clock error uncertainties are simply represented by a large, over-bounded diagonal process noise covariance matrix, the predicted height error standard deviation is expected to be large, too, which might be problematic for flight guidance. Therefore, a more realistic receiver clock error model is necessary to ease the distinction between both errors. A two-state oscillator error model regarding the white frequency, flicker frequency and random walk frequency noise of the receiver oscillator has been derived. Model parameters have been exemplarily estimated from the PSDs given in the datasheets of two typical TCXO that are used in current GNSS receiver designs.

- Receiver noise

It is new that the actual tracking loop structure is involved in the calculation of the pseudorange and range rate measurement noise covariances. Experience has shown that the time correlation of the pseudorange and range rate measurements depends on the tracking loop filter order and the chosen bandwidths. Furthermore, the chosen discriminators (for example fixed or variable gain) play a decisive role. A tracking loop model of a third order FLL assisted PLL and a first order carrier aided DLL has been derived, which outputs the current pseudorange and range rate variances and needs the current carrier-to-noise ratio estimate of the receiver as input. The consideration of the actual tracking loop structure outperforms the common standard deviation assessment formulas, requires however that the user has insight into the tracking filter design. The time correlation of the raw measurements is the higher, the larger the bandwidths. Generally, if the update rate is 1 Hz, the correlation has not be considered in the integrated navigation filter. If higher update rates larger than 10 Hz are used, correlation may be present and should be considered in the navigation filter. It is a misleading assumption that higher GNSS pseudorange and range update rates provide higher accuracy because if the correlation is not considered, the resulting position error might be larger than with a lower update rate where the error is not correlated and correlation does not have to be accounted for by the integrated navigation filter.

The used identification methods are however general and can also be applied to other sensors or model errors.

Table 6-1 repeats again the derived reality models and the main facts. The number of additionally required states in the simulation on the one hand and in the filter implementation on the other hand are given in the ‘Number states’ column. The errors with ‘-’ entries are not generated by ARMA models in the simulation but synthesized otherwise. The satellite clock and orbit errors for example result because precise ephemeris are applied in the simulation and broadcast ephemeris are used by the GPS receiver. The estimate (E) / consider (C) flag gives a recommendation whether the error should be estimated by the navigation filter or rather considered only as colored process or measurement noise. In the last column the proposed covariance propagation interval of the error models are given.

Table 6-1: Overview over derived reality models

Measurement / Modeling Error	Magnitude	Bias / Noise -like (B/N)	Corr. Length	Model	Number states		(Cross-) Cov. Prop. Interval	Remark
					Sim.	Filter		
Turn-on bias	~0.025 – 3 mg ~0.003 – 10 °/h	B	∞		-	(2x) 3	E	0.1 s
Bias instability	~10 µg ~0.002 – 0.1 °/h	N	~ 600 s		4	(6x) 1	C	0.1 s
IMU error	~0.008 – 0.25 mg/√Hz ~0.0025 – 0.3 °/√h	N	0 s	Table 3.7	-	0	C	0.1 s
Velocity/Angle random walk					-	(2x) 6	C	0.1 s
Misalignment	0.025 – 0.5 mrad	B	∞					
Scale factor error	5 – 1000 ppm	B	∞		-	(2x) 3	E/C	0.1 s
Gravity model error	0.35 mg (Alps) 0.08 mg (U.S.)	N	68 km (Alps) 90 km (U.S.)	(3.88) (3.89)	-	5	C	0.1 s
Random vibration	variable	N	< 0.1 s	(3.97)	7	(6x) 3	C	0.1 s
Satellite clock error	~ 1 – 4 m	B	< 30 h	Dataset cutover (3.111) Residual noise (3.113) (3.114)	-	(m×) 2	C	2 h
Radial satellite orbit error	~ 0.5 m	B	< 50 h	Coeff. a_{12}, a_2 (3.130) Coeff. b_{12}, b_2 (3.131) Coeff. c_{12}, c_2 (3.132)	(m×) 2 × 2 (m×) 2 × 2 (m×) 2 × 2	(m×) 2	C	30 s
Ionoosphere error	< 15 m	B	~ 12 h	(3.146)	-	1	E	0.1 s
Troposphere error	~ 0.1 m	B	~ 1 h	Short distance (3.167) Large distance (3.171)	5	2	C	0.1 s
GPS observable errors				Initial bias / drift (3.181) WF noise FF noise RWF noise	2	2	E	0.1 s
Receiver clock error	< 1 m/s	B	∞	Table 3.7	0	0	C	0.1 s
Receiver noise	~ 0.1 – 1 m	N	< 1 s	(3.274)	(m×) 5	(m×) 5	C	0.005 – 0.001 s

Efficiently increasing navigation filter consistency

- **Inertial navigation algorithms**

In general, it is assumed that the truncation error of the inertial navigation algorithm is adequately small and much smaller than all other errors. In high dynamic situations or under unfavorable coning and sculling motion, the numerical error can however have a large influence. A good understanding of the error characteristic of the chosen algorithm is essential. For that reason, three classes of inertial navigation algorithms with different truncation orders have been thoroughly investigated with respect to their numerical error propagation, robustness against noise and robustness against vibration. These are the conversion-integration-extrapolation algorithms, the newly derived single frequency algorithm and the well-known dual frequency algorithms for orientation. Generally, the higher the measurement noise, the less significant is the truncation order of the algorithm. Central conversion schemes of integrating IMU measurements in the CIE algorithms are more robust against noise and vibration than non-central integration schemes. The innovative single frequency approach turned out to be accurate and robust against noise and vibration. The dual frequency algorithms revealed even higher accuracy and robustness compared to the single frequency approach. In contrast to the single frequency approach the dual frequency algorithms provide only an output value every third or fourth input measurement value. Therefore, a combination of the dual frequency orientation algorithms, augmented by interim orientation extrapolation for example with the single frequency 1st order CIE, together with the position and velocity propagation equations of the single frequency approach would be a good possibility.

- **Navigation filter initialization**

The question has been answered how the orientation angles and the corresponding orientation error covariance are to be initialized in a statistically consistent manner either stationary before take-off or in flight. The focus was amongst others on the influence of the correlation between the initial orientation estimate and the IMU errors on the covariance prediction of the integrated navigation filter in the initial settlement period. If they are considered, predicted filter statistics are coherent from the beginning of the filter runtime. If they are neglected and the filter is initialized with a diagonal and over-bounded orientation error covariance matrix, it depends on the maneuvers how fast the filter settles to the correct covariances. It is therefore recommended to correctly initialize the orientation error covariance and the cross-covariances between orientation errors and accelerometer and gyroscope errors. The integrated acceleration measurement enables the initialization of orientation during highly accelerated flight sections. Simulations showed good performance with respect to accuracy and statistical consistency.

- **Out-of-sequence measurements**

Since the issue of measurement synchronization is central in an integrated navigation filter, which has to process measurements with different validity and delivery times, a Kalman filter for delayed measurements has been derived, which also copes with out-of-sequence arrivals of measurements. Three common techniques from literature, that is the state backward propagation, measurement prediction and history state update methods, have been compared and analyzed for their suitability in an integrated navigation filter. It has been found that the history state update method is most versatile since it can be easily embedded into the integrated navigation filter framework and is able to process delayed as well as out-of-sequence aiding measurements correctly from a statistical point of view. However, it has been seen that the additional effort is only necessary if the covariance changes considerably in the time between the actual measurement and the provision of the

measurement value. That might be the case if the measurement delay is long, or if the process noise input is large due to a low grade IMU, or in the initialization period, when the filter has not yet been settled and covariances still change a lot. If the IMU is high grade or the measurement delays are short, the innovation of each new aiding measurement is comparably small and the effect of the statistically correct update of the covariance matrix less significant. In this case, the history state update functionality can be switched off and it is sufficient to predict the measurement from the validity time to the current time with the current navigation state estimate and to use the current covariance estimate for the calculation of the innovation. It has been shown that the Kalman filter for delayed measurements is potentially unstable and measures to increase stability have to be taken.

- Navigation filter state augmentation and avoidance of state augmentation for noise-like errors

In order to make use of the derived error models and to improve filter performance with respect to statistical coherence, the integration of these error models into the navigation filter has been thoroughly analyzed. Bias-like errors with long correlation times can be estimated by the filter and for example used for in-flight calibration of sensor measurements. In this case, the core inertial navigation error model is simply augmented by the corresponding error models. For the noise-like errors with short correlation times, which are not estimated but only statistically considered, the well-known Schmidt-Kalman filter has been prepared for the usage in the context of integrated navigation. Templates are given for Schmidt-Kalman filters for process and measurement noise models. It is shown how more than one colored noise model in parallel is to be treated. The discussion is completed with a Schmidt-Kalman filter for out-of-sequence measurements.

Navigation filter performance prediction and stability

- Steady-state Kalman filter

Analytical formulas for the quick calculation of the expected steady-state navigation accuracy have been derived. They can be applied in online integrity monitoring algorithms to assess the coherence of the predicted navigation error covariance or in mission planning tools where a large number of different scenarios and trajectories shall be quickly analyzed. The given formulas require much less computations than the iterative solution of the Kalman filter covariance equations. The formulas have been derived by applying a general solution method for DAREs to the integrated navigation problem.

- Kalman filter stability

It is a well-known fact that the Kalman filter based navigation filter can become unstable due to numerical errors violating the positive definiteness of the state error covariance matrix. It has been thoroughly analyzed which conditions favor the destabilization and the thesis gives a mathematically substantiated answer. In the course of this analysis it has been discovered that the Kalman filter equations for delayed measurements of the measurement prediction and history state update methods are inherently unstable. Additional stabilization enhancement measures or the use of Joseph's form for the update are compulsory in this case. The analysis gives valuable hints for the robust implementation of the integrated navigation filter on an embedded system.

6.2 Significant Achievements of the Thesis beyond State-of-the-Art

Reality modeling

One of the key aspects of the reality modeling is the uniform and consistent description of the sensor and modeling errors of integrated navigation systems in a common mathematical context likewise suitable for simulation and real-time algorithms. The commonality and comprehensiveness of the presented methods for the analysis of the sensor and modeling data, identification of the model structure and estimation of the model parameters are the main achievement of the reality modeling chapter and cannot be found in the respective literature in this form to the best of the author's knowledge.

Moreover, the following models shall be highlighted once again since they are novel and have not been published to date in this way:

- The process noise model which is adapted in real-time to represent the current magnitude and time correlation of the process noise for example due to varying random vibration
- The satellite clock and orbit error models that consider amongst others the upload and dataset cutovers of the broadcast navigation messages and the revolution period of the satellites
- The NeQuick2 model for ionosphere compensation with the estimation of the scalar sunspot number parameter by the navigation filter
- The receiver clock model that considers the white frequency, flicker frequency and random walk frequency noise of the local oscillator
- The pseudorange noise model that does not use the well-known assessment formulas but the actual tracking loop structure and discriminators since these have amongst others a large influence on the magnitude and correlation of the noise

Efficiently increasing navigation filter consistency

- Inertial navigation algorithms

The single frequency inertial navigation algorithm for integrating IMU is novel. Its performance can keep up with the conventional dual frequency algorithms, which do not output a new state estimate for each new IMU sample and provide only orientation. The comprehensive comparison of the CIE algorithms, the single frequency approach and dual frequency algorithms is helpful for system designers.

- Navigation filter initialization

In literature, it is most often proposed to initialize the navigation error covariance with a diagonal, over-bounded covariance matrix and to let it settle in the initial settlement phase by means of the filter. The statistically correct initialization, which regards correlations, and the analysis of the effect of the diagonal simplification however provide a new contribution to the effort to increase consistency. Furthermore, the in-flight orientation angle estimation by means of the TRIAD method and the integrated acceleration observation is a novel approach that has not been published up to date.

- Out-of-sequence measurements

The three presented methods are in fact individually well described in literature. It is new to present them in the same mathematical context and to reveal the commonalities and differences. The templates help the system designer to implement the adapted filter algorithms even for a Schmidt-

Kalman filter with out-of-sequence-measurements. Moreover, to the author's best knowledge, the instability issue of the Kalman filter for delayed measurements has not been addressed in the respective literature so far. Furthermore, the judgement of the influence of delays and out-of-sequence arrivals is helpful.

- Navigation filter state augmentation and avoidance of state augmentation for noise-like errors

The Kalman filter and Schmidt-Kalman filter are well-known and widely used estimation tools. The thesis describes in detail the application of the general Schmidt-Kalman filter equations to the integrated navigation problem and derives the formulas for multiple simultaneous process and measurement noises to be statistically considered but not estimated. The focus is on the implementation in an embedded system, which is emphasized by the comprehensive templates. Interesting side aspects like the distinction of covariance propagation equations with and without preceding updates complete the discussion.

Navigation filter performance prediction and stability

- Steady-state Kalman filter

The analytical non-iterative formulas for the fast assessment of the steady-state navigation error covariance in quasi-stationary flight phases could not be found in literature during the research, but may be very useful for the quick computation of the expected steady-state solution of many different scenarios for example in mission planning tools.

- Kalman filter stability

Finally, the destabilization mechanisms, which are invoked if transposed covariance matrices are used in the Kalman filter equations, have not been analyzed so far in this manner to the author's best knowledge. Very interesting is the connection between this analysis and the Kalman filter equations for delayed measurements.

Bibliography

- [1] Federal Aviation Administration (FAA), "Airborne Navigation Sensors Using The Global Positioning System Augmented By The Satellite Based Augmentation System," TSO-C145d, 2013.
- [2] Federal Aviation Administration (FAA), "Stand-Alone Airborne Navigation Equipment Using The Global Positioning System Augmented By The Satellite Based Augmentation System," TSO-C146d, 2013.
- [3] Radio Technical Commission for Aeronautics (RTCA), "Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment," RTCA DO-229D, 2006.
- [4] Federal Aviation Administration (FAA), "Attitude and Heading Reference Systems (AHRS)," TSO-C201, 2012.
- [5] Radio Technical Commission for Aeronautics (RTCA), "Minimum Operational Performance Standards (MOPS) for Strapdown Attitude and Heading Reference System (AHRS)," RTCA DO-334, 2012.
- [6] Federal Aviation Administration (FAA), "Airborne Supplemental Navigation Sensors for Global Positioning System Equipment Using Aircraft-Based Augmentation," TSO-C196b, 2013.
- [7] Radio Technical Commission for Aeronautics (RTCA), "Minimum Operational Performance Standards for Global Positioning System / Aircraft Based Augmentation System Airborne Equipment," RTCA DO-316, 2009.
- [8] D. H. Titterton and J. L. Weston, Strapdown Inertial Navigation Technology, 2nd ed., The Institution of Electrical Engineers, 2004.
- [9] K. R. Britting, Inertial Navigation Systems Analysis, New York: John Wiley & Sons, 1971.
- [10] A. B. Chatfield, Fundamentals of High Accuracy Inertial Navigation, American Institute of Aeronautics & Astronautics (AIAA), 1997.
- [11] C. Jekeli, Inertial Navigation Systems with Geodetic Applications, Berlin: Walter de Gruyter, 2000.
- [12] A. Dachs, „Rechenzeitoptimierung, Robustifizierung und Tuning eines Kalmanfilters zur Datenfusion für Navigationsanwendungen,“ Dissertation, Universität der Bundeswehr München, Neubiberg, 2010.
- [13] D. Simon, Optimal State Estimation: Kalman, H^∞ , and Nonlinear Approaches, John Wiley & Sons, 2006.
- [14] W. J. Riley, "Handbook of Frequency Stability Analysis," National Institute of Standards and Technology, 2008.
- [15] IEEE Standards Coordinating Committee 27, "IEEE Standard Definitions of Physical Quantities for Fundamental Frequency and Time Metrology - Random Instabilities," IEEE Std 1139-2008, 2009.
- [16] J. S. Bendat and A. G. Piersol, Random Data: Analysis and Measurement Procedures, 3 ed., New York: John Wiley & Sons, 2000.
- [17] S. L. Marple, Digital Spectral Analysis with Applications, Englewood Cliffs: Prentice Hall PTR, 1987.
- [18] M. H. Hayes, Statistical Digital Signal Processing and Modeling, New York: John Wiley & Sons, Inc., 1996.
- [19] Y. Ferdi, "Efficient Generation of 1/f Noise Using Signal Modeling Techniques," *IEEE Transactions on Circuits and Systems*, vol. 55, no. 6, pp. 1704-1710, 2008.
- [20] T. W. Parks and C. S. Burrus, Digital Filter Design, New York: John Wiley & Sons, 1987, pp. 226-228.
- [21] K. Steiglitz and L. E. McBride, "A Technique for the Identification of Linear Systems," *IRE Transactions on Automatic Control*, vol. 6, no. 2, pp. 183-199, 1965.
- [22] M. A. Al-Aloui, "Novel Digital Integrator and Differentiator," *Electronics Letters*, vol. 29, no. 4, pp. 376-378, 1993.
- [23] P. Stoica and R. Moses, Spectral Analysis of Signals, Englewood Cliffs: Prentice Hall, 2005.
- [24] Institute of Electrical and Electronics Engineers (IEEE), "IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Laser Gyros," IEEE Std 647-2006, 2006.
- [25] Institute of Electrical and Electronics Engineers (IEEE), "IEEE Standard Inertial Sensor Terminology," IEEE Std 528-2001, 2001.
- [26] A. Lawrence, Modern Inertial Technology, Navigation, Guidance, and Control, 2nd ed., New York: Springer, 1998.

Bibliography

- [27] P. D. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, 2nd ed., Artech House, 2013.
- [28] H. Weinberg, "Gyro Mechanical Performance: The Most Important Parameter," Technical Article MS-2158, Analog Devices, September 2011.
- [29] S. Nassar, "Improving the Inertial Navigation System (INS) Error Model for INS and INS/DGPS Applications," Dissertation, University of Calgary, November 2003.
- [30] S. Nassar and N. El-Sheimy, "Accuracy Improvement of Stochastic Modeling of Inertial Sensor Errors," *Zeitschrift für Geodäsie, Geoinformation und Landmanagement (zgl)*, vol. 130. Jhg., no. 3/2005, pp. 146-155, 2005.
- [31] S. Nassar and N. El-Sheimy, "INS Error Model Improvement for Enhanced INS/GPS Navigation During GPS Signal Blockage Period," *Survey Review*, vol. 38, no. 301, pp. 563-572, July 2006.
- [32] Y. Yuksel, N. El-Sheimy and A. Noureldin, "Error Modeling and Characterization of Environmental Effects for Low Cost Inertial MEMS Units," in *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, Indian Wells, CA, 4-6 May 2010.
- [33] Institute of Electrical and Electronics Engineers (IEEE), "IEEE Standard Specification Format Guide and Test Procedure for Coriolis Vibratory Gyros," IEEE Std 1431-2004, 2004.
- [34] Northrop Grumman LITEF, " μ IMU-IC Micro Inertial Measurement Unit," March 2013.
- [35] Department of Defense (DoD), "World Geodetic System 1984," NIMA TR8350.2, Third Edition, Amendment 1, 2000.
- [36] R. G. Gottlieb, „Fast Gravity, Gravity Partials, Normalized Gravity, Gravity Gradient Torque and Magnetic Field: Derivation, Code and Data,“ NASA Contractor Report 188243, Houston, Texas, 1993.
- [37] J. M. Leonard, F. G. Nievinski and G. H. Born, "Gravity Error Compensation Using Second-Order Gauss-Markov Processes," *Journal of Spacecraft and Rockets*, vol. 50, no. 1, pp. 217-229, 2013.
- [38] B. Eifeller and P. Spietz, "Shaping Filter Design for the Anomalous Gravity Field by Means of Spectral Factorization," *Manuscripta Geodaetica*, vol. 14, pp. 183-192, 1989.
- [39] A. Gelb, Applied Optimal Estimation, The MIT Press, 1972.
- [40] K. H. Neumayer, "Shaping Filters and The Upward Continuation Problem," *Journal of Geodesy*, vol. 72, no. 12, pp. 698-704, December 1998.
- [41] National Geospatial-Intelligence Agency (NGA), "EGM2008 Gravity Anomalies and DOV Data," 29 April 2013. [Online]. Available: http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm2008/anomalies_dov.html. [Accessed 28 June 2015].
- [42] C. Zaiss, „IMU Design for High Vibration Environments with Special Consideration for Vibration Rectification,“ Master thesis, University of Calgary, 2012.
- [43] S. A. Spiewak and W. J. E. Teskey, "Attenuation of Low Frequency Distortions in 6-DOF Inertial MEMS Sensors," in *2005 SEM Annual Conference & Exposition on Experimental and Applied Mechanics*, 2005.
- [44] J. Lai, P. Lv, J. Liu and B. Jiang, "Noncommutativity Error Analysis of Strapdown Inertial Navigation System under the Vibration in UAVs," *International Journal of Advanced Robotic Systems*, vol. 9, no. 136, 2012.
- [45] D. J. Kocian, "Study of the Effects of Vibration on Inertial Navigation System Accuracy," Master Thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, June 1986.
- [46] Department of Defense (DoD), "Environmental Engineering Considerations and Laboratory Tests," MIL-STD-810G, 2008.
- [47] C. Cohenour and F. Van Graas, "GPS Orbit and Clock Error Distributions," *Journal of The Institute of Navigation*, vol. 58, no. 1, pp. 17-28, Spring 2011.
- [48] L. Heng, G. X. Gao, T. Walter and P. Enge, "Statistical Characterization of GPS Signal-In-Space Errors," in *Proceedings of the 2011 International Technical Meeting of The Institute of Navigation*, San Diego, CA, 2011.
- [49] J. Taylor and E. Barnes, "GPS Current Signal-in-Space Navigation Performance," in *Proceedings of the 2005 International Technical Meeting of The Institute of Navigation*, San Diego, CA, January 2005.
- [50] D. L. M. Warren and J. F. Raquet, "Broadcast vs. Precise GPS Ephemerides: a Historical Perspective," *GPS Solutions*, vol. 7, no. 3, pp. 151-156, December 2003.
- [51] J. F. Zumberge and W. I. Bertiger, "Ephemeris and Clock Navigation Message Accuracy," in *Global Positioning System: Theory and Applications, Volume I*, B. Parkinson, J. Spilker, P. Axelrad and P. Enge, Eds., Washington DC, American Institute of Aeronautics and Astronautics (AIAA), 1996, pp. 585-599.
- [52] Department of Defense (DoD), "Global Positioning Systems Directorate Systems Engineering & Integration Interface Specification," IS-GPS-200G, September 2012.

- [53] Department of Defense (DoD), "Global Positioning System Standard Positioning Service Performance Standard," 4th edition, September 2008.
- [54] International GNSS Service (IGS), [Online]. Available: <http://www.igs.org/>.
- [55] K. Senior, P. Koppang, D. Matsakis and J. Ray, "Developing an IGS Timescale," in *Proceedings of the 2001 IEEE International Frequency Control Symposium and PDA Exhibition*, Seattle, WA, 6-8 June 2001.
- [56] K. L. Senior, J. R. Ray and R. L. Beard, "Characterization of Periodic Variations in the GPS Satellite Clocks," *GPS Solutions*, vol. 12, no. 3, pp. 221-2225, July 2008.
- [57] "GNSS Timescale Description," [Online]. Available: <http://www.unoosa.org/pdf/icg/2012/Timescale-IGS.pdf>. [Accessed 1 July 2015].
- [58] International GNSS Service (IGS), "IGS Products," [Online]. Available: <http://igscb.jpl.nasa.gov/components/prods.html>. [Accessed 27 December 2013].
- [59] U.S. Coast Guard Navigation Center, "GPS OPS Advisories for Year 2012," [Online]. Available: <http://navcen.uscg.gov/?Do=gpsArchives&path=OPSADVISORIES&year=2012>. [Accessed 27 December 2013].
- [60] L. Ljung, *System Identification: Theory for the User*, 2nd ed., Upper Saddle River, New Jersey: Prentice Hall PTR, 1999.
- [61] J. S. Subirana, J. M. Zornoza and M. Hernández-Pajares, "Precise GNSS Satellite Coordinates Computation," *Navipedia*, 2011. [Online]. Available: http://www.navipedia.net/index.php/Precise_GNSS_Satellite_Coordinates_Computation. [Accessed 14 May 2014].
- [62] J. S. Subirana, J. M. Zorónza and M. Hernández-Pajares, "Satellite Antenna Phase Centre," 2011. [Online]. Available: http://www.navipedia.net/index.php/Satellite_Antenna_Phase_Centre. [Accessed 27 May 2013].
- [63] National Geospatial-Intelligence Agency (NGA), "NGA," [Online]. Available: <https://www.nga.mil>.
- [64] E. D. Kaplan and C. J. Hegarty, *Understanding GPS: Principles and Applications*, 2nd ed., Norwood: Artech House, 2006.
- [65] J. A. Klobuchar, "Ionospheric Time-Delay Algorithm for Single-Frequency GPS Users," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 23, no. 3, pp. 325-331, May 1987.
- [66] D. Bilitza, "International Reference Ionosphere," National Aeronautics and Space Administration (NASA), 23 April 2012. [Online]. Available: <http://iri.gsfc.nasa.gov/>. [Accessed 28 June 2015].
- [67] G. Di Giovanni and S. M. Radicella, "An Analytical Model of the Electron Density Profile in the Ionosphere," *Advances in Space Research*, vol. 10, no. 11, pp. 27-30, 1990.
- [68] S. M. Radicella, "The NeQuick Model Genesis, Uses and Evolution," *Annals of Geophysics*, vol. 52, no. 3, pp. 417-422, June-August 2009.
- [69] International Telecommunication Union (ITU), "Ionospheric Propagation Data and Prediction Methods required for the Design of Satellite Services and Systems," Recommendation ITU-R P.531-12 (09/2013), 2013.
- [70] European Union, "European GNSS (Galileo) Open Service Signal in Space Interface Control Document," OS SIS ICD, Issue 1.1, 2010.
- [71] A. Angrisano, S. Gaglione, C. Gioia, M. Massaro and S. Troisi, "Benefit of the NeQuick Galileo Version in GNSS Single-Point Positioning," *International Journal of Navigation and Observation*, vol. 2013, 2013.
- [72] A. Angrisano, S. Gaglione, C. Gioia, M. Massaro and U. Robustelli, "Assessment of NeQuick Ionospheric Model for Galileo Single-Frequency Users," *Acta Geophysica*, vol. 61, no. 6, pp. 1457-1476, December 2013.
- [73] B. Nava, P. Coisson and S. M. Radicella, "A new Version of the NeQuick Ionosphere Electron Density Model," *Journal of Atmospheric and Solar-Terrestrial Physics*, pp. 1856-1862, 2008.
- [74] European Union, "Ionospheric Correction Algorithm for Galileo Single Frequency Users," Issue 1.1, June 2015.
- [75] International Telecommunication Union (ITU), "ITU-R Reference Ionospheric Characteristics," Recommendation ITU-R P.1239-3 (02/2012), 2012.
- [76] J. S. Subirana, J. M. Zornoza and M. Hernández-Pajares, *GNSS Data Processing. Volume 1: Fundamentals and Algorithms*, European Space Agency (ESA), 2013.
- [77] Solar Influences Data Analysis Center (SIDC), "The International Sunspot Number," Royal Observatory of Belgium, [Online]. Available: <http://www.sidc.be/silso/>.
- [78] European Space Agency (ESA), "User Guide for EGNOS Application Developers," Ed 1.1, 2009.
- [79] S. R. Steffes, *Development and Analysis of SHEFEX-2 Hybrid Navigation System Experiment*, Universität Bremen, 2013.

- [80] J. Saastamoinen, "Atmospheric Correction for the Troposphere and Stratosphere in Radio Ranging of Satellites," *The Use of Artificial Satellites for Geodesy, Geophysical Monograph Series*, pp. 247-251, 1972.
- [81] A. E. Niell, "Global Mapping Functions for the Atmosphere Delay at Radio Wavelengths," *Journal of Geophysical Research*, vol. 101, no. B2, pp. 3227-3246, 10 February 1996.
- [82] L. Oolman, "Atmospheric Sounding," University of Wyoming, College of Engineering, [Online]. Available: <http://weather.uwyo.edu/upperair/sounding.html>. [Accessed 23 April 2013].
- [83] F. Kleijer, Troposphere Modeling and Filtering for Precise GPS Leveling, Delft: NCG, Netherlands Geodetic Commission, 2004.
- [84] J. P. Collins and R. B. Langley, "Nominal and Extreme Error Performance of the UNB3 Troposphere Delay Model," Technical Report No. 204, Department of Geodesy and Geomatics Engineering, University of New Brunswick, 1999.
- [85] Zarlink Semiconductor, "GP2015 GPS Receiver RF Front End," September 2007.
- [86] Zarlink Semiconductor, "GP4020 GPS Receiver Baseband Processor," 2005.
- [87] J. R. Vig, "Quartz Crystal Resonators and Oscillators for Frequency Control and Timing Applications - A Tutorial," 2004.
- [88] Rakon, "Datasheet IT225B," 2003. [Online]. Available: http://www.ko4bb.com/Manuals/05%29_GPS_Timing/Novatel/Novatel_-_Superstar_II/Rakon_-_IT225BE_10MHz_TCXO.pdf. [Accessed 3 July 2014].
- [89] Filtronetics Inc., "Datasheet TCXO ON-169A," [Online]. Available: <http://www.filtro.net/images/products/oscillators/ON-169A.pdf>. [Accessed 3 July 2014].
- [90] Novatel, "Receivers OEM628, Next Generation High Performance GNSS Receiver," 2014. [Online]. Available: <http://www.novatel.com/assets/Documents/Papers/OEM628.pdf>.
- [91] B. W. Parkinson and J. J. Spilker, Global Positioning System: Theory and Applications, Washington: American Institute of Aeronautics and Astronautics (AIAA), 1996.
- [92] Zarlink Semiconductor, "GP2021 GPS 12-Channel Correlator," August 2005.
- [93] A. J. van Dierendonck, P. Fenton and T. Ford, "Theory and Performance of Narrow Correlator Spacing in a GPS Receiver," *Journal of the Institute of Navigation*, vol. 39, no. 3, pp. 265-283, 1992.
- [94] R. B. Miller, "A New Strapdown Attitude Algorithm," *Journal of Guidance, Control & Dynamics*, vol. 6, no. 4, pp. 287-291, July-August 1983.
- [95] M. B. Ignagni, "Optimal Strapdown Attitude Integration Algorithms," *Journal of Guidance, Control & Dynamics*, vol. 13, no. 2, pp. 363-369, March-April 1990.
- [96] J. G. Lee, Y. J. Yoon, J. G. Mark and D. A. Tazartes, "Extension of Strapdown Attitude Algorithm for High-Frequency Base Motion," *Journal of Guidance, Control & Dynamics*, vol. 13, no. 4, pp. 738-743, 1990.
- [97] V. Z. Gusinsky, V. M. Lesyuchevsky, Y. A. Litmanovich, H. Musoff and G. T. Schmidt, "New Procedure for Deriving Optimized Strapdown Attitude Algorithms," *Journal of Guidance, Control & Dynamics*, vol. 20, no. 4, pp. 673-680, July-August 1997.
- [98] P. Savage, "Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms," *Journal of Guidance, Control & Dynamics*, vol. 21, no. 1, pp. 19-28, January-February 1998.
- [99] P. Savage, "Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms," *Journal of Guidance, Control & Dynamics*, vol. 21, no. 2, 1998.
- [100] J. E. Bortz, "A New Mathematical Formulation for Strapdown Inertial Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vols. AES-7, no. 1, pp. 61 - 66, January 1971.
- [101] J. Dambeck and B. Braun, "A Single Frequency Strapdown Algorithm for Integrating IMUs in ECEF-Frame," in *Advances in Aerospace Guidance, Navigation and Control*, Berlin Heidelberg, Springer, 2011, pp. 305-319.
- [102] Y. Bar-Shalom, X. Rong Li and K. Thiagalingam, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*, New York: John Wiley & Sons, 2001.
- [103] I. N. Bronstein, K. A. Semendjajew, G. Musiol und H. Mühlig, *Taschenbuch der Mathematik*, Teubner.
- [104] H. D. Black, "A Passive System for Determining the Attitude of a Satellite," *AIAA Journal*, vol. 2, no. 7, pp. 1350-1351, July 1964.
- [105] F. L. Markley, "Attitude Determination Using Vector Observations: A Fast Optimal Algorithm," *The Journal of the Astronautical Sciences*, vol. 41, no. 2, pp. 261-280, April-June 1993.
- [106] Y. Bar-Shalom, M. Mallick, C. Huimin and R. Washburn, "One-Step Solution for the General Out-of-Sequence-Measurement Problem in Tracking," *IEEE Aerospace Conference Proceedings*, vol. 4, pp. 1551-1559, 9-16 March 2002.

- [107] J. Wendel, *Integrierte Navigationssysteme*, München: Oldenbourg, 2007.
- [108] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*, 3rd ed., Hoboken, New Jersey: John Wiley & Sons, 2008.
- [109] S. F. Schmidt, "Application of State Space Methods to Navigation Problems," in *Advanced in Control Systems*, C. T. Leondes, Ed., New York, Academic Press, 1966.
- [110] M. S. Grewal, L. R. Weill and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, 2nd ed., Hoboken, New Jersey: John Wiley & Sons, 2007.
- [111] R. J. Fitzgerald, "Simple Tracking Filters: Closed-Form Solutions," *IEEE Transactions on Aerospace and Electronic Systems*, Vols. AES-17, no. 6, pp. 781-785, November 1981.
- [112] R. A. Singer, "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," *IEEE Transactions on Aerospace and Electronic Systems*, Vols. AES-6, no. 4, pp. 473-483, July 1970.
- [113] I. Klein, S. Filin, T. Toledo and I. Rusnak, "Assessment of Aided-INS Performance," *The Journal of Navigation*, vol. 65, no. 1, pp. 169-185, January 2012.
- [114] H. Abou-Kandil, G. Freiling, V. Ionescu and G. Jank, *Matrix Riccati Equations in Control and System Theory*, Birkhäuser Verlag, 2003.
- [115] C. Moler and C. van Loan, "Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later," *SIAM Review*, vol. 25, no. 1, 2003.
- [116] B. Chen and S. Peng, "Robust Stability Analysis of Kalman Filter under Parametric and Noise Uncertainties," in *Approximate Kalman Filtering*, Singapore, World Scientific Publishing, 1993, pp. 179-192.
- [117] T. H. Kerr, "Numerical Approximations and Other Structural Issues in Practical Implementations of Kalman Filtering," in *Approximate Kalman Filtering*, Singapore, World Scientific Publishing, 1993, pp. 193-220.
- [118] J. P. Collins, "Assessment and Development of a Tropospheric Delay Model for Aircraft Users of the Global Positioning System," Technical Report No. 203, Department of Geodesy and Geomatics Engineering, University of New Brunswick, 1999.
- [119] Deutsches Institut für Normung e.V., „Normatmosphäre,“ ISO 2533, 1979.
- [120] A. N. Kolmogorov, "Dissipation of Energy in the Locally Isotropic Turbulence," *Proceedings: Mathematical and Physical Sciences*, vol. 434, no. 1890, pp. 15-17, 8 July 1991.
- [121] R. N. Treuhaft and G. E. Lanyi, "The Effect of The Dynamic Wet Troposphere on Radio Interferometric Measurements," *Radio Science*, vol. 22, no. 2, pp. 251-265, March-April 1987.
- [122] A. F. Romero-Wolf and C. S. Jacobs, "Effects of Tropospheric Spatio-Temporal Correlated Noise on the Analysis of Space Geodetic Data," *IVS General Meeting Proceedings*, 5-8 March 2012.
- [123] C. J. Naudet, "Estimation of Tropospheric Fluctuations using GPS Data," TDA Progress Report 42-126, NASA Jet Propulsion Laboratory, August 1996.
- [124] R. A. McKern, "A Study of Transformation Algorithms for Use in a Digital Computer," Master's thesis, Massachusetts Institute of Technology, 1968.
- [125] Analog Devices, "Tactical Grade Ten Degrees of Freedom Inertial Sensor ADIS16488A Data Sheet," 2014. [Online]. Available: http://www.analog.com/static/imported-files/data_sheets/ADIS16488A.pdf.
- [126] iMAR Gesellschaft für Inertiale Mess-, Automatisierungs- und Regelsysteme, "iIMU-FCAI-E," 2013.
- [127] Northrop Grumman LITEF, "LCI-100C Inertial Measurement Unit," March 2013.
- [128] Northrop Grumman LITEF, "TPD Product Description μIMU-I," 145680-0000-100, Rev. B, 2012.

A Preliminaries

A.1 Quaternions

Quaternions \tilde{q} are hyper-complex numbers with three imaginary parts

$$\tilde{q} := (q_0, \mathbf{q}) = q_0 + q_1 \cdot i + q_2 \cdot j + q_3 \cdot k \quad (\text{A.1})$$

with the real part q_0 and the imaginary parts combined in the vector $\mathbf{q} = (q_1 \ q_2 \ q_3)^\top$. The mathematician Arthur Cayley found a way to use quaternions as rotation parameters by interpreting them as follows

$$\begin{aligned} q_0 &= \cos \frac{\alpha}{2} \\ \mathbf{q} &= \sin \frac{\alpha}{2} \cdot \mathbf{n} \end{aligned} \quad (\text{A.2})$$

Therein, \mathbf{n} is the unit vector about which the frame has to be rotated with angle α to obtain the target frame. The rotation matrix \mathbf{R} is expressed with the quaternion rotation parameters \tilde{q} with

$$\mathbf{R}(\tilde{q}) = \frac{1}{q_0^2 + q_1^2 + q_2^2 + q_3^2} \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix} \quad (\text{A.3})$$

It is easy to show that $\mathbf{R}(\tilde{q}) \cdot \mathbf{R}^\top(\tilde{q}) = \mathbf{I}_3$ and $\det(\mathbf{R}(\tilde{q})) = +1$ such that the matrix \mathbf{R} actually belongs to the special orthogonal group $\text{SO}(3)$, and thus fulfills all properties of a rotation matrix.

The quaternion multiplication, which is for example required in the rotation ODE (2.1), is given with

$$\tilde{q} \cdot \tilde{p} = (q_0, \mathbf{q}) \cdot (p_0, \mathbf{p}) = (q_0 p_0 - \mathbf{q}^\top \mathbf{p}, q_0 \mathbf{p} + p_0 \mathbf{q} + \mathbf{q} \times \mathbf{p}) \quad (\text{A.4})$$

The quaternion inverse \tilde{q}^{-1} is defined such that $\tilde{q} \cdot \tilde{q}^{-1} = (1, \mathbf{0})$

$$\tilde{q}^{-1} = \frac{1}{q_0^2 + q_1^2 + q_2^2 + q_3^2} (q_0, -\mathbf{q}) \quad (\text{A.5})$$

A.2 Derivation of the Inertial Navigation Error ODE

First, the orientation error differential equation is derived. The derivation begins with the direction cosine matrix orientation differential equation

$$\dot{\mathbf{R}}_{eb} = \mathbf{R}_{eb} \boldsymbol{\Omega}_{ib} - \boldsymbol{\Omega}_{ie} \mathbf{R}_{eb} \quad (\text{A.6})$$

with $\boldsymbol{\Omega}_{ib} = \text{veck}^{-1}(\boldsymbol{\omega}_{ib})$ and $\boldsymbol{\Omega}_{ie} = \text{veck}^{-1}(\boldsymbol{\omega}_{ie})$ which is alternative to the quaternion version in (2.1). The error-free direction cosine matrix \mathbf{R}_{eb} is split into the small erroneous rotation $\mathbf{R}_{e\bar{e}}$ and the erroneous rotation $\mathbf{R}_{\bar{e}b}$ and the gyroscope measurement $\boldsymbol{\omega}_{ib}$ is perturbed

$$\mathbf{R}_{eb} = \mathbf{R}_{ee} \mathbf{R}_{eb}, \quad \boldsymbol{\Omega}_{ib} = \tilde{\boldsymbol{\Omega}}_{ib} + \delta\boldsymbol{\Omega}_{ib} \quad (\text{A.7})$$

The Earth rate $\boldsymbol{\omega}_{ie}$ is assumed to be perfectly known. (A.6) becomes

$$\frac{d}{dt}(\mathbf{R}_{ee} \mathbf{R}_{eb}) = \mathbf{R}_{ee} \mathbf{R}_{eb} (\tilde{\boldsymbol{\Omega}}_{ib} + \delta\boldsymbol{\Omega}_{ib}) - \boldsymbol{\Omega}_{ie} \mathbf{R}_{ee} \mathbf{R}_{eb} \quad (\text{A.8})$$

The left hand side can be further expressed as

$$\frac{d}{dt}(\mathbf{R}_{ee} \mathbf{R}_{eb}) = \dot{\mathbf{R}}_{ee} \mathbf{R}_{eb} + \mathbf{R}_{ee} \dot{\mathbf{R}}_{eb} = \dot{\mathbf{R}}_{ee} \mathbf{R}_{eb} + \mathbf{R}_{ee} (\mathbf{R}_{eb} \tilde{\boldsymbol{\Omega}}_{ib} - \boldsymbol{\Omega}_{ie} \mathbf{R}_{eb}) \quad (\text{A.9})$$

The differential equation (A.8) can thus be written as

$$\dot{\mathbf{R}}_{ee} \mathbf{R}_{eb} = \mathbf{R}_{ee} \mathbf{R}_{eb} (\tilde{\boldsymbol{\Omega}}_{ib} + \delta\boldsymbol{\Omega}_{ib}) - \boldsymbol{\Omega}_{ie} \mathbf{R}_{ee} \mathbf{R}_{eb} - \mathbf{R}_{ee} (\mathbf{R}_{eb} \tilde{\boldsymbol{\Omega}}_{ib} - \boldsymbol{\Omega}_{ie} \mathbf{R}_{eb}) \quad (\text{A.10})$$

It is assumed that the erroneous values fulfill (A.6)

$$\dot{\mathbf{R}}_{eb} = \mathbf{R}_{eb} \tilde{\boldsymbol{\Omega}}_{ib} - \boldsymbol{\Omega}_{ie} \mathbf{R}_{eb} \quad (\text{A.11})$$

such that (A.10) simplifies to

$$\dot{\mathbf{R}}_{ee} = \mathbf{R}_{ee} (\mathbf{R}_{eb} \delta\boldsymbol{\Omega}_{ib} \mathbf{R}_{eb}^T + \boldsymbol{\Omega}_{ie}) - \boldsymbol{\Omega}_{ie} \mathbf{R}_{ee} \quad (\text{A.12})$$

Solving this matrix differential equation explicitly for the derivatives of the three angle errors yields

$$\begin{aligned} \dot{\varphi}_{ee} &= \delta\omega_{ib_x,e} + \sin\varphi_{ee} \tan\vartheta_{ee} \delta\omega_{ib_y,e} + \cos\varphi_{ee} \tan\vartheta_{ee} (\delta\omega_{ib_z,e} + \omega_{ie}) \\ \dot{\vartheta}_{ee} &= \cos\varphi_{ee} \delta\omega_{ib_y,e} - \sin\varphi_{ee} (\delta\omega_{ib_z,e} + \omega_{ie}) \\ \dot{\psi}_{ee} &= \frac{\sin\varphi_{ee}}{\cos\vartheta_{ee}} \delta\omega_{ib_y,e} + \frac{\cos\varphi_{ee}}{\cos\vartheta_{ee}} (\delta\omega_{ib_z,e} + \omega_{ie}) - \omega_{ie} \end{aligned} \quad (\text{A.13})$$

Next, small angle approximations are applied to the trigonometric functions

$$\begin{pmatrix} \dot{\varphi}_{ee} \\ \dot{\vartheta}_{ee} \\ \dot{\psi}_{ee} \end{pmatrix} = \begin{pmatrix} 0 & \delta\omega_{ib_z,e} + \omega_{ie} & 0 \\ -\delta\omega_{ib_z,e} - \omega_{ie} & 0 & 0 \\ \delta\omega_{ib_y,e} & 0 & 0 \end{pmatrix} \begin{pmatrix} \varphi_{ee} \\ \vartheta_{ee} \\ \psi_{ee} \end{pmatrix} + \begin{pmatrix} \delta\omega_{ib_x,e} \\ \delta\omega_{ib_y,e} \\ \delta\omega_{ib_z,e} \end{pmatrix} + h.o.t. \quad (\text{A.14})$$

and finally the bilinear navigation state and gyroscope error terms are neglected to obtain the linear equation

$$\dot{\psi}_{ee} \doteq -\text{veck}^{-1}(\boldsymbol{\omega}_{ie}) \psi_{ee} + \mathbf{R}_{eb} \delta\boldsymbol{\omega}_{ib} \quad (\text{A.15})$$

Now, the translational inertial navigation error ODE are derived. For that, the position and velocity ODE in (2.1) are perturbed with $\mathbf{x}_e = \tilde{\mathbf{x}}_e + \delta\mathbf{x}_e$, $\mathbf{v}_e = \tilde{\mathbf{v}}_e + \delta\mathbf{v}_e$ and $\mathbf{R}_{eb} = \mathbf{R}_{ee} \mathbf{R}_{eb}$

$$\begin{aligned} \dot{\tilde{\mathbf{x}}}_e + \delta\dot{\mathbf{x}}_e &= \tilde{\mathbf{v}}_e + \delta\mathbf{v}_e \\ \dot{\tilde{\mathbf{v}}}_e + \delta\dot{\mathbf{v}}_e &= \mathbf{R}_{ee} \mathbf{R}_{eb} (\tilde{\mathbf{f}}_b + \delta\mathbf{f}_b) + \tilde{\mathbf{Y}}_e (\tilde{\mathbf{x}}_e + \delta\mathbf{x}_e) + \delta\mathbf{Y}_e - 2\boldsymbol{\omega}_{ie} \times (\tilde{\mathbf{v}}_e + \delta\mathbf{v}_e) \end{aligned} \quad (\text{A.16})$$

As in the rotational case it is assumed that the approximations fulfill the basic equations (2.1)

$$\begin{aligned} \dot{\tilde{\mathbf{x}}}_e &= \tilde{\mathbf{v}}_e \\ \dot{\tilde{\mathbf{v}}}_e &= \mathbf{R}_{eb} \tilde{\mathbf{f}}_b + \tilde{\mathbf{Y}}_e (\tilde{\mathbf{x}}_e) - 2\boldsymbol{\omega}_{ie} \times \tilde{\mathbf{v}}_e \end{aligned} \quad (\text{A.17})$$

and the nonlinear position and velocity error ODE (A.16) become

$$\begin{aligned}\delta \dot{\mathbf{x}}_e &= \delta \mathbf{v}_e \\ \delta \dot{\mathbf{v}}_e &= (\mathbf{R}_{ee} - \mathbf{I}_3) \mathbf{R}_{eb} \tilde{\mathbf{f}}_b + \mathbf{R}_{ee} \mathbf{R}_{eb} \delta \mathbf{f}_b + \tilde{\mathbf{y}}_e (\tilde{\mathbf{x}}_e + \delta \mathbf{x}_e) - \tilde{\mathbf{y}}_e (\tilde{\mathbf{x}}_e) + \delta \mathbf{y}_e - 2 \boldsymbol{\omega}_{ie} \times \delta \mathbf{v}_e\end{aligned}\quad (\text{A.18})$$

The rotation error matrix is substituted by $\mathbf{R}_{ee} = \mathbf{I}_3 + \boldsymbol{\Psi}_{ee} + h.o.t.$ in the velocity error equation

$$\begin{aligned}\delta \dot{\mathbf{v}}_e &= (\boldsymbol{\Psi}_{ee} + h.o.t.) \mathbf{R}_{eb} \tilde{\mathbf{f}}_b + (\mathbf{I}_3 + \boldsymbol{\Psi}_{ee} + h.o.t.) \mathbf{R}_{eb} \delta \mathbf{f}_b \\ &\quad + \tilde{\mathbf{y}}_e (\tilde{\mathbf{x}}_e + \delta \mathbf{x}_e) - \tilde{\mathbf{y}}_e (\tilde{\mathbf{x}}_e) + \delta \mathbf{y}_e - 2 \boldsymbol{\omega}_{ie} \times \delta \mathbf{v}_e\end{aligned}\quad (\text{A.19})$$

The higher order error terms are neglected to obtain the linearized velocity error ODE and the first cross product is flipped

$$\delta \dot{\mathbf{v}}_e \doteq -\text{veck}^{-1}(\mathbf{R}_{eb} \tilde{\mathbf{f}}_b) \boldsymbol{\Psi}_{ee} + \mathbf{R}_{eb} \delta \mathbf{f}_b + \tilde{\mathbf{y}}_e (\tilde{\mathbf{x}}_e + \delta \mathbf{x}_e) - \tilde{\mathbf{y}}_e (\tilde{\mathbf{x}}_e) + \delta \mathbf{y}_e - 2 \boldsymbol{\omega}_{ie} \times \delta \mathbf{v}_e \quad (\text{A.20})$$

Finally, Earth's gravity model has to be linearized

$$\delta \dot{\mathbf{v}}_e \doteq -\left(\mathbf{R}_{eb} (\boldsymbol{\Psi}_{eb}) \tilde{\mathbf{f}}_b \right) \times \boldsymbol{\Psi}_{ee} + \mathbf{R}_{eb} \delta \mathbf{f}_b + \frac{\partial \tilde{\mathbf{y}}_e (\mathbf{x}_e)}{\partial \mathbf{x}_e^T} \Big|_{\tilde{\mathbf{x}}_e} \delta \mathbf{x}_e + \delta \mathbf{y}_e - 2 \boldsymbol{\omega}_{ie} \times \delta \mathbf{v}_e \quad (\text{A.21})$$

The error ODE for position, velocity and orientation are written in a linear system

$$\begin{pmatrix} \delta \dot{\mathbf{x}}_e \\ \delta \dot{\mathbf{v}}_e \\ \boldsymbol{\Psi}_{ee} \end{pmatrix} \doteq \begin{pmatrix} 0 & \mathbf{I}_3 & 0 \\ \boldsymbol{\Gamma}_e & -2 \boldsymbol{\Omega}_{ie} & -\text{veck}^{-1}(\tilde{\mathbf{f}}_e) \\ 0 & 0 & -\boldsymbol{\Omega}_{ie} \end{pmatrix} \cdot \begin{pmatrix} \delta \mathbf{x}_e \\ \delta \mathbf{v}_e \\ \boldsymbol{\Psi}_{ee} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ \mathbf{R}_{eb} & 0 \\ 0 & \mathbf{R}_{eb} \end{pmatrix} \cdot \begin{pmatrix} \delta \mathbf{f}_b \\ \delta \boldsymbol{\omega}_{ib} \end{pmatrix} \quad (\text{A.22})$$

A.3 Block Matrices of the n-frame Inertial Navigation Error ODE

In this section the block matrices \mathbf{F}_{11} , \mathbf{F}_{21} , \mathbf{F}_{22} , \mathbf{F}_{31} , \mathbf{F}_{32} and \mathbf{F}_{33} of the system matrix in (2.7) are given for the sake of completeness without derivation

$$\begin{aligned}\mathbf{F}_{11} &= \frac{\partial (\mathbf{D}^{-1}(\boldsymbol{\lambda}) \cdot \tilde{\mathbf{v}}_n)}{\partial \boldsymbol{\lambda}^T} \Big|_{\tilde{\boldsymbol{\lambda}}} \\ \mathbf{F}_{21} &= \frac{\partial \tilde{\mathbf{y}}_n(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}^T} \Big|_{\tilde{\boldsymbol{\lambda}}} - \frac{\partial [(2 \mathbf{R}_{ne}(\boldsymbol{\lambda}) \boldsymbol{\Omega}_{ie} \mathbf{R}_{ne}^T(\boldsymbol{\lambda}) + \boldsymbol{\Omega}_{en}(\boldsymbol{\lambda}, \tilde{\mathbf{v}}_n)) \cdot \tilde{\mathbf{v}}_n]}{\partial \boldsymbol{\lambda}^T} \Big|_{\tilde{\boldsymbol{\lambda}}} \\ \mathbf{F}_{22} &= - \frac{\partial [(2 \mathbf{R}_{ne}(\tilde{\boldsymbol{\lambda}}) \boldsymbol{\Omega}_{ie} \mathbf{R}_{ne}^T(\tilde{\boldsymbol{\lambda}}) + \boldsymbol{\Omega}_{en}(\tilde{\boldsymbol{\lambda}}, \mathbf{v}_n)) \cdot \mathbf{v}_n]}{\partial \mathbf{v}_n^T} \Big|_{\tilde{\mathbf{v}}_n} \\ \mathbf{F}_{31} &= - \frac{\partial \boldsymbol{\omega}_{in}(\boldsymbol{\lambda}, \tilde{\mathbf{v}}_n)}{\partial \boldsymbol{\lambda}^T} \Big|_{\tilde{\boldsymbol{\lambda}}} \\ \mathbf{F}_{32} &= - \frac{\partial \boldsymbol{\omega}_{in}(\tilde{\boldsymbol{\lambda}}, \mathbf{v}_n)}{\partial \mathbf{v}_n^T} \Big|_{\tilde{\mathbf{v}}_n} \\ \mathbf{F}_{33} &= -\boldsymbol{\Omega}_{in}(\tilde{\boldsymbol{\lambda}}, \tilde{\mathbf{v}}_n)\end{aligned}\quad (\text{A.23})$$

where $\boldsymbol{\lambda} = (\phi, \lambda, h)^T$ is a vector with the WGS84 geodetic coordinates, $\boldsymbol{\Omega}_{en}$ is the skew-symmetric matrix of the transport rate, $\boldsymbol{\Omega}_{en} = \text{veck}^{-1}(\boldsymbol{\omega}_{en})$, $\boldsymbol{\omega}_{in}$ is the sum of Earth rate and transport rate, $\boldsymbol{\omega}_{in} = \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}$, and $\boldsymbol{\Omega}_{in}$ is the skew-symmetric matrix of $\boldsymbol{\omega}_{in}$, $\boldsymbol{\Omega}_{in} = \text{veck}^{-1}(\boldsymbol{\omega}_{in})$.

B Reality Modeling

B.1 Autocorrelation Matrix of the Gravity Disturbance Vector

It can be shown that the autocorrelation function of the gravity disturbance in north direction is the negative second derivative of the disturbing gravity potential autocorrelation function (3.75) with respect to the north direction x_n

$$R_{\delta\gamma_n\delta\gamma_n}(r) = -\frac{\partial^2 R_{TT}(r)}{\partial x_n^2} = -\frac{\partial^2 R_{TT}(r)}{\partial r^2} \cdot \left(\frac{\partial r}{\partial x_n} \right)^2 - \frac{\partial R_{TT}(r)}{\partial r} \cdot \frac{\partial^2 r}{\partial x_n^2} \quad (\text{B.1})$$

The autocorrelation function of the gravity disturbance in east direction is accordingly the negative second derivative of the disturbing gravity potential autocorrelation function (3.75) with respect to the east direction x_e

$$R_{\delta\gamma_e\delta\gamma_e}(r) = -\frac{\partial^2 R_{TT}(r)}{\partial x_e^2} = -\frac{\partial^2 R_{TT}(r)}{\partial r^2} \cdot \left(\frac{\partial r}{\partial x_e} \right)^2 - \frac{\partial R_{TT}(r)}{\partial r} \cdot \frac{\partial^2 r}{\partial x_e^2} \quad (\text{B.2})$$

The cross-correlation function of the gravity disturbance in north and east direction is the negative partial derivative of the disturbing gravity potential autocorrelation function (3.75) with respect to the north and east directions x_n and x_e

$$R_{\delta\gamma_n\delta\gamma_e}(r) = -\frac{\partial^2 R_{TT}(r)}{\partial x_n \partial x_e} = -\frac{\partial^2 R_{TT}(r)}{\partial r^2} \cdot \frac{\partial r}{\partial x_n} \cdot \frac{\partial r}{\partial x_e} - \frac{\partial R_{TT}(r)}{\partial r} \cdot \frac{\partial}{\partial x_n} \left(\frac{\partial r}{\partial x_e} \right) \quad (\text{B.3})$$

The autocorrelation function of the gravity disturbance in down direction is given by the Vening-Meinesz formula

$$R_{\delta\gamma_d\delta\gamma_d}(r) = R_{\delta\gamma_n\delta\gamma_n}(r) + R_{\delta\gamma_e\delta\gamma_e}(r) \quad (\text{B.4})$$

In [38] the different correlation functions are derived in detail. The spatial autocorrelation function matrix of the gravity disturbance vector at height $h=0$ is given by

$$\begin{aligned} \mathbf{R}_{\delta\gamma_n\delta\gamma_n}(r) &= \begin{pmatrix} R_{\delta\gamma_n\delta\gamma_n}(r) & R_{\delta\gamma_n\delta\gamma_e}(r) & 0 \\ R_{\delta\gamma_e\delta\gamma_n}(r) & R_{\delta\gamma_e\delta\gamma_e}(r) & 0 \\ 0 & 0 & R_{\delta\gamma_d\delta\gamma_d}(r) \end{pmatrix} \\ &= \sigma_T^2 \beta^2 e^{-\beta|r|} \begin{pmatrix} 1 - \beta|r|\cos^2 \chi & -\beta|r|\sin \chi \cos \chi & 0 \\ -\beta|r|\sin \chi \cos \chi & 1 - \beta|r|\sin^2 \chi & 0 \\ 0 & 0 & 2 - \beta|r| \end{pmatrix} \end{aligned} \quad (\text{B.5})$$

where the north and east directions x_n and x_e have been replaced by the radial distance r and the flight track angle χ

$$\begin{aligned} x_n &= r \cos \chi \\ x_e &= r \sin \chi \end{aligned} \quad (\text{B.6})$$

With the transformation matrix

$$\mathbf{R}_3(\chi) = \begin{pmatrix} \cos \chi & -\sin \chi & 0 \\ \sin \chi & \cos \chi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{B.7})$$

the spatial autocorrelation function matrix $\mathbf{R}_{\delta\mathbf{v}_n \delta\mathbf{v}_n}$ can be expressed with the autocorrelation function matrix that is aligned with the along- and cross-track directions of the flight trajectory

$$\mathbf{R}_{\delta\mathbf{v}_n \delta\mathbf{v}_n}(r) = \sigma_T^2 \beta^2 e^{-\beta|r|} \cdot \mathbf{R}_3(\chi) \cdot \begin{pmatrix} 1-\beta|r| & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2-\beta|r| \end{pmatrix} \cdot \mathbf{R}_3^T(\chi) \quad (\text{B.8})$$

It can be seen that the along- and cross-track errors are not correlated.

B.2 Satellite Track Frame

In this section the satellite track frame (t -frame) is introduced. The origin P_t of the t -frame is the APC of the satellite. As illustrated in Figure B-1, the x -axis unit vector \mathbf{t}_1 points into along-track direction, that is the direction of the current satellite velocity w.r.t. Earth \mathbf{V}_e . The y -axis unit vector \mathbf{t}_2 points into cross-track direction, thus orthogonal to the orbit plane, which is spanned by the current satellite velocity and position w.r.t. Earth. The z -axis unit vector \mathbf{t}_3 in radial direction completes the right hand system. The three unit vectors \mathbf{t}_1 , \mathbf{t}_2 , \mathbf{t}_3 and the corresponding transformation matrix between t - and e -frame \mathbf{R}_{et} are

$$\mathbf{R}_{et} = (\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3)_e = \left(\frac{\mathbf{V}_e}{\|\mathbf{V}_e\|}, \frac{\mathbf{V}_e \times \mathbf{X}_e}{\|\mathbf{V}_e \times \mathbf{X}_e\|}, \frac{\mathbf{V}_e \times (\mathbf{V}_e \times \mathbf{X}_e)}{\|\mathbf{V}_e \times (\mathbf{V}_e \times \mathbf{X}_e)\|} \right) \quad (\text{B.9})$$

Note that the z -axis generally coincides neither with the direction of the satellite position vector \mathbf{X}_e nor with the line-of-sight direction between the user and the APC e but slightly differs. Only in apogee and perigee the position vector \mathbf{X}_e and the radial unit vector \mathbf{t}_3 match.

Splitting the satellite position error $\delta\mathbf{X}_e$ into along-track, cross-track and radial directions yields

$$\delta\mathbf{X}_e = \frac{\mathbf{V}_e}{\|\mathbf{V}_e\|} \delta X_a + \frac{\mathbf{V}_e \times \mathbf{X}_e}{\|\mathbf{V}_e \times \mathbf{X}_e\|} \delta X_c + \frac{\mathbf{V}_e \times (\mathbf{V}_e \times \mathbf{X}_e)}{\|\mathbf{V}_e \times (\mathbf{V}_e \times \mathbf{X}_e)\|} \delta X_r \quad (\text{B.10})$$

The along-track error δX_a is

$$\delta X_a = \mathbf{t}_{1,e}^T \delta\mathbf{X}_e = \frac{\mathbf{V}_e^T \delta\mathbf{X}_e}{\|\mathbf{V}_e\|} \quad (\text{B.11})$$

The cross-track error δX_c is

$$\delta X_c = \mathbf{t}_{2,e}^T \delta\mathbf{X}_e = \frac{(\mathbf{V}_e \times \mathbf{X}_e)^T \delta\mathbf{X}_e}{\|\mathbf{V}_e \times \mathbf{X}_e\|} \quad (\text{B.12})$$

The radial error δX_r is

$$\delta X_r = \mathbf{t}_{3,e}^T \delta\mathbf{X}_e = \frac{[\mathbf{V}_e \times (\mathbf{V}_e \times \mathbf{X}_e)]^T \delta\mathbf{X}_e}{\|\mathbf{V}_e \times (\mathbf{V}_e \times \mathbf{X}_e)\|} \quad (\text{B.13})$$

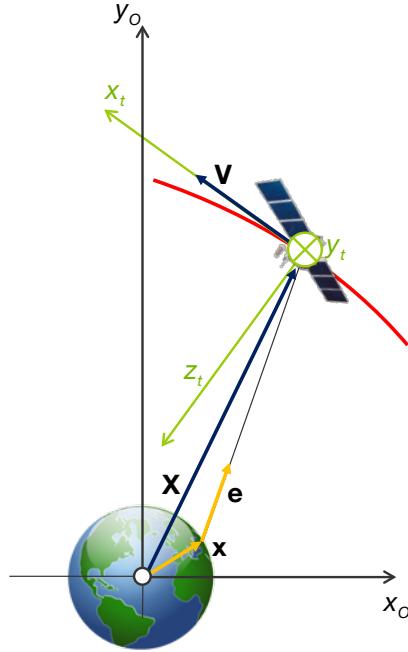


Figure B-1: Definition of the satellite track frame (t -frame)

B.3 Ionosphere Error

B.3.1 Ionospheric Pierce Point

The Klobuchar model as well as the TEC map model substitute the ionospheric layer with the electron density profile over height by a spherical shell of zero height extent in which the whole number of free electrons along the vertical path through the ionosphere is imaginarily concentrated. The Ionospheric Pierce Point (IPP) where the signal path from the satellite to the user intersects with this shell as well as the intersection angle are decisive for the computation of the ionospheric delay with one of the models. The height of the substitute shell above the Earth ellipsoid is, for example, chosen to be 350 km. The corresponding geometry is illustrated in Figure B-2.

The IPP can be calculated with

$$\mathbf{x}_{IPP_i} = \mathbf{x} + \left(-\mathbf{x}^T \mathbf{e}_i + \sqrt{\left(\mathbf{x}^T \mathbf{e}_i\right)^2 - \left(\|\mathbf{x}\|^2 - (R_e + h_i)^2\right)} \right) \mathbf{e}_i \quad (\text{B.14})$$

where R_e is the mean radius of the Earth, h_i is the effective height of the ionospheric shell above the Earth ellipsoid and \mathbf{e}_i is the normalized line-of-sight vector between the user antenna position \mathbf{x} and the i^{th} satellite position \mathbf{X}_i . Alternatively, the position of the IPP can be calculated by means of the formulas given in the GPS interface control document [52].

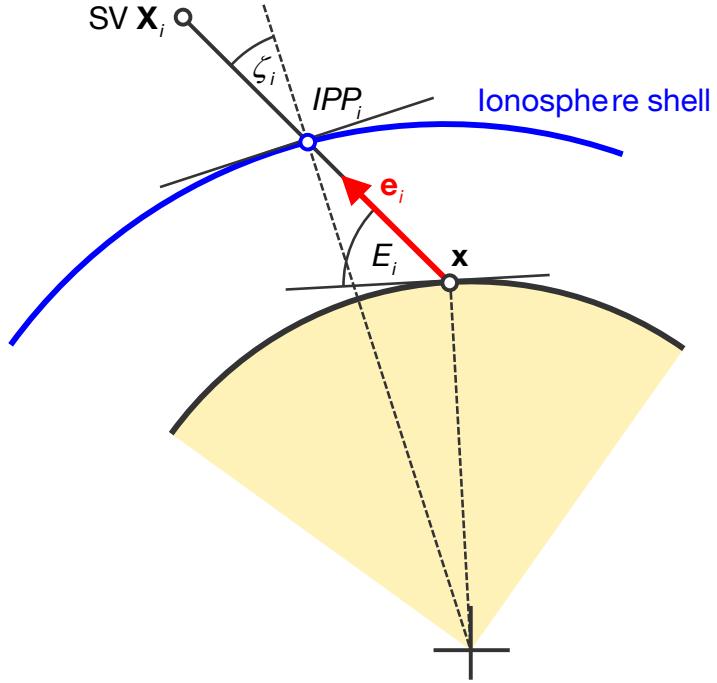


Figure B-2: Ionospheric shell, ionospheric pierce point (IPP) and ionospheric intersection angle ζ

B.3.2 Obliquity Factor

The obliquity factor F is defined as

$$F = \frac{1}{\cos \zeta} \quad (\text{B.15})$$

ζ is the angle between the line-of-sight and the normal of the local tangent plane at the IPP. F describes the prolongation of the signal path through the ionosphere due to the oblique piercing angle of satellites that are not in the zenith. The angle ζ is found from the scalar product of the negative line-of-sight vector transformed into the local n -frame at the IPP with the local normal direction

$$\cos \zeta = -\mathbf{e}_{i,n}^T \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad (\text{B.16})$$

The obliquity factor F then becomes

$$F = -\frac{1}{\mathbf{e}_{i,n,z}} \quad (\text{B.17})$$

Alternatively, the obliquity factor F can be computed by means of

$$F = 1 + 16 \left(0.53 - \pi^{-1} E_i \right)^3 \quad (\text{B.18})$$

given in the GPS interface control document [52] or by

$$F = \sqrt{1 - \left(\frac{R_e \cos E_i}{R_e + h_i} \right)^2}^{-1} \quad (\text{B.19})$$

given in the RTCA DO-229D standard [3] where E_i is the elevation angle of the i^{th} satellite at the user antenna position \mathbf{x} , $R_e = 6378.136$ km is the mean radius of the Earth and $h_i = 350$ km is the ionospheric height.

B.4 Troposphere Error

B.4.1 Zenith Delay Models

In general, the tropospheric zenith delay $\Delta\rho_{tropo,z}$ is caused by the refraction of the air in the atmosphere. The refraction N depends on the partial pressure of the dry air p_d , the partial pressure of the water vapor e as well as the temperature T

$$\begin{aligned}\Delta\rho_{tropo,z} &= 10^{-6} \int_{h_0}^{\infty} N dh \\ N &= k_1 \frac{p_d}{T} + k_2 \frac{e}{T} + k_3 \frac{e}{T^2}\end{aligned}\quad (\text{B.20})$$

where h_0 is the height of the receiver/antenna and k_1 , k_2 , and k_3 are the refractivity constants. The tropospheric delay $\Delta\rho_{tropo}$ is composed of the hydrostatic and the wet delay. The mapping functions represent the slant effect

$$\Delta\rho_{tropo} = m_h(\zeta) \cdot \Delta\rho_{h,z} + m_w(\zeta) \cdot \Delta\rho_{w,z} \quad (\text{B.21})$$

where $\Delta\rho_{h,z}$ is the hydrostatic zenith delay, $\Delta\rho_{w,z}$ is the wet zenith delay, $m_h(\zeta)$ is the hydrostatic mapping function, $m_w(\zeta)$ is the wet mapping function and ζ is the zenith angle.

Current hydrostatic zenith delay models base upon the same structure

$$\Delta\rho_{h,z}(p) = 10^{-6} \frac{k_1 R_d}{g_m} p \quad (\text{B.22})$$

The static pressure p at height h is the variable input of the function whereas k_1 is a constant, R_d is the specific gas constant of dry air and g_m is the mean gravity. The available models solely differ in the model constants k_1 , R_d and g_m . The most popular model is certainly the Saastamoinen hydrostatic zenith delay model [80], which uses

$$k_1 = 77.624 \cdot 10^{-2} \frac{\text{Km}^2}{\text{N}}, \quad R_d = 287.04 \frac{\text{J}}{\text{kgK}}, \quad g_m = 9.784 \frac{\text{m}}{\text{s}^2} \quad (\text{B.23})$$

The zenith hydrostatic delay model proposed by the RTCA standard DO-316 [3] for WAAS receivers uses the UNB3 model parameters

$$k_1 = 77.604 \cdot 10^{-2} \frac{\text{Km}^2}{\text{N}}, \quad R_d = 287.054 \frac{\text{J}}{\text{kgK}}, \quad g_m = 9.784 \frac{\text{m}}{\text{s}^2} \quad (\text{B.24})$$

A comprehensive essay about the UNB3 model can be found in [118]. The wet part of the tropospheric model can be represented by the Saastamoinen wet zenith delay model. It additionally depends on the temperature and the partial pressure of water vapor in the air

$$\Delta\rho_{w,z}(T, e) = 10^{-6} \frac{k_1 R_d}{g_m} \left(\frac{k_3}{k_1 \left(\lambda + 1 - \beta \frac{R_d}{g_m} \right) T} + \frac{k'_2}{k_1 (\lambda + 1)} \right) e \quad (\text{B.25})$$

The temperature T in $[\text{K}]$ and the partial pressure of water vapor e at the antenna height h in $[\text{N/m}^2]$ are the variables. Constants additional to those of the hydrostatic zenith delay model are

$\beta = 0.0062$	$[\frac{K}{m}]$	Temperature lapse rate
$\lambda = 3$	$[-]$	Water vapor lapse rate
$k_2 = 64.7 \cdot 10^{-2}$	$[\frac{Km^2}{N}]$	Constant
$k'_2 = k_2 - 0.622 k_1$	$[\frac{Km^2}{N}]$	Constant
$k_3 = 371900 \cdot 10^{-2}$	$[\frac{K^2 m^2}{N}]$	Constant

The WAAS wet zenith delay model has a similar but simpler structure

$$\Delta\rho_{w,z}(T, e) = 10^{-6} \frac{k_2 R_d}{g_m (\lambda + 1) - \beta R_d} \frac{e}{T} \quad (B.26)$$

It requires only one additional model constant, $k_2 = 382000 \cdot 10^{-2} \frac{K^2 m^2}{N}$. A comprehensive and thorough description of different troposphere models can be found in [83].

B.4.2 Temperature, Static Pressure and Partial Pressure of Water Vapor

If the temperature T and the static pressure p are not measured, they can be calculated from the standard atmosphere (w/o or with considering seasonal variations) or actual meteorological values given at mean sea level. Temperature, static pressure and partial pressure of water vapor change with height according to

$$\begin{aligned} T &= T_0 - \beta H \\ p &= p_0 \left(\frac{T}{T_0} \right)^{\frac{g}{R\beta}} = p_0 \left(1 - \frac{\beta}{T_0} H \right)^{\frac{g}{R\beta}} \\ e &= e_0 \left(\frac{T}{T_0} \right)^{\frac{(\lambda+1)g}{R\beta}} = e_0 \left(1 - \frac{\beta}{T_0} H \right)^{\frac{(\lambda+1)g}{R\beta}} \end{aligned} \quad (B.27)$$

where the geopotential height and the gravity are

$$\begin{aligned} H &= \frac{r h}{r + h} \\ g &= g_n \left(\frac{r}{r + h} \right)^2 \end{aligned} \quad (B.28)$$

and $g_n = 9.80665 \frac{m}{s^2}$, $r = 6356766 \text{ m}$ and $R = 287.05287 \frac{m^2}{K s^2}$. The standard atmosphere at mean sea level at mid-latitudes of the northern hemisphere is defined in [119] by the parameters

$p_0 [\text{N/m}^2]$	$T_0 [\text{K}]$	$e_0 [\text{N/m}^2]$	$\beta_0 [\text{K/m}]$	$\lambda_0 [-]$
$1013.25 \cdot 10^2$	288.15	$11.7 \cdot 10^2$	$6.50 \cdot 10^{-3}$	-

Average values at mean sea level for different latitudes are [3]

$\phi [\text{°}]$	$p_0 [\text{N/m}^2]$	$T_0 [\text{K}]$	$e_0 [\text{N/m}^2]$	$\beta_0 [\text{K/m}]$	$\lambda_0 [-]$
15	$1013.25 \cdot 10^2$	299.65	$26.31 \cdot 10^2$	$6.30 \cdot 10^{-3}$	2.77
30	$1017.25 \cdot 10^2$	294.15	$21.79 \cdot 10^2$	$6.05 \cdot 10^{-3}$	3.15
45	$1015.75 \cdot 10^2$	283.15	$11.66 \cdot 10^2$	$5.58 \cdot 10^{-3}$	2.57
60	$1011.75 \cdot 10^2$	272.15	$6.78 \cdot 10^2$	$5.39 \cdot 10^{-3}$	1.81
75	$1013.00 \cdot 10^2$	263.65	$4.11 \cdot 10^2$	$4.53 \cdot 10^{-3}$	1.55

Typical seasonal variations of the standard atmosphere parameters at mean sea level are given with

ϕ [°]	Δp_0 [N/m ²]	ΔT_0 [K]	Δe_0 [N/m ²]	$\Delta \beta_0$ [K/m]	$\Delta \lambda_0$ [-]	
15	$0.00 \cdot 10^2$	0.00	$0.00 \cdot 10^2$	$0.00 \cdot 10^{-3}$	0.00	
30	$-3.75 \cdot 10^2$	7.00	$8.85 \cdot 10^2$	$0.25 \cdot 10^{-3}$	0.33	
45	$-2.25 \cdot 10^2$	11.00	$7.24 \cdot 10^2$	$0.32 \cdot 10^{-3}$	0.46	
60	$-1.75 \cdot 10^2$	15.00	$5.36 \cdot 10^2$	$0.81 \cdot 10^{-3}$	0.74	
75	$-0.50 \cdot 10^2$	14.50	$3.39 \cdot 10^2$	$0.62 \cdot 10^{-3}$	0.30	

(B.31)

[3]. For a given latitude the average parameters $\xi_0(\phi)$ as well as the seasonal variations $\Delta\xi(\phi)$ are simply calculated by linear interpolation of the tabulated values

$$\begin{aligned} \xi_0(\phi) &= \xi_0(\phi_i) + \frac{\xi_0(\phi_{i+1}) - \xi_0(\phi_i)}{\phi_{i+1} - \phi_i} (\phi - \phi_i) , & \xi = \{p, T, e, \beta, \lambda\}, \quad \phi_i \leq \phi \leq \phi_{i+1} \\ \Delta\xi(\phi) &= \Delta\xi(\phi_i) + \frac{\Delta\xi(\phi_{i+1}) - \Delta\xi(\phi_i)}{\phi_{i+1} - \phi_i} (\phi - \phi_i) \end{aligned} \quad (B.32)$$

The interpolated seasonal variation $\Delta\xi(\phi)$ is then added to the interpolated average value $\xi_0(\phi)$ using a harmonic function and the number of the day in the year d [-]

$$\begin{aligned} \xi(\phi, d) &= \xi_0(\phi) - \Delta\xi(\phi) \cos\left(2\pi \frac{d - d_{min}}{365.25}\right) , & \xi = \{p, T, e, \beta, \lambda\} \\ d_{min} &= \begin{cases} 28 & \text{for } \phi \geq 0 \\ 211 & \text{for } \phi < 0 \end{cases} \end{aligned} \quad (B.33)$$

B.4.3 Mapping Functions

Mapping functions consider the slant effect and provide the oblique factor for the zenith delays. A common function for the hydrostatic and wet mapping, which depends on the elevation angle E and the coefficients a , b and c , is

$$m_i(E) = \frac{1 + \frac{a_i}{1 + \frac{b_i}{1 + c_i}}}{\sin E + \frac{a_i}{\sin E + \frac{b_i}{\sin E + c_i}}} , \quad i = \{h, w\} \quad (B.34)$$

The coefficients of the hydrostatic mapping function a_h , b_h and c_h are composed of a constant and a seasonal part

$$\begin{aligned} \xi_h(\phi, d) &= \xi_{h,0}(\phi) + \xi_{h,d}(\phi) \cos\left(2\pi \frac{d - d_{min}}{365.25}\right) , & \xi = \{a, b, c\} \\ d_{min} &= \begin{cases} 28 & \text{for } \phi \geq 0 \\ 211 & \text{for } \phi < 0 \end{cases} \end{aligned} \quad (B.35)$$

The various mapping functions merely differ in the a , b and c coefficients. The New Mapping Function (NMF) of Niell [81] gets along without current values for the static temperature and static pressure but makes use of tabulated average values for the hydrostatic mapping function

	$\phi = 15^\circ$	$\phi = 30^\circ$	$\phi = 45^\circ$	$\phi = 60^\circ$	$\phi = 75^\circ$	
$a_{h,0}$	$1.2769934 \cdot 10^{-3}$	$1.2683230 \cdot 10^{-3}$	$1.2465397 \cdot 10^{-3}$	$1.2196049 \cdot 10^{-3}$	$1.2045996 \cdot 10^{-3}$	
$b_{h,0}$	$2.9153695 \cdot 10^{-3}$	$2.9152299 \cdot 10^{-3}$	$2.9288445 \cdot 10^{-3}$	$2.9022565 \cdot 10^{-3}$	$2.9024912 \cdot 10^{-3}$	
$c_{h,0}$	$62.610505 \cdot 10^{-3}$	$62.837393 \cdot 10^{-3}$	$63.721774 \cdot 10^{-3}$	$63.824265 \cdot 10^{-3}$	$64.258455 \cdot 10^{-3}$	(B.36)
$a_{h,0}$	0.0	$1.2709626 \cdot 10^{-5}$	$2.6523662 \cdot 10^{-5}$	$3.4000452 \cdot 10^{-5}$	$4.1202191 \cdot 10^{-5}$	
$b_{h,0}$	0.0	$2.1414979 \cdot 10^{-5}$	$3.0160779 \cdot 10^{-5}$	$7.2562722 \cdot 10^{-5}$	$11.723375 \cdot 10^{-5}$	
$c_{h,0}$	0.0	$9.0128400 \cdot 10^{-5}$	$4.3497037 \cdot 10^{-5}$	$84.795348 \cdot 10^{-5}$	$170.37206 \cdot 10^{-5}$	

and for the wet mapping function

	$\phi = 15^\circ$	$\phi = 30^\circ$	$\phi = 45^\circ$	$\phi = 60^\circ$	$\phi = 75^\circ$	
a_w	$5.8021897 \cdot 10^{-4}$	$5.6794847 \cdot 10^{-4}$	$5.8118019 \cdot 10^{-4}$	$5.9727542 \cdot 10^{-4}$	$6.1641693 \cdot 10^{-4}$	
b_w	$1.4275268 \cdot 10^{-3}$	$1.5138625 \cdot 10^{-3}$	$1.4572752 \cdot 10^{-3}$	$1.5007428 \cdot 10^{-3}$	$1.7599082 \cdot 10^{-3}$	
c_w	$4.3472961 \cdot 10^{-2}$	$4.6729510 \cdot 10^{-2}$	$4.3908931 \cdot 10^{-2}$	$4.4626982 \cdot 10^{-2}$	$5.4736038 \cdot 10^{-2}$	

The WAAS mapping function [3] is

$$m(E) = \frac{1.001}{\sqrt{0.002001 + \sin^2 E}} \cdot \left(1 + 0.015 (\max(0, 4^\circ - E))^2 \right) \quad (\text{B.38})$$

The WAAS model is valid for satellite elevation angles not less than 2° .

B.4.4 Atmospheric Turbulence Structure Function

Structure functions are commonly used to describe random spatial scalar fields $z(\mathbf{x})$

$$D_z(\mathbf{x}, \mathbf{r}) = E[z(\mathbf{x} + \mathbf{r}) - z(\mathbf{x})]^2 \quad (\text{B.39})$$

\mathbf{x} is an arbitrary location and \mathbf{r} is the line-of-sight vector between two locations in the field. Structure functions describe the correlation of value differences of neighboring locations. In the special case of isotropic and ergodic spatial stochastic fields the structure function simplifies to a scalar function, which depends only on the relative distance r between two arbitrary locations in the field and not on any absolute position \mathbf{x}

$$D_z(r) = E[z(r) - z_0]^2 \quad (\text{B.40})$$

where $z_0 = z(r=0)$. The corresponding autocorrelation function is

$$R_z(\mathbf{x}, \mathbf{r}) = E[z(\mathbf{x} + \mathbf{r})z(\mathbf{x})] \quad (\text{B.41})$$

The relation between the structure function and the autocorrelation function is given with

$$\begin{aligned} D_z(\mathbf{x}, \mathbf{r}) &= E[z^2(\mathbf{x} + \mathbf{r}) - 2z(\mathbf{x} + \mathbf{r})z(\mathbf{x}) + z^2(\mathbf{x})] \\ &= E[z^2(\mathbf{x} + \mathbf{r})] - 2E[z(\mathbf{x} + \mathbf{r})z(\mathbf{x})] + E[z^2(\mathbf{x})] \\ &= E[z^2(\mathbf{x} + \mathbf{r})] - 2R_z(\mathbf{x}, \mathbf{r}) + E[z^2(\mathbf{x})] \end{aligned} \quad (\text{B.42})$$

Solving for the autocorrelation function yields

$$R_z(\mathbf{x}, \mathbf{r}) = \frac{1}{2} (E[z^2(\mathbf{x} + \mathbf{r})] + E[z^2(\mathbf{x})] - D_z(\mathbf{x}, \mathbf{r})) \quad (\text{B.43})$$

For isotropic and ergodic spatial stochastic fields the spatial autocorrelation function becomes

$$R_z(r) = E[z_0^2] - \frac{1}{2}D_z(r) = R_z(0) - \frac{1}{2}D_z(r) \quad (\text{B.44})$$

The spatial fluctuations of the refractivity N , which are mainly caused by turbulent flow of water vapor in the air, cannot be represented by the global deterministic zenith delay models. In turn, they can be described by the turbulence theory, which was firstly elaborated by Kolmogorov in 1941 [120]. The structure function $D_{\delta N}$ of turbulent fluctuations follows a power-law

$$D_{\delta N}(r) = C_{\delta N}^2 r^\rho \quad (\text{B.45})$$

with the coefficient $C_{\delta N} = 0.24 \text{ m}^{-\frac{1}{3}}$ [121]. In the near field (up to two kilometers), which is dominated by the small scale eddies, the exponent is $\rho = \frac{5}{3}$. In the far field (larger than two kilometers) with large scale scatter the exponent is $\rho = \frac{2}{3}$ [121]. The refraction fluctuations δN are converted to a radio signal zenith path delay by integrating along the signal path

$$\delta\rho_{tropo,z} = 10^{-6} \int_{h_0}^{\infty} \delta N \, dh \quad (\text{B.46})$$

The structure function of the zenith delay with scaling factor $C_{\delta\rho_{tropo,z}}^2$ and normalization with the effective height of the troposphere H , usually $H = 2 \text{ km}$, is given by

$$D_{\delta\rho_{tropo,z}}(r) = C_{\delta\rho_{tropo,z}}^2 \left(\frac{r}{H} \right)^\rho \quad (\text{B.47})$$

The coefficient $C_{\delta\rho_{tropo,z}}^2$ is calculated as

$$C_{\delta\rho_{tropo,z}}^2 = 10^{-12} C_{\delta N}^2 H^{\frac{8}{3}} \quad (\text{B.48})$$

The spatial structure function is converted into a temporal structure function by applying the frozen troposphere model as illustrated in Figure B-3 [122]. The model assumes that the overall turbulent structure of the troposphere is isotropic and spatially ergodic. Due to the wind and due to the motion of the receiver, the turbulent structures move with respect to the receiver and the spatial correlation translates into a correlation with time. The same approach has been used for the stochastic gravity error model. Mathematically, the conversion from the spatial random field to the temporal random process is accomplished by replacing the lag r by a product of velocity and time. The aerodynamic velocity $v_{A,h}$ to be used for that consists of the horizontal wind component and of the horizontal velocity of the platform

$$r = v_{A,h} \tau = (v_w + v_h) \tau \quad (\text{B.49})$$

The horizontal wind velocity v_w is usually assumed as 10 m/s. This goes quite well with the real data in Figure 3-84 for heights up to five kilometers. The horizontal component of the platform velocity is given by $v_h = \sqrt{v_n^2 + v_e^2}$. The structure function for the tropospheric zenith residual error (B.47) is then

$$D_{\delta\rho_{tropo,z}}(\tau) = C_{\delta\rho_{tropo,z}}^2 \left(\frac{v_{A,h} \tau}{H} \right)^\rho \quad (\text{B.50})$$

Further examples can be found in [123].

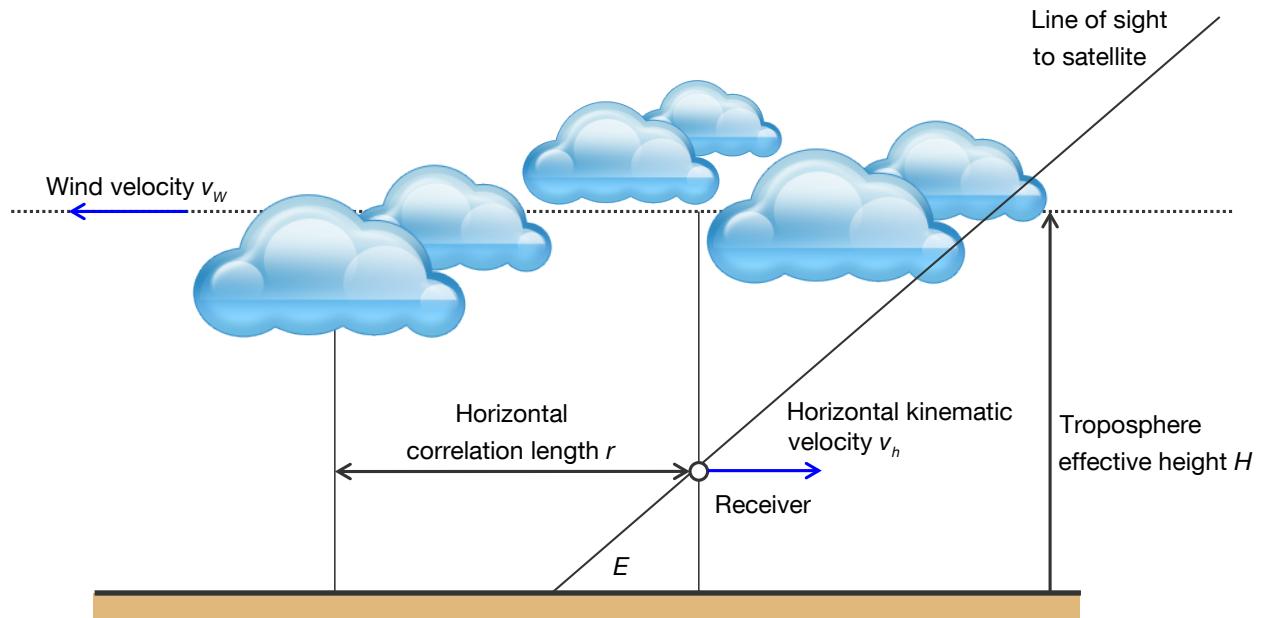


Figure B-3: Frozen troposphere assumption

C Efficiently Increasing Navigation Filter Consistency

C.1 Inertial Navigation Algorithms

C.1.1 Solution of the Orientation ODE using Laning-Bortz Parameters

The dual frequency algorithms for orientation base upon the Laning-Bortz parameter representation. The Laning-Bortz orientation vector σ can be interpreted as follows: its norm is the rotation angle and its direction represents the rotation axis. The non-linear orientation differential equation in Laning-Bortz parameters is

$$\dot{\sigma}_{ib} = \omega_{ib} + \frac{1}{2}\sigma_{ib} \times \omega_{ib} + \frac{1}{\|\sigma_{ib}\|^2} \left(1 - \frac{\|\sigma_{ib}\| \sin \|\sigma_{ib}\|}{2(1 - \cos \|\sigma_{ib}\|)} \right) \sigma_{ib} \times (\sigma_{ib} \times \omega_{ib}) \quad (C.1)$$

The $\sin \|\sigma_{ib}\|$ and $\cos \|\sigma_{ib}\|$ terms can be expanded in series

$$\frac{\|\sigma_{ib}\| \sin \|\sigma_{ib}\|}{2(1 - \cos \|\sigma_{ib}\|)} = \frac{\|\sigma_{ib}\| \left(\|\sigma_{ib}\| - \frac{\|\sigma_{ib}\|^3}{3!} + \dots \right)}{2 \left[1 - \left(1 - \frac{\|\sigma_{ib}\|^2}{2!} + \frac{\|\sigma_{ib}\|^4}{4!} - \dots \right) \right]} \approx \frac{1 - \frac{\|\sigma_{ib}\|^2}{6}}{1 - \frac{\|\sigma_{ib}\|^2}{12}} \quad (C.2)$$

The series are truncated after the second element since subsequent terms feature alternating signs. The denominator in turn can be expressed as power series, yielding

$$\begin{aligned} 1 - \frac{\|\sigma_{ib}\| \sin \|\sigma_{ib}\|}{2(1 - \cos \|\sigma_{ib}\|)} &\approx 1 - \frac{1 - \frac{\|\sigma_{ib}\|^2}{6}}{1 - \frac{\|\sigma_{ib}\|^2}{12}} = 1 - \left(1 - \frac{\|\sigma_{ib}\|^2}{6} \right) \left(1 + \frac{\|\sigma_{ib}\|^2}{12} + \frac{\|\sigma_{ib}\|^4}{144} + \dots \right) \\ &\approx \frac{\|\sigma_{ib}\|^2}{12} + \frac{\|\sigma_{ib}\|^4}{72} - \frac{\|\sigma_{ib}\|^6}{864} + \dots \approx \frac{\|\sigma_{ib}\|^2}{12} \end{aligned} \quad (C.3)$$

The first term of the series is positive and all other subsequent terms have alternating signs, which justifies the approximation above. The orientation differential equation (C.1) simplifies to

$$\dot{\sigma}_{ib} \approx \omega_{ib} + \frac{1}{2}\sigma_{ib} \times \omega_{ib} + \frac{1}{12}\sigma_{ib} \times (\sigma_{ib} \times \omega_{ib}) \quad (C.4)$$

According to [98], the two cross product terms can be replaced by

$$\frac{1}{2}\sigma_{ib}(t) \times \omega_{ib}(t) + \frac{1}{12}\sigma_{ib}(t) \times (\sigma_{ib}(t) \times \omega_{ib}(t)) \approx \frac{1}{2} \int_{t_0}^t \omega_{ib}(\tau) d\tau \times \omega_{ib}(t) \quad (C.5)$$

Integration of the differential equation with the simplification yields the orientation solution

$$\boldsymbol{\sigma}_{ib}(t) = \boldsymbol{\sigma}_{ib}(t_0) + \int_{t_0}^t \boldsymbol{\omega}_{ib}(\tau) d\tau + \frac{1}{2} \int_{t_0}^t \int_{t_0}^\xi \boldsymbol{\omega}_{ib}(\tau) d\tau \times \boldsymbol{\omega}_{ib}(\xi) d\xi + h.o.t. \quad (C.6)$$

The first integral term is just the sum of the integrating gyroscope angle increments. The double integral term has to be solved by efficient numerical integration schemes, which have for example been developed by Miller [94], Lee [96] or Gusinsky [97]. These algorithms are summarized in Template 4-1.

C.1.2 Derivation of the Single Frequency Approach

This section is dedicated to the derivation of the single frequency approach, which consists of the orientation quaternion difference equation and the velocity and position difference equations.

Orientation Quaternion Difference Equation

The orientation quaternion difference equation is derived at first. In continuous time the quaternion differential equation in e -frame is according to (2.1)

$$\dot{\tilde{q}}_{eb}(t) = \frac{1}{2} (\tilde{q}_{eb}(t) \tilde{\omega}_{ib}(t) - \tilde{\omega}_{ie} \tilde{q}_{eb}(t)) , \quad \tilde{q}_{eb}(t_0) = \tilde{q}_{eb_0} \quad (C.7)$$

It can be decomposed into a differential equation for the orientation of the b -frame with respect to the i -frame and a differential equation for the orientation of the e -frame with respect to the i -frame. The orientation quaternion \tilde{q}_{eb} is then obtained by multiplying the inverse of the solution of the latter differential equation \tilde{q}_{ie}^{-1} with the solution of the former differential equation \tilde{q}_{ib}

$$\begin{aligned} \dot{\tilde{q}}_{ib}(t) &= \frac{1}{2} \tilde{q}_{ib}(t) \tilde{\omega}_{ib}(t) , \quad \tilde{q}_{ib}(t_0) = \tilde{q}_{ib_0} \\ \dot{\tilde{q}}_{ie}(t) &= \frac{1}{2} \tilde{q}_{ie}(t) \tilde{\omega}_{ie} , \quad \tilde{q}_{ie}(t_0) = \tilde{q}_{ie_0} \\ \tilde{q}_{eb}(t) &= \tilde{q}_{ie}^{-1}(t) \cdot \tilde{q}_{ib}(t) \end{aligned} \quad (C.8)$$

The corresponding orientation difference equations are

$$\begin{aligned} \tilde{q}_{ib}(t + \Delta t) &= \tilde{q}_{ib}(t) \cdot \tilde{p}_b(t) , \quad \tilde{q}_{ib}(t_0) = \tilde{q}_{ib_0} \\ \tilde{q}_{ie}(t + \Delta t) &= \tilde{q}_{ie}(t) \cdot \tilde{p}_e(t) , \quad \tilde{q}_{ie}(t_0) = \tilde{q}_{ie_0} \\ \tilde{q}_{eb}(t + \Delta t) &= \tilde{p}_e^{-1}(t) \cdot \underbrace{\tilde{q}_{ie}^{-1}(t) \cdot \tilde{q}_{ib}(t)}_{\tilde{q}_{eb}(t)} \cdot \tilde{p}_b(t) = \tilde{p}_e^{-1}(t) \cdot \tilde{q}_{eb}(t) \cdot \tilde{p}_b(t) \end{aligned} \quad (C.9)$$

Therein, $\tilde{p}_e(t)$ and $\tilde{p}_b(t)$ are universal “transition” quaternions, which have to be determined next. For that, an expression for a general transition quaternion $\tilde{p}(t)$ is derived at first. The Hamilton quaternion \tilde{q} at time $t + \Delta t$ can be expanded in a Taylor series to describe the relation between the previous time step t and $t + \Delta t$

$$\tilde{q}(t + \Delta t) = \tilde{q}(t) + \dot{\tilde{q}}(t) \cdot \Delta t + \ddot{\tilde{q}}(t) \cdot \frac{\Delta t^2}{2!} + \dddot{\tilde{q}}(t) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4) \quad (C.10)$$

The derivatives of the quaternion with time are given by

$$\begin{aligned} \dot{\tilde{q}} &= \frac{1}{2} \tilde{q} \tilde{\omega} \\ \ddot{\tilde{q}} &= \frac{1}{2} \left(\dot{\tilde{q}} \tilde{\omega} + \tilde{q} \dot{\tilde{\omega}} \right) = \frac{1}{2} \tilde{q} \cdot \left(\frac{1}{2} \tilde{\omega}^2 + \dot{\tilde{\omega}} \right) \\ \dddot{\tilde{q}} &= \frac{1}{2} \dot{\tilde{q}} \cdot \left(\frac{1}{2} \tilde{\omega}^2 + \dot{\tilde{\omega}} \right) + \frac{1}{2} \tilde{q} \cdot \left(\frac{1}{2} \dot{\tilde{\omega}} \tilde{\omega} + \frac{1}{2} \tilde{\omega} \dot{\tilde{\omega}} + \ddot{\tilde{\omega}} \right) = \frac{1}{2} \tilde{q} \cdot \left(\frac{1}{4} \tilde{\omega}^3 + \tilde{\omega} \dot{\tilde{\omega}} + \frac{1}{2} \dot{\tilde{\omega}} \tilde{\omega} + \ddot{\tilde{\omega}} \right) \end{aligned} \quad (C.11)$$

By inserting the derivatives into the Taylor series expansion (C.10) the transition quaternion $\check{p}(t)$ can be generally expressed by the angular rate quaternion $\check{\omega}(t)$ and its time derivatives

$$\begin{aligned}\check{q}(t + \Delta t) &= \check{q}(t) \cdot \left[\check{1} + \frac{1}{2} \check{\omega}(t) \cdot \Delta t + \frac{1}{2} \left(\frac{1}{2} \check{\omega}^2(t) + \dot{\check{\omega}}(t) \right) \cdot \frac{\Delta t^2}{2!} + \right. \\ &\quad \left. + \frac{1}{2} \left(\frac{1}{4} \check{\omega}^3(t) + \check{\omega}(t) \dot{\check{\omega}}(t) + \frac{1}{2} \dot{\check{\omega}}(t) \check{\omega}(t) + \ddot{\check{\omega}}(t) \right) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4) \right] \\ &=: \check{p}(t)\end{aligned}\quad (\text{C.12})$$

where the quaternion $\check{1} := (1, \mathbf{0})$. The transition quaternion \check{p}_e of the differential equation $\dot{\check{q}}_{ie}(t) = \frac{1}{2} \check{q}_{ie}(t) \check{\omega}_{ie}$ is consequently with the constant Earth rate quaternion $\check{\omega}_{ie} = (0, \boldsymbol{\omega}_{ie})$

$$\check{p}_e = \check{1} + \frac{1}{2} \check{\omega}_{ie} \cdot \Delta t + \frac{1}{4} \check{\omega}_{ie}^2 \cdot \frac{\Delta t^2}{2!} + \frac{1}{8} \check{\omega}_{ie}^3 \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4) = \sum_{n=0}^{\infty} \frac{1}{n!} \left(\check{\omega}_{ie} \frac{\Delta t}{2} \right)^n \quad (\text{C.13})$$

which is exactly the Taylor series expansion of the exponential function. Thus

$$\check{p}_e = e^{\check{\omega}_{ie} \frac{\Delta t}{2}} \quad (\text{C.14})$$

The transition quaternion \check{p}_e can be computed with Euler's formula that is likewise valid for complex and hyper complex numbers

$$\check{p}_e = e^{\omega_{ie} \frac{\Delta t}{2} (0, \boldsymbol{\omega}_{ie}/\omega_{ie})} = \left(\cos\left(\omega_{ie} \frac{\Delta t}{2}\right), \sin\left(\omega_{ie} \frac{\Delta t}{2}\right) \frac{\boldsymbol{\omega}_{ie}}{\omega_{ie}} \right) \quad (\text{C.15})$$

The transition quaternion \check{p}_b of the differential equation $\dot{\check{q}}_{ib}(t) = \frac{1}{2} \check{q}_{ib}(t) \check{\omega}_{ib}(t)$ is with the angular rate quaternion $\check{\omega}_{ib}(t) = (0, \boldsymbol{\omega}_{ib}(t))$

$$\begin{aligned}\check{p}_b(t) &= \check{1} + \frac{1}{2} \check{\omega}_{ib}(t) \cdot \Delta t + \frac{1}{2} \left(\frac{1}{2} \check{\omega}_{ib}^2(t) + \dot{\check{\omega}}_{ib}(t) \right) \cdot \frac{\Delta t^2}{2!} + \\ &\quad \frac{1}{2} \left(\frac{1}{4} \check{\omega}_{ib}^3(t) + \check{\omega}_{ib}(t) \dot{\check{\omega}}_{ib}(t) + \frac{1}{2} \dot{\check{\omega}}_{ib}(t) \check{\omega}_{ib}(t) + \ddot{\check{\omega}}_{ib}(t) \right) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4)\end{aligned}\quad (\text{C.16})$$

The angular rates $\check{\omega}_{ib}(t)$ in turn can be obtained from the Taylor series expansions of the angle increment measurements $\Delta\check{\theta}_{ib}^+$ and $\Delta\check{\theta}_{ib}^-$ of the integrating gyroscope

$$\begin{aligned}\Delta\check{\theta}_{ib}^+ &= \int_t^{t+\Delta t} \check{\omega}_{ib}(\tau) d\tau = \underbrace{\int_t^t \check{\omega}_{ib}(\tau) d\tau}_{0} + \check{\omega}_{ib}(t) \cdot \Delta t + \dot{\check{\omega}}_{ib}(t) \cdot \frac{\Delta t^2}{2!} + \ddot{\check{\omega}}_{ib}(t) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4) \\ \Delta\check{\theta}_{ib}^- &= \int_{t-\Delta t}^t \check{\omega}_{ib}(\tau) d\tau = - \int_t^{t+(-\Delta t)} \check{\omega}_{ib}(\tau) d\tau = \check{\omega}_{ib}(t) \cdot \Delta t - \dot{\check{\omega}}_{ib}(t) \cdot \frac{\Delta t^2}{2!} + \ddot{\check{\omega}}_{ib}(t) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4)\end{aligned}\quad (\text{C.17})$$

In order to express the quadratic, cubic and mixed product terms of the angular rate and its derivatives in (C.16), the double products of the integrating gyroscope measurements are additionally required

$$\begin{aligned}\Delta\check{\theta}_{ib}^+ \cdot \Delta\check{\theta}_{ib}^+ &= \check{\omega}_{ib}^2(t) \cdot \Delta t^2 + (\check{\omega}_{ib}(t) \dot{\check{\omega}}_{ib}(t) + \dot{\check{\omega}}_{ib}(t) \check{\omega}_{ib}(t)) \cdot \frac{\Delta t^3}{2} + O(\Delta t^4) \\ \Delta\check{\theta}_{ib}^+ \cdot \Delta\check{\theta}_{ib}^- &= \check{\omega}_{ib}^2(t) \cdot \Delta t^2 - (\check{\omega}_{ib}(t) \dot{\check{\omega}}_{ib}(t) - \dot{\check{\omega}}_{ib}(t) \check{\omega}_{ib}(t)) \cdot \frac{\Delta t^3}{2} + O(\Delta t^4) \\ \Delta\check{\theta}_{ib}^- \cdot \Delta\check{\theta}_{ib}^+ &= \check{\omega}_{ib}^2(t) \cdot \Delta t^2 + (\check{\omega}_{ib}(t) \dot{\check{\omega}}_{ib}(t) - \dot{\check{\omega}}_{ib}(t) \check{\omega}_{ib}(t)) \cdot \frac{\Delta t^3}{2} + O(\Delta t^4)\end{aligned}\quad (\text{C.18})$$

$$\Delta\bar{\theta}_{ib}^- \cdot \Delta\bar{\theta}_{ib}^- = \bar{\omega}_{ib}^2(t) \cdot \Delta t^2 - (\bar{\omega}_{ib}(t) \dot{\bar{\omega}}_{ib}(t) + \dot{\bar{\omega}}_{ib}(t) \bar{\omega}_{ib}(t)) \cdot \frac{\Delta t^3}{2} + O(\Delta t^4)$$

The triple product of $\Delta\bar{\theta}_{ib}^+$ is given with

$$\Delta\bar{\theta}_{ib}^+ \cdot \Delta\bar{\theta}_{ib}^+ \cdot \Delta\bar{\theta}_{ib}^+ = \bar{\omega}_{ib}^3(t) \cdot \Delta t^3 + O(\Delta t^4) \quad (C.19)$$

The single terms of the transition quaternion \bar{p}_b are expressed by linearly combining the Taylor series of $\Delta\bar{\theta}_{ib}^+$ and $\Delta\bar{\theta}_{ib}^-$ (C.17), the double products (C.18) and the triple product (C.19)

$$\bar{p}_b(t) = \bar{1} + \mathbf{a}_1 \Delta\bar{\theta}_{ib}^+ + \mathbf{a}_2 \Delta\bar{\theta}_{ib}^- + \mathbf{a}_3 \Delta\bar{\theta}_{ib}^{+2} + \mathbf{a}_4 \Delta\bar{\theta}_{ib}^+ \Delta\bar{\theta}_{ib}^- + \mathbf{a}_5 \Delta\bar{\theta}_{ib}^- \Delta\bar{\theta}_{ib}^+ + \mathbf{a}_6 \Delta\bar{\theta}_{ib}^{-2} + \mathbf{a}_7 \Delta\bar{\theta}_{ib}^{+3} + O(\Delta t^4) \quad (C.20)$$

The comparison of the coefficients that are linear in $\bar{\omega}_{ib}(t)$ and its time derivatives gives

$$\begin{aligned} \bar{\omega}_{ib}(t) \cdot \Delta t &: \mathbf{a}_1 + \mathbf{a}_2 = \frac{1}{2} \\ \dot{\bar{\omega}}_{ib}(t) \cdot \Delta t^2 &: \frac{1}{2}\mathbf{a}_1 - \frac{1}{2}\mathbf{a}_2 = \frac{1}{4} \Rightarrow \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix} \\ \ddot{\bar{\omega}}_{ib}(t) \cdot \Delta t^3 &: \frac{1}{6}\mathbf{a}_1 + \frac{1}{6}\mathbf{a}_2 = \frac{1}{12} \end{aligned} \quad (C.21)$$

The comparison of the coefficients that are quadratic in $\bar{\omega}_{ib}(t)$ and its time derivatives gives

$$\begin{aligned} \bar{\omega}_{ib}^2(t) \cdot \Delta t^2 &: \mathbf{a}_3 + \mathbf{a}_4 + \mathbf{a}_5 + \mathbf{a}_6 = \frac{1}{8} \\ \bar{\omega}_{ib}(t) \dot{\bar{\omega}}_{ib}(t) \cdot \Delta t^3 &: \frac{1}{2}\mathbf{a}_3 - \frac{1}{2}\mathbf{a}_4 + \frac{1}{2}\mathbf{a}_5 - \frac{1}{2}\mathbf{a}_6 = \frac{1}{12} \Rightarrow \begin{pmatrix} \mathbf{a}_3 \\ \mathbf{a}_4 \\ \mathbf{a}_5 \end{pmatrix} = \begin{pmatrix} \frac{1}{8} \\ -\frac{1}{48} \\ \frac{1}{48} \end{pmatrix} \\ \dot{\bar{\omega}}_{ib}(t) \bar{\omega}_{ib}(t) \cdot \Delta t^3 &: \frac{1}{2}\mathbf{a}_3 + \frac{1}{2}\mathbf{a}_4 - \frac{1}{2}\mathbf{a}_5 - \frac{1}{2}\mathbf{a}_6 = \frac{1}{24} \end{aligned} \quad (C.22)$$

The comparison of the coefficients that are cubic in $\bar{\omega}_{ib}(t)$ gives

$$\bar{\omega}_{ib}^3(t) \cdot \Delta t^3 : \mathbf{a}_7 = \frac{1}{48} \quad (C.23)$$

Inserting the determined coefficients in (C.20) yields

$$\bar{p}_b(t) = \bar{1} + \frac{1}{2} \Delta\bar{\theta}_{ib}^+ + \frac{1}{8} \Delta\bar{\theta}_{ib}^{+2} - \frac{1}{48} (\Delta\bar{\theta}_{ib}^+ \Delta\bar{\theta}_{ib}^- - \Delta\bar{\theta}_{ib}^- \Delta\bar{\theta}_{ib}^+) + \frac{1}{48} \Delta\bar{\theta}_{ib}^{+3} + O(\Delta t^4) \quad (C.24)$$

The angle increment quaternion products are explicitly calculated

$$\begin{aligned} \Delta\bar{\theta}_{ib}^{+2} &= \left(-\|\Delta\boldsymbol{\Theta}_{ib}^+\|^2, \mathbf{0} \right), \quad \Delta\bar{\theta}_{ib}^{+3} = \left(\mathbf{0}, -\|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 \Delta\boldsymbol{\Theta}_{ib}^+ \right) \\ \Delta\bar{\theta}_{ib}^+ \Delta\bar{\theta}_{ib}^- &= \left(-\Delta\boldsymbol{\Theta}_{ib}^{+T} \Delta\boldsymbol{\Theta}_{ib}^-, \Delta\boldsymbol{\Theta}_{ib}^+ \times \Delta\boldsymbol{\Theta}_{ib}^- \right), \quad \Delta\bar{\theta}_{ib}^- \Delta\bar{\theta}_{ib}^+ = \left(-\Delta\boldsymbol{\Theta}_{ib}^{-T} \Delta\boldsymbol{\Theta}_{ib}^+, \Delta\boldsymbol{\Theta}_{ib}^- \times \Delta\boldsymbol{\Theta}_{ib}^+ \right) \end{aligned} \quad (C.25)$$

and the transition quaternion becomes

$$\begin{aligned} \bar{p}_b(t) &= \left(\bar{1} - \frac{1}{8} \|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 + O(\Delta t^4), \frac{1}{2} \Delta\boldsymbol{\Theta}_{ib}^+ - \frac{1}{24} \Delta\boldsymbol{\Theta}_{ib}^+ \times \Delta\boldsymbol{\Theta}_{ib}^- - \frac{1}{48} \|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 \Delta\boldsymbol{\Theta}_{ib}^+ + O(\Delta t^4) \right) \\ &= \left(\bar{1} - \frac{1}{8} \|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 + O(\Delta t^4), \frac{1}{2} \left(1 - \frac{1}{24} \|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 \right) \Delta\boldsymbol{\Theta}_{ib}^+ - \frac{1}{48} \Delta\boldsymbol{\Theta}_{ib}^+ \times \Delta\boldsymbol{\Theta}_{ib}^- + O(\Delta t^4) \right) \end{aligned} \quad (C.26)$$

The derivation in [124] comes to the same result. With the derived transition quaternions \bar{p}_e (C.15) and $\bar{p}_b(t)$ the quaternion orientation equation for integrating IMU (C.9) is

$$\begin{aligned} \bar{q}_{eb}(t + \Delta t) &= \left(\cos\left(\omega_{ie} \frac{\Delta t}{2}\right), -\sin\left(\omega_{ie} \frac{\Delta t}{2}\right) \frac{\boldsymbol{\omega}_{ie}}{\omega_{ie}} \right) \cdot \bar{q}_{eb}(t) \cdot \\ &\quad \left(\bar{1} - \frac{1}{8} \|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 + O(\Delta t^4), \frac{1}{2} \left(1 - \frac{1}{24} \|\Delta\boldsymbol{\Theta}_{ib}^+\|^2 \right) \Delta\boldsymbol{\Theta}_{ib}^+ - \frac{1}{24} \Delta\boldsymbol{\Theta}_{ib}^+ \times \Delta\boldsymbol{\Theta}_{ib}^- + O(\Delta t^4) \right) \end{aligned} \quad (C.27)$$

Velocity Difference and Position Difference Equation

The velocity difference equation is given by the Taylor series expansion of the velocity \mathbf{v}_e

$$\mathbf{v}_e(t + \Delta t) = \mathbf{v}_e(t) + \dot{\mathbf{v}}_e(t) \cdot \Delta t + \ddot{\mathbf{v}}_e(t) \cdot \frac{\Delta t^2}{2!} + O(\Delta t^3) \quad (\text{C.28})$$

The time derivative of the velocity is just the velocity differential equation of (2.1) and with the acceleration \mathbf{f}_b replaced by the corresponding “1 left / 1 right” conversion (4.8)

$$\dot{\mathbf{v}}_e(t) = \mathbf{R}(\tilde{q}_{eb}(t)) \frac{\Delta \mathbf{v}_b^+ + \Delta \mathbf{v}_b^-}{2\Delta t} + \mathbf{y}_e(\mathbf{x}_e(t)) - 2\boldsymbol{\omega}_{ie} \times \mathbf{v}_e(t) + O(\Delta t^2) \quad (\text{C.29})$$

The second time derivative of the velocity is then

$$\begin{aligned} \ddot{\mathbf{v}}_e(t) &= \mathbf{R}(\tilde{q}_{eb}(t)) \boldsymbol{\Omega}_{eb}(t) \mathbf{f}_b(t) + \mathbf{R}(\tilde{q}_{eb}(t)) \dot{\mathbf{f}}_b(t) + \frac{\partial \mathbf{y}_e(\mathbf{x}_e(t))}{\partial \mathbf{x}_e^\top} \mathbf{v}_e(t) - 2\boldsymbol{\omega}_{ie} \times \dot{\mathbf{v}}_e(t) \\ &= \mathbf{R}(\tilde{q}_{eb}(t)) \left[\boldsymbol{\omega}_{eb}(t) \times \frac{\Delta \mathbf{v}_b^+ + \Delta \mathbf{v}_b^-}{2\Delta t} \right] + \mathbf{R}(\tilde{q}_{eb}(t)) \frac{\Delta \mathbf{v}_b^+ - \Delta \mathbf{v}_b^-}{\Delta t^2} \\ &\quad + \frac{\partial \mathbf{y}_e(\mathbf{x}_e(t))}{\partial \mathbf{x}_e^\top} \mathbf{v}_e(t) - 2\boldsymbol{\omega}_{ie} \times \dot{\mathbf{v}}_e(t) + O(\Delta t^2) \end{aligned} \quad (\text{C.30})$$

Therein, the conversion of the acceleration derivative $\dot{\mathbf{f}}_b$ is from (4.15). The position difference equation is given by the Taylor series expansion of the position \mathbf{x}_e

$$\mathbf{x}_e(t + \Delta t) = \mathbf{x}_e(t) + \dot{\mathbf{x}}_e(t) \cdot \Delta t + \ddot{\mathbf{x}}_e(t) \cdot \frac{\Delta t^2}{2!} + \ddot{\mathbf{x}}_e(t) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4) \quad (\text{C.31})$$

or with the position derivatives substituted by the corresponding velocity derivatives

$$\mathbf{x}_e(t + \Delta t) = \mathbf{x}_e(t) + \mathbf{v}_e(t) \cdot \Delta t + \dot{\mathbf{v}}_e(t) \cdot \frac{\Delta t^2}{2!} + \ddot{\mathbf{v}}_e(t) \cdot \frac{\Delta t^3}{3!} + O(\Delta t^4) \quad (\text{C.32})$$

The rate vector $\boldsymbol{\omega}_{eb}(t)$ is required for the second derivative of the velocity (C.30)

$$\boldsymbol{\omega}_{eb}(t) = \boldsymbol{\omega}_{ib}(t) - \mathbf{R}_{eb}^\top(t) \boldsymbol{\omega}_{ie} \quad (\text{C.33})$$

The angular rate $\boldsymbol{\omega}_{ib}(t)$ is calculated in 1st order accuracy from the measured gyro triad increments $\Delta \boldsymbol{\theta}_{ib}^-$ and $\Delta \boldsymbol{\theta}_{ib}^+$ with the “1 left / 1 right” conversion (4.8). Inserting into (C.33) yields

$$\boldsymbol{\omega}_{eb}(t) = \frac{1}{2\Delta t} (\Delta \boldsymbol{\theta}_{ib}^+ + \Delta \boldsymbol{\theta}_{ib}^-) - \mathbf{R}^\top(\tilde{q}_{eb}(t)) \boldsymbol{\omega}_{ie} + O(\Delta t^2) \quad (\text{C.34})$$

The equations of the single frequency approach are now readily derived and summarized in Template 4-3.

Template C-1: Single frequency approach (n-frame, 3rd order orientation accuracy)

$$\begin{aligned}
\phi(t + \Delta t) &= \phi(t) + \dot{\phi}(t) \Delta t + \ddot{\phi}(t) \frac{\Delta t^2}{2!} + \dddot{\phi}(t) \frac{\Delta t^3}{3!} + O(\Delta t^4) \\
\lambda(t + \Delta t) &= \lambda(t) + \dot{\lambda}(t) \Delta t + \ddot{\lambda}(t) \frac{\Delta t^2}{2!} + \dddot{\lambda}(t) \frac{\Delta t^3}{3!} + O(\Delta t^4) \\
h(t + \Delta t) &= h(t) + \dot{h}(t) \Delta t + \ddot{h}(t) \frac{\Delta t^2}{2!} + \dddot{h}(t) \frac{\Delta t^3}{3!} + O(\Delta t^4) \\
\mathbf{v}_n(t + \Delta t) &= \mathbf{v}_n(t) + \dot{\mathbf{v}}_n(t) \Delta t + \ddot{\mathbf{v}}_n(t) \frac{\Delta t^2}{2!} + O(\Delta t^3) \\
\bar{q}_{nb}(t + \Delta t) &= \bar{p}_n^{-1}(t) \bar{q}_{nb}(t) \bar{p}_b(t) + O(\Delta t^4)
\end{aligned}$$

where

$$\begin{aligned}
\begin{pmatrix} \dot{\phi}(t) \\ \dot{\lambda}(t) \\ \dot{h}(t) \end{pmatrix} &= \mathbf{D}^{-1}(\phi, h) \mathbf{v}_n(t), \quad \begin{pmatrix} \ddot{\phi}(t) \\ \ddot{\lambda}(t) \\ \ddot{h}(t) \end{pmatrix} = -\mathbf{D}^{-1}(\phi, h) \dot{\mathbf{D}}(\phi, h) \begin{pmatrix} \dot{\phi}(t) \\ \dot{\lambda}(t) \\ \dot{h}(t) \end{pmatrix} + \mathbf{D}^{-1}(\phi, h) \dot{\mathbf{v}}_n(t) \\
\begin{pmatrix} \ddot{\phi}(t) \\ \ddot{\lambda}(t) \\ \ddot{h}(t) \end{pmatrix} &= -\mathbf{D}^{-1}(\phi, h) \ddot{\mathbf{D}}(\phi, h) \begin{pmatrix} \dot{\phi}(t) \\ \dot{\lambda}(t) \\ \dot{h}(t) \end{pmatrix} - 2\mathbf{D}^{-1}(\phi, h) \dot{\mathbf{D}}(\phi, h) \begin{pmatrix} \ddot{\phi}(t) \\ \ddot{\lambda}(t) \\ \ddot{h}(t) \end{pmatrix} + \mathbf{D}^{-1}(\phi, h) \ddot{\mathbf{v}}_n(t) \\
\mathbf{D}(\phi, h) &= \begin{pmatrix} M(\phi) + h & 0 & 0 \\ 0 & (N(\phi) + h) \cos \phi & 0 \\ 0 & 0 & -1 \end{pmatrix} \\
\dot{\mathbf{D}}(\phi, h) &= \begin{pmatrix} \dot{h} & 0 & 0 \\ 0 & \dot{h} \cos \phi - (N(\phi) + h) \sin \phi \dot{\phi} & 0 \\ 0 & 0 & 0 \end{pmatrix} \\
\ddot{\mathbf{D}}(\phi, h) &\approx \begin{pmatrix} \ddot{h} & 0 & 0 \\ 0 & \ddot{h} \cos \phi - 2\dot{h} \sin \phi \dot{\phi} - (N(\phi) + h) \cos \phi \dot{\phi}^2 - (N(\phi) + h) \sin \phi \ddot{\phi} & 0 \\ 0 & 0 & 0 \end{pmatrix}
\end{aligned}$$

$$\begin{aligned}
\dot{\mathbf{v}}_n(t) &= \mathbf{R}(\bar{q}_{nb}(t)) \frac{\Delta \mathbf{v}_b^+ + \Delta \mathbf{v}_b^-}{2\Delta t} + \mathbf{V}_n(\phi, h) - (2\mathbf{R}_{en}^\top(\lambda, \phi) \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}(t)) \times \mathbf{v}_n(t) \\
\ddot{\mathbf{v}}_n(t) &= \mathbf{R}(\bar{q}_{nb}(t)) \left(\boldsymbol{\omega}_{nb} \times \frac{\Delta \mathbf{v}_b^+ + \Delta \mathbf{v}_b^-}{2\Delta t} \right) + \mathbf{R}(\bar{q}_{nb}(t)) \frac{\Delta \mathbf{v}_b^+ - \Delta \mathbf{v}_b^-}{\Delta t^2} + \\
&\quad \frac{\partial \mathbf{V}_n(\phi, h)}{\partial (\phi, h)} \begin{pmatrix} \dot{\phi} \\ \dot{h} \end{pmatrix} - (2\mathbf{R}_{en}^\top(\lambda, \phi) \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}(t)) \times \dot{\mathbf{v}}_n(t) - \dot{\boldsymbol{\omega}}_{en}(t) \times \mathbf{v}_n(t) \\
\boldsymbol{\omega}_{nb}(t) &= \frac{1}{2\Delta t} (\Delta \boldsymbol{\theta}_{ib}^+ + \Delta \boldsymbol{\theta}_{ib}^-) - \mathbf{R}^\top(\bar{q}_{nb}(t)) \boldsymbol{\omega}_{in}(t) \\
\bar{p}_n(t) &= \left(1 - \frac{1}{8} \|\boldsymbol{\omega}_{in}(t)\|^2 \Delta t^2, \frac{1}{2} \boldsymbol{\omega}_{in}(t) \cdot \Delta t + \frac{1}{4} \dot{\boldsymbol{\omega}}_{in}(t) \Delta t^2 \right) \\
\bar{p}_b(t) &= \left(1 - \frac{1}{8} \|\Delta \boldsymbol{\theta}_{ib}^+\|^2, \frac{1}{2} \left(1 - \frac{1}{24} \|\Delta \boldsymbol{\theta}_{ib}^+\|^2 \right) \Delta \boldsymbol{\theta}_{ib}^+ - \frac{1}{24} \Delta \boldsymbol{\theta}_{ib}^+ \times \Delta \boldsymbol{\theta}_{ib}^- \right)
\end{aligned}$$

IMU Measurements :

$$\Delta \boldsymbol{\theta}_{ib}^+ = \int_t^{t+\Delta t} \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta \boldsymbol{\theta}_{ib}^- = \int_{t-\Delta t}^t \boldsymbol{\omega}_{ib}(\tau) d\tau, \quad \Delta \mathbf{v}_b^+ = \int_t^{t+\Delta t} \mathbf{f}_b(\tau) d\tau, \quad \Delta \mathbf{v}_b^- = \int_{t-\Delta t}^t \mathbf{f}_b(\tau) d\tau$$

C.2 Navigation Filter Initialization

C.2.1 Orientation Angle Expected Values

The velocity differential equation in n -frame is given in (2.3) with

$$\dot{\mathbf{v}}_n = \mathbf{R}_{nb} \mathbf{f}_b + \mathbf{y}_n - (2 \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} + \boldsymbol{\omega}_{en}) \times \mathbf{v}_n \quad (\text{C.35})$$

Stationarity means that

$$\mathbf{v}_n = 0, \quad \dot{\mathbf{v}}_n = 0 \quad (\text{C.36})$$

which yields, when inserted into the velocity differential equation (C.35),

$$0 = \mathbf{R}_{nb} \mathbf{f}_b + \mathbf{y}_n \quad (\text{C.37})$$

With the transformation matrix \mathbf{R}_{nb} between n - and b -frame

$$\mathbf{R}_{nb} = \begin{pmatrix} \cos \vartheta_{nb} \cos \psi_{nb} & \sin \varphi_{nb} \sin \vartheta_{nb} \cos \psi_{nb} - \cos \varphi_{nb} & \cos \varphi_{nb} \sin \vartheta_{nb} \cos \psi_{nb} + \sin \varphi_{nb} \sin \psi_{nb} \\ \sin \varphi_{nb} \cos \vartheta_{nb} \sin \psi_{nb} & \sin \varphi_{nb} \sin \vartheta_{nb} \sin \psi_{nb} + \cos \varphi_{nb} \cos \psi_{nb} & \cos \varphi_{nb} \sin \vartheta_{nb} \sin \psi_{nb} - \sin \varphi_{nb} \cos \psi_{nb} \\ -\sin \vartheta_{nb} & \sin \varphi_{nb} \cos \vartheta_{nb} & \cos \varphi_{nb} \cos \vartheta_{nb} \end{pmatrix} \quad (\text{C.38})$$

and the gravity vector

$$\mathbf{y}_n = \begin{pmatrix} 0 \\ 0 \\ \gamma \end{pmatrix} \quad (\text{C.39})$$

the relation (C.37) solved for \mathbf{f}_b becomes

$$\mathbf{f}_b = -\gamma \begin{pmatrix} -\sin \vartheta_{nb} \\ \sin \varphi_{nb} \cos \vartheta_{nb} \\ \cos \varphi_{nb} \cos \vartheta_{nb} \end{pmatrix} \quad (\text{C.40})$$

The roll angle φ_{nb} and the pitch angle ϑ_{nb} are calculated as

$$\varphi_{nb} = \text{atan2}(-f_{b,y}, -f_{b,z}) \in [-\pi, \pi) \quad (\text{C.41})$$

$$\vartheta_{nb} = \arcsin \frac{f_{b,x}}{\gamma} \in \left[-\frac{\pi}{2}, \frac{\pi}{2} \right] \quad (\text{C.42})$$

Generally, the angle about the gravity vector direction cannot be observed. Since gravity is pointing towards the z -axis of the n -frame, it is just the heading angle ψ_{nb} that is not observable by the acceleration measurement \mathbf{f}_b .

When stationary, the gyroscope measures only the Earth rate $\boldsymbol{\omega}_{ie}$, here transformed to n -frame

$$\mathbf{R}_{nb} \boldsymbol{\omega}_{ib} = \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} \quad (\text{C.43})$$

The transformation matrix \mathbf{R}_{nb} can be split into the three single rotations about the z -, y - and x -axes. The attitude angles of the first and second rotations about the roll and the pitch axis have already been determined and can be directly used to transform the measured angular rate from the b -frame to the leveled h -frame, which is already parallel to the local tangent plane but not yet adjusted to north

$$\underbrace{\mathbf{R}_3(\psi_{nb})\mathbf{R}_2(\vartheta_{nb})\mathbf{R}_1(\varphi_{nb})}_{=: \boldsymbol{\omega}_{ib,h}} \cdot \boldsymbol{\omega}_{ib} = \mathbf{R}_{ne} \boldsymbol{\omega}_{ie} \quad (C.44)$$

Therein, the angular rate transformed to the horizontal h -frame is given with

$$\boldsymbol{\omega}_{ib,h} = \begin{pmatrix} \omega_{ib,x} \cos \vartheta_{nb} + \omega_{ib,y} \sin \vartheta_{nb} \sin \varphi_{nb} + \omega_{ib,z} \sin \vartheta_{nb} \cos \varphi_{nb} \\ \omega_{ib,y} \cos \varphi_{nb} - \omega_{ib,z} \sin \varphi_{nb} \\ -\omega_{ib,x} \sin \vartheta_{nb} + \omega_{ib,y} \cos \vartheta_{nb} \sin \varphi_{nb} + \omega_{ib,z} \cos \vartheta_{nb} \cos \varphi_{nb} \end{pmatrix} \quad (C.45)$$

(C.44) is further

$$\begin{pmatrix} \cos \psi_{nb} & -\sin \psi_{nb} & 0 \\ \sin \psi_{nb} & \cos \psi_{nb} & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \boldsymbol{\omega}_{ib,h} = \omega_{ie} \cdot \begin{pmatrix} \cos \phi \\ 0 \\ -\sin \phi \end{pmatrix} \quad (C.46)$$

Only the two first equations contain the searched heading angle ψ_{nb} . Considering only these two equations and exchanging the angular rate and the trigonometric terms yields

$$\begin{pmatrix} -\omega_{ib,h,y} & \omega_{ib,h,x} \\ \omega_{ib,h,x} & \omega_{ib,h,y} \end{pmatrix} \cdot \begin{pmatrix} \sin \psi_{nb} \\ \cos \psi_{nb} \end{pmatrix} = \omega_{ie} \cdot \begin{pmatrix} \cos \phi \\ 0 \end{pmatrix} \quad (C.47)$$

Only the north component of the Earth rate vector $\boldsymbol{\omega}_{ie}$ and the horizontal components of the measured angular rate $\boldsymbol{\omega}_{ib}$ are required to observe the heading angle ψ_{nb} . The down component is aligned to the gravity vector and does not contribute to the observability of the heading angle. The angular rate matrix is brought to the right and is expanded

$$\begin{pmatrix} \sin \psi_{nb} \\ \cos \psi_{nb} \end{pmatrix} = \frac{-\omega_{ie} \cos \phi}{\omega_{ib,h,x}^2 + \omega_{ib,h,y}^2} \cdot \begin{pmatrix} \omega_{ib,h,y} \\ -\omega_{ib,h,x} \end{pmatrix} \quad (C.48)$$

and the heading angle ψ_{nb} is finally found by calculating the double-argument arctangent

$$\begin{aligned} \psi_{nb} = \text{atan2}\left(-\omega_{ib,y} \cos \varphi_{nb} + \omega_{ib,z} \sin \varphi_{nb}, \dots \right. \\ \left. \omega_{ib,x} \cos \vartheta_{nb} + \omega_{ib,y} \sin \vartheta_{nb} \sin \varphi_{nb} + \omega_{ib,z} \cos \vartheta_{nb} \sin \vartheta_{nb}\right) \in [-\pi, \pi) \end{aligned} \quad (C.49)$$

It is remarkable that only the entries of the measured angular rate vector $\boldsymbol{\omega}_{ib}$ are required, but not the Earth rate $\boldsymbol{\omega}_{ie}$.

C.2.2 IMU Error Cross-Covariances

If the IMU turn-on biases, scale factor errors and misalignments shall be estimated or at least considered by the navigation error filter, the cross-covariances between the initial orientation error $\boldsymbol{\Psi}_{n\bar{n}}$ and these errors are required rather than the cross-covariances with the overall IMU errors (4.39) and (4.40). Return to mind that the acceleration error $\delta \mathbf{f}_b$ and the angular rate error $\delta \boldsymbol{\omega}_{ib}$ are described in section 3.2.2 by (3.42) and (3.47), respectively. The cross-covariances between the orientation error $\boldsymbol{\Psi}_{n\bar{n}}$ and the turn-on biases \mathbf{b}_0 , scale factor errors \mathbf{s}_0 and misalignments \mathbf{m} of the accelerometer and gyroscope are given with

$$\begin{aligned} \mathbb{E}[\boldsymbol{\Psi}_{n\bar{n}} \mathbf{b}_{0,f}^\top] &= \mathbf{H}_f \mathbb{E}[\mathbf{b}_{0,f} \mathbf{b}_{0,f}^\top] & \mathbb{E}[\boldsymbol{\Psi}_{n\bar{n}} \mathbf{b}_{0,\omega}^\top] &= \mathbf{H}_\omega \mathbb{E}[\mathbf{b}_{0,\omega} \mathbf{b}_{0,\omega}^\top] \\ \mathbb{E}[\boldsymbol{\Psi}_{n\bar{n}} \mathbf{s}_{0,f}^\top] &= \mathbf{H}_f \text{diag}(\bar{\mathbf{f}}_b) \mathbb{E}[\mathbf{s}_{0,f} \mathbf{s}_{0,f}^\top] & \mathbb{E}[\boldsymbol{\Psi}_{n\bar{n}} \mathbf{s}_{0,\omega}^\top] &= \mathbf{H}_\omega \text{diag}(\bar{\boldsymbol{\omega}}_{ib}) \mathbb{E}[\mathbf{s}_{0,\omega} \mathbf{s}_{0,\omega}^\top] \\ \mathbb{E}[\boldsymbol{\Psi}_{n\bar{n}} \mathbf{m}_f^\top] &= \mathbf{H}_f \mathbf{F}_{M_f} \mathbb{E}[\mathbf{m}_f \mathbf{m}_f^\top] & \mathbb{E}[\boldsymbol{\Psi}_{n\bar{n}} \mathbf{m}_\omega^\top] &= \mathbf{H}_\omega \mathbf{F}_{M_\omega} \mathbb{E}[\mathbf{m}_\omega \mathbf{m}_\omega^\top] \end{aligned} \quad (C.50)$$

C.2.3 Recursive Averaging

The influence of the accelerometer and gyroscope measurement noise on the orientation angles is reduced by averaging a series of recorded measurement values before doing the stationary initialization. Recursive averaging is appropriate for real-time navigation systems since the mean values and the corresponding variances are immediately updated as soon as a new measurement is available. The algorithm can decide online to stop the averaging when the continuously updated variances of the accelerometer and gyroscope measurements drop below predefined thresholds. The mean value $\bar{\mathbf{f}}_{b,k-1}$ is updated by the new measurement $\tilde{\mathbf{f}}_{b,k}$ with

$$\bar{\mathbf{f}}_{b,k} = \frac{k-1}{k} \bar{\mathbf{f}}_{b,k-1} + \frac{1}{k} \tilde{\mathbf{f}}_{b,k} \quad (\text{C.51})$$

where k is the number of collected measurements until now. The mean value is initialized by $\bar{\mathbf{f}}_{b,0} = \mathbf{0}$. The covariance $E[\delta\bar{\mathbf{f}}_{b,k} \delta\bar{\mathbf{f}}_{b,k}^T]$ continuously decreases with each new measurement sample by

$$\begin{aligned} E[\delta\bar{\mathbf{f}}_{b,k} \delta\bar{\mathbf{f}}_{b,k}^T] &= \left(\frac{k-1}{k}\right)^2 E[\delta\bar{\mathbf{f}}_{b,k-1} \delta\bar{\mathbf{f}}_{b,k-1}^T] \\ &\quad + \frac{k-1}{k^2} \left(E[\delta\bar{\mathbf{f}}_{b,k-1} \delta\mathbf{f}_{b,k}^T] + E[\delta\mathbf{f}_{b,k} \delta\bar{\mathbf{f}}_{b,k-1}^T] \right) + \frac{1}{k^2} E[\delta\mathbf{f}_{b,k} \delta\mathbf{f}_{b,k}^T] \end{aligned} \quad (\text{C.52})$$

Looking at the acceleration error model (3.42) and the angular rate error model (3.47), the recursive averaging becomes

$$\begin{aligned} E[\delta\bar{\mathbf{f}}_{b,k} \delta\bar{\mathbf{f}}_{b,k}^T] &= \left(\frac{k-1}{k}\right)^2 E[\delta\bar{\mathbf{f}}_{b,k-1} \delta\bar{\mathbf{f}}_{b,k-1}^T] \\ &\quad + \frac{1}{k^2} \left(\sum_{i \in \{N, B\}} \mathbf{Q}_{b_i f} + E[\mathbf{b}_{0,f} \mathbf{b}_{0,f}^T] + \text{diag}(\tilde{\mathbf{f}}_{b,k}) E[\mathbf{s}_{0,f} \mathbf{s}_{0,f}^T] \text{diag}(\tilde{\mathbf{f}}_{b,k}) + \mathbf{F}_{M_f} E[\mathbf{m}_f \mathbf{m}_f^T] \mathbf{F}_{M_f}^T \right) \\ &\quad + \frac{k-1}{k^2} \left(E[\mathbf{b}_{0,f} \delta\bar{\mathbf{f}}_{b,k-1}^T] + \text{diag}(\tilde{\mathbf{f}}_{b,k}) E[\mathbf{s}_{0,f} \delta\bar{\mathbf{f}}_{b,k-1}^T] + \mathbf{F}_{M_f} E[\mathbf{m}_f \delta\bar{\mathbf{f}}_{b,k-1}^T] \right. \\ &\quad \left. + E[\delta\bar{\mathbf{f}}_{b,k-1} \mathbf{b}_{0,f}^T] + E[\delta\bar{\mathbf{f}}_{b,k-1} \mathbf{s}_{0,f}^T] \text{diag}(\tilde{\mathbf{f}}_{b,k}) + E[\delta\bar{\mathbf{f}}_{b,k-1} \mathbf{m}_f^T] \mathbf{F}_{M_f}^T \right), \quad E[\delta\bar{\mathbf{f}}_{b,0} \delta\bar{\mathbf{f}}_{b,0}^T] = \mathbf{0} \end{aligned} \quad (\text{C.53})$$

neglecting the bias and scale factor temperature sensitivities as well as the bias instability. The recursive cross-covariance propagation equations between averaged accelerometer measurement and single random constant accelerometer errors are given with

$$\begin{aligned} E[\delta\bar{\mathbf{f}}_{b,k} \mathbf{b}_{0,f}^T] &= \frac{k-1}{k} E[\delta\bar{\mathbf{f}}_{b,k-1} \mathbf{b}_{0,f}^T] + \frac{1}{k} E[\mathbf{b}_{0,f} \mathbf{b}_{0,f}^T], & E[\delta\bar{\mathbf{f}}_{b,0} \mathbf{b}_{0,f}^T] &= \mathbf{0} \\ E[\delta\bar{\mathbf{f}}_{b,k} \mathbf{s}_{0,f}^T] &= \frac{k-1}{k} E[\delta\bar{\mathbf{f}}_{b,k-1} \mathbf{s}_{0,f}^T] + \frac{1}{k} \text{diag}(\tilde{\mathbf{f}}_{b,k}) E[\mathbf{s}_{0,f} \mathbf{s}_{0,f}^T], & E[\delta\bar{\mathbf{f}}_{b,0} \mathbf{s}_{0,f}^T] &= \mathbf{0} \\ E[\delta\bar{\mathbf{f}}_{b,k} \mathbf{m}_f^T] &= \frac{k-1}{k} E[\delta\bar{\mathbf{f}}_{b,k-1} \mathbf{m}_f^T] + \frac{1}{k} \mathbf{F}_{M_f} E[\mathbf{m}_f \mathbf{m}_f^T], & E[\delta\bar{\mathbf{f}}_{b,0} \mathbf{m}_f^T] &= \mathbf{0} \end{aligned} \quad (\text{C.54})$$

The same equations can be applied for the recursive averaging of the angular rate measurements.

C.3 Out-of-Sequence Measurements

C.3.1 State Backward Propagation

To begin with, the true state propagation between two subsequent time steps t_{k-1} and t_k is described by the discrete-time propagation equation

$$\mathbf{z}_k = \Phi_{k-1} \mathbf{z}_{k-1} + \Gamma_{k-1} \boldsymbol{\omega}_{k-1} \quad (\text{C.55})$$

with $\boldsymbol{\omega}_k \sim WN(0, \mathbf{Q}_k)$. The true state propagation between time t_j and time t_k is

$$\mathbf{z}_k = \underbrace{\left(\prod_{i=j}^{k-1} \Phi_i \right)}_{=: \Phi_{kj}} \mathbf{z}_j + \sum_{l=j}^{k-1} \left(\prod_{i=l+1}^{k-1} \Phi_i \right) \Gamma_l \boldsymbol{\omega}_l \quad (\text{C.56})$$

The true state \mathbf{z}_k that is back-propagated to time t_j is then

$$\mathbf{z}_j = \Phi_{kj}^{-1} \left(\mathbf{z}_k - \sum_{l=j}^{k-1} \Phi_{k(l+1)} \Gamma_l \boldsymbol{\omega}_l \right) \quad (\text{C.57})$$

Analogously, the back-propagated estimated state $\hat{\mathbf{z}}_{j,k}$ that is obtained from the current a priori state estimate $\hat{\mathbf{z}}_k^-$ is given with

$$\hat{\mathbf{z}}_{j,k} = \Phi_{kj}^{-1} \hat{\mathbf{z}}_k^- \quad (\text{C.58})$$

The estimated state error $\delta \mathbf{z}_{j,k} = \mathbf{z}_j - \hat{\mathbf{z}}_{j,k}$ is calculated by subtracting (C.57) and (C.58)

$$\delta \mathbf{z}_{j,k} = \Phi_{kj}^{-1} \left(\delta \mathbf{z}_k^- - \sum_{l=j}^{k-1} \Phi_{k(l+1)} \Gamma_l \boldsymbol{\omega}_l \right) \quad (\text{C.59})$$

The covariance matrix $\mathbf{P}_{j,k}$ is hence

$$\begin{aligned} \mathbf{P}_{j,k} &= E[\delta \mathbf{z}_{j,k} \delta \mathbf{z}_{j,k}^T] \\ &= \Phi_{kj}^{-1} \left(\mathbf{P}_k^- + \sum_{l=j}^{k-1} \Phi_{k(l+1)} \Gamma_l \mathbf{Q}_l \Gamma_l^T \Phi_{k(l+1)}^T - \sum_{l=j}^{k-1} \Phi_{k(l+1)} \Gamma_l E[\boldsymbol{\omega}_l \delta \mathbf{z}_k^{-T}] - \sum_{l=j}^{k-1} E[\delta \mathbf{z}_k^- \boldsymbol{\omega}_l^T] \Gamma_l^T \Phi_{k(l+1)}^T \right) \Phi_{kj}^{-T} \end{aligned} \quad (\text{C.60})$$

In the period between the measurement time t_j and the arrival time t_k measurements from other sensors are processed. The cross-covariance between the process noise $\boldsymbol{\omega}_l$ and the state estimate error $\delta \mathbf{z}_k^-$ is thus given with

$$E[\delta \mathbf{z}_k^- \boldsymbol{\omega}_l^T] = \prod_{i=l+1}^{k-1} [\Phi_i (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i)] \Gamma_l \mathbf{Q}_l \quad (\text{C.61})$$

It is a reasonable assumption that in most situations the process noise covariance \mathbf{Q}_k does not change much in the waiting time between the measurement validity time t_j and arrival time t_k , meaning $\mathbf{Q} = \mathbf{const} = \mathbf{Q}_j$ for $t_j \leq t \leq t_k$. Then, without loss of generality, the covariance matrix $\mathbf{P}_{j,k}$ simplifies to

$$\mathbf{P}_{j,k} = \Phi_{kj}^{-1} \left(\mathbf{P}_k^- + \Gamma_{kj} \mathbf{Q}_j \Gamma_{kj}^T - \Gamma_{kj} \mathbf{Q}_j \bar{\Gamma}_{kj}^T - \bar{\Gamma}_{kj} \mathbf{Q}_j \Gamma_{kj}^T \right) \Phi_{kj}^{-T} \quad (\text{C.62})$$

with the definitions of the overall input matrices

$$\begin{aligned}\boldsymbol{\Gamma}_{kj} &:= \sum_{l=j}^{k-1} \left(\prod_{i=l+1}^{k-1} \boldsymbol{\Phi}_i \right) \boldsymbol{\Gamma}_l \\ \bar{\boldsymbol{\Gamma}}_{kj} &:= \sum_{l=j}^{k-1} \left(\prod_{i=l+1}^{k-1} [\boldsymbol{\Phi}_i (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i)] \right) \boldsymbol{\Gamma}_l\end{aligned}\quad (C.63)$$

Finally, the cross-covariance $\mathbf{P}_{jk}^- = \mathbb{E}[\delta \mathbf{z}_{j,k} \delta \mathbf{z}_k^{-T}]$ is derived using (C.59), (C.61) and (C.63)

$$\begin{aligned}\mathbf{P}_{jk}^- &= \mathbb{E} \left[\boldsymbol{\Phi}_{kj}^{-1} \left(\delta \mathbf{z}_k^- - \sum_{l=j}^{k-1} \boldsymbol{\Phi}_{k(l+1)} \boldsymbol{\Gamma}_l \boldsymbol{\omega}_l \right) \delta \mathbf{z}_k^{-T} \right] \\ &= \boldsymbol{\Phi}_{kj}^{-1} (\mathbf{P}_k^- - \boldsymbol{\Gamma}_{kj} \mathbf{Q}_j \bar{\boldsymbol{\Gamma}}_{kj}^T)\end{aligned}\quad (C.64)$$

With the covariance propagation from time t_j to time t_k written as

$$\mathbf{P}_k^- = \bar{\boldsymbol{\Phi}}_{kj} \mathbf{P}_j^- \boldsymbol{\Phi}_{kj}^T + \bar{\boldsymbol{\Gamma}}_{kj} \mathbf{Q}_j \boldsymbol{\Gamma}_{kj}^T \quad (C.65)$$

and with the definition of the transition matrix containing interim updates

$$\bar{\boldsymbol{\Phi}}_{kj} := \sum_{l=j}^{k-1} \left(\prod_{i=l}^{k-1} [\boldsymbol{\Phi}_i (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i)] \right) \quad (C.66)$$

the cross-covariance \mathbf{P}_{jk}^- can be written alternatively to (C.64) as

$$\begin{aligned}\mathbf{P}_{jk}^- &= \boldsymbol{\Phi}_{kj}^{-1} (\bar{\boldsymbol{\Phi}}_{kj} \mathbf{P}_j^- \boldsymbol{\Phi}_{kj}^T + \bar{\boldsymbol{\Gamma}}_{kj} \mathbf{Q}_j \boldsymbol{\Gamma}_{kj}^T - \boldsymbol{\Gamma}_{kj} \mathbf{Q}_j \bar{\boldsymbol{\Gamma}}_{kj}^T) && \text{due to symmetry} \\ &= \boldsymbol{\Phi}_{kj}^{-1} \bar{\boldsymbol{\Phi}}_{kj} \mathbf{P}_j^- \boldsymbol{\Phi}_{kj}^T && \text{due to symmetry} \\ &= \boldsymbol{\Phi}_{kj}^{-1} \bar{\boldsymbol{\Phi}}_{kj} \mathbf{P}_j^- \bar{\boldsymbol{\Phi}}_{kj}^T \\ &= \mathbf{P}_j^- \bar{\boldsymbol{\Phi}}_{kj}^T\end{aligned}\quad (C.67)$$

With (C.58), (C.62) and (C.64) or (C.67) the required inputs $\hat{\mathbf{z}}_{j,k}$, $\mathbf{P}_{j,k}$ and \mathbf{P}_{jk}^- for the Kalman filter update for delayed measurements according to Template 4-14 are readily available.

Their calculation in the instant when the measurement value arrives would mean a high computational burden. However, if the sensor notifies the filter of a new measurement before the measurement value is actually available to the filter, the required matrices \mathbf{P}_{jk}^- , $\boldsymbol{\Phi}_{kj}$, $\boldsymbol{\Gamma}_{kj}$ and $\bar{\boldsymbol{\Gamma}}_{kj}$ can be already computed in parallel looking forward to the arrival of the measurement. The computing time is thus better spread and reduced in the moment of the actual measurement processing.

As soon as the sensor indicates a new measurement or the navigation computer actively polls a measurement from a sensor, the cross-covariance $\mathbf{P}_{jj}^- = \mathbf{P}_j^-$ can be created and consequently propagated and updated, parallel to the regular Kalman filter steps, when other, newer measurements arrive. Later, at time t_k , when the measurement $\tilde{\mathbf{y}}_j$ is available in the navigation computer, \mathbf{P}_{kj}^- can be readily used to compute the Kalman gain and update the state estimate covariance matrix. The propagation step between two subsequent time steps is carried out with

$$\mathbf{P}_{(i+1)j}^- = \boldsymbol{\Phi}_i \mathbf{P}_{ij}^- \quad (C.68)$$

The interim update of the cross-covariance matrix with newer measurement at time t_i is analog to the usual state covariance update

$$\mathbf{P}_{ij}^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^- \quad (C.69)$$

A final remark about efficiency: since $\mathbf{P}_{kj}^- \mathbf{H}_j^T$ is required in the Kalman gain equation and its transpose in the covariance update equation in Template 4-14, it suffices to propagate and update only $\mathbf{P}_{kj}^- \mathbf{H}_j^T$ which is more efficient because of the reduced dimension of $(n \times m)$ instead of $(n \times n)$. Therein, n is the dimension of state vector and m the dimension of the measurement vector. In the navigation application the dimension of the measurement vector m is considerably smaller than the dimension of the state vector n . In a navigation error filter the state vector is usually of size $n = 15$ to $n = 30$, whereas, for example, a three-dimensional position measurement vector is of size $m = 3$.

The method is summarized in Template 4-15.

C.3.2 Measurement Prediction

In the following, a sketch of the derivation in [107] is given. The measurement $\tilde{\mathbf{y}}_j$ is extrapolated to the current time t_k using the current state estimate $\hat{\mathbf{z}}_k^-$ and the state estimate $\hat{\mathbf{z}}_j^-$ of time t_j . The extrapolated measurement $\tilde{\mathbf{y}}_k^*$ is

$$\tilde{\mathbf{y}}_k^* = \tilde{\mathbf{y}}_j + \mathbf{H}_k \hat{\mathbf{z}}_k^- - \mathbf{H}_j \hat{\mathbf{z}}_j^- \quad (\text{C.70})$$

Inserting the observation equation (4.113) and using the definitions of the state estimate errors $\delta\mathbf{z}_j^- = \mathbf{z}_j - \hat{\mathbf{z}}_j^-$ as well as $\delta\mathbf{z}_k^- = \mathbf{z}_k - \hat{\mathbf{z}}_k^-$, the extrapolated measurement $\tilde{\mathbf{y}}_k^*$ can also be expressed by

$$\tilde{\mathbf{y}}_k^* = \mathbf{H}_k \mathbf{z}_k + \mathbf{H}_j \delta\mathbf{z}_j^- - \mathbf{H}_k \delta\mathbf{z}_k^- + \mathbf{v}_{\text{f}} \quad (\text{C.71})$$

Identifying the effective noise of the extrapolated measurement as $\mathbf{v}_k^* = \mathbf{H}_j \delta\mathbf{z}_j^- - \mathbf{H}_k \delta\mathbf{z}_k^- + \mathbf{v}_{\text{f}}$, an observation equation similar to (4.113) is obtained

$$\tilde{\mathbf{y}}_k^* = \mathbf{H}_k \mathbf{z}_k + \mathbf{v}_k^* \quad (\text{C.72})$$

The innovation $\mathbf{s}_{j,k}$ of the extrapolated measurement is then

$$\mathbf{s}_{j,k} = \tilde{\mathbf{y}}_k^* - \mathbf{H}_k \hat{\mathbf{z}}_k^- \quad (\text{C.73})$$

and with (C.70) the innovation becomes

$$\begin{aligned} \mathbf{s}_{j,k} &= \tilde{\mathbf{y}}_j + \cancel{\mathbf{H}_k \hat{\mathbf{z}}_k^-} - \cancel{\mathbf{H}_j \hat{\mathbf{z}}_j^-} - \cancel{\mathbf{H}_k \hat{\mathbf{z}}_j^-} \\ &= \tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_j^- \end{aligned} \quad (\text{C.74})$$

The result is remarkable because the innovation of the predicted measurement $\mathbf{s}_{j,k}$ is simply the innovation \mathbf{s}_j that would have been valid at time t_j when the measurement $\tilde{\mathbf{y}}_j$ would have been immediately available. The measurement prediction method obviously supposes that the later corrections of the newer measurements of the other sensors and the dynamic behavior of the state estimation error only have minor influence on the innovation. This assumption is certainly justified if the Kalman filter is more or less in steady state. Then, the innovation formed with the old state and the innovation formed with the newest state incorporating all available information up to now do not differ much. It is clear that this method is not optimal and not statistically correct because the state information content is not correctly assumed when forming the innovation. From this point of view, it would probably have been better to name the prediction measurement method as “history state” method in dependence on the “history state update” method that is derived in the next section.

Comparing the innovation of the Kalman filter for delayed measurements in Template 4-14 with the innovation of the measurement prediction (C.74), it can be seen that the involved state $\hat{\mathbf{z}}_{j,k}$ equals the

state estimate $\hat{\mathbf{z}}_j^-$. Correspondingly, the covariance $\mathbf{P}_{j,k}$ is just the covariance \mathbf{P}_j^- . Between two subsequent time steps t_{k-1} and t_k the true state \mathbf{z}_k , the state estimate $\hat{\mathbf{z}}_k$ and the state estimate error $\delta\mathbf{z}_k$ propagate according

$$\begin{aligned}\mathbf{z}_k &= \Phi_{k-1} \mathbf{z}_{k-1} + \Gamma_{k-1} \boldsymbol{\omega}_{k-1} \\ \hat{\mathbf{z}}_k^- &= \Phi_{k-1} \hat{\mathbf{z}}_{k-1}^+ \\ \delta\mathbf{z}_k^- &= \Phi_{k-1} \delta\mathbf{z}_{k-1}^+ + \Gamma_{k-1} \boldsymbol{\omega}_{k-1}\end{aligned}\quad (\text{C.75})$$

When measurements from other sensors come in, $\hat{\mathbf{z}}_k^-$ and $\delta\mathbf{z}_k^-$ are updated, taking (4.113) into account, by

$$\begin{aligned}\hat{\mathbf{z}}_k^+ &= \hat{\mathbf{z}}_k^- + \mathbf{K}_k (\mathbf{H}_k \delta\mathbf{z}_k^- + \mathbf{v}_k) \\ \delta\mathbf{z}_k^+ &= (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \delta\mathbf{z}_k^- - \mathbf{K}_k \mathbf{v}_k\end{aligned}\quad (\text{C.76})$$

Combining all propagation and update steps between t_j and t_k yields the overall evolution of the state estimate error $\delta\mathbf{z}_j^-$ towards $\delta\mathbf{z}_k^-$

$$\delta\mathbf{z}_k^- = \bar{\Phi}_{kj} \delta\mathbf{z}_j^- - \sum_{i=j+1}^{k-1} \bar{\Phi}_{ki} \mathbf{K}_i \mathbf{v}_i + \sum_{i=j}^{k-1} \bar{\Gamma}_{ki} \boldsymbol{\omega}_i \quad (\text{C.77})$$

With (C.77) the estimated state error cross-covariance $\mathbf{P}_{kj}^- = \mathbb{E}[\delta\mathbf{z}_k^- \delta\mathbf{z}_j^{-T}]$ is computed straightforwardly

$$\begin{aligned}\mathbf{P}_{kj}^- &= \mathbb{E}\left[\left(\bar{\Phi}_{kj} \delta\mathbf{z}_j^- - \sum_{i=j+1}^{k-1} \bar{\Phi}_{ki} \mathbf{K}_i \mathbf{v}_i + \sum_{i=j}^{k-1} \bar{\Gamma}_{ki} \boldsymbol{\omega}_i\right) \delta\mathbf{z}_j^{-T}\right] \\ &= \bar{\Phi}_{kj} \mathbf{P}_j^- - \sum_{i=j+1}^{k-1} \bar{\Phi}_{ki} \mathbf{K}_i \mathbb{E}[\mathbf{v}_i \delta\mathbf{z}_j^{-T}] + \sum_{i=j}^{k-1} \bar{\Gamma}_{ki} \mathbb{E}[\boldsymbol{\omega}_i \delta\mathbf{z}_j^{-T}]\end{aligned}\quad (\text{C.78})$$

The noise \mathbf{v}_i of the other measurements that are newer than the measurement $\tilde{\mathbf{y}}_j$ is not correlated with the state estimate $\hat{\mathbf{z}}_j^-$ and thus also not with the previous state estimate error $\delta\mathbf{z}_j^-$, that is $\mathbb{E}[\mathbf{v}_i \delta\mathbf{z}_j^{-T}] = \mathbf{0}$. The same is valid for the process noise $\boldsymbol{\omega}_i$. The process noise occurring later than the measurement time t_j is not correlated with the state estimate and the error, meaning $\mathbb{E}[\boldsymbol{\omega}_i \delta\mathbf{z}_j^{-T}] = \mathbf{0}$. The cross-covariance \mathbf{P}_{kj}^- between the state error $\delta\mathbf{z}_j^-$ and the state error $\delta\mathbf{z}_k^-$ abbreviates to

$$\mathbf{P}_{kj}^- = \bar{\Phi}_{kj} \mathbf{P}_j^- \quad (\text{C.79})$$

As in the state backward propagation method, the cross-covariance \mathbf{P}_{kj}^- can be calculated in parallel if the sensor indicates a new measurement to the navigation computer.

The computation of $\mathbf{P}_{kj}^- \mathbf{H}_j^T$ with the measurement prediction method for the case when the measurement validity time is already known is recapitulated in Template 4-16. The final update when the measurement value is available is done with the Kalman filter update step for delayed measurements according to Template 4-14.

C.3.3 History State Update

At interim times $t = t_i$ the augmented state vector (4.129) is propagated by

$$\begin{pmatrix} \hat{\mathbf{z}}_i^- \\ \hat{\mathbf{z}}_{j,i}^- \end{pmatrix} = \begin{pmatrix} \Phi_{i-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix} \begin{pmatrix} \hat{\mathbf{z}}_{i-1}^+ \\ \hat{\mathbf{z}}_{j,i-1}^- \end{pmatrix} \quad (\text{C.80})$$

Therein, only the current state estimate $\hat{\mathbf{z}}_{i-1}^+$ is propagated with the system matrix Φ_{i-1} . The history state $\hat{\mathbf{z}}_{j,i-1}^-$ is not modified. Thus, the distinction between the a priori history state $\hat{\mathbf{z}}_{j,i}^-$ and the posterior history state $\hat{\mathbf{z}}_{j,i}^+$ is omitted in the following. The covariance matrix is analogously propagated

$$\begin{pmatrix} \mathbf{P}_i^- & \mathbf{P}_{ji}^- \\ \mathbf{P}_{ji}^- & \mathbf{P}_{j,i}^- \end{pmatrix} = \begin{pmatrix} \Phi_{i-1} & 0 \\ 0 & \mathbf{I}_n \end{pmatrix} \begin{pmatrix} \mathbf{P}_{i-1}^+ & \mathbf{P}_{(i-1)j}^+ \\ \mathbf{P}_{j(i-1)}^+ & \mathbf{P}_{j,i-1}^+ \end{pmatrix} \begin{pmatrix} \Phi_{i-1}^\top & 0 \\ 0 & \mathbf{I}_n \end{pmatrix} + \begin{pmatrix} \Gamma_{i-1} \\ 0 \end{pmatrix} \mathbf{Q}_{i-1} \begin{pmatrix} \Gamma_{i-1}^\top & 0 \end{pmatrix} \quad (\text{C.81})$$

Since the history state covariance $\mathbf{P}_{j,i}^-$ is also not modified by the propagation, the minus and plus superscripts are left, too. When a measurement of another sensor arrives in the interim time period, the innovation is calculated according to the CKF

$$\begin{aligned} \mathbf{s}_i &= \tilde{\mathbf{y}}_i - (\mathbf{H}_i \quad 0) \begin{pmatrix} \hat{\mathbf{z}}_i^- \\ \hat{\mathbf{z}}_{j,i-1}^- \end{pmatrix} \\ \mathbf{S}_i &= \mathbf{R}_i + (\mathbf{H}_i \quad 0) \begin{pmatrix} \mathbf{P}_i^- & \mathbf{P}_{ji}^- \\ \mathbf{P}_{ji}^- & \mathbf{P}_{j,i-1}^- \end{pmatrix} \begin{pmatrix} \mathbf{H}_i^\top \\ 0 \end{pmatrix} \end{aligned} \quad (\text{C.82})$$

The Kalman gain is straightforwardly

$$\begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix} = \begin{pmatrix} \mathbf{P}_i^- & \mathbf{P}_{ji}^- \\ \mathbf{P}_{ji}^- & \mathbf{P}_{j,i-1}^- \end{pmatrix} \begin{pmatrix} \mathbf{H}_i^\top \\ 0 \end{pmatrix} \mathbf{S}_i^{-1} \quad (\text{C.83})$$

and the update of the augmented state vector and the covariance matrix, using the short form of the covariance update equation, is given by

$$\begin{aligned} \begin{pmatrix} \hat{\mathbf{z}}_i^+ \\ \hat{\mathbf{z}}_{j,i} \end{pmatrix} &= \begin{pmatrix} \hat{\mathbf{z}}_i^- \\ \hat{\mathbf{z}}_{j,i-1}^- \end{pmatrix} + \begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix} \mathbf{s}_i \\ \begin{pmatrix} \mathbf{P}_i^+ & \mathbf{P}_{ji}^+ \\ \mathbf{P}_{ji}^+ & \mathbf{P}_{j,i}^+ \end{pmatrix} &= \left(\begin{pmatrix} \mathbf{I}_n & 0 \\ 0 & \mathbf{I}_n \end{pmatrix} - \begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix} (\mathbf{H}_i \quad 0) \right) \begin{pmatrix} \mathbf{P}_i^- & \mathbf{P}_{ji}^- \\ \mathbf{P}_{ji}^- & \mathbf{P}_{j,i-1}^- \end{pmatrix} \end{aligned} \quad (\text{C.84})$$

The long, symmetry preserving, Joseph form of the covariance update equation is

$$\begin{aligned} \begin{pmatrix} \mathbf{P}_i^+ & \mathbf{P}_{ji}^+ \\ \mathbf{P}_{ji}^+ & \mathbf{P}_{j,i}^+ \end{pmatrix} &= \left(\begin{pmatrix} \mathbf{I}_n & 0 \\ 0 & \mathbf{I}_n \end{pmatrix} - \begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix} (\mathbf{H}_i \quad 0) \right) \begin{pmatrix} \mathbf{P}_i^- & \mathbf{P}_{ji}^- \\ \mathbf{P}_{ji}^- & \mathbf{P}_{j,i-1}^- \end{pmatrix} \left(\begin{pmatrix} \mathbf{I}_n & 0 \\ 0 & \mathbf{I}_n \end{pmatrix} - \begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix} (\mathbf{H}_i \quad 0) \right)^\top \\ &\quad + \begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix} \mathbf{R}_i \begin{pmatrix} \mathbf{K}_i \\ \mathbf{K}_i^* \end{pmatrix}^\top \end{aligned} \quad (\text{C.85})$$

As soon as the longingly awaited measurement value, which has initially provoked the state augmentation, comes in at the current time t_k , the innovation can be computed by

$$\begin{aligned} \mathbf{s}_k &= \tilde{\mathbf{y}}_j - (0 \quad \mathbf{H}_j) \begin{pmatrix} \hat{\mathbf{z}}_k^- \\ \hat{\mathbf{z}}_{j,k-1}^- \end{pmatrix} \\ \mathbf{S}_k &= \mathbf{R}_j + (0 \quad \mathbf{H}_j) \begin{pmatrix} \mathbf{P}_k^- & \mathbf{P}_{kj}^- \\ \mathbf{P}_{jk}^- & \mathbf{P}_{j,k-1}^- \end{pmatrix} \begin{pmatrix} 0 \\ \mathbf{H}_j^\top \end{pmatrix} \end{aligned} \quad (\text{C.86})$$

It can be easily seen that the continuously updated history state estimate $\hat{\mathbf{z}}_{j,k}$ and covariance matrix $\mathbf{P}_{j,k}$ can now be directly used for the innovation. The Kalman gain is consequently

$$\begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix} = \begin{pmatrix} \mathbf{P}_k^- & \mathbf{P}_{kj}^- \\ \mathbf{P}_{jk}^- & \mathbf{P}_{j,k-1}^- \end{pmatrix} \begin{pmatrix} 0 \\ \mathbf{H}_j^\top \end{pmatrix} \mathbf{S}_k^{-1} \quad (\text{C.87})$$

The final update of the state estimate and covariance is

$$\begin{aligned} \begin{pmatrix} \hat{\mathbf{z}}_k^+ \\ \hat{\mathbf{z}}_{j,k} \end{pmatrix} &= \begin{pmatrix} \hat{\mathbf{z}}_k^- \\ \hat{\mathbf{z}}_{j,k-1} \end{pmatrix} + \begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix} \mathbf{s}_k \\ \begin{pmatrix} \mathbf{P}_k^+ & \mathbf{P}_{kj}^+ \\ \mathbf{P}_{jk}^+ & \mathbf{P}_{j,k} \end{pmatrix} &= \left(\begin{pmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix} - \begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix} \begin{pmatrix} \mathbf{0} & \mathbf{H}_j \end{pmatrix} \right) \begin{pmatrix} \mathbf{P}_k^- & \mathbf{P}_{kj}^- \\ \mathbf{P}_{jk}^- & \mathbf{P}_{j,k-1}^- \end{pmatrix} \end{aligned} \quad (\text{C.88})$$

or

$$\begin{aligned} \begin{pmatrix} \mathbf{P}_k^+ & \mathbf{P}_{kj}^+ \\ \mathbf{P}_{jk}^+ & \mathbf{P}_{j,k} \end{pmatrix} &= \left(\begin{pmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix} - \begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix} \begin{pmatrix} \mathbf{0} & \mathbf{H}_j \end{pmatrix} \right) \begin{pmatrix} \mathbf{P}_k^- & \mathbf{P}_{kj}^- \\ \mathbf{P}_{jk}^- & \mathbf{P}_{j,k-1}^- \end{pmatrix} \left(\begin{pmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix} - \begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix} \begin{pmatrix} \mathbf{0} & \mathbf{H}_j \end{pmatrix} \right)^T \\ &\quad + \begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix} \mathbf{R}_j \begin{pmatrix} \mathbf{K}_k \\ \mathbf{K}_k^* \end{pmatrix}^T \end{aligned} \quad (\text{C.89})$$

respectively. Obviously, the history state $\hat{\mathbf{z}}_{j,k}$, the history covariance $\mathbf{P}_{j,k}$ and the cross-covariance \mathbf{P}_{kj}^+ are no longer required after the update. Only the state $\hat{\mathbf{z}}_k^+$ and covariance \mathbf{P}_k^+ are further processed. The augmented states and covariance entries are thus discarded and the original form of the state vector and covariance matrix is restored. When the next delayed measurement is announced by the sensor, the described procedure starts again. Since the augmented state and covariance are not used anymore, it does not make much sense to update them in (C.88). The computational effort can be reduced by expanding the block matrix representation and then deleting the non-relevant equations

$$\begin{aligned} \hat{\mathbf{z}}_k^+ &= \hat{\mathbf{z}}_k^- + \mathbf{K}_k \mathbf{s}_k \\ \mathbf{P}_k^+ &= \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^- \end{aligned} \quad (\text{C.90})$$

Joseph's form (C.89) becomes

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^- - \mathbf{P}_{kj}^- \mathbf{H}_j^T \mathbf{K}_k^T + \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T \quad (\text{C.91})$$

The same is done with the Kalman gain (C.87)

$$\mathbf{K}_k = \mathbf{P}_{kj}^- \mathbf{H}_j^T \mathbf{S}_k^{-1} \quad (\text{C.92})$$

and the innovation (C.86)

$$\begin{aligned} \mathbf{s}_k &= \tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k-1} \\ \mathbf{S}_k &= \mathbf{R}_j + \mathbf{H}_j \mathbf{P}_{j,k-1} \mathbf{H}_j^T \end{aligned} \quad (\text{C.93})$$

The propagation equations (C.80) and (C.81) are explicitly written in compact form

$$\begin{aligned} \hat{\mathbf{z}}_i^- &= \Phi_{i-1} \hat{\mathbf{z}}_{i-1}^+ \\ \mathbf{P}_i^- &= \Phi_{i-1} \mathbf{P}_{i-1}^+ \Phi_{i-1}^T + \Gamma_{i-1} \mathbf{Q}_{i-1} \Gamma_{i-1}^T \\ \mathbf{P}_{ij}^- &= \Phi_{i-1} \mathbf{P}_{(i-1)j}^+ \end{aligned} \quad (\text{C.94})$$

The expanded equations of the interim update for innovation (C.82), Kalman gain (C.83) and state and covariance update (C.84) and (C.85), respectively, can be split into those equations that accrue anyway, as in the CKF,

$$\begin{aligned}
\mathbf{s}_i &= \tilde{\mathbf{y}}_i - \mathbf{H}_i \hat{\mathbf{z}}_i^- \\
\mathbf{S}_i &= \mathbf{R}_i + \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^\top \\
\mathbf{K}_i &= \mathbf{P}_i^- \mathbf{H}_i^\top \mathbf{S}_i^{-1} \\
\hat{\mathbf{z}}_i^+ &= \hat{\mathbf{z}}_i^- + \mathbf{K}_i \mathbf{s}_i \\
\mathbf{P}_i^+ &= (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_i^- \quad \text{or} \quad \mathbf{P}_i^+ = (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_i^- (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i)^\top + \mathbf{K}_i \mathbf{R}_i \mathbf{K}_i^\top
\end{aligned} \tag{C.95}$$

and the additional equations that are required to update the history state $\hat{\mathbf{z}}_{j,i}$, the history covariance $\mathbf{P}_{j,i}$ and the corresponding cross-covariance \mathbf{P}_{ij}^+

$$\begin{aligned}
\mathbf{K}_i^* &= \mathbf{P}_{ji}^- \mathbf{H}_i^\top \mathbf{S}_i^{-1} \\
\hat{\mathbf{z}}_{j,i} &= \hat{\mathbf{z}}_{j,i-1} + \mathbf{K}_i^* \mathbf{s}_i \\
\mathbf{P}_{j,i} &= \mathbf{P}_{j,i-1} - \mathbf{K}_i^* \mathbf{H}_i \mathbf{P}_{ij}^- \\
\mathbf{P}_{ij}^+ &= (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^-
\end{aligned} \tag{C.96}$$

The longer forms of the covariance update equations are

$$\begin{aligned}
\mathbf{P}_{j,i} &= \mathbf{P}_{j,i-1} - \mathbf{K}_i^* \mathbf{H}_i \mathbf{P}_{ij}^- - \mathbf{P}_{ji}^- \mathbf{H}_i^\top \mathbf{K}_i^{*\top} + \mathbf{K}_i^* \mathbf{S}_i \mathbf{K}_i^{*\top} \\
\mathbf{P}_{ij}^+ &= (\mathbf{I}_n - \mathbf{K}_i \mathbf{H}_i) \mathbf{P}_{ij}^- - \mathbf{P}_i^- \mathbf{H}_i^\top \mathbf{K}_i^{*\top} + \mathbf{K}_i \mathbf{S}_i \mathbf{K}_i^{*\top}
\end{aligned} \tag{C.97}$$

The history state update method is summarized in Template 4-17. In contrast to the measurement prediction method of Template 4-16, the history state $\hat{\mathbf{z}}_{j,i}$ and covariance $\mathbf{P}_{j,i}$ are additionally updated at an interim update step.

C.4 Filter Augmentation by IMU Biases and Scale Factor Errors

In this section, it is exemplarily illustrated how to augment the core navigation error state vector by states for the accelerometer and gyroscope turn-on biases and scale factor errors to be estimated by the navigation filter and to be used for in-flight calibration of the IMU measurements.

The true, error-free acceleration \mathbf{f}_b and the accelerometer measurement error $\delta\mathbf{f}_b$ are given with

$$\begin{aligned}
\mathbf{f}_b &= \tilde{\mathbf{f}}_b + \tilde{\mathbf{b}}_{0,f} + \text{diag}(\tilde{\mathbf{f}}_b) \tilde{\mathbf{s}}_{0,f} + \delta\mathbf{f}_b \\
\delta\mathbf{f}_b &= \delta\mathbf{b}_{0,f} + \text{diag}(\tilde{\mathbf{f}}_b) \delta\mathbf{s}_{0,f} + \mathbf{b}_{N,f}
\end{aligned} \tag{C.98}$$

$\tilde{\mathbf{b}}_{0,f}$ and $\tilde{\mathbf{s}}_{0,f}$ are the current estimates of the turn-on biases and scale factor errors that are used for the in-flight calibration of the current IMU measurements. They deviate from the actual values $\mathbf{b}_{0,f}$ and $\mathbf{s}_{0,f}$ by residual errors $\delta\mathbf{b}_{0,f}$ and $\delta\mathbf{s}_{0,f}$ that have not been estimated so far. $\mathbf{b}_{N,f}$ is the white noise on the accelerometer measurements. Other errors are neglected to simplify matters. The model matrices for both the turn-on bias and scale factor error are

$$\Phi_{b_{0,f}/s_{0,f}} = \mathbf{I}_3, \quad \Gamma_{b_{0,f}/s_{0,f}} = 0, \quad \mathbf{H}_{b_{0,f}/s_{0,f}} = \mathbf{I}_3, \quad \mathbf{D}_{b_{0,f}/s_{0,f}} = 0 \tag{C.99}$$

The model matrices for the white noise are

$$\Phi_{b_{N,f}} = 0, \quad \Gamma_{b_{N,f}} = 0, \quad \mathbf{H}_{b_{N,f}} = 0, \quad \mathbf{D}_{b_{N,f}} = \mathbf{I}_3 \tag{C.100}$$

The gyroscope error models are analogous to the accelerometer error models. The navigation error model and the IMU error model are combined in the augmented system

$$\begin{pmatrix} \mathbf{z}_k \\ \mathbf{z}_{b_{0,f},k} \\ \mathbf{z}_{b_{0,\omega},k} \\ \mathbf{z}_{s_{0,f},k} \\ \mathbf{z}_{s_{0,\omega},k} \end{pmatrix} = \begin{pmatrix} \Phi_{k-1} & \Gamma_{f,k-1} & \Gamma_{\omega,k-1} & \Gamma_{f,k-1} \text{diag}(\tilde{\mathbf{f}}_{b,k-1}) & \Gamma_{\omega,k-1} \text{diag}(\tilde{\mathbf{\omega}}_{ib,k-1}) \\ 0 & \mathbf{I}_3 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{I}_3 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I}_3 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I}_3 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{z}_{k-1} \\ \mathbf{z}_{b_{0,f},k-1} \\ \mathbf{z}_{b_{0,\omega},k-1} \\ \mathbf{z}_{s_{0,f},k-1} \\ \mathbf{z}_{s_{0,\omega},k-1} \end{pmatrix} + \begin{pmatrix} \Gamma_{f,k-1} & \Gamma_{\omega,k-1} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{n}_{b_{N,f},k-1} \\ \mathbf{n}_{b_{N,\omega},k-1} \end{pmatrix} \quad (\text{C.101})$$

The input matrix of the navigation error model Γ_k has been split into the input of the accelerations $\Gamma_{f,k}$ and the input of the angular rates $\Gamma_{\omega,k}$ according to $\Gamma_k = (\Gamma_{f,k} \quad \Gamma_{\omega,k})$. The output equation is not matter of this example and can be considered separately.

C.5 Schmidt-Kalman Filter for Noise-Like Errors

C.5.1 Correspondence with Kalman Filter for State Cross-Correlated Measurements

The Schmidt-Kalman filter for time-correlated measurement noise corresponds to a Kalman filter that accounts for known cross-correlation between the navigation error state and the measurement error $\mathbf{P}_{zy,k} = E[\delta \mathbf{z}_k \delta \mathbf{y}_k^T]$ with the Kalman gain and covariance update equations as follows

$$\begin{aligned} \mathbf{K}_k &= (\mathbf{P}_{zz,k}^- \mathbf{H}_k^T - \mathbf{P}_{zy,k}) (\mathbf{H}_k \mathbf{P}_{zz,k}^- \mathbf{H}_k^T + \mathbf{R}_k - \mathbf{H}_k \mathbf{P}_{zy,k} - \mathbf{P}_{yz,k} \mathbf{H}_k^T)^{-1} \\ \mathbf{P}_{zz,k}^+ &= (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{zz,k}^- + \mathbf{K}_k \mathbf{P}_{yz,k} \end{aligned} \quad (\text{C.102})$$

Correlation between the current state estimate error and the current measurement can for example occur in an aircraft when high frequent aerodynamic disturbances similarly affect the accelerometer and gyroscope measurements due to unresolved vibrations and the angle of attack measurement.

Comparing (C.102) with the equations in Template 4-22 reveals that

$$\mathbf{P}_{zy,k} = - \sum_{i=1}^r (\mathbf{P}_{zv_i,k}^- \mathbf{H}_{v_i}^T) \quad (\text{C.103})$$

Therein, the different definition of the measurement error $\delta \mathbf{y}_k$ and the noise \mathbf{v}_k in

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \delta \mathbf{y}_k = \mathbf{H}_k \mathbf{z}_k + \mathbf{v}_k \quad (\text{C.104})$$

explains the negative sign in (C.103). This means that the time correlation of the measurement error leads to the gradual cross-correlation of the navigation state error and the measurement error.

C.5.2 Schmidt-Kalman Filter for Out-of-Sequence Measurements

In order to complete the discussion of delayed measurements of section 4.3, the question how to consider colored process and measurement noise shall be answered. The procedure is principally the same as in the history state update method. In the instant of the new measurement the state vector, which is augmented by the considered states, is extended by a copy of itself. In the sequel the Kalman gains of the considered states $\mathbf{K}_{\omega,i}$ and $\mathbf{K}_{v,i}$ as well as of the history considered states $\mathbf{K}_{\omega,i}^*$ and $\mathbf{K}_{v,i}^*$

are set to zero at interim updates. In Template C-2 the expanded equations that are additionally required to the Kalman filter equations for colored process and measurement noise are listed for the interim and final updates. For the process noise only the cross-covariance matrix between the current process noise states and the history state $\mathbf{P}_{\omega_s z_j}$ is required for the propagation of the covariance matrix between the current state and the history state \mathbf{P}_{k_j} . For the colored measurement noise three additional cross-covariance matrices have to be used: $\mathbf{P}_{zv_{t,j}}$, $\mathbf{P}_{v_t z_j}$ and $\mathbf{P}_{v_t v_{t,j}}$. These are the cross-covariance between the current state and the history considered measurement noise state, the cross-covariance between the current considered measurement noise state and the history state and the cross-covariance between the current and history considered measurement noise states. The distinction between prior and posterior matrices is omitted for those matrices that are only propagated but not corrected by the interim update.

It has to be noticed that interim updates are assumed to be other measurements than the delayed measurement with other, independent colored measurement noise models. Therefore, the measurement noise models that belong to the delayed measurement do not affect the interim update. Only the cross-covariances of the considered measurement noise states that are related to the current interim measurement have to be considered in the interim update equations. The same is valid for the final update: only the cross-covariances that are related to the delayed measurement are to be used in the update equations.

The z -subindex of the only remaining Kalman gain $\mathbf{K}_{z,k}$ is omitted for convenience, $\mathbf{K}_{z,k} = \mathbf{K}_k$ in the templates and in the following of this chapter.

In many cases the process noise is in steady-state, that is $\mathbf{P}_{\omega_s \omega_s} = \text{const.}$ The bias instability of accelerometer and gyroscope measurements are typically stationary process noise errors. The covariance has then not to be propagated. The same is valid for the measurement noise.

Template C-2: History state update method with colored process and measurement noise
Initialization (at measurement time $t = t_j$)

$\hat{\mathbf{z}}_{j,j} = \hat{\mathbf{z}}_j^-$	History state initialization
$\mathbf{P}_{j,j}^- = \mathbf{P}_j^-$	History state covariance initialization
$\mathbf{P}_{jj}^- = \mathbf{P}_j^-$	State error cross - covariance initialization
$\mathbf{P}_{\omega_s z_j, j} = \mathbf{P}_{\omega_s z_j}$	Process noise & history state cross-covariance initialization
$\mathbf{P}_{zv_{t,j}, j}^- = \mathbf{P}_{zv_{t,j}}$	State & history meas. noise cross-covariance initialization
$\mathbf{P}_{v_t z_j, j}^- = \mathbf{P}_{v_t z_j}$	Meas. noise & history state cross-covariance initialization
$\mathbf{P}_{v_t v_{t,j}, j} = \mathbf{P}_{v_t v_{t,j}}$	Measurement noise cross-covariance initialization

Propagation (at time $t = t_k$)

$\mathbf{P}_{kj}^- = \Phi_{k-1} \mathbf{P}_{(k-1)j}^+ + \sum_{i=1}^q (\mathbf{\Gamma}_{i,k-1} \mathbf{H}_{\omega_i} \mathbf{P}_{\omega_i z_j, k-1})$	State error cross - covariance propagation
$\mathbf{P}_{\omega_s z_j, k} = \Phi_{\omega_s} \mathbf{P}_{\omega_s z_j, k-1}$	Process noise & history state cross-covariance propagation
$\mathbf{P}_{zv_{t,j}, k}^- = \Phi_{k-1} \mathbf{P}_{zv_{t,j}, k-1}^+$	State & history meas. noise cross-covariance propagation
$\mathbf{P}_{v_t z_j, k} = \Phi_{v_t} \mathbf{P}_{v_t z_j, k-1}$	Meas. noise & history state cross-covariance propagation
$\mathbf{P}_{v_t v_{t,j}, k} = \Phi_{v_t} \mathbf{P}_{v_t v_{t,j}, k-1}$	Measurement noise cross-covariance propagation

Interim update (other measurement at time $t = t_k$)

$\mathbf{P}_{kj}^+ = (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{kj}^- - \mathbf{K}_k \sum_{i=1}^p \mathbf{H}_{v_i} \mathbf{P}_{v_i z_j, k}$	State error cross-covariance update	Only the state & history meas. noise cross-covariances that are related to the interim measurement are to be used in the sums!
$\mathbf{K}_k^* = (\mathbf{P}_{jk}^- \mathbf{H}_k^\top + \sum_{i=1}^p \mathbf{P}_{zv_{i,j}, k} \mathbf{H}_{v_i}^\top) \mathbf{S}_k^{-1}$	History state Kalman gain	
$\hat{\mathbf{z}}_{j,k} = \hat{\mathbf{z}}_{j,k-1} + \mathbf{K}_k^* \mathbf{s}_k$	History state update	
$\mathbf{P}_{j,k} = \mathbf{P}_{j,k-1} - \mathbf{K}_k^* (\mathbf{H}_k \mathbf{P}_{kj}^- - \sum_{i=1}^p \mathbf{H}_{v_i} \mathbf{P}_{v_i z_j, k})$	History state covariance update	
$\mathbf{P}_{zv_{t,j}, k}^+ = \begin{cases} (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{zv_{t,j}, k}^- - \mathbf{K}_k \mathbf{H}_{v_t} \mathbf{P}_{v_t z_j, k} \\ (\mathbf{I}_n - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{zv_{t,j}, k}^- \end{cases}$	State & history meas. noise cross - covariance update related to the interim measurement	
	all others	

Final update (at arrival time $t = t_k$)

$\mathbf{R}_j = \sum_{i=1}^r (\mathbf{H}_{v_i} \mathbf{P}_{v_i v_{i,j}} \mathbf{H}_{v_i}^\top + \mathbf{D}_{v_i} \mathbf{Q}_{v_i} \mathbf{D}_{v_i}^\top)$	Measurement covariance	Only the state & history meas. noise cross-covariances that are related to the delayed measurement are to be used in the sums!
$\mathbf{s}_{j,k} = \tilde{\mathbf{y}}_j - \mathbf{H}_j \hat{\mathbf{z}}_{j,k}$	Innovation	
$\mathbf{S}_{j,k} = \mathbf{H}_j \mathbf{P}_{j,k}^- \mathbf{H}_j^\top + \mathbf{R}_j$	Innovation covariance	
$+ \sum_{i=1}^r (\mathbf{H}_{v_i} \mathbf{P}_{v_i, j}^-) \mathbf{H}_j^\top + \mathbf{H}_j \sum_{i=1}^r \mathbf{P}_{zv_{i,j}, k}^- \mathbf{H}_{v_i}^\top$		
$\mathbf{K}_k = (\mathbf{P}_{kj}^- \mathbf{H}_j^\top + \sum_{i=1}^r \mathbf{P}_{zv_{i,j}, k}^- \mathbf{H}_{v_i}^\top) \mathbf{S}_{j,k}^{-1}$	Kalman gain	
$\hat{\mathbf{z}}_k^+ = \hat{\mathbf{z}}_k^- + \mathbf{K}_k \mathbf{s}_{j,k}$	Navigation state update	
$\mathbf{P}_{zz, k}^+ = \mathbf{P}_{zz, k}^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{jk}^- - \mathbf{K}_k \sum_{i=1}^r \mathbf{H}_{v_i} \mathbf{P}_{v_i, j}^-$	Navigation covariance update	
$\mathbf{P}_{zv_{t,k}}^+ = \begin{cases} \mathbf{P}_{zv_{t,k}}^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{zv_{t,k}}^- - \mathbf{K}_k \mathbf{H}_{v_t} \mathbf{P}_{v_t, j}^- \\ \mathbf{P}_{zv_{t,k}}^- - \mathbf{K}_k \mathbf{H}_j \mathbf{P}_{zv_{t,k}}^- \end{cases}$	State & measurement noise cross-covariance update related to the interim measurement	
	all others	

D Navigation Filter

Performance Prediction and Stability

D.1 Eigenvalues and Eigenvectors of the Hamiltonian Matrix

In this section, the eigenvalues and eigenvectors of the Hamiltonian matrix Ψ (5.32) are calculated analytically. At first, the rows and columns of Ψ are permuted to obtain a favorable almost upper triangular matrix. This form simplifies the Gauss elimination later on. With the permutation matrix

$$\mathbf{T} = \left(\begin{array}{ccc|ccc} 0 & 0 & 0 & 0 & 0 & \mathbf{I}_2 \\ 0 & 0 & \mathbf{I}_2 & 0 & 0 & 0 \\ 0 & \mathbf{I}_3 & 0 & 0 & 0 & 0 \\ \hline \mathbf{I}_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I}_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I}_3 & 0 \end{array} \right) \quad (\text{D.1})$$

the permuted Hamiltonian matrix is

$$\mathbf{T}\Psi\mathbf{T}^T = \left(\begin{array}{ccc|ccc} 0 & 0 & 0 & 0 & 0 & \text{veck}^{-1}(\tilde{\mathbf{f}}_{\delta})_{3 \times 2}^T \\ \mathbf{Q}_{\omega} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\text{veck}^{-1}(\tilde{\mathbf{f}}_{\delta})_{3 \times 2} & 0 & 0 & 0 & \mathbf{Q}_f \\ \hline 0 & 0 & \mathbf{I}_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{R}_{pos}^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\mathbf{I}_3 & 0 \end{array} \right) \quad (\text{D.2})$$

Next, the characteristic equation

$$\det(\lambda \mathbf{I}_{16} - \mathbf{T}\Psi\mathbf{T}^T) = 0 \quad (\text{D.3})$$

which is required for the calculation of the eigenvalues, is derived. $\lambda \mathbf{I}_{16} - \mathbf{T}\Psi\mathbf{T}^T$ is given with

$$\lambda \mathbf{I}_{16} - \mathbf{T}\Psi\mathbf{T}^T = \left(\begin{array}{ccc|ccc} \lambda \mathbf{I}_2 & 0 & 0 & 0 & 0 & -\text{veck}^{-1}(\tilde{\mathbf{f}}_{\delta})_{3 \times 2}^T \\ -\mathbf{Q}_{\omega} & \lambda \mathbf{I}_2 & 0 & 0 & 0 & 0 \\ 0 & \text{veck}^{-1}(\tilde{\mathbf{f}}_{\delta})_{3 \times 2} & \lambda \mathbf{I}_3 & 0 & 0 & -\mathbf{Q}_f \\ \hline 0 & 0 & -\mathbf{I}_3 & \lambda \mathbf{I}_3 & 0 & 0 \\ 0 & 0 & 0 & -\mathbf{R}_{pos}^{-1} & \lambda \mathbf{I}_3 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I}_3 & \lambda \mathbf{I}_3 \end{array} \right) \quad (\text{D.4})$$

Determinants and therefore the characteristic equation are invariant with respect to Gauss elimination. Thus, $\lambda \mathbf{I}_{16} - \mathbf{T}\Psi\mathbf{T}^T$ is converted to

$$\left(\begin{array}{ccc|cc} \lambda \mathbf{I}_2 & 0 & 0 & 0 & 0 \\ 0 & \lambda \mathbf{I}_2 & 0 & 0 & 0 \\ 0 & 0 & \lambda \mathbf{I}_3 & 0 & 0 \\ \hline 0 & 0 & 0 & \lambda \mathbf{I}_3 & 0 \\ 0 & 0 & 0 & 0 & \lambda \mathbf{I}_3 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right) \begin{array}{l} -\text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \\ -\lambda^{-1} \mathbf{Q}_\omega \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \\ -\mathbf{Q}_f + \lambda^{-2} \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} \mathbf{Q}_\omega \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \\ \hline \lambda^{-1} \left(-\mathbf{Q}_f + \lambda^{-2} \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} \mathbf{Q}_\omega \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \right) \\ \lambda^{-2} \mathbf{R}_{pos}^{-1} \left(-\mathbf{Q}_f + \lambda^{-2} \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} \mathbf{Q}_\omega \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \right) \\ \lambda^{-3} \mathbf{R}_{pos}^{-1} \left(-\mathbf{Q}_f + \lambda^{-2} \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} \mathbf{Q}_\omega \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \right) \end{array} \quad (D.5)$$

Due to the upper triangular matrix structure the characteristic equation is the product of the determinants of the main diagonal set to zero

$$\lambda^{-2} \det \left(\lambda^6 \mathbf{I}_3 + \lambda^2 \mathbf{R}_{pos}^{-1} \mathbf{Q}_f - \mathbf{R}_{pos}^{-1} \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2} \mathbf{Q}_\omega \text{veck}^{-1}(\tilde{\mathbf{f}}_o)_{3 \times 2}^T \right) = 0 \quad (D.6)$$

The characteristic equation is in expanded form

$$\lambda^{-2} \det \left(\begin{array}{ccc} \lambda^6 + \frac{\sigma_f^2}{\sigma_{pos}^2} \lambda^2 - \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2 & 0 & 0 \\ 0 & \lambda^6 + \frac{\sigma_f^2}{\sigma_{pos}^2} \lambda^2 - \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2 & 0 \\ 0 & 0 & \lambda^6 + \frac{\sigma_f^2}{\sigma_{pos}^2} \lambda^2 \end{array} \right) = 0 \quad (D.7)$$

or becomes with the determinant calculated

$$\left(\lambda^6 + \frac{\sigma_f^2}{\sigma_{pos}^2} \lambda^2 - \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2 \right)^2 \left(\lambda^4 + \frac{\sigma_f^2}{\sigma_{pos}^2} \right) = 0 \quad (D.8)$$

Solving the characteristic equation (D.8) yields the eigenvalues of the Hamiltonian matrix Ψ . The first bracket of (D.8) provides six eigenvalues with algebraic multiplicity two, thus 12 of the 16 eigenvalues of Ψ

$$\lambda^6 + \frac{\sigma_f^2}{\sigma_{pos}^2} \lambda^2 - \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2 = 0 \quad (D.9)$$

With the substitution $u = \lambda^2$ the 6th order polynomial is replaced by a cubic equation

$$u^3 + \frac{\sigma_f^2}{\sigma_{pos}^2} u - \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2 = 0 \quad (D.10)$$

The solution of this equation can be found with Cardano's method [103]. p and q are defined as

$$p = \frac{1}{3} \frac{\sigma_f^2}{\sigma_{pos}^2}, \quad q = -\frac{1}{2} \frac{\sigma_\omega^2}{\sigma_{pos}^2} \|\tilde{\mathbf{f}}_b\|^2 \quad (D.11)$$

Since $q^2 + p^3 > 0$, there are one real and two complex roots. The three roots of the cubic polynomial (D.10) are given with

$$\begin{aligned} u_1 &= A + B \\ u_2 &= A \left(-\frac{1}{2} + \frac{\sqrt{3}}{2} i \right) + B \left(-\frac{1}{2} - \frac{\sqrt{3}}{2} i \right) \\ u_3 &= A \left(-\frac{1}{2} - \frac{\sqrt{3}}{2} i \right) + B \left(-\frac{1}{2} + \frac{\sqrt{3}}{2} i \right) \end{aligned} \quad (\text{D.12})$$

where

$$A = \sqrt[3]{-q + \sqrt{q^2 + p^3}}, \quad B = \sqrt[3]{-q - \sqrt{q^2 + p^3}} \quad (\text{D.13})$$

The six eigenvalues λ_{1-6} are then

$$\begin{aligned} \lambda_{1/2} &= \pm \sqrt{A + B} \\ \lambda_{3/4} &= \pm \sqrt{A \left(-\frac{1}{2} + \frac{\sqrt{3}}{2} i \right) + B \left(-\frac{1}{2} - \frac{\sqrt{3}}{2} i \right)} \\ \lambda_{5/6} &= \pm \sqrt{A \left(-\frac{1}{2} - \frac{\sqrt{3}}{2} i \right) + B \left(-\frac{1}{2} + \frac{\sqrt{3}}{2} i \right)} \end{aligned} \quad (\text{D.14})$$

The eigenvalues λ_{7-12} are equal to the eigenvalues λ_{1-6} . The second bracket in (D.8) provides the remaining four eigenvalues of Ψ

$$\lambda^4 + \frac{\sigma_f^2}{\sigma_{pos}^2} = 0 \quad (\text{D.15})$$

Solving for λ yields the eigenvalues λ_{13-16}

$$\lambda_{13/14} = \pm \sqrt{\frac{\sigma_f}{\sigma_{pos}}} i, \quad \lambda_{15/16} = \pm \sqrt{-\frac{\sigma_f}{\sigma_{pos}}} i \quad (\text{D.16})$$

Next, the 16 eigenvectors \mathbf{v}_i that correspond to the found eigenvalues λ_i are searched. Eigenvalue and eigenvector pairs fulfill

$$(\lambda_i \mathbf{I}_{16} - \mathbf{T} \Psi \mathbf{T}^\top) \mathbf{v}_i = 0 \quad (\text{D.17})$$

For that, the matrix $\lambda_i \mathbf{I}_{16} - \mathbf{T} \Psi \mathbf{T}^\top$ (D.5) is consulted again. The eigenvectors \mathbf{v}_i with 16 elements are subdivided into six subvectors, $\mathbf{v}_i = (\mathbf{v}_{i,1}^\top \quad \mathbf{v}_{i,2}^\top \quad \dots \quad \mathbf{v}_{i,6}^\top)^\top$.

The sixth subvectors $\mathbf{v}_{i,6}$ are chosen as

$$\mathbf{v}_{i,6} = \lambda_i^5 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \text{for } \lambda_1 \dots \lambda_6, \quad \mathbf{v}_{i,6} = \lambda_i^5 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{for } \lambda_7 \dots \lambda_{12}, \quad \mathbf{v}_{i,6} = \lambda_i^5 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad \text{for } \lambda_{13} \dots \lambda_{16} \quad (\text{D.18})$$

The direction of these vectors is arbitrary as long as the vectors are multiplied by λ_i^5 . Then, the characteristic equation appears again in the last row of (D.5), which is just zero for the eigenvalue λ_i . The other subvectors are, depending on the last subvector $\mathbf{v}_{i,6}$

$$\begin{aligned} \mathbf{v}_{i,1} &= \lambda_i^{-1} \text{veck}^{-1} \left(\tilde{\mathbf{f}}_\delta \right)_{3 \times 2}^\top \mathbf{v}_{i,6} \\ \mathbf{v}_{i,2} &= \lambda_i^{-2} \mathbf{Q}_\omega \text{veck}^{-1} \left(\tilde{\mathbf{f}}_\delta \right)_{3 \times 2}^\top \mathbf{v}_{i,6} \\ \mathbf{v}_{i,3} &= \lambda_i^{-1} \left(\mathbf{Q}_f - \lambda_i^{-2} \text{veck}^{-1} \left(\tilde{\mathbf{f}}_\delta \right)_{3 \times 2} \mathbf{Q}_\omega \text{veck}^{-1} \left(\tilde{\mathbf{f}}_\delta \right)_{3 \times 2}^\top \right) \mathbf{v}_{i,6} \end{aligned} \quad (\text{D.19})$$

$$\begin{aligned}\mathbf{v}_{i,4} &= \lambda_i^{-2} \left(\mathbf{Q}_f - \lambda_i^{-2} \text{veck}^{-1} \left(\tilde{\mathbf{f}}_{\delta} \right)_{3 \times 2} \mathbf{Q}_{\omega} \text{veck}^{-1} \left(\tilde{\mathbf{f}}_{\delta} \right)_{3 \times 2}^T \right) \mathbf{v}_{i,6} \\ \mathbf{v}_{i,5} &= \lambda_i^{-3} \mathbf{R}_{pos}^{-1} \left(\mathbf{Q}_f - \lambda_i^{-2} \text{veck}^{-1} \left(\tilde{\mathbf{f}}_{\delta} \right)_{3 \times 2} \mathbf{Q}_{\omega} \text{veck}^{-1} \left(\tilde{\mathbf{f}}_{\delta} \right)_{3 \times 2}^T \right) \mathbf{v}_{i,6}\end{aligned}$$

The vectors $\mathbf{v}_{i,1}$ and $\mathbf{v}_{i,2}$ are of size two whereas all others are of size three. Finally, the eigenvectors \mathbf{v}_i are permuted back to the original order with $\mathbf{T}^T \mathbf{v}_i$. The back-permuted eigenvectors are explicitly written

Eigenvectors $\lambda_1 \dots \lambda_6$	Eigenvectors $\lambda_7 \dots \lambda_{12}$	Eigenvectors $\lambda_{13} \dots \lambda_{16}$
$\mathbf{v}_i = \begin{pmatrix} \lambda_i^3 \sigma_f^2 - \lambda_i \sigma_{\omega}^2 \ \tilde{\mathbf{f}}_b\ ^2 \\ 0 \\ 0 \\ \hline \lambda_i^4 \sigma_f^2 - \lambda_i^2 \sigma_{\omega}^2 \ \tilde{\mathbf{f}}_b\ ^2 \\ 0 \\ 0 \\ \hline 0 \\ \lambda_i^3 \sigma_{\omega}^2 \ \tilde{\mathbf{f}}_b\ \\ \hline \lambda_i^2 \frac{\sigma_f^2}{\sigma_{pos}^2} - \frac{\sigma_{\omega}^2}{\sigma_{pos}^2} \ \tilde{\mathbf{f}}_b\ ^2 \\ 0 \\ 0 \\ \hline \lambda_i^5 \\ 0 \\ 0 \\ \hline 0 \\ \lambda_i^4 \ \tilde{\mathbf{f}}_b\ \end{pmatrix}$	$\mathbf{v}_i = \begin{pmatrix} 0 \\ \lambda_i^3 \sigma_f^2 - \lambda_i \sigma_{\omega}^2 \ \tilde{\mathbf{f}}_b\ ^2 \\ 0 \\ \hline 0 \\ \lambda_i^4 \sigma_f^2 - \lambda_i^2 \sigma_{\omega}^2 \ \tilde{\mathbf{f}}_b\ ^2 \\ 0 \\ \hline -\lambda_i^3 \sigma_{\omega}^2 \ \tilde{\mathbf{f}}_b\ \\ 0 \\ \hline 0 \\ \lambda_i^2 \frac{\sigma_f^2}{\sigma_{pos}^2} - \frac{\sigma_{\omega}^2}{\sigma_{pos}^2} \ \tilde{\mathbf{f}}_b\ ^2 \\ 0 \\ \hline 0 \\ \lambda_i^5 \\ 0 \\ \hline -\lambda_i^4 \ \tilde{\mathbf{f}}_b\ \\ 0 \end{pmatrix}$	$\mathbf{v}_i = \begin{pmatrix} 0 \\ 0 \\ \lambda_i^3 \sigma_f^2 \\ 0 \\ 0 \\ \hline \lambda_i^4 \sigma_f^2 \\ 0 \\ 0 \\ \hline 0 \\ \lambda_i^2 \frac{\sigma_f^2}{\sigma_{pos}^2} \\ 0 \\ 0 \\ \hline \lambda_i^5 \\ 0 \\ 0 \\ \hline 0 \\ 0 \end{pmatrix}$

(D.20)

D.2 Kalman Filter Stability Enhancement

In section 5.2, it has been revealed that an adequate countermeasure against the growth of asymmetric errors is indispensable, even if the filter is implemented in the correct way, because numerical errors occur in all computing systems. Especially if the units of the navigation states and thus their magnitude orders are chosen inappropriately, the rank of the involved matrices is low and the influence of round-off errors is high. The simplest way is to symmetrize the covariance matrix from time to time by

$$\bar{\mathbf{P}} = \frac{1}{2} (\mathbf{P} + \mathbf{P}^T) \quad (D.21)$$

Alternatively to (D.21) the covariance propagation equation of the discrete-time Kalman filter in (5.38) could be extended by $\alpha \mathbf{P}^T - \alpha \mathbf{P}$

$$\mathbf{P}_k^- = \Phi \mathbf{P}_{k-1}^+ \Phi^T + \Gamma \mathbf{Q}_k \Gamma^T + \alpha \mathbf{P}_{k-1}^T - \alpha \mathbf{P}_{k-1} \quad (D.22)$$

$\alpha > 0$ is a selectable factor. The corresponding continuous-time differential covariance equation (5.54) becomes

$$\dot{\mathbf{P}} = -\mathbf{P}\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\mathbf{P} + \mathbf{F}\mathbf{P} + \mathbf{P}\mathbf{F}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T + \alpha\mathbf{P}^T - \alpha\mathbf{P} \quad (\text{D.23})$$

and the decisive matrix \mathbf{M} of the asymmetric covariance error differential equation (5.68) is

$$\mathbf{M} = \mathbf{F} - \tilde{\mathbf{P}}_s \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} - \alpha \mathbf{I}_n \quad (\text{D.24})$$

instead of (5.72). The factor α has to be chosen such that the eigenvalues of \mathbf{M} lie in the left-hand side of the complex plane. Then, the asymmetric covariance error differential equation is guaranteed stable.

Independently, the navigation states should always be adequately scaled. The orientation error states Ψ_{ee} or Ψ_{nn} should be expressed in mrad instead of rad, thus scaled by the factor 1000, in order to get numbers in the range of one, which corresponds well to the expected position and velocity errors. If the navigation error equations in n -frame are used, the latitude and longitude errors $\delta\phi$ and $\delta\lambda$ should be scaled by the factor 10^6 to obtain metric like errors in the range of one in lateral direction that correspond to the range of the height error δh .

E Simulation Scenarios

E.1 Trajectory

E.1.1 Generic Lissajous

For the simulation and analysis of integrated navigation algorithms appropriate reference trajectories have to be specified. Analytical curves that can be described by mathematical functions are appealing because non-integrating as well as integrating IMU measurements can be easily derived from them. So called Lissajous figures provide realistic closed trajectories. The position is represented by harmonic functions

$$\begin{aligned}\phi &= \phi_0 + \frac{A_\phi}{M} \sin(\omega_\phi t + \varphi_\phi) \\ \lambda &= \lambda_0 + \frac{A_\lambda}{(N+h)\cos\phi} \sin(\omega_\lambda t + \varphi_\lambda) \\ h &= h_0 + A_h \sin(\omega_h t + \varphi_h)\end{aligned}\tag{E.1}$$

with amplitudes A_ϕ/M , $A_\lambda/[(N+h)\cos\phi]$ and A_h , frequencies ω_ϕ , ω_λ , and ω_h and phase offsets φ_ϕ , φ_λ and φ_h . Velocities and accelerations with respect to Earth specified in n -frame are calculated by derivation of (E.1) with time.

It is assumed that the angles of attack and sideslip are zero such that the pitch angle ϑ_{nb} is equal to the angle of climb and the heading angle ψ_{nb} corresponds to the track angle

$$\begin{aligned}\vartheta_{nb} &= -\arcsin \frac{v_d}{\|\mathbf{v}_n\|} \\ \psi_{nb} &= \text{atan2}(v_e, v_n)\end{aligned}\tag{E.2}$$

The roll angle φ_{nb} results from the equilibrium of lateral forces as illustrated in Figure E-1 since it is supposed that the aircraft flies coordinated curves

$$\begin{aligned}\tan \varphi_{nb} &= \frac{m\|\mathbf{v}_n\| \cos \vartheta_{nb} \dot{\psi}_{nb}}{m\gamma} \\ \varphi_{nb} &= \text{atan2}(\|\mathbf{v}_n\| \cos \vartheta_{nb} \dot{\psi}_{nb}, \gamma)\end{aligned}\tag{E.3}$$

The following settings are chosen:

$$\begin{aligned}\phi_0 &= 47^\circ & \lambda_0 &= 11^\circ & h_0 &= 5000 \text{ m} \\ A_\phi &= 24000 \text{ m} & A_\lambda &= 12000 \text{ m} & A_h &= 5000 \text{ m} \\ \omega_\phi &= 0.01 \text{ rad/s} & \omega_\lambda &= 0.02 \text{ rad/s} & \omega_h &= 0.015 \text{ rad/s} \\ \varphi_\phi &= 0 \text{ rad} & \varphi_\lambda &= 0 \text{ rad} & \varphi_h &= 0 \text{ rad}\end{aligned}\tag{E.4}$$

Figure E-2 shows the resulting trajectory. The velocity is illustrated in Figure E-3, the orientation angles in Figure E-4, the accelerations in Figure E-5 and the angular rates in Figure E-6.

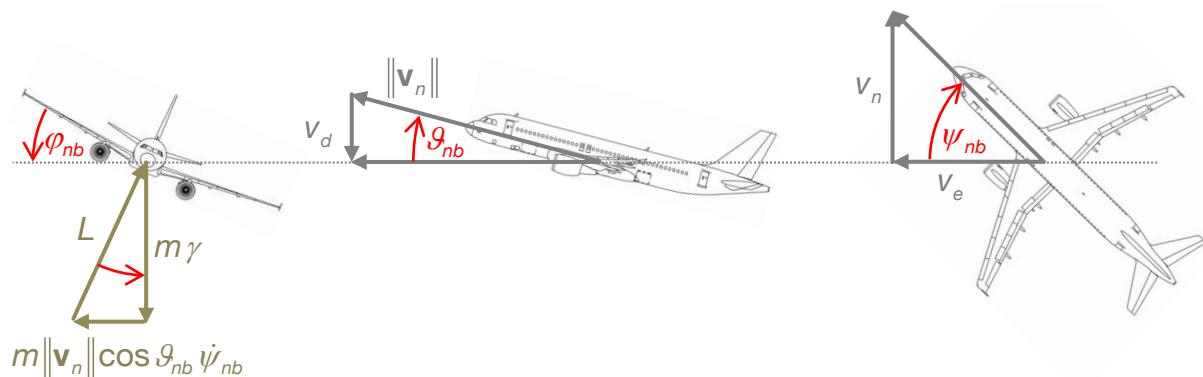


Figure E-1: Lateral equilibrium of forces

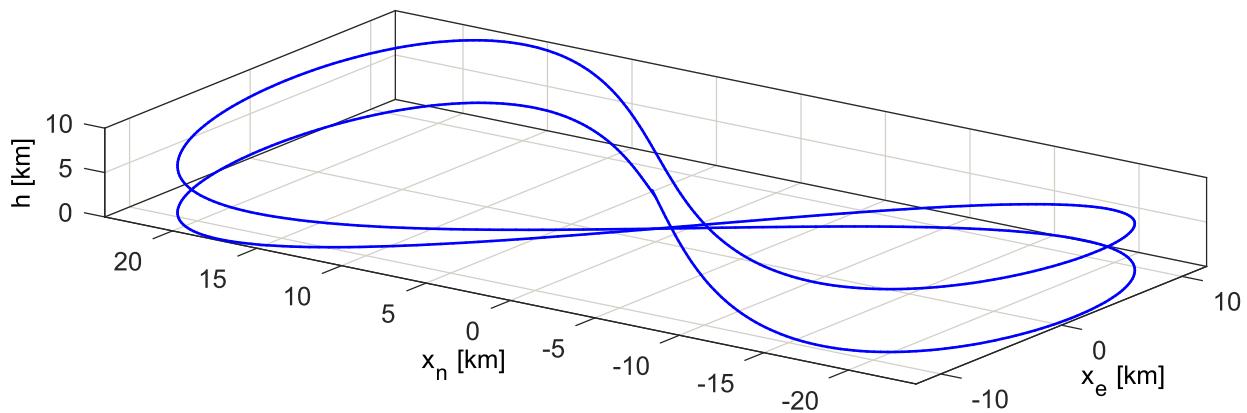


Figure E-2: Lissajous trajectory

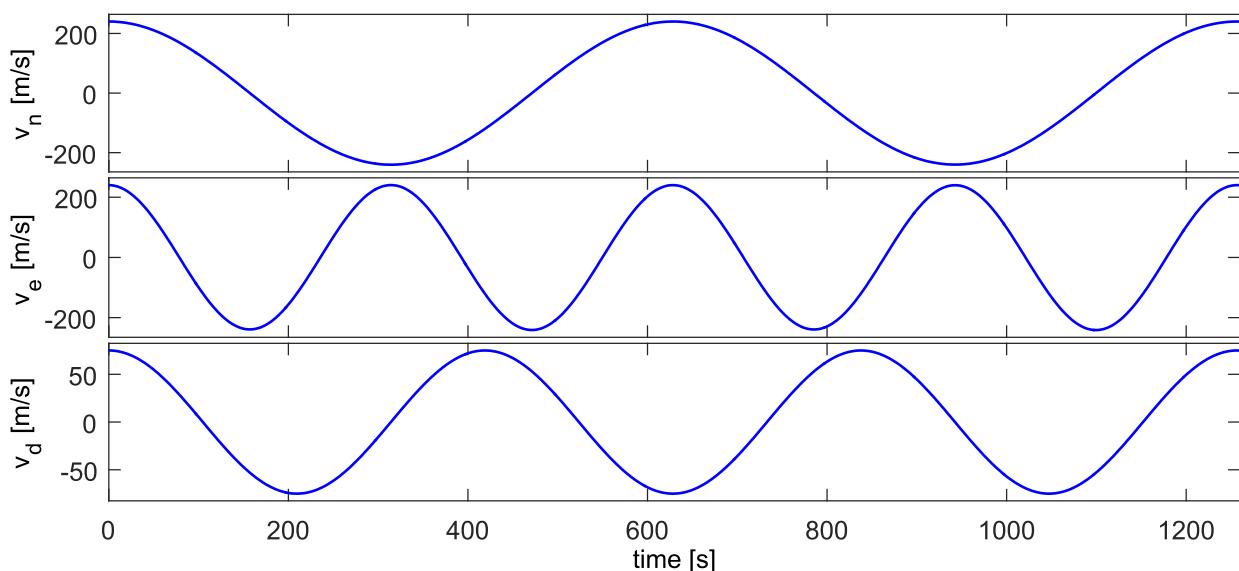
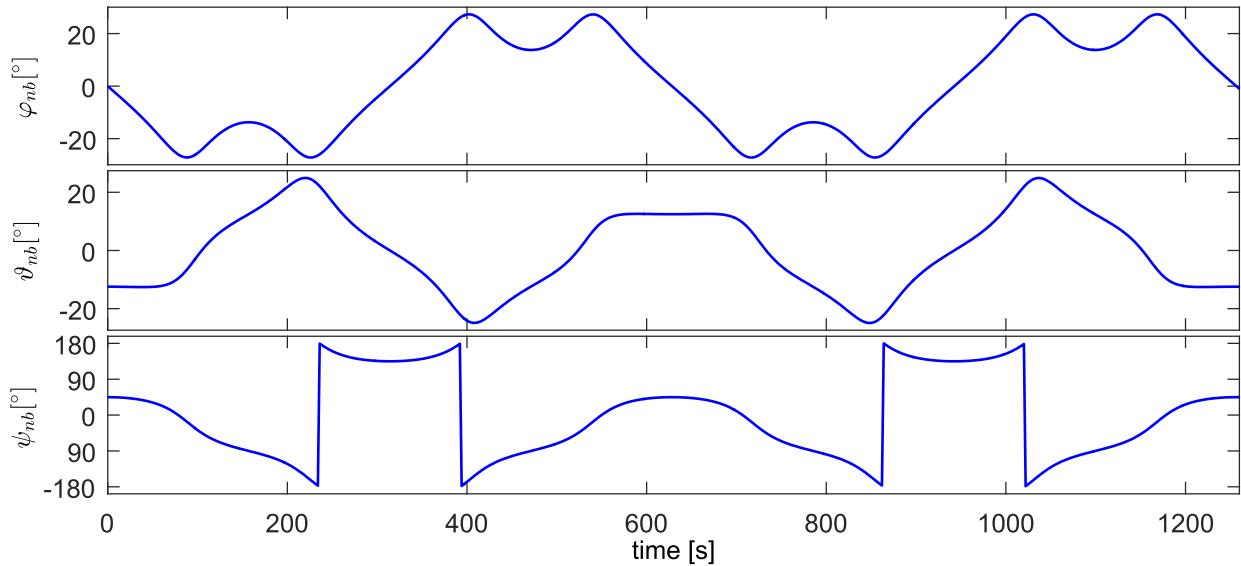
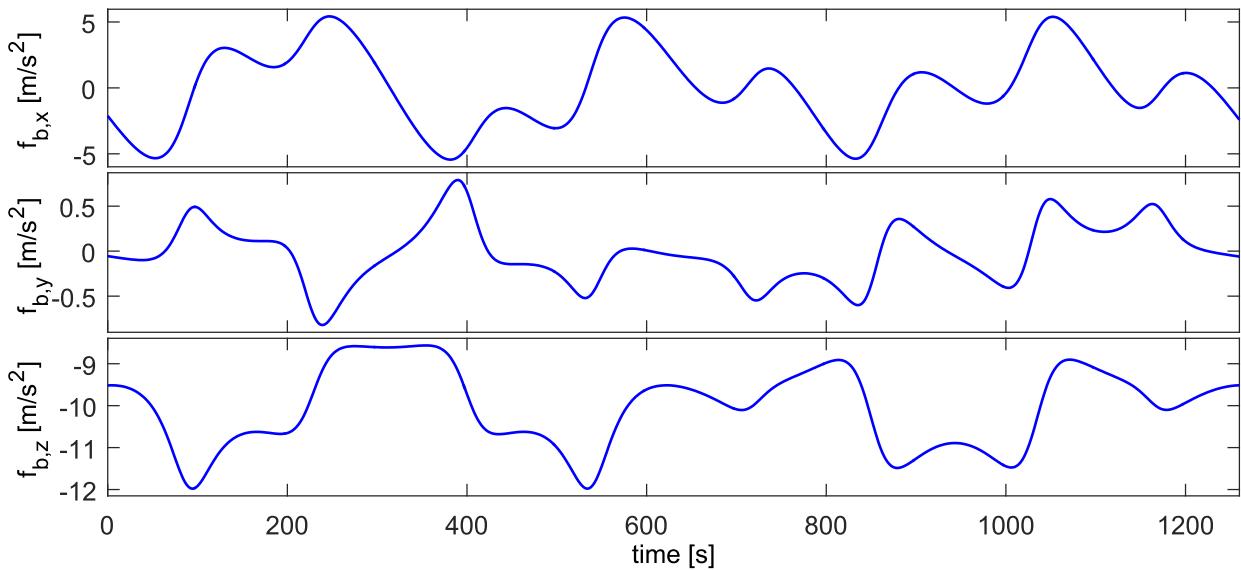
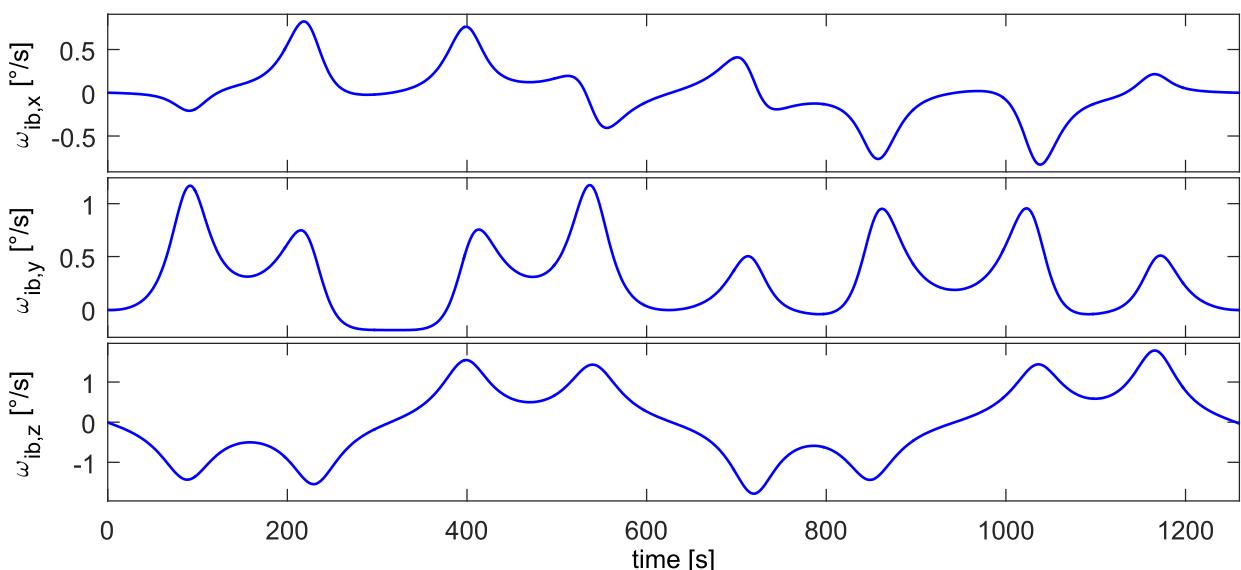


Figure E-3: Velocity

**Figure E-4: Orientation****Figure E-5: Acceleration****Figure E-6: Angular rate**

E.1.2 General Aviation Aircraft

As second scenario a more realistic flight of a general aviation aircraft from Munich airport (ICAO-code EDDM) to Oberpfaffenhofen airport (ICAO-code EDMO) has been chosen. The initial trajectory has been recorded with the Diamond DA-42 Flight Training Device at the Institute of Flight System Dynamics. Analytic curves have been derived by quintic spline interpolation of the position and orientation. Quantization steps in the latitude and longitude have been removed before by low-pass filtering of the recorded trajectory. From the analytic splines IMU measurements with arbitrary sample rates can be synthesized. Figure E-7 shows the trajectory. The velocity is plotted in Figure E-8, the orientation in Figure E-9, the acceleration in Figure E-10 and the angular rate in Figure E-11. It can be seen that the dynamics are comparatively low.

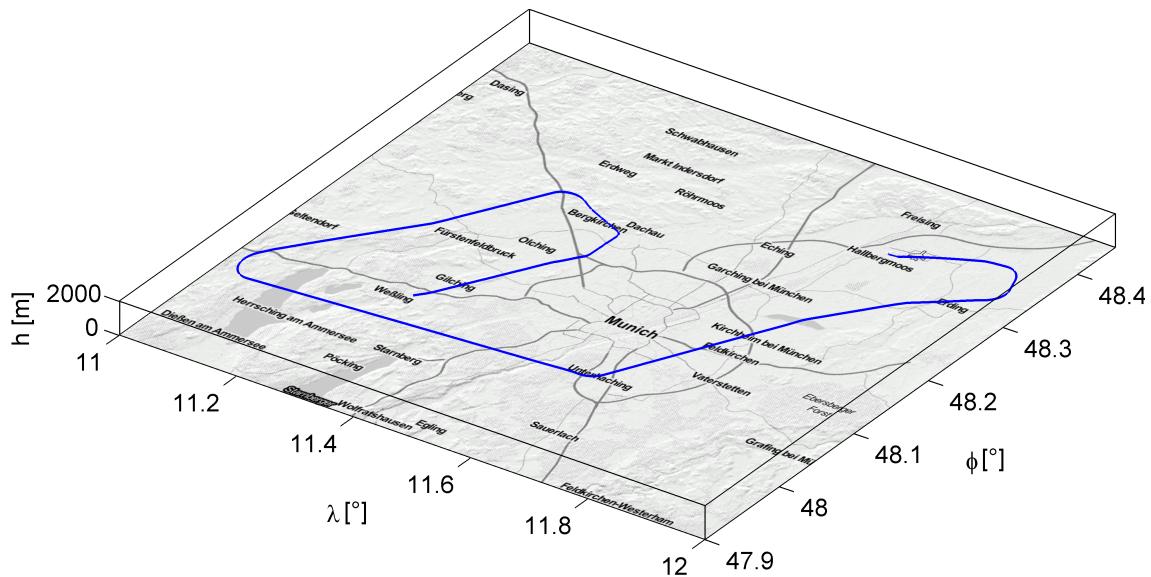


Figure E-7: Trajectory

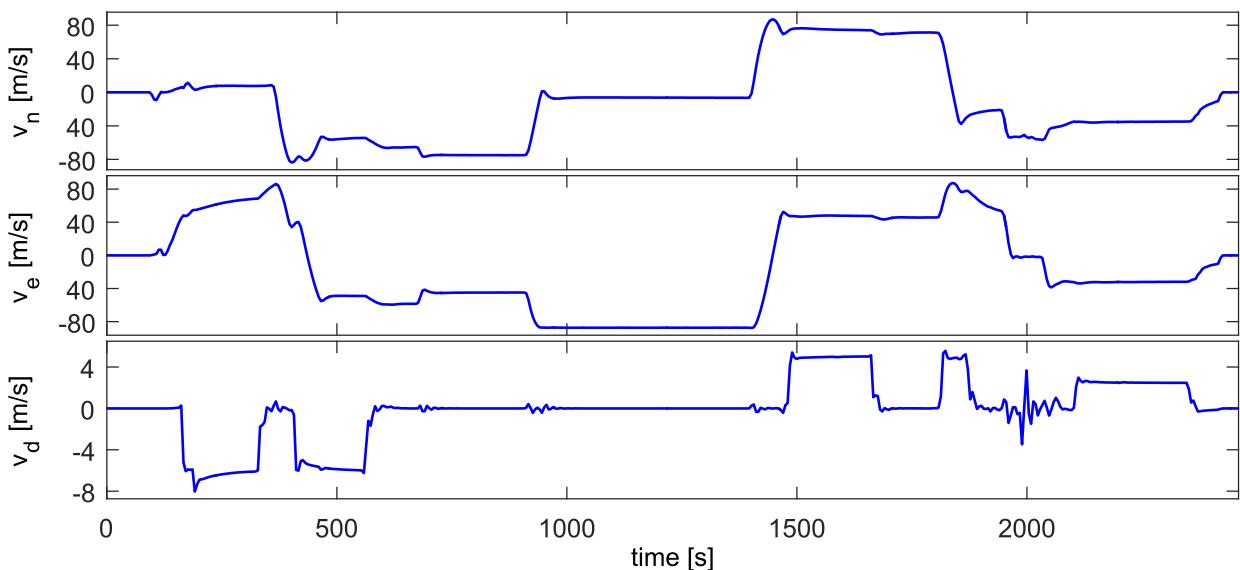
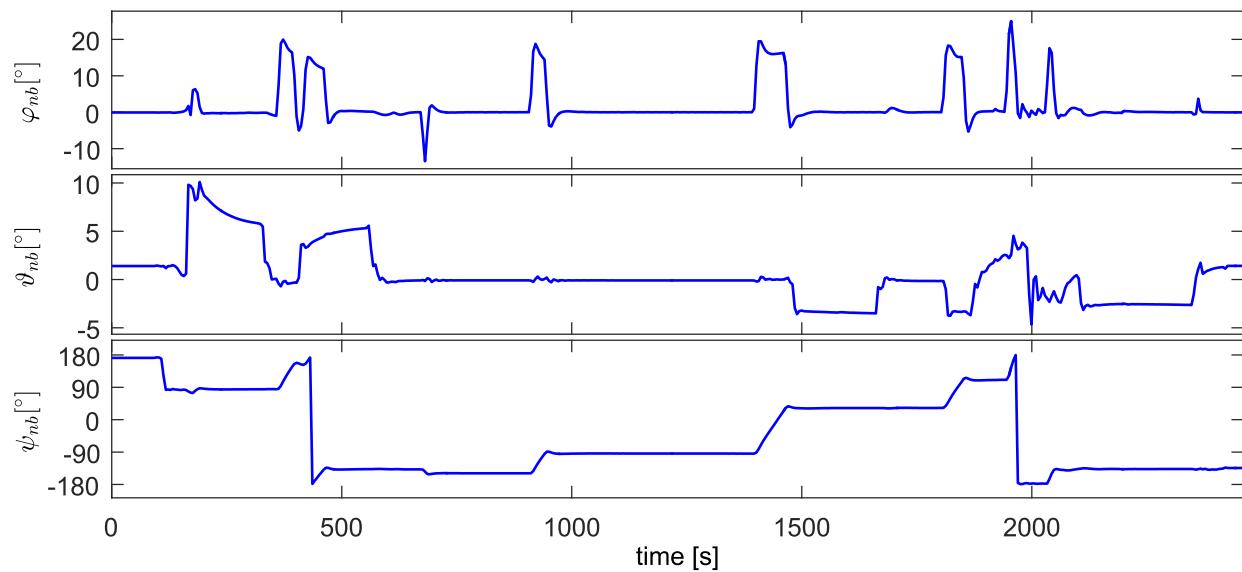
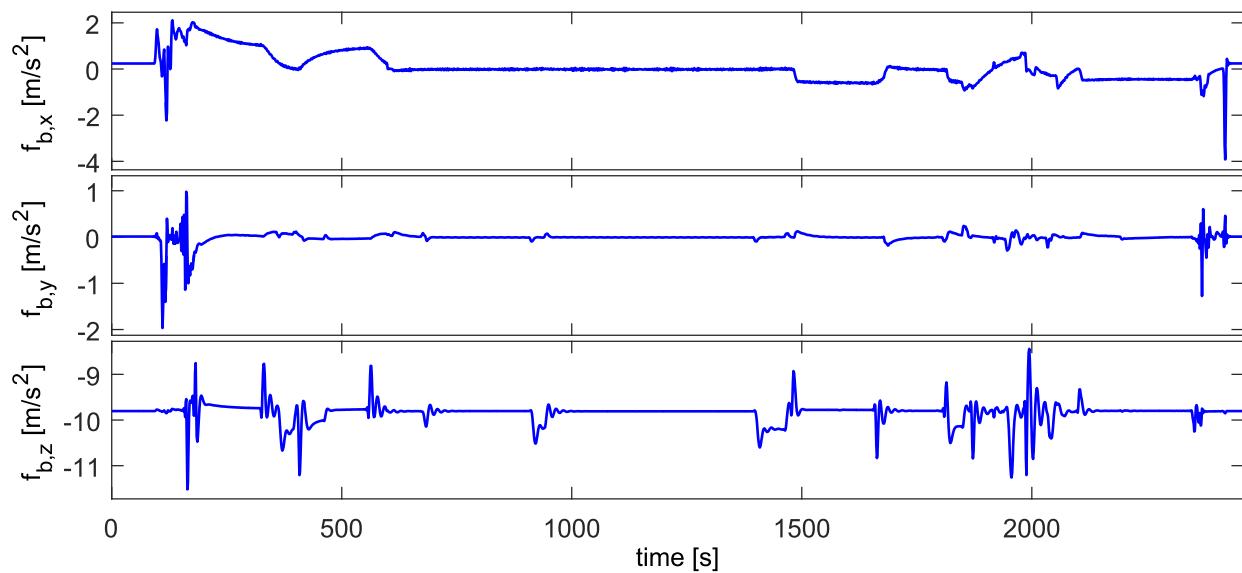
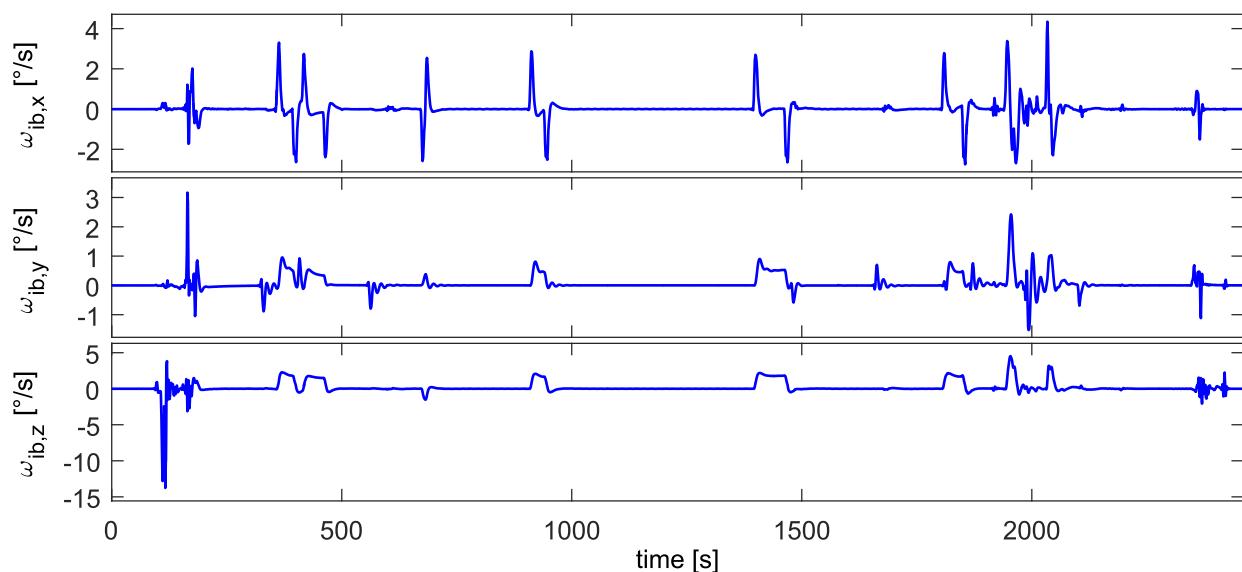


Figure E-8: Velocity

**Figure E-9: Orientation****Figure E-10: Acceleration****Figure E-11: Angular rate**

E.2 Sensors

E.2.1 IMU

IMU sensors of four different grades have been chosen for the simulations of this thesis. The first device (IMU 1) is an Analog Devices ADIS16488 [125], which incorporates a MEMS gyroscope triad and a MEMS accelerometer triad within a compact housing. The second device (IMU 2) is a Northrop Grumman LITEF μIMU-IC [34], which also bases upon MEMS technology like the ADIS16488 but features better gyroscope performance about one order of magnitude. The third device (IMU 3) is an iMAR iIMU-FCAI-E [126], incorporating three fiber optical gyroscopes (FOG) and three quartz accelerometers. The fourth device (IMU 4) is a Northrop Grumman LITEF LCI-100C using FOG as the FCAI-E but of higher grade. The performance specifications are given in Table E-5.

Table E-2: Specification of IMU sensors

		Analog Devices ADIS16488A [125] (IMU 1)	NG LITEF μIMU-IC [34] [128] (IMU 2)	iMAR iIMU-FCAI-E [126] (IMU 3)	NG LITEF LCI100C [127] (IMU 4)
Technology		MEMS	MEMS	solid	MEMS
Range	g	±18	±15	±10	±10
Turn-on bias	mg	16	3	2	1.25
Bias temperature sensitivity	mg/K	0.1	n/a	n/a	0.0033
Scale factor error	ppm	1667	1500	1500	100
Scale factor temperature sensitivity	ppm/K	25	n/a	n/a	3.3
Nonlinearity	ppm	1000	n/a	300	100
Misalignment axis-to-axis	mrad	0.6	0.5	0.17	0.17
Velocity random walk	$\mu\text{g}/\sqrt{\text{Hz}}$	49	83	20	33
Bias instability	μg	70	10	10	100
Vibration rectification	mg	n/a	2.5	n/a	n/a
Bandwidth	Hz	330	100	200	400
Technology		MEMS	MEMS	FOG	FOG
Range	°/s	±450	±499	±450	±495
Turn-on bias	°/h	720	10	2	0.1
Bias temperature sensitivity	°/h/K	9	0.05	n/a	0.002
Scale factor error	ppm	3333	1000	500	100
Scale factor temperature sensitivity	ppm/K	35	n/a	n/a	3.3
Nonlinearity	ppm	100	1800	300	100
Misalignment axis-to-axis	mrad	0.9	0.5	0.17	0.17
axis-to-frame	mrad	17	0.13	n/a	n/a
Angular random walk	$^{\circ}/\sqrt{\text{h}}$	0.26	0.1	0.02	0.004
Bias instability	°/h	5.1	0.1	0.02	0.05
G sensitivity	°/h/g	32.4	2	n/a	n/a
Vibration rectification	°/h/g ²	n/a	0.3	n/a	n/a
Bandwidth	Hz	330	100	500	400

E.2.2 GPS

For all simulations in this thesis, the same GPS satellite constellation is used. Unless otherwise specified, 19:00 (GPS time) at 06/22/2012 is chosen as start time T0 of the simulations. The corresponding sky plot at position N47° E11° between T0 and T0+3600 is shown in Figure E-1. The squares with the PRN are at the end of the trajectories. The grey shaded area marks the northern polar region without satellites. Generally, the satellites in view with the largest elevation angles are used for the solution. The minimum elevation is set to ten degree.

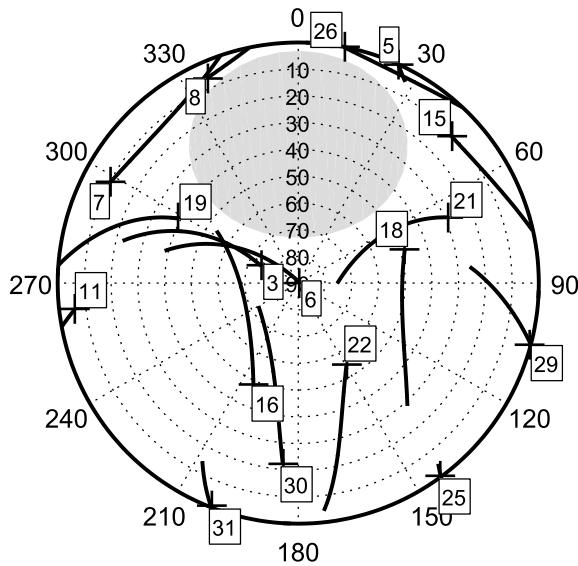


Figure E-12: Sky plot at N47° E11°, 06/22/2012, 7pm to 8pm