Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C Dissertationen Heft Nr. 850

Björn Friedrich Reuper

Multi-Frequency GNSS Sensor Fusion With Quality Assessment for Automotive Applications

München 2020

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325 ISBN 978-3-7696-5262-8

Diese Arbeit ist gleichzeitig veröffentlicht in:
Schriftenreihe der Fachrichtung Geodäsie der Technischen Universität Darmstadt
ISBN 978-3-935631-46-4, Heft 57, Darmstadt 2020

Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C Dissertationen Heft Nr. 850

Multi-Frequency GNSS Sensor Fusion With Quality Assessment for Automotive Applications

Vom Fachbereich Bau- und Umweltingenieurwissenschaften
der Technischen Universität Darmstadt
zur Erlangung des Grades
Doktor-Ingenieur (Dr.-Ing.)
genehmigte Dissertation

Vorgelegt von

M.Sc. Björn Friedrich Reuper

Geboren am 12.12.1987 in Göttingen, Deutschland

München 2020

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325 ISBN 978-3-7696-5262-8

Diese Arbeit ist gleichzeitig veröffentlicht in:
Schriftenreihe der Fachrichtung Geodäsie der Technischen Universität Darmstadt
ISBN 978-3-935631-46-4, Heft 57, Darmstadt 2020

Adresse der DGK:

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften (DGK)
Alfons-Goppel-Straße 11 ● D – 80 539 München
Telefon +49 – 331 – 288 1685 ● Telefax +49 – 331 – 288 1759
E-Mail post@dgk.badw.de ● http://www.dgk.badw.de

Prüfungskommission:

Vorsitzender: Prof. Dr.-Ing. Jens Schneider

Referent: Prof. Dr.-Ing. Matthias Becker

Korreferenten: Prof. Dr. rer. nat. Hermann Winner
Prof. Dr.-Ing. Andreas Eichhorn

Tag der mündlichen Prüfung: 19.12.2019

© 2020 Bayerische Akademie der Wissenschaften, München

Alle Rechte vorbehalten. Ohne Genehmigung der Herausgeber ist es auch nicht gestattet,
die Veröffentlichung oder Teile daraus auf photomechanischem Wege (Photokopie, Mikrokopie) zu vervielfältigen

ISSN 0065-5325 ISBN 978-3-7696-5262-8

Acknowledgments

I would like to thank my supervisor, *Prof. Dr.-Ing. Matthias Becker*, for his guidance throughout the years and for granting me the opportunity to work on a very interesting topic, even though I did not know that much about satellite geodesy in the beginning.

I want to thank my co-supervisors, *Prof. Dr. rer. nat. Hermann Winner* and *Prof. Dr.-Ing. Andreas Eichhorn*, for providing helpful advice regarding the content of this thesis and the preparation of my oral examination. In addition, I want to thank Prof. Winner for offering me the chance to work at his institute right after finishing my master's degree and for fostering my scientific progress subsequently.

Thanks to my colleagues at the Chair of Physical and Satellite Geodesy for the friendly and pleasant work environment. Special thanks to *Dr.-Ing. Stefan Leinen* for his continuous support since being my advisor during my master's thesis, lending an ear to my questions about parameter estimation and providing the voice of reason whenever necessary. Thanks to *Dr.-Ing. David Becker* for lots of advice regarding Kalman filtering and data collection as well as granting me access to his vast arsenal of useful MATLAB toolboxes. Thanks to my long-term office roommate *Dr.-Ing. Florian Reckeweg* for sharing his knowledge about GNSS in general and differential code biases in particular with me. The term "fruitful discussions" is slightly overused in thesis acknowledgments, but it is spot-on in this case.

Finally, I would like to express my gratitude to my family, especially to my wife for having my back at all times and to my daughter for reminding me that some things in life are more important than work.

Zusammenfassung

Durch die Modernisierung der Satellitenkonstellationen von GPS und GLONASS sowie die Inbetriebnahme von Galileo und BeiDou steht zivilen Nutzern eine deutlich erhöhte Anzahl an Navigationssatelliten mit nutzbaren Signalen auf mehreren Trägerfrequenzen zur Verfügung. Gleichzeitig steigen die Qualitätsanforderungen an Lokalisierungsalgorithmen, die im Bereich der Kraftfahrzeugtechnik Verwendung finden, da diese Algorithmen nicht mehr ausschließlich zu Navigationszwecken eingesetzt werden. Stattdessen kommt ihnen eine tragende Rolle bei der Entwicklung autonomer Fahrzeuge zu. Diese gestiegenen Anforderungen sind durch GNSS alleine nicht zu erfüllen, sondern erfordern die Fusion von Daten mehrerer Sensortypen.

In dieser Arbeit werden daher Messgrößen von drei verschiedenen Sensortypen mit Hilfe eines erweiterten Kalman-Filters miteinander fusioniert. Bei diesen drei Sensortypen handelt es sich um einen Mehrfrequenz-GNSS-Empfänger, eine inertiale Messeinheit sowie fahrzeugeigene Odometriesensoren. Der Schwerpunkt der Arbeit liegt in der optimalen Verarbeitung von Pseudostrecken, die zu Satelliten unterschiedlicher Konstellationen auf jeweils mehreren Trägerfrequenzen gemessen werden. In diesem Rahmen werden ionosphärenfreie Linearkombinationen eingesetzt, wodurch der größte Fehlereinfluss, der bei der Verwendung von Einfrequenzmessungen vorliegt, eliminiert wird. Durch den gleichzeitigen Einsatz verschiedener Satellitengenerationen und die wechselnden Empfangsbedingungen entsteht hierbei der Bedarf, zu jeder Messepoche gleichzeitig Pseudostrecken oder deren Linearkombinationen auf mehreren Trägerfrequenzen zu berücksichtigen. Dies setzt eine Kalibrierung der signalabhängigen differenziellen Codebiase voraus, die bei der Pseudostreckenmessung auftreten. Eine solche Kalibrierung wird für den verwendeten Mehrfrequenz-GNSS-Empfänger für die GPS-Signale L1 C/A, L2C und L5 sowie für die Galileo-Signale E1, E5a und E5b durchgeführt, wodurch die entsprechenden Pseudostrecken und deren Linearkombinationen im Integrationsfilter einsetzbar sind. Außerdem wird ein Modell zur quantitativen Beschreibung des Messrauschens entwickelt und parametriert, was eine optimale Gewichtung der verschiedenen Beobachtungsgrößen ermöglicht. Als weitere GNSS-Messgrößen kommen zeitlich differenzierte Trägerphasenmessungen auf GPS L1 C/A und Galileo E1 zum Einsatz.

Die fahrzeugeigenen Odometriesensoren stellen die Drehraten der vier Räder sowie den Lenkradwinkel zur Verfügung. Aus diesen Größen werden die horizontalen Geschwindigkeitsvektoren an den Radaufstandspunkten ermittelt, wobei unterschiedliche Reifenmodelle zur Kompensation von Längsschlupf und Schräglauf verwendet werden. Ein wesentlicher Entwicklungsaspekt ist die Einbeziehung von Korrelationen in die Kovarianzmatrix des Messrauschens. Diese Korrelationen sind in Fahrzeugquerrichtung so stark ausgeprägt, dass die Quergeschwindigkeiten der vier Räder letztlich zu je einem Messwert pro Achse zusammengefasst werden.

Nachdem die Beobachtungen aus GNSS-Empfänger und Odometriesensorik in dieser Weise vorverarbeitet wurden, werden sie mit den Messgrößen einer MEMS-IMU in enger Kopplung fusioniert. Die Modellierung der IMU-Messfehler zählt nicht zum Kern der Forschung in dieser Arbeit und erfolgt daher mit konventionellen Modellen.

Zur Qualitätsbeurteilung der vom Lokalisierungsalgorithmus bereitgestellten Fusionslösung werden Metriken aus den Bereichen Genauigkeit und Integrität ausgewählt. Diese Metriken werden anhand von Testszenarien, die unterschiedlichste GNSS-Empfangsbedingungen abdecken, ausgewertet. Die Referenzlösung wird durch die Integration eines RTK-fähigen GNSS-Empfängers mit einer Ringlaserkreisel-IMU generiert. Hierdurch wird der Nachweis erbracht, dass die Verwendung von Mehrfrequenzmessungen zu einer deutlichen Genauigkeitssteigerung bei allen betrachteten Szenarien führt. Bei ungestörtem GNSS-Empfang wird ein horizontaler Positionsfehler von 0,5 m oder besser in 95 % der Zeit erreicht.

Abstract

The modernization of the satellite constellations of GPS and GLONASS as well as the commissioning of Galileo and BeiDou provide civil users with a significantly increased amount of navigation satellites broadcasting usable signals on multiple carrier frequencies. Simultaneously, the quality requirements for automotive localization algorithms increase, because these algorithms are no longer used exclusively for navigation. Instead, they play an important role in the development of autonomous vehicles. These increased requirements cannot be met with GNSS alone, but necessitate the fusion of data from multiple sensor types.

This thesis describes the fusion of measurements from three different sensor types with an extended Kalman filter. These three sensor types are a multi-frequency GNSS receiver, an inertial measurement unit and the vehicle's built-in odometry sensors. The focus lies on the optimal processing of pseudoranges obtained from satellites of various constellations on multiple carrier frequencies. This is achieved by forming ionosphere-free linear combinations, which eliminate the largest error source of single-frequency observations. The current deployment of different active satellite generations and the varying reception conditions create the necessity to process pseudoranges or linear combinations thereof on multiple carrier frequencies simultaneously in each epoch. This requires the calibration of signal-dependent differential code biases occurring in pseudorange measurements. A suitable calibration is performed for the employed multi-frequency GNSS receiver for the GPS signals L1 C/A, L2C and L5 as well as for the Galileo signals E1, E5a and E5b. This enables the utilization of the respective pseudoranges and their linear combinations within the integration filter. In addition, a quantitative model of the measurement noise is developed and parametrized, permitting the optimal weighting of the different observations. Additional GNSS observables are time-differenced carrier phase measurements on GPS L1 C/A and Galileo E1.

The vehicle's built-in odometry sensors provide the rotation rates of the four wheels and the steering wheel angle. From these quantities, the horizontal velocity vectors at the wheel contact patches are computed. During this computation, the compensation of longitudinal and lateral slip is carried out with various tire models. A major development aspect is the inclusion of correlation into the measurement noise covariance matrix. The magnitude of these correlations in lateral direction is so large that the lateral velocities of the four wheels are subsumed into a single observation per axle.

After the observations from GNSS receiver and odometry sensors have been preprocessed in this way, they are fused with the measurements of a MEMS IMU in a tightly coupled integration filter. IMU error modeling is not a key aspect of this thesis and is therefore performed with conventional models.

In order to assess the quality of the integrated solution supplied by the localization algorithm, performance metrics concerning accuracy and integrity are chosen. These metrics are evaluated with the help of test scenarios covering different GNSS reception conditions. The reference solution is obtained by integrating data from a ring laser gyroscope IMU and from a GNSS receiver capable of RTK positioning. The results verify that the utilization of multi-frequency observations leads to a significant accuracy improvement in all considered test scenarios. During unobstructed GNSS reception, a horizontal position error of 0.5 m or better is achieved in 95 % of epochs.

Contents

Notation **v**

List of Symbols **vii**

List of Acronyms **xi**

List of Figures **xvi**

List of Tables **xviii**

1 Introduction **1**
- 1.1 Motivation 1
- 1.2 Aim of the Thesis 2
- 1.3 Thesis Outline 4

2 Background and State of the Art **7**
- 2.1 Coordinate Frames 7
  - 2.1.1 Notation for Quantities Within Coordinate Frames 7
  - 2.1.2 Earth-Centered Inertial Frame 8
  - 2.1.3 Earth-Centered Earth-Fixed Frame 8
  - 2.1.4 Body Frame 8
  - 2.1.5 Navigation Frame 9
- 2.2 GNSS 11
- 2.3 Odometry 14
- 2.4 GNSS/INS Integration 15
- 2.5 State of Current Research 18
  - 2.5.1 GNSS 18
  - 2.5.2 GNSS/INS Integration for Automotive Applications 22

3 GNSS Preprocessing **25**
- 3.1 Observables 25
- 3.2 Multi-Constellation GNSS 26
- 3.3 Multi-Frequency GNSS 27
- 3.4 Pseudorange Biases 31
- 3.5 Precise Satellite Orbits and Clocks 37
- 3.6 Measurement Noise 39
  - 3.6.1 Satellite Position and Clock Bias 40
  - 3.6.2 Unmodeled Ionospheric Delay 41
  - 3.6.3 Unmodeled Tropospheric Delay 43
  - 3.6.4 Code Multipath and Code Tracking Noise 44
  - 3.6.5 Total Pseudorange Error 48
  - 3.6.6 Range Rates 50

4 Odometry Preprocessing 53
4.1 Observables . . . 53
4.2 Tire Slip Estimation . . . 55
4.2.1 Linear Tire Model . . . 57
4.2.2 Magic Formula Tire Model . . . 60
4.2.3 Side Slip Estimation Based on Gyroscope Measurements . . . 61
4.3 Measurement Noise . . . 63
4.3.1 Noise Determination for Individual Wheels . . . 63
4.3.2 Correlation in Between the Wheels . . . 68

5 Quality Assessment 73
5.1 Performance Metrics for Localization Algorithms . . . 73
5.1.1 Accuracy . . . 73
5.1.2 Integrity . . . 74
5.1.3 Availability . . . 75
5.1.4 Continuity . . . 75
5.1.5 Control Engineering Metrics for Closed-Loop Systems . . . 76
5.2 Internal Quality Assessment . . . 77
5.3 External Quality Assessment . . . 82

6 Integration Filter 85
6.1 Overall Architecture . . . 85
6.2 System Model . . . 88
6.2.1 Initialization . . . 89
6.2.2 System Noise . . . 90
6.3 Measurement Models . . . 92
6.3.1 Pseudorange Update . . . 92
6.3.2 Range Rate Update . . . 93
6.3.3 Odometry Update . . . 93
6.3.4 Delayed Availability of Measurement Data . . . 94
6.4 Outlier Detection . . . 95
6.4.1 Innovation Monitoring . . . 95
6.4.2 Residual Monitoring . . . 96

7 Results 99
7.1 Test Scenarios . . . 99
7.2 Accuracy . . . 100
7.2.1 Reference Solution Quality . . . 100
7.2.2 Improvements Through Multi-Frequency/Multi-Constellation GNSS . 101
7.2.3 Improvements Through Enhanced Odometry Processing . . . 106
7.3 Integrity . . . 113

8 Conclusion and Outlook 121
8.1 Conclusion . . . 121
8.2 Outlook . . . 122

References 130

Own Publications 131

Appendix 133

A Test Scenario Details 133
- A.1 Maps of the Routes 133
- A.2 GNSS Reception Conditions 138
  - A.2.1 PDOP and Number of Available Satellites 138
  - A.2.2 Code Multipath Error 140
  - A.2.3 Range Rate Error 147
  - A.2.4 Single-Epoch Position Solution Accuracy 148
  - A.2.5 Reference Solution Quality 154

B Additional Results 157
- B.1 Differential Code Biases 157
- B.2 GNSS Measurement Noise 161
- B.3 Odometry Measurement Noise 168
- B.4 Test Scenario Results 170
  - B.4.1 Position Accuracy 170
  - B.4.2 Velocity Accuracy 180
  - B.4.3 Yaw Angle Accuracy 183
  - B.4.4 Dynamic Tire Radius Accuracy 186
  - B.4.5 Position Integrity 192
  - B.4.6 Velocity Integrity 195
  - B.4.7 Yaw Angle Integrity 198

C Additional Equations 201
- C.1 System Model 201
- C.2 Kalman Filter Residual Covariance 202

Notation

$\mathbf{a}$	a vector
$\mathbf{a}^{o}$	a vector **a**, resolved in the *o*-frame
$\mathbf{a}^{o}_{pq}$	vector quantity **a** of q w. r. t. p, resolved in the *o*-frame

$\mathbf{A}$	a matrix
$\mathbf{A}^{\mathsf{T}}$	matrix transpose
$\mathbf{A}^{-1}$	matrix inverse
$\mathbf{0}$	zero matrix
$\mathbf{I}$	identity matrix

$\lVert\mathbf{a}\rVert$	Euclidean norm of vector **a**
$\mathbf{a} \times \mathbf{b}$	cross product
$[\mathbf{a}\times]$	cross product-forming matrix, $[\mathbf{a}\times]\,\mathbf{b} = \mathbf{a} \times \mathbf{b}$
$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}$	a block matrix, consisting of four submatrices

$\breve{a}$	a true quantity (usually unknown)
$\tilde{a}$	a measured quantity
$\hat{a}$	an estimated quantity
$\delta\tilde{a} = \tilde{a} - \breve{a}$	a measurement error
$\delta\hat{a} = \hat{a} - \breve{a}$	an estimation error
$\bar{a}$	arithmetic mean of a
a^{-}	an a-priori quantity
a^{+}	an a-posteriori quantity
a^{c}	a corrected quantity
a_{k}	a quantity a, given at a discrete time epoch k

$\dot{a}$,$\ddot{a}$,$\dddot{a}$	time derivatives of a
$\left.\frac{\partial \mathbf{a}}{\partial \mathbf{b}}\right\rvert_{\mathbf{c}}$	partial derivative of **a** w. r. t. **b**, evaluated at **c**

List of Symbols

Coordinate Frames

b	body frame, ordered front-left-up
e	Earth-centered Earth-fixed frame
f, r	individual wheel frame front/rear for single-track model
fl, fr, rl, rr	individual wheel frame front left/front right/rear left/rear right
i	Earth-centered inertial frame
n	local navigation frame, ordered east-north-up
w	arbitrary wheel frame

$y^b_{pq,F}$, $y^b_{pq,L}$, $y^b_{pq,U}$	front/left/up component of vector $\mathbf{y}^b_{pq}$
$y^n_{pq,E}$, $y^n_{pq,N}$, $y^n_{pq,U}$	east/north/up component of vector $\mathbf{y}^n_{pq}$
$y^o_{pq,x}$, $y^o_{pq,y}$, $y^o_{pq,z}$	$x/y/z$-component of vector $\mathbf{y}^o_{pq}$

GNSS

B_p	pseudorange/code bias for signal p
B^R_p	receiver-specific part of B_p
B^S_p	satellite-specific part of B_p
c	speed of light
c/n_0	carrier-to-noise ratio (in $\frac{\text{W}}{\text{W/Hz}}$)
C/N_0	carrier-to-noise ratio (in dB-Hz)
$d_{p,q}$	differential code bias between the signals p and q
$d^R_{p,q}$	receiver-specific part of $d_{p,q}$
$d^S_{p,q}$	satellite-specific part of $d_{p,q}$
$D_{p,q}$	inter-system bias between the constellations p and q
f	frequency
f_{ca}, f_{co}, f_{nav}	carrier frequency/code chipping rate/navigation data rate
f_{sca}, f_{sco}	subcarrier frequency/secondary code chipping rate
$\mathbf{G}$	geometry matrix
N_{ca}	carrier phase wavelength ambiguity
$\mathbf{u}_{as}$	unit vector pointing from receiver antenna to satellite antenna

$\gamma_{p,q}$	squared ratio of two (carrier) frequencies f_p and f_q, $\gamma_{pq} = \left(\frac{f_p}{f_q}\right)^2$
δt_R	receiver clock bias
$\delta \dot{t}_R$	receiver clock drift
δt_S	satellite clock bias
$\delta \dot{t}_S$	satellite clock drift
$\delta\rho_I$	ionospheric delay
$\delta\rho_M$	pseudorange/code multipath error
$\delta\rho_N$	pseudorange/code tracking noise
$\delta\rho_T$	tropospheric delay

$\delta\phi_M$	carrier phase multipath error
$\delta\phi_N$	carrier phase tracking noise
ζ	azimuth
θ	elevation
λ_{ca}	carrier wavelength
ρ	pseudorange
$\dot{\rho}$	pseudorange rate
ϕ	carrier phase (in rad)

IMU

$\mathbf{b}_a$	3-D vector of accelerometer biases
$\mathbf{b}_\omega$	3-D vector of gyroscope biases
$\mathbf{f}_{ib}$	specific force acting on the *b*-frame w. r. t. the *i*-frame
g_e	normal gravity at the equator
g_φ	normal gravity at ellipsoidal latitude φ
$\mathbf{g}_{ib}$	gravity vector of the *b*-frame w. r. t. the *i*-frame
$\check{\mathbf{g}}_{ib}$	gravitation vector of the *b*-frame w. r. t. the *i*-frame
$\mathbf{w}_a$	accelerometer measurement noise
$\mathbf{w}_\omega$	gyroscope measurement noise

Kalman Filter

Unless stated otherwise, all quantities related to the Kalman filter are given as their discrete-time version.

$\mathbf{h}$	measurement function, $\tilde{\mathbf{z}} = \mathbf{h}(\mathbf{x}) + \mathbf{w}_m$
$\mathbf{H}$	measurement matrix, $\mathbf{H} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}$
$\mathbf{K}$	Kalman gain matrix
$\mathbf{n}$	state function, $\dot{\mathbf{x}} = \mathbf{n}(\mathbf{x}) + \check{\mathbf{w}}_s$
$\mathbf{N}$	continuous-time system matrix, $\mathbf{N} = \frac{\partial \mathbf{n}}{\partial \mathbf{x}}$
$\mathbf{P}$	error covariance matrix
$\mathbf{Q}$, $\check{\mathbf{Q}}$	system noise/continuous-time system noise covariance matrix
$\mathbf{R}$	measurement noise covariance matrix
$\mathbf{w}_m$	measurement noise vector
$\mathbf{w}_s$, $\check{\mathbf{w}}_s$	system noise/continuous-time system noise vector
$\mathbf{x}$	total state vector
$\mathbf{z}$, $\tilde{\mathbf{z}}$	error state formulation/total measurement vector
$\delta\mathbf{x}$	error state vector
$\delta\mathbf{z}^-$	measurement innovation, $\delta\mathbf{z}^- = \mathbf{z} - \mathbf{h}(\hat{\mathbf{x}}^-)$
$\delta\mathbf{z}^+$	measurement residual, $\delta\mathbf{z}^+ = \mathbf{z} - \mathbf{h}(\hat{\mathbf{x}}^+)$
τ	time interval
τ_s	state propagation interval
τ_g, τ_i, τ_o	measurement interval of GNSS/IMU/odometry
$\boldsymbol{\Phi}$	transition matrix

Kinematics

$\mathbf{a}_{pq}$	acceleration vector of q w. r. t. p
$\mathbf{C}_q^p$	direction-cosine matrix, $\mathbf{v}^p = \mathbf{C}_q^p \mathbf{v}^q$, $\mathbf{v}^q = \mathbf{C}_q^{p\mathsf{T}} \mathbf{v}^p$
h_e	ellipsoidal height
$\mathbf{p}_{pq}$	position vector of q w. r. t. p
R_E	ellipsoidal radius of curvature in east-west direction, i. e. transverse radius of curvature
R_N	ellipsoidal radius of curvature in north-south direction, i. e. meridian radius of curvature
$\mathbf{v}_{pq}$	velocity vector of q w. r. t. p

η_{pq}	roll angle of the q-frame w. r. t. the p-frame
ϑ_{pq}	pitch angle of the q-frame w. r. t. the p-frame
λ_e	ellipsoidal longitude
φ_e	ellipsoidal latitude
ψ_{pq}	yaw angle of the q-frame w. r. t. the p-frame
$\boldsymbol{\psi}_{pq}$	attitude vector of the q-frame w. r. t. the p-frame, $\boldsymbol{\psi}_{pq} = \begin{pmatrix} \eta_{pq} & \vartheta_{pq} & \psi_{pq} \end{pmatrix}^{\mathsf{T}}$
ω	a scalar rotation rate
$\boldsymbol{\omega}_{pq}$	rotation rate vector of q w.r.t. p
$\mathbf{\Omega}_{pq} = [\boldsymbol{\omega}_{pq} \times]$	skew-symmetric matrix of rotation rate vector $\boldsymbol{\omega}_{pq}$
$\boldsymbol{\omega}_{en}$	transport rate (due to motion in a curvilinear frame)
ω_{ie}	scalar Earth rotation rate
$\boldsymbol{\omega}_{ie}$	Earth rotation rate vector

Odometry

$\mathbf{f}_{pq}$	specific force acting on q w. r. t. p
$\mathbf{F}_{pq}$	force acting on q w. r. t. p
h_{CG}	height of center of gravity above road surface (along z^b-axis)
l	wheelbase
l_f, l_r	distance from center of gravity to front/rear axle (along x^b-axis)
m_v	vehicle mass
r_d	dynamic tire radius
$\mathbf{r}_d$	vector containing all dynamic tire radii

α	tire side slip angle
β	vehicle side slip angle
δ_f, $\Delta\delta_f$	mean/difference steering angle at front axle
δ_{fl}, δ_{fr}	front left wheel/front right wheel steering angle
δ_{sw}	steering wheel angle
κ_x	longitudinal tire slip correction factor
λ_x	longitudinal tire slip
μ	friction coefficient
ω_w	wheel rotation rate

Parameter Estimation

$\mathbf{A}$	design matrix (in a Gauss-Markov model)
$\mathbf{d\hat{x}}$	vector of unknown parameters (in a Gauss-Markov model)
$\mathbf{l}$	observation vector (in a Gauss-Markov model)
p	percentage
$\bar{q}$	arithmetic mean of q
r_c	redundancy component
S	power spectral density

$\boldsymbol{\Xi}_{\mathbf{q}}$	matrix of correlation coefficients of $\mathbf{q}$
σ_q	standard deviation of q
$\hat{\sigma}_0^2$	a-posteriori variance factor
$\boldsymbol{\Sigma}_{\mathbf{q}}$	covariance matrix of $\mathbf{q}$
$\boldsymbol{\upsilon}$	improvement vector (in a Gauss-Markov model)

Performance Metrics

e_∞	steady-state error
o_p	peak overshoot
T_p, T_r, T_{s,p_s}	peak time/rise time/settling time for percentage p_s

$\lvert\delta\hat{y}\rvert_{p_q}$	p_q quantile of $\lvert\delta\hat{y}\rvert$
$\varepsilon_{\delta\hat{\mathbf{y}}}$	test quantity for test of agreement between estimated covariance matrix and actual error
$\xi_{\delta\hat{\mathbf{y}},p_\varepsilon}$	limit of the p_ε confidence interval for $\varepsilon_{\delta\hat{\mathbf{y}}}$

List of Acronyms

1-D, 2-D, 3-D, ...	one-dimensional, two-dimensional, three-dimensional, ...
ABS	anti-lock braking system
ANTEX	Antenna Exchange Format
APC	antenna phase center
ARAIM	advanced receiver autonomous integrity monitoring
BGD	Broadcast Group Delay
BOC	binary offset carrier
BPSK	binary phase-shift keying
CAN	Controller Area Network
CDF	cumulative distribution function
CDMA	code-division multiple access
CG	center of gravity
CNAV	Civil Navigation Message
CNES	Centre National d'Etudes Spatiales
C/A	coarse/acquisition
DCB	differential code bias
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DOP	dilution of precision
DOY	day of year
ECEF	Earth-centered Earth-fixed
ECI	Earth-centered inertial
EGNOS	European Geostationary Navigation Overlay Service
EKF	extended Kalman filter
ENU	east-north-up
eRAIM	extended receiver autonomous integrity monitoring
ESC	electronic stability control
ESCAPE	European Safety Critical Applications Positioning Engine
FCDF	folded cumulative distribution function
FLU	front-left-up
F/NAV	Free Navigation Message
GFZ	Deutsches GeoForschungsZentrum
GGTO	GPS to Galileo Time Offset
GLONASS	Global'naya Navigatsionnaya Sputnikova Sistema
GNSS	global navigation satellite system
GPS	Global Positioning System
GPT	Global Pressure and Temperature Model
GRS	Geodetic Reference System
HAS	High Accuracy Service
ICD	interface control document
IERS	International Earth Rotation and Reference Systems Service
IF	ionosphere-free
IGS	International GNSS Service
IMU	inertial measurement unit
INS	inertial navigation system

IONEX	Ionosphere Map Exchange Format
ISB	inter-system bias
ISC	Inter-Signal Correction Term
ISM	integrity support message
I/NAV	Integrity Navigation Message
LNAV	Legacy Navigation Message
MEMS	microelectromechanical systems
MGEX	Multi-GNSS Experiment
NLOS	non-line-of-sight
PDOP	position dilution of precision
PPP	precise point positioning
PRN	pseudo-random noise
PSD	power spectral density
QZSS	Quasi-Zenith Satellite System
RAIM	receiver autonomous integrity monitoring
RINEX	Receiver Independent Exchange Format
RMS	root mean square
RTK	real time kinematic
SBAS	satellite-based augmentation system
SINEX	Solution Independent Exchange Format
SISRE	signal-in-space ranging error
TGD	Timing Group Delay
UEE	user equipment error
UERE	user equivalent range error
WAAS	Wide Area Augmentation System
WGS	World Geodetic System
ZARU	zero angular rate update

List of Figures

1.1 Overall localization algorithm architecture. 4

2.1 ECEF frame and ellipsoidal coordinates. 9
2.2 ENU navigation coordinate frame. 10
2.3 Euler angles: roll η_{nb}, pitch ϑ_{nb} and yaw ψ_{nb}. 11
2.4 Single-track model. 14
2.5 Pacejka's Magic Formula tire model. 16
2.6 Closed-loop GNSS/INS integration architecture. 16

3.1 Receiver-specific DCBs on DOY 64, computed with satellite-specific DCBs from DLR. 34
3.2 Satellite position and clock bias error. 40
3.3 Difference between reference value from IONEX and estimated value for ionospheric delay in the L1 band. 42
3.4 Difference between reference value from SINEX and estimated value for tropospheric delay. 43
3.5 Code multipath and code tracking noise for GPS. 44
3.6 Fitted standard deviation of code tracking noise. 46
3.7 Fitted standard deviation of code multipath measurement noise. 47
3.8 CDF of total pseudorange error for Galileo with precise satellite orbits and clocks. 49
3.9 Relative frequency of total pseudorange error for Galileo with precise satellite orbits and clocks. 49
3.10 Range rate error in static data for varying GNSS measurement intervals. . . . 50
3.11 Range rate error and vehicle speed in kinematic data for varying GNSS reception conditions. 51
3.12 Range rate error in kinematic data and its fitted standard deviation. 52

4.1 Steering angles as a function of the steering wheel angle. 55
4.2 Fitted longitudinal tire slip models. 58
4.3 Fitted lateral tire slip models. 59
4.4 Fitted standard deviation of wheel rotation rate. 64
4.5 Longitudinal slip estimation accuracy with linear tire model. 65
4.6 Longitudinal slip estimation accuracy with Magic Formula tire model. 65
4.7 Side slip angle estimation accuracy with linear tire model, based on accelerometer measurements. 66
4.8 Side slip angle estimation accuracy with Magic Formula tire model, based on accelerometer measurements. 66
4.9 Side slip angle estimation accuracy with Magic Formula tire model, based on gyroscope measurements. 67
4.10 Longitudinal odometry velocity error, front and rear left wheel. 69

4.11 Lateral odometry velocity error when side slip is estimated based on accelerometer measurements. 71
4.12 Lateral odometry velocity error when side slip is estimated based on gyroscope measurements. 71

5.1 Command step response of a closed-loop system. 76
5.2 A-posteriori variance factor for all three types of measurement updates. . . . 78
5.3 Redundancy components and a-posteriori variance factors for yaw angle, velocity and position. 80
5.4 Original and smoothed a-posteriori variance factors for yaw angle, velocity and position. 81

6.1 Overall localization algorithm architecture. 86

7.1 Position error CDF for Griesheim airfield data set, moderate driving style. . . 102
7.2 Position error CDF for Griesheim airfield data set, dynamic driving style. . . 102
7.3 Position error CDF for Odenwald data set. 103
7.4 Position error CDF for Darmstadt data set. 103
7.5 Position error CDF for Frankfurt data set. 104
7.6 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Frankfurt data set. 109
7.7 Horizontal projection of position solution during tunnel in Odenwald data set. 111
7.8 Position error during tunnel in Odenwald data set. 111
7.9 Position error during tunnel in Darmstadt data set. 112
7.10 Position error and its estimated 95 % confidence interval, Griesheim airfield data set. 115
7.11 Horizontal velocity error and its estimated 95 % confidence interval, Griesheim airfield data set. 117
7.12 Yaw angle error, its estimated 95 % confidence interval and yaw rate, Griesheim airfield data set. 119

A.1 Test scenario 1: Griesheim airfield. 133
A.2 Test scenario 2: Odenwald. 134
A.3 Test scenario 3: Darmstadt. 135
A.4 Test scenario 4: Frankfurt. 136
A.5 Test scenario 4: Frankfurt, detailed view. 137
A.6 PDOP and number of available satellites for Griesheim airfield data sets, moderate driving style. 138
A.7 PDOP and number of available satellites for Griesheim airfield data sets, dynamic driving style. 138
A.8 PDOP and number of available satellites for Odenwald data sets. 139
A.9 PDOP and number of available satellites for Darmstadt data sets. 139
A.10 PDOP and number of available satellites for Frankfurt data sets. 139
A.11 FCDF of multipath combination ρ_{MC} on GPS L1 for all test scenarios. 141
A.12 FCDF of multipath combination ρ_{MC} on GPS L2C for all test scenarios. . . . 142
A.13 FCDF of multipath combination ρ_{MC} on GPS L5 for all test scenarios. 143
A.14 FCDF of multipath combination ρ_{MC} on Galileo E1 for all test scenarios. . . 144
A.15 FCDF of multipath combination ρ_{MC} on Galileo E5a for all test scenarios. . . 145
A.16 FCDF of multipath combination ρ_{MC} on Galileo E5b for all test scenarios. . 146
A.17 FCDF of range rate error $\delta\tilde{\dot{\rho}}$ for all test scenarios. 147
A.18 Position error CDF for Griesheim airfield data sets from single-epoch solution, moderate driving style. 149

A.19 Position error CDF for Griesheim airfield data sets from single-epoch solution, dynamic driving style. 149
A.20 Position error CDF for Odenwald data sets from single-epoch solution. 150
A.21 Position error CDF for Darmstadt data sets from single-epoch solution. . . . 150
A.22 Position error CDF for Frankfurt data sets from single-epoch solution. 150
A.23 Position error CDF for Frankfurt data sets from single-epoch solution, logarithmic scale. 151
A.24 Horizontal projection of position solution inside Frankfurt city center, run 1. . 151
A.25 Horizontal projection of position solution inside Frankfurt city center, run 2. . 152
A.26 Horizontal projection of position solution inside Frankfurt city center, run 3. . 153
A.27 Reference solution quality factor distribution for all test scenario runs. 155

B.1 Receiver-specific DCBs on DOY 64, computed with satellite-specific DCBs from DLR. 157
B.2 Receiver-specific DCBs on DOY 65, computed with satellite-specific DCBs from DLR. 158
B.3 Receiver-specific DCBs on DOY 66, computed with satellite-specific DCBs from DLR. 158
B.4 Receiver-specific DCBs on DOY 64, computed with satellite-specific DCBs from the navigation data messages. 159
B.5 Receiver-specific DCBs on DOY 65, computed with satellite-specific DCBs from the navigation data messages. 159
B.6 Receiver-specific DCBs on DOY 66, computed with satellite-specific DCBs from the navigation data messages. 160
B.7 Difference between reference value from IONEX and estimated value for ionospheric delay in the L1 band. 161
B.8 Code multipath and code tracking noise for GPS. 162
B.9 Code multipath and code tracking noise for Galileo. 162
B.10 CDF of total pseudorange error for GPS with precise satellite orbits and clocks. 163
B.11 Relative frequency of total pseudorange error for GPS with precise satellite orbits and clocks. 163
B.12 CDF of total pseudorange error for Galileo with precise satellite orbits and clocks. 164
B.13 Relative frequency of total pseudorange error for Galileo with precise satellite orbits and clocks. 164
B.14 CDF of total pseudorange error for GPS with LNAV satellite orbits and clocks. 165
B.15 Relative frequency of total pseudorange error for GPS with LNAV satellite orbits and clocks. 165
B.16 CDF of total pseudorange error for Galileo with F/NAV satellite orbits and clocks. 166
B.17 Relative frequency of total pseudorange error for Galileo with F/NAV satellite orbits and clocks. 166
B.18 Side slip angle estimation accuracy with linear tire model, based on accelerometer measurements. 168
B.19 Side slip angle estimation accuracy with linear tire model, based on gyroscope measurements. 168
B.20 Longitudinal odometry velocity error, front and rear left wheel. 169
B.21 Longitudinal odometry velocity error, front and rear right wheel. 169
B.22 Position error CDF for Griesheim airfield data set, moderate driving style, run 1. 171
B.23 Position error CDF for Griesheim airfield data set, moderate driving style, run 2. 171
B.24 Position error CDF for Griesheim airfield data set, moderate driving style, run 3. 171
B.25 Position error CDF for Griesheim airfield data set, dynamic driving style, run 1. 172

B.26 Position error CDF for Griesheim airfield data set, dynamic driving style, run 2. 172
B.27 Position error CDF for Griesheim airfield data set, dynamic driving style, run 3. 172
B.28 Position error CDF for Odenwald data set, run 1. 173
B.29 Position error CDF for Odenwald data set, run 2. 173
B.30 Position error CDF for Darmstadt data set, run 1. 174
B.31 Position error CDF for Darmstadt data set, run 2. 174
B.32 Position error CDF for Darmstadt data set, run 3. 174
B.33 Position error CDF for Frankfurt data set, run 1. 175
B.34 Position error CDF for Frankfurt data set, run 2. 175
B.35 Position error CDF for Frankfurt data set, run 3. 175
B.36 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Griesheim airfield data set, moderate driving style, run 1. 186
B.37 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Griesheim airfield data set, moderate driving style, run 2. 187
B.38 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Griesheim airfield data set, dynamic driving style, run 3. 187
B.39 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Odenwald data set, run 1. 188
B.40 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Darmstadt data set, run 2. 188
B.41 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Frankfurt data set, run 1. 189
B.42 Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Frankfurt data set, run 2. 189

List of Tables

3.1 Contributions to the GNSS UERE 28
3.2 RINEX-Style Codes for the GNSS Signals Utilized in This Thesis 30
3.3 Receiver-Specific DCBs, Computed With Satellite-Specific DCBs From DLR . 35
3.4 Receiver-Specific DCBs, Computed With Satellite-Specific DCBs From the Navigation Data Messages 35
3.5 Accuracy and Latency of GPS Satellite Orbits and Clocks 38
3.6 Fitted Parameters of Code Tracking Noise Magnitude Function 46
3.7 Fitted Parameters of Code Multipath Error Magnitude Function 47
3.8 Total Pseudorange Error With Precise Satellite Orbits and Clocks 50

4.1 Estimated Parameters and Their Standard Deviations for Magic Formula Tire Model 61

7.1 Position Accuracy Metrics 105
7.2 Velocity Accuracy Metrics 107
7.3 Yaw Angle Accuracy Metrics 108
7.4 Dynamic Tire Radius Accuracy Metrics 110
7.5 Position Integrity Metrics 113
7.6 Velocity Integrity Metrics 116
7.7 Yaw Angle Integrity Metrics 118

A.1 RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on GPS L1 for all Test Scenarios 141
A.2 RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on GPS L2C for all Test Scenarios 142
A.3 RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on GPS L5 for all Test Scenarios 143
A.4 RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on Galileo E1 for all Test Scenarios 144
A.5 RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on Galileo E5a for all Test Scenarios 145
A.6 RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on Galileo E5b for all Test Scenarios 146
A.7 RMS, 99 % Quantile and Maximum of Range Rate Error $\delta\tilde{\dot{\rho}}$ for all Test Scenarios 147
A.8 Quality Factor Description and Associated 3-D Accuracy for Novatel's Waypoint - Inertial Explorer Software 154

B.1 Total Pseudorange Error With Precise Satellite Orbits and Clocks 167
B.2 Total Pseudorange Error With Broadcast Satellite Orbits and Clocks 167
B.3 Position Accuracy Metrics for Griesheim Airfield Data Sets 176
B.4 Position Accuracy Metrics for Odenwald Data Sets 177
B.5 Position Accuracy Metrics for Darmstadt Data Sets 178

B.6 Position Accuracy Metrics for Frankfurt Data Sets 179
B.7 Velocity Accuracy Metrics for Griesheim Airfield Data Sets 180
B.8 Velocity Accuracy Metrics for Odenwald Data Sets 181
B.9 Velocity Accuracy Metrics for Darmstadt Data Sets 181
B.10 Velocity Accuracy Metrics for Frankfurt Data Sets 182
B.11 Yaw Angle Accuracy Metrics for Griesheim Airfield Data Sets 183
B.12 Yaw Angle Accuracy Metrics for Odenwald Data Sets 184
B.13 Yaw Angle Accuracy Metrics for Darmstadt Data Sets 185
B.14 Yaw Angle Accuracy Metrics for Frankfurt Data Sets 185
B.15 Dynamic Tire Radius Accuracy Metrics for Griesheim Airfield Data Sets . . . 190
B.16 Dynamic Tire Radius Accuracy Metrics for Odenwald Data Sets 190
B.17 Dynamic Tire Radius Accuracy Metrics for Darmstadt Data Sets 191
B.18 Dynamic Tire Radius Accuracy Metrics for Frankfurt Data Sets 191
B.19 Position Integrity Metrics for Griesheim Airfield Data Sets 192
B.20 Position Integrity Metrics for Odenwald Data Sets 193
B.21 Position Integrity Metrics for Darmstadt Data Sets 193
B.22 Position Integrity Metrics for Frankfurt Data Sets 194
B.23 Velocity Integrity Metrics for Griesheim Airfield Data Sets 195
B.24 Velocity Integrity Metrics for Odenwald Data Sets 196
B.25 Velocity Integrity Metrics for Darmstadt Data Sets 196
B.26 Velocity Integrity Metrics for Frankfurt Data Sets 197
B.27 Yaw Angle Integrity Metrics for Griesheim Airfield Data Sets 198
B.28 Yaw Angle Integrity Metrics for Odenwald Data Sets 199
B.29 Yaw Angle Integrity Metrics for Darmstadt Data Sets 199
B.30 Yaw Angle Integrity Metrics for Frankfurt Data Sets 200

Chapter 1

Introduction

This thesis details the elements of a localization algorithm that inputs data from *global navigation satellite systems* (GNSS), an *inertial navigation system* (INS) and odometry sensors. New GNSS signals are available to civil users, offering many new possibilities and challenges regarding their processing. This chapter gives an introduction by first laying out the motives behind the underlying research and potential application areas in Section 1.1. Afterwards, the aim of the thesis is presented in Section 1.2. Main aspects of that section are the thesis' design novelties in comparison to already existing designs and a basic outline of how the improvements brought forth by these novelties are evaluated. The chapter closes with an outlook on the remaining chapters in Section 1.3.

1.1 Motivation

In the wake of the advent of autonomously driving cars, the role of localization algorithms for automotive applications changes. They are no longer used exclusively for navigation, but provide input for the control of the vehicle's movement, as well. Therefore, the requirements that these localization algorithms have to meet increase drastically. Moreover, localization algorithms become safety-critical systems, so new requirements in addition to the ones concerning accuracy are necessary. The integration of data stemming from a GNSS receiver on the one hand and an *inertial measurement unit* (IMU) on the other hand is common practice for navigation systems, because these two sensor types offer complementary benefits and drawbacks [*Groves*, 2013, p. 559]. Traditionally, a single-frequency GNSS receiver tracking signals only in the L1 band is applied, with the *Global Positioning System* (GPS) as only employed constellation and its *coarse/acquisition* (C/A) code on L1 as the only employed signal. This approach is suitable for navigation purposes, but it is limited in terms of accuracy and satellite availability. The modernized satellite constellations of GPS and its Russian counterpart, the *Global'naya Navigatsionnaya Sputnikova Sistema* (GLONASS), offer multiple carrier frequencies modulated with signals designated for civilian use. The same is true for the European Galileo and the Chinese BeiDou constellations, both of which are currently under commissioning, but feature active and usable satellites already. Due to these developments, the classical single-frequency/single-constellation approach can be replaced with a multi-frequency/multi-constellation architecture, offering increased performance but posing new challenges, as well.

Most of the research concerning processing techniques for multi-frequency/multi-constellation GNSS observations addresses geodetic applications, because the equipment cost of the corresponding receivers made them infeasible for automotive applications in the past. This is no longer the case since multi-frequency/multi-constellation GNSS receivers are now available for mass-market applications down to smartphone level [*Mongrédien et al.*, 2018; *Warnant et al.*, 2018]. The optimal fusion of GNSS observations from several constellations on multiple frequencies, IMU measurements and data from additional sensors is therefore a research area with outstanding relevance for future automotive localization algorithms. An obvious choice for such additional data are odometry sensors that are standard equipment in any modern car anyway, namely wheel rotation rate and steering wheel angle sensors. For an optimal sensor fusion, detailed information about the underlying measurement models of all employed sensors and their associated errors is necessary.

Current research in the field of automotive localization algorithms deals with positioning techniques based on carrier phase measurements, mainly *real time kinematic* (RTK) positioning and *precise point positioning* (PPP). While a lot of effort is put into making these techniques more robust to imperfect GNSS reception conditions, they are still unable to work reliably in a substantial percentage of surroundings that occur in everyday road traffic. Moreover, they require a steady stream of external data such as observations from a reference station or precise satellite orbits and clocks. These data are typically distributed via mobile internet connection, which cannot be guaranteed in all areas. In order to remain operational, localization algorithms for automotive applications have to be able to work without external input data and without techniques that require the constant availability of high-quality carrier phase measurements. Depending on the desired application scenario, this ability might be sufficient to fulfill all requirements, or it might be designed to function as a backup for time periods during which the more accurate techniques do not work properly due to difficult surrounding conditions or unavailability of external input data streams. A decision on whether a system consisting of a multi-frequency/multi-constellation GNSS receiver, an IMU and odometry sensors is able to achieve the performance necessary for an envisioned application without the help of additional sensors or external input data is only possible if that system's behavior is examined thoroughly and quantified via suitable metrics.

1.2 Aim of the Thesis

This thesis' main goal is the development and evaluation of an automotive localization algorithm that inputs data from a multi-frequency/multi-constellation GNSS receiver, a *micro-electromechanical systems* (MEMS) IMU and the vehicle's built-in odometry sensors. The envisioned algorithm works without a steady stream of external input data, therefore it does not utilize carrier phase positioning techniques such as RTK and PPP. The only input data that do not stem from the sensors themselves or from the broadcast satellite signals are predicted precise satellite orbits and clocks. These are downloaded once at the beginning of each measurement scenario. They can be replaced by data from the satellites' navigation data messages if this download fails for some reason, however. Due to its intended employment in the automotive sector, the localization algorithm is required to work in real time, i. e. it needs to output a navigation solution based solely on data that are available at the current point in time.

One aim is to identify the limitations of this setup, i. e. to find out what performance can be achieved without carrier phase positioning techniques or supplementary sensors (e. g. cameras, lidar, radar, etc.) and under which circumstances some form of additional aiding is necessary. In order to facilitate the later inclusion of such data sources, the developed algorithm has a modular architecture, allowing easy replacement and addition of components for maximum flexibility regarding the sensor setup.

This thesis' most important contribution is the inclusion of pseudorange measurements from two constellations (GPS and Galileo) on multiple carrier frequencies into a tightly coupled integration filter. Additional constellations increase the number of available satellites, which is especially relevant when a large portion of the sky is invisible due to surrounding buildings, trees, etc. Pseudoranges from multiple frequencies allow for the formation of *ionosphere-free* (IF) linear combinations, thereby eliminating the largest individual contributor to the overall pseudorange error budget. Because the signals beyond GPS L1 C/A feature a more modern design, they also offer benefits in terms of multipath characteristics and received signal power due to properties like higher code chipping rate, pilot signals free of navigation data and advanced modulation techniques. Due to their dissimilar design and their different frequencies, a receiver can acquire and track some satellite signals easier than others. Because signal propagation properties depend on the surroundings, the acquisition and tracking abilities of a particular receiver vary over time. Consequently, amount and type of available pseudorange observations from a specific satellite are not constant. Due to the stepwise modernization of the GPS constellation, there are currently active GPS satellites from four different generations in orbit [*Navigation Center*, 2019]. Satellites from newer generations broadcast more signals than the older ones, which increases the variation in the amount and type of available pseudoranges even further. Therefore, the localization algorithm needs to be able to process different types of pseudorange observations simultaneously in order to maximize the number of available satellites in each measurement epoch.

Each type of pseudorange observation exhibits a different error characteristic, so a dedicated error model that takes these varying characteristics into account is required to obtain an optimal integrated navigation solution. Therefore, another major aspect of this thesis is to develop and parametrize error models for all observations that are processed in the localization algorithm. For GNSS processing, this applies to pseudoranges and linear combinations of them as well as range rates, which are derived from carrier phase observations at consecutive measurement epochs. For odometry processing, it applies to the velocity vectors at the wheel contact patches, which are derived from the steering wheel angle and wheel rotation rate measurements. An important aspect of the odometry error model is the inclusion of correlation in between the velocity errors at different wheels, which is typically neglected in existing models. The error model for the IMU measurements is comparatively basic, because the developed localization algorithm applies established techniques for the processing of IMU observations, enabling the utilization of an established error model, as well.

The performance improvements provided by the novelties developed in this thesis need to be evaluated and compared with the existing techniques in order to characterize the envisioned algorithm's potential as well as its limitations. This evaluation process requires the identification of suitable metrics to quantify the algorithm's capabilities. Part of this thesis therefore focuses on the determination of a proper set of metrics to achieve this task. While the concept of accuracy remains relevant, other classes of metrics such as integrity, availability and continuity are also crucial for safety-critical systems. Of all quantities that are output by an automotive localization algorithm, the position is currently the one that exhibits the largest gap between demanded and achievable performance in terms of both accuracy and integrity. Positioning performance is thus the most relevant metric for quality assessment in this field. Besides position, horizontal velocity and yaw angle are important quantities, as well, because they are typical inputs for an autonomous car's motion controller. All aforementioned contributions regarding multi-frequency/multi-constellation GNSS and advanced error models aim to achieve an integration filter that is optimal in the sense of accuracy, i. e. they try to minimize the estimation error. Integrity monitoring aims to overbound the remaining error and provide information about the output solution's trustworthiness. The localization algorithm presented in this thesis does not include a dedicated integrity monitoring feature. However, it employs measurement consistency checks and dedicated procedures to check the agreement between the parametrized error models and the actual errors of the input quantities. The aim of these techniques is to estimate the current error of the output solution as good as possible. In the end, a conclusion is drawn about whether the combination of these procedures is able to achieve a useful degree of integrity or if a dedicated integrity monitoring feature is necessary to accomplish this task.

1.3 Thesis Outline

The thesis consist of eight chapters in total. After this introduction, Chapter 2 lays out the groundwork by defining the necessary nomenclature and the utilized coordinate frames. Afterwards, the relevant principles of GNSS, odometry and GNSS/INS integration are explained before the chapter closes with an overview about the current research in the fields of GNSS in general and GNSS/INS integration for automotive applications in particular. The partitioning into GNSS, odometry and GNSS/INS integration originates from the localization algorithm's architecture as depicted in Figure 1.1. The three sensor groups on the left provide the input data, which get preprocessed by the modules in the middle column before they are fused by the integration filter. The same structure is applied to the chapters of this thesis, as well. There is one chapter dedicated to GNSS preprocessing, odometry preprocessing and the integration filter, respectively. No chapter dedicated to INS exists because the IMU measurements are processed with established methods without novel aspects. The application of these methods is part of Chapter 6. All results regarding the processing of data within a particular module are featured in that module's chapter. The results achieved through the collaboration of all modules are presented in the dedicated Chapter 7.

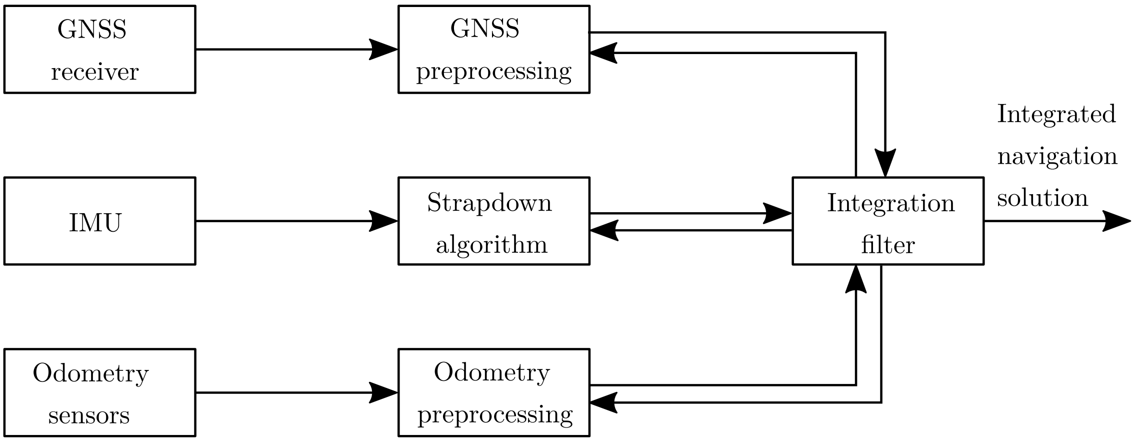

Figure 1.1: Overall localization algorithm architecture.

Chapter 3 details how the GNSS receiver's measurements get preprocessed on their way to the integration filter. This encompasses the calculation of range rates from carrier phase measurements, the compensation of a-priori known errors and the calculation of the measurement noise covariance matrix. Due to the inclusion of multi-frequency/multi-constellation pseudoranges, some effects have to be considered that do not play a role for single-frequency/single-constellation schemes, with pseudorange biases being the most prominent example of such effects.

The analogous steps for the odometry observations are described in Chapter 4. Central parts of this preprocessing module are measurement noise variance computation as well as the estimation of longitudinal and lateral tire slip. The latter is performed with tire models of varying complexity to examine the effect that these models have on the overall accuracy of the integrated navigation solution. During measurement noise variance computation, special emphasis is placed on the correlation of errors in between the wheels.

Chapter 5 deals with methods concerning quality assessment. After reviewing existing metrics for performance assessment of localization algorithms, a suitable set of metrics for the algorithm designed in this thesis is chosen based on previously defined criteria. An important part of that chapter are techniques employed by the localization algorithm to estimate the error magnitude of its own output quantities.

The fusion of all measurement data in the central integration filter is the topic of Chapter 6. It encompasses a description of the strapdown algorithm, which is responsible for the preprocessing of IMU data, as well as details about the integration filter's system and measurement models. To prevent contamination of the integrated navigation solution by faulty observations, methods for the detection and elimination of outliers are featured in that chapter, too.

Chapter 7 presents the results achieved with the developed algorithm in numerous test scenarios. These test scenarios reflect the range of operating conditions faced by automotive localization algorithms, especially in terms of GNSS reception conditions. The improvements brought forth by the design novelties from the previous chapters are highlighted through comparison with the existing techniques. These improvements are quantified with the help of the relevant metrics identified in Chapter 5.

The thesis ends by drawing conclusions about its achievements and giving an outlook on potential further developments in Chapter 8.

Chapter 2

Background and State of the Art

This chapter introduces the basic principles concerning the work presented in this thesis. These basics include coordinate frames (Section 2.1), GNSS (Section 2.2), odometry (Section 2.3) and GNSS/INS integration (Section 2.4). The focus lies on the aspects of the respective topics that are relevant to the following chapters. As the main novelties of this thesis regard GNSS processing and integrated navigation systems for automotive applications, an overview of the research that is currently done in these fields and relevant to this thesis is provided in Section 2.5.

2.1 Coordinate Frames

As this thesis deals with the estimation of an object's position, attitude and movement, different coordinate frames are necessary to express these quantities quantitatively. This section defines the most important coordinates frames that appear within this thesis in its Sections 2.1.2-2.1.5. Before introducing the coordinate frames, the notation to describe quantities within these frames is defined in a generalized way in Section 2.1.1. The content of this section is based on *Groves* [2013, Chapter 2].

2.1.1 Notation for Quantities Within Coordinate Frames

Quantities such as position or velocity describe the spatial relationship of coordinate frames w. r. t. each other. Each such quantity $\mathbf{y}$ usually involves three (not necessarily distinct) coordinate frames:

1. The reference frame p, w. r. t. which the quantity $\mathbf{y}$ is described.
2. The object frame q, whose kinematics are described.
3. The resolving frame o, in which the quantity $\mathbf{y}$ is expressed.

In order to specify a quantity $\mathbf{y}$, the symbols for reference and object frame are notated as subscripts, while the symbol for the resolving frame is notated as superscript:

$$\mathbf{y}_{pq}^{o}. \tag{2.1}$$

Position $\mathbf{p}$, velocity $\mathbf{v}$, acceleration $\mathbf{a}$ and rotation rate $\boldsymbol{\omega}$ require the specification of all three frames. For attitude $\boldsymbol{\psi}$, only reference frame p and object frame q are meaningful specifications. Hence, the resolving frame o is omitted. In case any of the involved coordinate frames is irrelevant, it may be omitted, as well. While this notation is defined here for quantities that describe spatial relationships, it is applied in a generalized way to include other vector quantities, e. g. forces, in the course of this thesis.

2.1.2 Earth-Centered Inertial Frame

An inertial coordinate frame is defined as having zero acceleration and zero rotation w. r. t. the rest of the Universe. An approximation of this concept is the *Earth-centered inertial* (ECI) frame, denoted by the letter i. Its origin lies in the Earth's center of mass. The z^i-axis points along the Earth's mean axis of rotation toward the north pole, implying that the other two axes lie in the equatorial plane. The x^i-axis is defined as pointing toward the Sun at the vernal equinox. Finally, the y^i-axis completes the right-handed set. Due to the movement of the Earth around the Sun, the slow variations of the Earth's axis of rotation and other effects, the ECI frame is not strictly an inertial frame. It is treated as an inertial frame in this thesis because the deviations from a true inertial frame are well below the measurement noise level exhibited by the employed navigation sensors. This frame is primarily important due to two aspects: On the one hand, the measurements of an IMU are made relative to this frame. On the other hand, the orbital plane of a satellite does not change its orientation w. r. t. the ECI frame in a two-body Kepler problem.

2.1.3 Earth-Centered Earth-Fixed Frame

The *Earth-centered Earth-fixed* (ECEF) frame has the same origin and the same z-axis as the ECI frame. In contrast to the ECI frame, the axes of the ECEF frame are fixed w. r. t. the Earth. That's why it is denoted by the letter e. The x^e-axis points to the intersection of the equator with the prime meridian, which is taken to be the reference meridian of the *International Earth Rotation and Reference Systems Service* (IERS). The y^e-axis completes the right-handed set and points toward the 90° east meridian. The ECEF frame rotates w. r. t. the ECI frame along the common z-axis with the rotation rate ω_{ie}.

To get a more descriptive depiction of navigation problems than with Cartesian coordinates, an approximation to the Earth's surface called reference ellipsoid is defined. As *IS-GPS-200* [2018] dictates, the reference ellipsoid to be used by GPS users is the one of the *World Geodetic System* (WGS) 84. Figure 2.1 depicts how positions are specified relative to the ellipsoid by ellipsoidal longitude $\lambda_e \in (-\pi, \pi]$, ellipsoidal latitude $\varphi_e \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and ellipsoidal height h_e. Sign conventions in this thesis are: $\lambda_e > 0$ in the eastern hemisphere, $\varphi_e > 0$ in the northern hemisphere and $h_e > 0$ for points above the surface of the reference ellipsoid. Together, these three quantities form the position vector $\mathbf{p}^e$ in the ECEF frame:

$$\mathbf{p}^e = \begin{pmatrix} \lambda_e \\ \varphi_e \\ h_e \end{pmatrix} . \tag{2.2}$$

2.1.4 Body Frame

The origin of the body frame is the object whose motion is described by the navigation solution. In this thesis, the body frame's origin is placed in the IMU's reference point and the moving object is a vehicle. The body frame's axes are fixed w. r. t. this vehicle and the frame is denoted by the letter b. The x^b-axis points forward, the y^b-axis to the left and the z^b-axis upward. Consequently, the body frame defined in this way is called a *front-left-up* (FLU) frame. The IMU's sensor axes are assumed to be aligned with the vehicle's body axes. The three components of a vector $\mathbf{y}^b_{pq}$ are denoted by $y^b_{pq,F}$, $y^b_{pq,L}$ and $y^b_{pq,U}$.

In addition to the main body frame, special frames for specific objects may be defined, e. g. an individual wheel, one of the vehicle's axles or the GNSS antenna. These special frames are defined whenever necessary and they are described in the respective chapters.

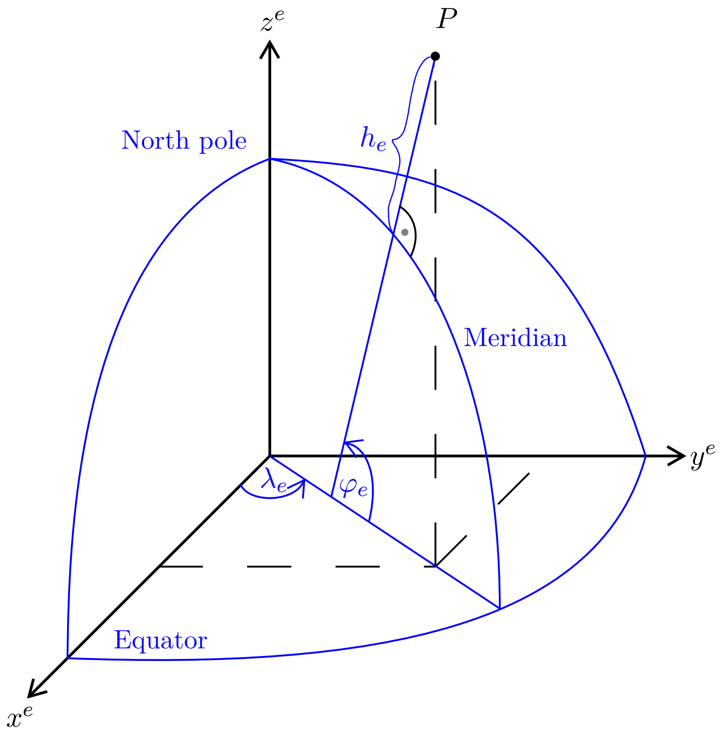

Figure 2.1: ECEF frame and ellipsoidal coordinates.

2.1.5 Navigation Frame

The navigation frame's origin coincides with the one of the body frame, i. e. the IMU's reference point. This frame is denoted by the letter n. As shown in Figure 2.2, the navigation frame's z^n-axis is defined as the normal to the reference ellipsoid's surface, pointing away from the center of the Earth. The y^n-axis points north, the x^n-completes the right-handed set and points east. This definition of the navigation frame is called an *east-north-up* (ENU) frame. The three components of a vector $\mathbf{y}^n_{pq}$ are denoted by $y^n_{pq,E}$, $y^n_{pq,N}$ and $y^n_{pq,U}$.

The specific force $\mathbf{f}_{ib}$ measured by an IMU w. r. t. the ECI frame is the sum of acceleration of the body frame w. r. t. to the ECI frame $\mathbf{a}_{ib}$ and a gravitational component $\check{\mathbf{g}}_{ib}$:

$$\mathbf{f}_{ib} = \mathbf{a}_{ib} + \check{\mathbf{g}}_{ib}. \tag{2.3}$$

When stationary w. r. t. the ECEF frame, the specific force sensed by the IMU stems from the reaction force due to gravity. Gravity comprises both a gravitational and a centrifugal component. In order to determine the navigation solution, the reaction force due to gravity needs to be compensated by a gravity model. The gravity model utilized in this thesis assumes that the gravity vector $\mathbf{g}_{ib}$ is parallel to the z^n-axis.

$$\mathbf{g}^n_{ib} = \begin{pmatrix} 0 \\ 0 \\ g^n_{ib,U} \end{pmatrix} \tag{2.4}$$

The value for $g^n_{ib,U}$ is given by the *Geodetic Reference System* (GRS) 80 [*Moritz*, 1980] and includes a height correction term according to *Li and Götze* [2001]:

$$g^n_{ib,U} = g_\varphi - (c_{h1} - c_{\varphi h}\sin^2\varphi_e)h_e + c_{h2}h_e^2 \tag{2.5}$$

$$g_\varphi = g_e(1 + c_{\varphi 2}\sin^2\varphi_e + c_{\varphi 4}\sin^4\varphi_e + c_{\varphi 6}\sin^6\varphi_e + c_{\varphi 8}\sin^8\varphi_e), \tag{2.6}$$

where g_e is the normal gravity at the equator.

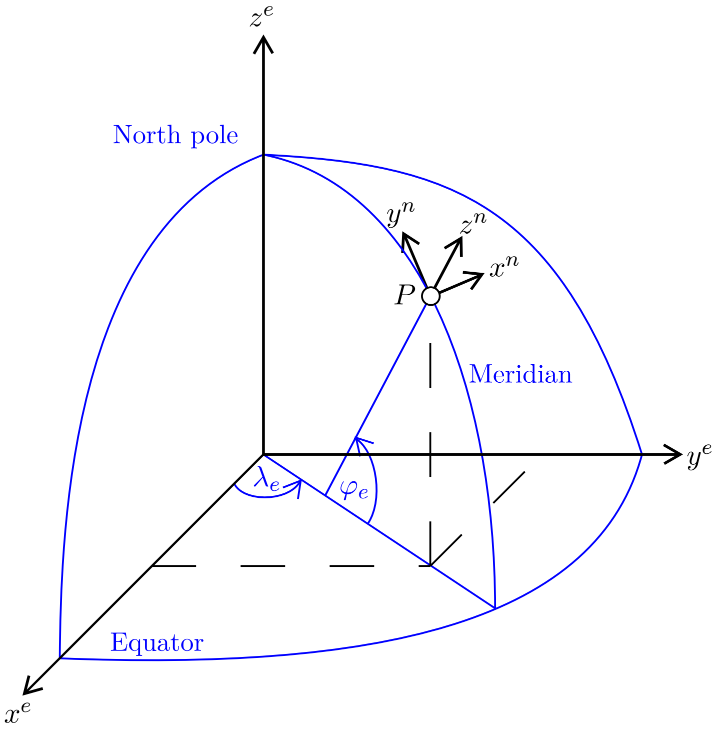

Figure 2.2: ENU navigation coordinate frame.

Body and navigation frame share a common origin, but are oriented differently. These different orientations are described by a set of three Euler angles: roll angle $\eta_{nb} \in (-\pi, \pi]$, pitch angle $\vartheta_{nb} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and yaw angle $\psi_{nb} \in (-\pi, \pi]$. Together, these three angles form the attitude vector $\boldsymbol{\psi}_{nb}$:

$$\boldsymbol{\psi}_{nb} = \begin{pmatrix} \eta_{nb} \\ \vartheta_{nb} \\ \psi_{nb} \end{pmatrix} . \tag{2.7}$$

In order to be utilized for coordinate frame rotation, the direction-cosine matrix $\mathbf{C}_b^n$ is constructed from $\boldsymbol{\psi}_{nb}$. With $\mathbf{C}_b^n$, the representation of any vector $\mathbf{y}$ in the b-frame is transformed into the representation of the identical vector $\mathbf{y}$ in the n-frame and vice versa:

$$\mathbf{y}^n = \mathbf{C}_b^n \mathbf{y}^b \tag{2.8}$$

$$\mathbf{y}^b = \mathbf{C}_n^b \mathbf{y}^n = \mathbf{C}_b^{n\mathsf{T}} \mathbf{y}^n . \tag{2.9}$$

The concept of Euler angles is generalized to apply to any two coordinate frames p and q by replacing the indices n and b to form the attitude vector $\boldsymbol{\psi}_{pq}$ and the direction-cosine matrix $\mathbf{C}_q^p$.

Figure 2.3 depicts the process of transforming the n-frame into the b-frame. Because rotations do not commute, their order is important. First, the n-frame is rotated around the z^n-axis by the yaw angle ψ_{nb}, forming an intermediate $'$-frame. In the second step, the $'$-frame is rotated around the y'-axis by the pitch angle ϑ_{nb}, forming another intermediate $''$-frame. Finally, the $''$-frame is rotated around the x''-axis by the roll angle η_{nb} to form the b-frame. Due to the way body and navigation frame are defined, the following sign conventions hold:

- $\eta_{nb} > 0$ indicates that the vehicle is tilted to the right.
- $\vartheta_{nb} > 0$ indicates that the vehicle's front is tilted down, e. g. when driving downhill.
- $\psi_{nb} > 0$ indicates that the vehicle's front is turned toward north.

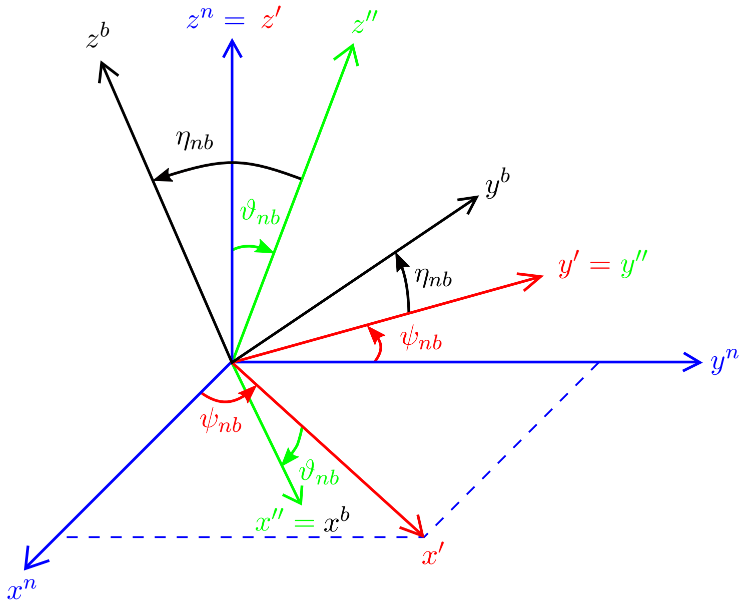

Figure 2.3: Euler angles: roll η_{nb}, pitch ϑ_{nb} and yaw ψ_{nb}.

2.2 GNSS

GNSS satellites orbit the Earth and broadcast navigation signals. These signals feature a spreading code modulated on a carrier wave with frequency f_{ca}. All GNSS satellites relevant to this thesis utilize multiple different carrier frequencies in the L band (1 GHz to 2 GHz) simultaneously. The spreading code is a *pseudo-random noise* (PRN) sequence with chipping rate f_{co}. For current GNSS signals, f_{co} is in the range of 511.5 kHz to 10.23 MHz [*Teunissen and Montenbruck*, 2017, Part B]. Some GNSS signals feature *binary phase-shift keying* (BPSK), while others include additional subcarrier sequences with frequency f_{sca}, resulting in a modulation known as *binary offset carrier* (BOC). Other components that may be modulated onto a signal are so-called secondary codes and navigation data messages with their respective frequencies f_{sco} and f_{nav}. The relevant aspects of the signals utilized in this thesis are detailed in Chapter 3.

By synchronizing an internal replica of the spreading code with the received satellite signal, a GNSS receiver is able to determine the signal's transmission time t^S_{tx} unambiguously. The superscript S indicates that this time is measured by the satellite's on-board clock. The transmission time is subtracted from the reception time t^R_{rx} (measured by the receiver clock) to form the pseudorange ρ:

$$\rho = \left(t^R_{rx} - t^S_{tx}\right) c, \tag{2.10}$$

where c is the speed of light. ρ is called a **pseudo**range because it differs from the true range between transmitting and receiving antenna, mainly due to synchronization errors between the satellite and receiver clocks. To synchronize them, the errors of both clocks w. r. t. a common clock Γ are formed:

$$\delta t_R = t^R - t^\Gamma \qquad\qquad \delta t_S = t^S - t^\Gamma. \tag{2.11}$$

δt_R is the receiver clock bias, δt_S is the satellite clock bias. The common clock Γ is usually a GNSS system time, maintained by the respective constellation's control segment. While δt_R needs to be estimated as part of the navigation solution, δt_S is provided to the navigation algorithm for each satellite, usually via the navigation data message broadcast by the satellite or via precise orbit and clock files, computed by a GNSS analysis center (see Section 3.5).

While pseudorange measurements are based on the spreading code, carrier phase measurements ϕ are based on the signal's carrier wave. In this work, ϕ is an angle and therefore expressed in radians or degrees. To get a length expressed in meters, ϕ is multiplied by the respective carrier wavelength λ_{ca} and divided by 2π or $360°$. Due to the higher frequency, carrier phase measurements are more precise than pseudorange measurements: While the code tracking noise standard deviation of modern receivers is in the order of decimeters, the carrier tracking noise standard deviation is typically less than a millimeter [*Teunissen and Montenbruck*, 2017, p. 579]. But while pseudorange measurements are unambiguous, carrier phase measurements suffer from an ambiguity of an integer multiple of 2π, because the receiver cannot distinguish between different cycles of the carrier wave.

Both pseudorange and carrier phase are range measurements. In order to obtain velocity observations from GNSS signals, different possibilities exist:

1. Forming time derivatives of carrier phase measurements. This eliminates the carrier phase ambiguity as long as no cycle slips occur.
2. Forming time derivatives of code measurements. These are comparatively noisy due to the higher standard deviation of the code tracking noise.
3. Measuring the Doppler shift of the incoming signal.

Independently of the method used to obtain velocity observations, the resulting measurement is the range rate $\dot{\rho}$. It represents the line-of-sight velocity between transmitting and receiving antenna. Range rate measurements utilized in this thesis are based on time derivatives of carrier phase measurements.

Besides pseudorange, carrier phase and range rate measurements, GNSS receivers output the estimated signal power of each tracked signal. This power is expressed in terms of carrier-to-noise ratio. It is either expressed in $\frac{\mathrm{W}}{\mathrm{W/Hz}}$ or in decibel form with the unit dB-Hz. For clarity, the former is denoted by c/n_0 and the latter by the upper case version C/N_0 in this thesis. Conversion between the two is done according to (2.12) [*Groves*, 2013, p. 363].

$$C/N_0 = 10\log_{10}\left(c/n_0\right) \qquad\qquad c/n_0 = 10^{\frac{C/N_0}{10}} \tag{2.12}$$

Pseudorange, carrier phase and range rate measurements are modeled according to (2.13)-(2.15). Because all three are scalars, it does not matter in which coordinate frame the vectors on the right hand side are resolved, as long as all vectors in one equation are resolved in the same coordinate frame. For simplification, the same resolving frames as in Chapter 6 are chosen here. Both ECEF frame and navigation frame rotate. Consequently, the orientation of their axes at t^{Γ}_{rx} differs from the one at t^{Γ}_{tx}. This has to be accounted for when evaluating (2.13)-(2.15).

$$\rho = \left\|\mathbf{p}^{e}_{es}(t^{\Gamma}_{tx}) - \mathbf{p}^{e}_{ea}(t^{\Gamma}_{rx})\right\| + c\delta t_R - c\delta t_S + \delta\rho_I + \delta\rho_T + \delta\rho_M + B + \delta\rho_N \tag{2.13}$$

$$\phi = \frac{2\pi}{\lambda_{ca}}\left(\left\|\mathbf{p}^{e}_{es}(t^{\Gamma}_{tx}) - \mathbf{p}^{e}_{ea}(t^{\Gamma}_{rx})\right\| + c\delta t_R - c\delta t_S - \delta\rho_I + \delta\rho_T\right) + \delta\phi_M + 2\pi N_{ca} + \delta\phi_N \tag{2.14}$$

$$\dot{\rho} = \mathbf{u}^{n\,\top}_{as}\left(\mathbf{v}^{n}_{es}(t^{\Gamma}_{tx}) - \mathbf{v}^{n}_{ea}(t^{\Gamma}_{rx})\right) + c\delta\dot{t}_R - c\delta\dot{t}_S + \delta\dot{\rho}_N \tag{2.15}$$

$\mathbf{p}^{e}_{es}(t^{\Gamma}_{tx})$ is the satellite's position w. r. t. the ECEF frame at the time of signal transmission. $\mathbf{p}^{e}_{ea}(t^{\Gamma}_{rx})$ is the receiver antenna's position w. r. t. the ECEF frame at the time of signal reception. $\delta\rho_I$ and $\delta\rho_T$ are ionospheric and tropospheric delay. The sign of the ionospheric delay is reversed for carrier phase measurements. ρ_M and ϕ_M are the multipath errors for pseudorange and carrier phase. B is the code bias. The corresponding bias term for carrier phase measurements is neglected. $N_{ca} \in \mathbb{Z}$ is the integer wavelength ambiguity. The terms

$\delta\rho_N$ and $\delta\phi_N$ for the code and carrier tracking noise subsume all error terms that have not been listed explicitly. In the range rate model (2.15), $\mathbf{u}^n_{as}$ is the unit vector pointing from receiver to satellite antenna. $\mathbf{v}^n_{es}(t^\Gamma_{tx})$ is the satellite's velocity w. r. t. the ECEF frame at the time of signal transmission. $\mathbf{v}^n_{ea}(t^\Gamma_{rx})$ is the receiver antenna's velocity w. r. t. the ECEF frame at the time of signal reception. $\delta\dot{t}_R$ is the receiver clock drift, $\delta\dot{t}_R$ is the satellite clock drift. As before, the range rate noise term $\delta\dot{\rho}_N$ subsumes all error terms not listed explicitly.

A satellite's position relative to the receiver antenna is described by elevation θ and azimuth ζ. These are related to the unit vector $\mathbf{u}^n_{as}$ by (2.16)-(2.18) [*Groves*, 2013, p. 344].

$$\theta = \arcsin u^n_{as,U} \tag{2.16}$$

$$\zeta = \arctan_2 \left(u^n_{as,E}, u^n_{as,N} \right) \tag{2.17}$$

$$\mathbf{u}^n_{as} = \begin{pmatrix} \cos\theta \sin\zeta \\ \cos\theta\cos\zeta \\ \sin\theta \end{pmatrix} \tag{2.18}$$

Another important quantity related to $\mathbf{u}^n_{as}$ is the *dilution of precision* (DOP). DOP is solely based on the geometric distribution of the received satellites. In order to obtain the DOP, the unit vectors pointing from the receiver antenna to the j received satellites are arranged in the geometry matrix $\mathbf{G}$:

$$\mathbf{G}^n = \begin{pmatrix} -\mathbf{u}^n_{as,1}{}^\intercal & 1 \\ -\mathbf{u}^n_{as,2}{}^\intercal & 1 \\ \vdots & \vdots \\ -\mathbf{u}^n_{as,j}{}^\intercal & 1 \end{pmatrix}. \tag{2.19}$$

Each row of $\mathbf{G}$ is the partial derivative of the respective pseudorange measurement (2.13) w. r. t. the four unknowns: 3-D receiver antenna position and the receiver clock bias. It is commonly resolved in either the e- or the n-frame, depending on the resolving frame of the unit vectors. The DOP values are extracted from the cofactor matrix $\left(\mathbf{G}^\intercal\mathbf{G}\right)^{-1}$ of the navigation solution:

$$\left(\mathbf{G}^{n\intercal}\mathbf{G}^n\right)^{-1} = \begin{pmatrix} \mathrm{EDOP}^2 & \cdot & \cdot & \cdot \\ \cdot & \mathrm{NDOP}^2 & \cdot & \cdot \\ \cdot & \cdot & \mathrm{VDOP}^2 & \cdot \\ \cdot & \cdot & \cdot & \mathrm{TDOP}^2 \end{pmatrix} \tag{2.20}$$

$$\mathrm{HDOP}^2 = \mathrm{EDOP}^2 + \mathrm{NDOP}^2 \tag{2.21}$$

$$\mathrm{PDOP}^2 = \mathrm{EDOP}^2 + \mathrm{NDOP}^2 + \mathrm{VDOP}^2 \tag{2.22}$$

$$\mathrm{GDOP}^2 = \mathrm{EDOP}^2 + \mathrm{NDOP}^2 + \mathrm{VDOP}^2 + \mathrm{TDOP}^2. \tag{2.23}$$

The first letters stand for **e**ast, **n**orth, **v**ertical, **t**ime, **h**orizontal, **p**osition and **g**eometric, respectively. The other matrix entries are omitted for simplicity. The DOP values relate the uncertainty of a single pseudorange measurement, expressed by the standard deviation σ_ρ, to the uncertainty of the navigation solution [*Groves*, 2013, pp. 424-427].

The *Receiver Independent Exchange Format* (RINEX) enables the storage of GNSS observations and navigation message data for post-processing. It defines unambiguous identifiers for GNSS signals. These identifiers consist of a three-character code for the satellite and another three-character code for the observation. The satellite codes are of type **SNN**, where **S** identifies the constellation (e. g. G for GPS, E for Galileo) and **NN** identifies the satellite's PRN or slot number. The observation codes are of type **tna**, where **t** identifies the observation type (**C** for pseudorange, **L** for carrier phase, **D** for Doppler, **S** for signal strength), **n** identifies the frequency band (e. g. 1 for L1/E1, 5 for L5/E5a) and **a** identifies the tracking mode or channel (e. g. **I** for tracking of the in-phase component, **Q** for tracking of the quadra-phase component). In this thesis, RINEX-style codes of type **Sna** are used as indices to identify the GNSS signals and the observation type is specified by the symbols ρ, ϕ and C/N_0. The full format specification of the current RINEX version 3.04 is *IGS* [2018]. Because this version does not yet cover the navigation message data broadcast on the GPS L2C and L5 signals, an extension format 4.00 including these data was developed [*Montenbruck and Langley*, 2013]. GNSS observations are stored in RINEX 3.03 or 3.04 before they are input into the algorithm. Navigation message data in RINEX 3.03/3.04 and 4.00 serve as backup to the precise orbit and clock products as explained in Section 3.5.

2.3 Odometry

"Odometry is the determination of a land vehicle's speed and distance traveled by measuring the rotation of its wheels" [*Groves*, 2013, p. 233]. Modern vehicles equipped with *electronic stability control* (ESC) typically employ one wheel speed sensor on each wheel to measure the wheel's rotation rate ω_w. To get information about the driver's input, a steering wheel angle sensor is installed. These data are output onto a data bus, usually a *Controller Area Network* (CAN).

In order to transform the information about the wheels' rotation into information about the vehicle's speed and distance traveled, odometry models are used. The complexity of these models covers a large range. Simple models utilize only one wheel speed sensor of a non-driven axle and assume that the vehicle's velocity vector is parallel to the x^b-axis [*Gao et al.*, 2006; *Li et al.*, 2010], neglecting longitudinal and lateral tire slip. Complex multi-body models account for nonlinearities within the vehicle's springs, dampers and tires [*Schramm et al.*, 2013]. In this thesis, a single-track model is applied to model the vehicle's dynamics.

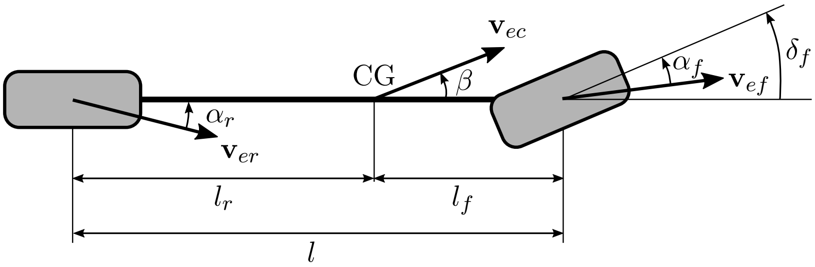

Figure 2.4: Single-track model.

Figure 2.4 depicts the single-track model. The forces necessary to generate lateral acceleration $a^b_{eb,L}$ stem from the tires. They are proportional to the side slip angles α_f and α_r in the classical single-track model. The vehicle's *center of gravity* (CG) is located on the line connecting the two wheels. Its velocity $\mathbf{v}_{ec}$ and the x^b-axis encompass the vehicle side slip angle β. The front wheel is turned by the steering angle δ_f, the rear wheel is unsteered and therefore parallel to the x^b-axis. This idealized single-track model is suited to model the vehicle's lateral dynamics up to a lateral acceleration of $|a^b_{eb,L}| = 4\,\mathrm{m\,s^{-2}}$ [*Schramm et al.*, 2013, p. 224].

One of the reasons for this limitation to moderate accelerations are nonlinear tire characteristics. The tires transfer force from the road to the vehicle, thereby creating longitudinal and lateral acceleration. According to Coulomb's Law of Friction, longitudinal tire force $F^w_{ew,x}$ and lateral tire force $F^w_{ew,y}$ are proportional to the tire force normal to the road surface $F^w_{ew,z}$.

The respective ratios are called friction coefficients μ_x and μ_y:

$$\mu_x = \frac{F^w_{ew,x}}{F^w_{ew,z}} \qquad\qquad \mu_y = \frac{F^w_{ew,y}}{F^w_{ew,z}}. \tag{2.24}$$

The index w denotes the wheel coordinate frame. This frame has its origin in the center of the tire contact patch. The z^w-axis is normal to the road, pointing upward. The y^w-axis is parallel to the wheel hub's rotation axis and points to the left. The x^w-axis completes the right-handed set and points forward, i. e. it is parallel to the x^b-axis if the wheel's steering angle is zero. A tire's friction coefficient is not constant. It varies due to temperature, road surface, tire pressure and tire slip, among others. To model the dependency on tire slip, the longitudinal slip λ_x and the side slip angle α are defined [*Rajamani*, 2006, pp. 29, 100]:

$$\lambda_x = \begin{cases} \frac{\omega_w r_d - v^w_{ew,x}}{v^w_{ew,x}} & \text{for } \omega_w r_d \leq v^w_{ew,x} \quad \text{(braking)} \\ \frac{\omega_w r_d - v^w_{ew,x}}{\omega_w r_d} & \text{for } \omega_w r_d > v^w_{ew,x} \quad \text{(acceleration)} \end{cases} \tag{2.25}$$

$$\alpha = -\arctan \frac{v^w_{ew,y}}{v^w_{ew,x}}. \tag{2.26}$$

r_d is the wheel's dynamic tire radius, $v^w_{ew,x}$ and $v^w_{ew,y}$ are the components of the wheel's velocity w. r. t. the ECEF frame along the x^w- and y^w-axis. The discontinuity for a wheel at rest ($\omega_w = v^w_{ew,x} = v^w_{ew,y} = 0$) is resolved by defining $\lambda_x = \alpha = 0$ for this condition. Linear tire models assume $\mu_x \propto \lambda_x$ and $\mu_y \propto \alpha$. The proportionality constants are longitudinal slip stiffness c_λ and side slip stiffness c_α:

$$\mu_x = c_\lambda \lambda_x \qquad\qquad \mu_y = c_\alpha \alpha. \tag{2.27}$$

In reality, the dependencies of μ_x on λ_x and of μ_y on α are degressive. Figure 2.5 depicts some exemplary curves. They are obtained with Pacejka's Magic Formula tire model [*Pacejka*, 2006, Section 4.3]:

$$\begin{aligned} \mu_x &= D_\lambda \sin\left\{C_\lambda \arctan\left[B_\lambda \lambda_x - E_\lambda \left(B_\lambda \lambda_x - \arctan B_\lambda \lambda_x\right)\right]\right\} + G_\lambda \\ \mu_y &= D_\alpha \sin\left\{C_\alpha \arctan\left[B_\alpha \alpha - E_\alpha \left(B_\alpha \alpha - \arctan B_\alpha \alpha\right)\right]\right\} + G_\alpha, \end{aligned} \tag{2.28}$$

with stiffness factor $B_{\lambda/\alpha}$, shape factor $C_{\lambda/\alpha}$, peak value $D_{\lambda/\alpha}$, curvature factor $E_{\lambda/\alpha}$ and a potential vertical shift $G_{\lambda/\alpha}$. These factors are determined empirically and depend on conditions such as weather, road surface and tire wear.

2.4 GNSS/INS Integration

According to the definitions in *Groves* [2013, Section 1.2], an IMU is a set of inertial sensors providing measurements of specific force $\mathbf{f}_{ib}$ and rotation rate $\boldsymbol{\omega}_{ib}$. An INS on the other hand is the combination of an IMU with a navigation processor, which integrates these measurements over time and outputs a navigation solution comprising position, velocity and attitude. The navigation solution provided by an INS offers a high update rate (typically 50 Hz or more) and exhibits only small short-term errors. INS require no external input apart from an initial navigation solution and are hence invulnerable to interference, jamming and weather conditions. However, the long-term accuracy of their stand-alone navigation solution is poor due to the IMU's sensor errors. Since these errors are integrated over time, they cause the navigation solution to drift away from the truth more and more as time progresses. GNSS receivers on the other hand feature a low update rate of typically 10 Hz or less. Single-epoch positions computed from pseudoranges exhibit comparatively large errors of several meters, even under optimal reception conditions. Moreover, the navigation solution's accuracy is susceptible

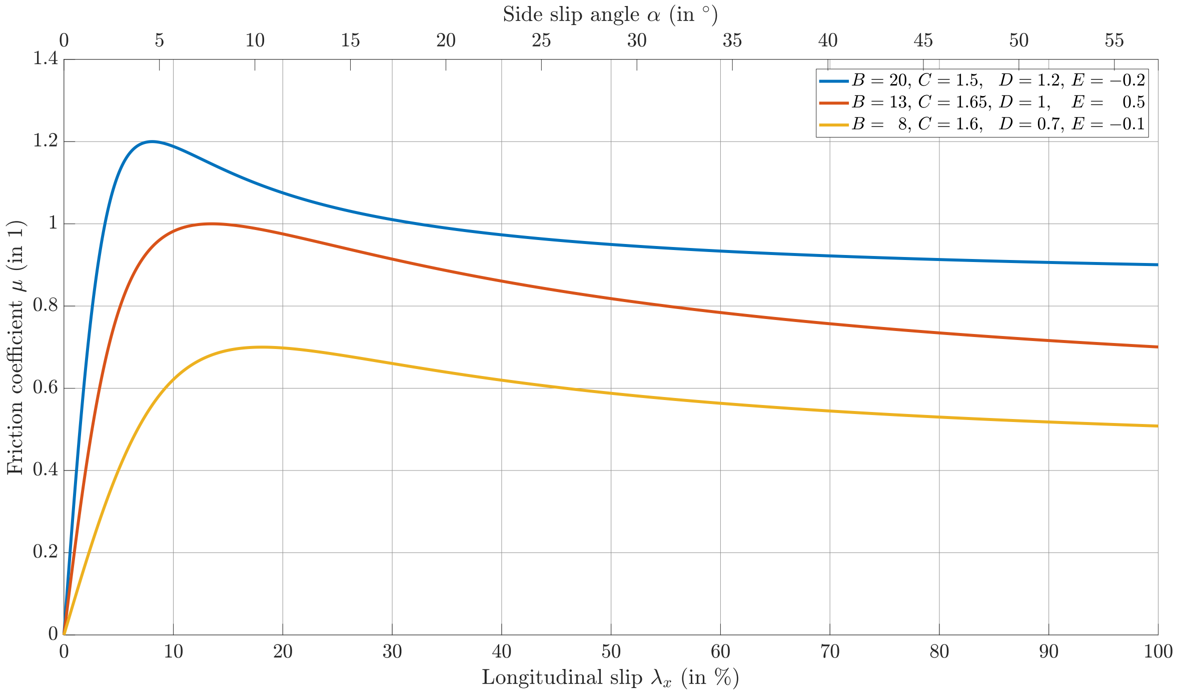

Figure 2.5: Pacejka's Magic Formula tire model.

to interference and signal obstructions, yielding poor continuity. However, the long-term accuracy of GNSS receivers is excellent and they are capable of providing a stand-alone navigation solution without the need for initialization from external sources. The combination of GNSS receivers with IMUs in a sensor data fusion algorithm profits from these complementary characteristics and is capable of combining the advantages of the two sensor types while simultaneously reducing the effects of their respective shortcomings [*Groves*, 2013, p. 559].

GNSS/INS integration algorithms differ in the types of GNSS observables that are input into the fusion filter. Loosely coupled algorithms input the 3-D position and velocity solution from the GNSS receiver and therefore perform position domain integration. Tightly coupled algorithms input GNSS pseudoranges and range rates, therefore performing range domain integration. Deeply coupled algorithms input the correlator outputs from the GNSS receiver, combining GNSS signal tracking and navigation solution in a common fusion filter and therefore perform tracking domain integration [*Groves*, 2013, p. 561]. The integration algorithm developed in this thesis is a tightly coupled one.

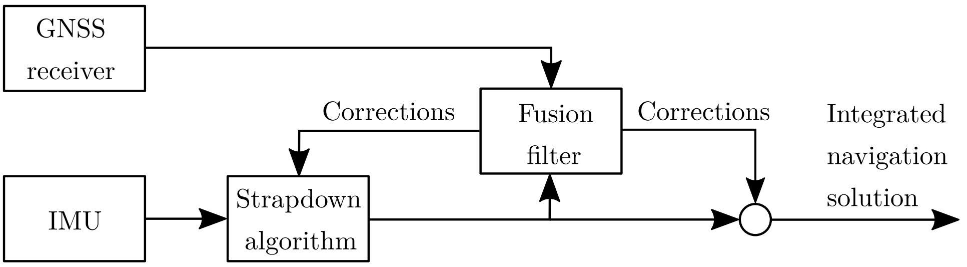

Figure 2.6: Closed-loop GNSS/INS integration architecture.

The integrated navigation solution is obtained via a closed-loop integration architecture as shown in Figure 2.6. IMU measurements are processed in a strapdown algorithm, resulting in the a-priori navigation solution. This navigation solution is fed into a fusion filter, which in this thesis is an *extended Kalman filter* (EKF), working in discrete time. Based on the a-priori navigation solution and the input from the GNSS receiver, the fusion filter calculates corrections that are applied to the a-priori solution to form the integrated navigation solution. In addition, correction values obtained by the fusion filter are fed back to the strapdown algorithm, forming the eponymous closed correction loop.

Since the EKF in a closed-loop architecture does not estimate the full navigation solution, but the errors of the externally computed a-priori solution, it is known as an error state filter. Consequently, the error state vector $\delta\mathbf{x}$ is estimated instead of the total state vector $\mathbf{x}$. $\delta\mathbf{x}$ is the difference between the estimated value of the total state vector $\hat{\mathbf{x}}$ and its true value $\breve{\mathbf{x}}$:

$$\delta\mathbf{x} = \hat{\mathbf{x}} - \breve{\mathbf{x}}. \tag{2.29}$$

$\breve{\mathbf{x}}$ and thereby $\delta\mathbf{x}$ are usually unknown. The EKF's task is to determine $\delta\hat{\mathbf{x}}$, an estimate of $\delta\mathbf{x}$, and its error covariance matrix $\mathbf{P}$ via predictions and measurement updates.

The EKF's prediction step is:

$$\delta\hat{\mathbf{x}}_k^- = \mathbf{0} \tag{2.30}$$

$$\mathbf{P}_k^- = \mathbf{\Phi}_k \mathbf{P}_{k-1}^+ \mathbf{\Phi}_k^\mathsf{T} + \mathbf{Q}_k, \tag{2.31}$$

where ˆ indicates an estimated quantity, the superscript − denotes an a-priori quantity and the superscript + denotes an a-posteriori quantity. $\mathbf{\Phi}$ is the state transition matrix and $\mathbf{Q}$ is the covariance matrix of the system noise vector $\mathbf{w}_s$. The subscript k denotes that the respective quantity is given at a discrete time epoch k. In case of a nonlinear system model (2.32), the matrix $\mathbf{\Phi}_k$ is obtained via a partial derivative of the state function $\mathbf{n}$ w. r. t. the total state vector $\mathbf{x}$ [*Groves*, 2013, Section 3.4.1]:

$$\dot{\mathbf{x}} = \mathbf{n}(\mathbf{x}) + \breve{\mathbf{w}}_s \tag{2.32}$$

$$\mathbf{N}_k = \left.\frac{\partial \mathbf{n}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_k^-} \tag{2.33}$$

$$\mathbf{\Phi}_k = \mathrm{e}^{\mathbf{N}_k \tau_{s,k}}. \tag{2.34}$$

$\breve{\mathbf{w}}_s$ is the system noise vector in continuous time, $\mathbf{N}$ is the system matrix and $\tau_{s,k} = t_k - t_{k-1}$ is the state propagation interval at epoch k. For the evaluation of (2.34), the matrix exponential function has to be used. The integration filter described in Chapter 6 features nonlinear system and measurement models.

The EKF's measurement update is:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^\mathsf{T} \left(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^\mathsf{T} + \mathbf{R}_k \right)^{-1} \tag{2.35}$$

$$\delta\hat{\mathbf{x}}_k^+ = \delta\hat{\mathbf{x}}_k^- + \mathbf{K}_k \left(\mathbf{z}_k - \mathbf{H}_k \delta\hat{\mathbf{x}}_k^- \right) \tag{2.36}$$

$$\mathbf{P}_k^+ = \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k\right) \mathbf{P}_k^- \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k\right)^\mathsf{T} + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^\mathsf{T}, \tag{2.37}$$

with the Kalman gain matrix $\mathbf{K}$, the measurement matrix $\mathbf{H}$, the measurement vector $\mathbf{z}$ and the covariance matrix $\mathbf{R}$ of the measurement noise vector $\mathbf{w}_m$. Equation (2.37) is known as Joseph's form of the covariance update and is utilized because it is numerically more robust than the mathematically equivalent formulation given by (2.38) [*Wendel*, 2007, p. 133].

$$\mathbf{P}_k^+ = \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k\right) \mathbf{P}_k^- \tag{2.38}$$

In the error state formulation, $\mathbf{z}_k$ is the difference between $\hat{\mathbf{z}}_k$ and $\tilde{\mathbf{z}}_k$, where $\tilde{\mathbf{z}}_k$ is the total measurement vector and $\hat{\mathbf{z}}_k$ is a prediction of that measurement vector, obtained from $\hat{\mathbf{x}}_k^-$, the a-priori estimate of the total state:

$$\mathbf{z}_k = \hat{\mathbf{z}}_k - \tilde{\mathbf{z}}_k \tag{2.39}$$

$$= \mathbf{h}(\hat{\mathbf{x}}_k^-) - \tilde{\mathbf{z}}_k. \tag{2.40}$$

$\mathbf{h}$ is called the measurement function. The partial derivative of $\mathbf{h}$ w. r. t. $\mathbf{x}$, evaluated at $\hat{\mathbf{x}}_k^-$, forms the measurement matrix $\mathbf{H}$:

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_k^-} . \tag{2.41}$$

The difference between total measurement vector and measurement vector prediction is called measurement innovation $\delta\mathbf{z}^-$ if the prediction is based on the a-priori state vector. It is called measurement residual $\delta\mathbf{z}^+$ if the prediction is based on the a-posteriori state vector.

$$\delta\mathbf{z}^- = \tilde{\mathbf{z}} - \mathbf{h}(\hat{\mathbf{x}}^-) \tag{2.42}$$

$$\delta\mathbf{z}^+ = \tilde{\mathbf{z}} - \mathbf{h}(\hat{\mathbf{x}}^+) \tag{2.43}$$

In the error state formulation, $\delta\mathbf{z}_k^-$ is the negative of the measurement vector $\mathbf{z}_k$ unless a sequential measurement update is performed [*Groves*, 2013, Section 3.2.7].

2.5 State of Current Research

The purpose of this section is to provide an overview of the research currently done in the fields relevant to this thesis. Due to the abundance of research results, especially in the context of GNSS, only some of the most relevant contributions are mentioned. The section is divided into two parts, dealing with recent contributions in the fields of GNSS in general and GNSS/INS integration for automotive applications in particular, respectively.

2.5.1 GNSS

Due to the new possibilities put forth by new satellite constellations and signals, a lot of research has been carried out in this field over the last years. This research still continues as a variety of approaches for different applications is developed, refined and tested. Most contributions focus on accuracy, integrity and/or robustness and how these quantities can be enhanced with the help of multi-frequency/multi-constellation GNSS.

The groundwork for applications with multiple GNSS constellations is performed by the *Multi-GNSS Experiment* (MGEX) of the *International GNSS Service* (IGS). Many authors have contributed to this experiment. The work done until October 2016 is summarized in *Montenbruck et al.* [2017]. MGEX employs a large network of reference stations (≈ 170 in October 2016) around the Earth and receives contributions from all IGS analysis centers. Some of the main achievements are: collection and storage of observation data and navigation messages, investigation of various modeling aspects for the computation of precise orbit and clock products, quality assessment of the computed orbit and clock products, analysis of signal biases that become relevant when multiple signals are utilized as well as introduction of standards and conventions for the exchange of all these data. The MGEX results are important for this thesis because they detail the effects that need to be considered for multi-frequency/multi-constellation GNSS and provide input data such as navigation messages, precise orbit and clock files, satellite biases as well as atmospheric data. They also show the achievable performance when high-quality geodetic equipment is employed, GNSS reception conditions are ideal and the antenna is static. As the requirements for automotive localization algorithms are quite different in terms of equipment cost and GNSS reception conditions, these performance achievements have very limited significance for the application of multi-frequency/multi-constellation GNSS in automotive applications.

Two main techniques are applied to reach the accuracy required by autonomous cars via GNSS: RTK and PPP. The main conceptual difference between these two approaches is that RTK requires a continuous stream of data from a nearby reference station, while PPP does not. PPP's major drawback are long convergence times ($\geq 10\,\mathrm{min}$), which have rendered it infeasible for automotive applications so far. However, utilizing more constellations and signals offers the potential to decrease convergence times, so PPP might be a valid option for automotive applications in the future. Current research regarding PPP is presented first, followed by recent advances concerning RTK.

Duong et al. [2019] employ multi-frequency/multi-constellation GNSS measurements to accelerate PPP convergence. Dual- and triple-frequency combinations for GPS, Galileo and BeiDou are formed to facilitate ambiguity resolution. The test data consist of static observations from five reference stations in Australia. Precise orbit and clock data are taken from a stream broadcast by the *Centre National d'Etudes Spatiales* (CNES) in real time. The solutions require at least 6 min to converge to within 10 cm horizontally and at least 10 min to reach this threshold vertically. The average time to first fix is 6 min for the best of the examined models.

Similar performance is reached by *Laurichesse and Blot* [2016] with triple-frequency observations on GPS, Galileo and BeiDou in static data sets. Convergence to 20 cm is achieved in 2 min instead of 5 min with dual-frequency observations. With the utilization of external tropospheric correction data, the 20 cm threshold is reached almost instantly. These results show that although convergence times for PPP are improved, they are still too high for automotive applications, even when triple-frequency observations from static data sets are used.

Besides convergence times, the distribution of precise orbit and clock data to the PPP user has to be addressed. The IGS real time service provides products for GPS and GLONASS [*Teunissen and Montenbruck*, 2017, Section 33.4.1], intended for usage with dual-frequency IF combinations. An extension to multi-frequency/multi-constellation GNSS is planned for the future. Some of the IGS analysis centers already provide corresponding real time streams.

The quality of one of these streams is evaluated in *Kazmierski et al.* [2018] by comparing the computed satellite positions and clock biases to the ones computed with final IGS products. The study obtains 3-D standard deviations of 4.8 cm, 9.8 cm, 18.3 cm and 28.2 cm for GPS, GLONASS, Galileo and BeiDou, respectively. This indicates that some work still has to be done before PPP with real time products can be considered a viable alternative to RTK in terms of accuracy.

Another method of distributing the precise orbit and clock data to the users is outlined in *Fernández Hernández* [2018]: The Galileo *High Accuracy Service* (HAS) will deliver these data via the Galileo E6-B signal with a target accuracy of less than 20 cm. The design is not finalized yet, but due to the utilization of an additional frequency band, the inclusion into automotive localization algorithms in the near future is unlikely because every additional frequency increases hardware costs.

Pullen et al. [2018] propose to distribute precise orbit and clock data via satellite, as well. In this case, the satellites are either geostationary or in low Earth orbit. The utilization of geostationary satellites is an established technique, known from *satellite-based augmentation systems* (SBAS) such as the *European Geostationary Navigation Overlay Service* (EGNOS) in Europe or the *Wide Area Augmentation System* (WAAS) in North America. The low Earth orbit satellites are provided by the Globalstar network. This approach is specifically designed with autonomous cars in mind and also addresses the important issue of integrity. One of the main conclusions is that while obtaining a horizontal position error of less than 25 cm for at least 95 % of epochs is possible with established techniques, extending this nominal accuracy to useful protection levels of several meters or less is very challenging and requires additional input data. Another conclusion is that multipath and *non-line-of-sight* (NLOS) reception pose the largest threat to accuracy and integrity in dense urban environments. To meet autonomous driving requirements, ionospheric errors have to be corrected via a dense network of reference stations (distance in between stations < 20 km) instead of dual-frequency IF combinations, because linear combinations further amplify multipath and NLOS errors, which are already critically large. As a long-term solution for reducing the PPP convergence time, additional ranging augmentation to low Earth orbit satellites is proposed. While *Pullen* *et al.* [2018] treat the challenges faced by localization algorithms for autonomous cars very thoroughly and give an overview of the state of currently available techniques as well as proposals to overcome their shortcomings, these proposals require additional ground and space segment infrastructure. Consequently, they do not offer a short-term solution.

In the context of RTK positioning for automotive applications, *Humphreys et al.* [2018] offer a similar level of completeness as *Pullen et al.* [2018] do for PPP. A GNSS positioning system consisting of a software receiver (GPS L1 C/A, GPS L2C, Galileo E1, WAAS L1) and an RTK positioning engine is designed and tested. The software receiver employs a vector tracking architecture for robust and fast carrier phase recovery after signal degradation, e. g. when the vehicle is passing under an overpass. A variety of design details is discussed and evaluated with regard to their influence on the localization algorithm's capability to perform successful ambiguity fixing. Prediction of the navigation message data bits and a very dense network of reference stations turn out to be the most important aspects in urban surroundings. One major conclusion is that a distance of 10 km in between rover and reference station is already too large and increases the probability of large errors, presumably due to incorrectly fixed ambiguities. The best of all studied configurations is able to provide a solution with fixed carrier ambiguities in 98.9 % of epochs with a probability of an incorrect fix of approx. 1 % in an urban environment which "is not an especially challenging one". The assignment of an accurate value for the false fix probability is complicated by the fact that the reference system cannot deliver a solution with the required accuracy in all epochs due to the urban setting, so the lower and upper bounds for the false fix probability are calculated as 0.5 % and 2.3 %, respectively. These results are achieved with RTK and a single GNSS antenna alone. The proposed setup for autonomous vehicles also includes an IMU, stereo cameras, a radar unit and a second GNSS antenna.

Another area of current research aims to make positioning algorithms based on pseudoranges more robust to the presence of multipath and NLOS reception, thereby increasing position accuracy in urban areas. Multipath mitigation may happen in all stages of signal processing within GNSS equipment, prompting its classification into antenna-based, receiver-based and navigation-processor-based techniques [*Groves*, 2013, Section 10.4]. Due to the abundance of existing techniques, only a limited selection can be presented here.

Hsu et al. [2015] employ a vector tracking loop for code tracking to achieve the goal of multipath and NLOS mitigation. This approach aims to reduce the multipath error already in the tracking domain to limit its influence on pseudorange observations. To evaluate the performance increase, a kinematic data set featuring urban canyons in Tokyo is recorded. The vector tracking loop reduces the mean of the 3-D position error from 11.6 m to 8.7 m and its standard deviation from 9.4 m to 4.3 m.

A technique that tries to detect and mitigate multipath errors at the observation level is proposed in *Strode and Groves* [2016]. It leverages the fact that interference from multipath can be destructive or constructive, depending on the phase lag. Because this lag varies with the carrier frequency, multi-frequency observations can be utilized to identify multipath errors. In this case, the carrier-to-noise ratios of three frequencies are subtracted from each other pairwise and these differences are compared to an elevation-dependent threshold. Static and kinematic data sets are recorded. The results verify the method's capability to detect multipath and NLOS reception. The authors recommend deploying the method "as part of a portfolio of multipath and NLOS mitigation techniques".

While all aforementioned techniques for the detection and mitigation of multipath and NLOS reception work autonomously, recent efforts have examined the utilization of external information, e. g. 3-D maps, for this purpose. Because autonomous cars need to monitor their surroundings, it can safely be assumed that localization algorithms for these cars have a detailed 3-D map available, either from previously mapped databases or generated from their own sensor data. The two-part paper *Groves and Adjrad* [2019]; *Adjrad et al.* [2019] reviews various algorithms and the effects that user equipment, the surrounding environment and the map's level of detail have on the achievable performance. The best performance is achieved by combining several of the discussed techniques, bringing down the horizontal *root mean square* (RMS) position error from 23.6 m to 3.5 m for a geodetic receiver and from 26.4 m to 4.7 m for a single-frequency mass market receiver. These results stem from single-epoch processing, so they are likely to improve when the navigation solution is filtered in time.

The last topic to be mentioned in this section is integrity, which characterizes the amount of trust that can be placed in the output of a localization algorithm (see Section 5.1 for a formal definition). The concept of integrity originates from the aviation industry. In order to quantify the risk of unusually large errors, dedicated error models for all input quantities need to be derived first. These error models have to encompass both the nominal case, when all components work within their specified error bounds and GNSS reception conditions are good, as well as the performance under faulty conditions, when at least one of the components operates outside its specified error bounds and/or GNSS measurement errors are unusually large due to poor reception conditions.

Pseudorange errors under nominal conditions are analyzed in *Cohenour and van Graas* [2011], *Perea et al.* [2016], *Salos et al.* [2010] and *Khanafseh et al.* [2018]. The latter two focus especially on error models for automotive applications, which differ significantly from the existing models for aviation due to the higher impact of multipath and NLOS errors. All mentioned authors use overbounding techniques in order to derive parameters of a Gaussian error distribution function that estimates the actual errors in a conservative way.

While analysis of the nominal errors is an important step in integrity monitoring, it is not sufficient for safety-critical systems, e. g. autonomous cars, because real measurements "exhibit large errors with a much higher probability than even a conservative Gaussian model would predict" [*Humphreys et al.*, 2018, p. 462]. To achieve the high levels of integrity that are necessary for safety-critical systems, a dedicated analysis of the tails of the error distributions needs to be performed. These analyses focus on GNSS performance under faulty conditions and its impact on the navigation solution.

Gunning et al. [2017] examine anomalies in the broadcast navigation messages of all four GNSS constellations. Because these anomalies occur very seldom, a large amount of data has to be analyzed in order to obtain a statistically substantiated error model.

Medina et al. [2018] focus on the other end of the signal transmission chain, where large multipath and NLOS errors are the main threat to integrity. In that paper, different multipath models are parametrized and compared to each other, taking observables such as elevation and carrier-to-noise ratio into account. The four parametrized models lead to availability percentages between 91.9 % and 96.4 %, with empirically assessed integrity risks between 0 % and 0.2 %. Clearly, this technique alone is insufficient to fulfill the availability and integrity requirements of autonomous cars. Dedicated integrity monitoring algorithms are required instead, some of which are summarized subsequently.

As a preliminary remark for the upcoming paragraphs, it should be noted that the provided numerical values for protection levels cannot be compared to each other directly. In order to do so, the corresponding integrity risk would have to be identical in all publications. Moreover, matching quantiles or the complete empirical distribution function of the protection level would need to be provided. While some authors use very conservative values (e. g. the horizontal protection level for an integrity risk of 10^{-7} per minute is ≤ 150 m in 99.9 % of epochs), others provide rather optimistic ones (e. g. the horizontal protection level for an integrity risk of 10^{-4} per epoch is ≤ 5 m in 95 % of epochs). The matter is further complicated by the fact that some authors account for the possible exclusion of one satellite, while others assume nominal conditions. Finally, the test scenarios can either stem from simulations regarding the number of available satellites or from real measurement data, making valid comparisons impossible.

One of the most common techniques concerning integrity in the field of GNSS is *receiver autonomous integrity monitoring* (RAIM). In very brief terms, RAIM performs consistency checks on the pseudorange measurements in order to detect and exclude faulty observations. RAIM is widely used in aviation applications, but is designed to work with single-frequency/single-constellation receivers. Its extension to multi-frequency/multi-constellation GNSS is called *advanced receiver autonomous integrity monitoring* (ARAIM). The potential and importance of ARAIM are large enough to justify the establishment of a dedicated technical subgroup for ARAIM within the GPS-Galileo Agreement Working Group C (Design and Development of the Next Generation of Systems) [*Working Group C - ARAIM Technical Subgroup*, 2016]. Some of the most recent scientific publications regarding ARAIM are: *Blanch et al.* [2018], *Gunning et al.* [2018], *Pullen et al.* [2018] and *Rippl et al.* [2018].

Blanch et al. [2018] focus on the concept of the *integrity support message* (ISM). Because the performance characteristics of GNSS constellations are expected to change over time, the most important parameters regarding their quality are provided by this ISM. These changes are slow, so providing a new set of ISM parameters once a year should be sufficient. With the current ARAIM performance, a horizontal protection level of below 185 m can be achieved "almost always". While this is a major improvement compared to RAIM, where the horizontal protection level "often exceeds 556 m", it is still not in an acceptable range for automotive applications. Moreover, these protection levels are computed for aircraft, so they assume unimpaired sky view.

This issue is addressed in *Gunning et al.* [2018]: By designing an integrity monitoring algorithm specifically tailored to be used in conjunction with PPP in kinematic applications, horizontal and vertical protection levels of less than 5 m are achieved in kinematic suburban scenarios. The algorithm requires a convergence time in the order of a few minutes, however.

Both *Pullen et al.* [2018] and *Rippl et al.* [2018] propose the inclusion of SBAS data for integrity monitoring. While *Rippl et al.* [2018] mainly examine modernized SBAS constellations with multi-frequency/multi-constellation support for aviation applications, *Pullen et al.* [2018] is specifically geared toward autonomous cars. The accuracy aspect of this paper is discussed earlier in this section when recent publications concerning PPP are summarized. The outlined integrity monitoring algorithm is set to achieve typical along-track and cross-track protection levels of 2 m to 3 m when used with PPP and SBAS corrections in good reception conditions. Additional corrections from a dense network of reference stations are required to achieve these protection levels in urban areas due to the detrimental effects of multipath and signal blockage.

Additional noteworthy contributions dealing with integrity are *Tijero et al.* [2017, 2018]. Because these papers address integrity monitoring specifically for GNSS/INS integration algorithms that are based on a Kalman filter, they are included in Section 2.5.2.

2.5.2 GNSS/INS Integration for Automotive Applications

This section summarizes some of the recent contributions in the field of GNSS/INS integration, as far as they concern automotive applications. Because this is a very active field, an abundance of material exist. Therefore, all publications that focus on the inclusion of additional sensors beyond GNSS, INS and odometry without providing substantial innovations for GNSS/INS/odometry integration itself are disregarded. The same applies to contributions that investigate various types of integration filters, e. g. unscented Kalman filters, cubature Kalman filters or particle filters. Just as for stand-alone GNSS, major trends are to increase accuracy, robustness and integrity. An additional aspect is the utilization of low-cost equipment, so research regarding single-frequency receivers is common and receivers tracking three or more carrier frequencies are virtually non-existent.

The work of *Arribas et al.* [2017] is one example of such a low-cost implementation. Data from a single-frequency/multi-constellation GNSS receiver, a MEMS IMU and the vehicle's speedometer are fused in real time on a Raspberry PI. Non-holonomic constraints are utilized based on the assumption that $v^b_{eb,L}$ and $v^b_{eb,U}$ are normally distributed zero-mean random variables. The designed algorithm also features dynamic noise covariance matrices, which adapt the measurement noise based on the EKF's innovation sequence. A loosely and a tightly coupled version of the algorithm are compared to each other, with the tightly coupled version outperforming the loosely coupled one. No numerical values for the accuracy are provided because the employed reference system was unable to provide a satisfactory reference solution in the tested urban scenario. This publication features the most basic versions of algorithm and measurement equipment that are discussed in current research activities.

Another low-cost approach is developed by *Elsheikh et al.* [2018]. A single-frequency PPP algorithm for GPS and GLONASS observations is designed and integrated loosely coupled with a MEMS IMU. Once again, non-holonomic constraints are included for $v^b_{eb,L}$ and $v^b_{eb,U}$, but this time without any odometry input data. The algorithm achieves a horizontal RMS error of 0.8 m in suburban areas and 0.7 m on a highway while passing under several overpasses. The reconvergence time to sub-meter accuracy after these overpasses is specified as a few seconds.

With the help of a dual-frequency receiver for GPS and GLONASS, *Liu et al.* [2016] integrate PPP with fixed ambiguities in a tightly coupled GNSS/INS architecture. The fixing strategy aims to resolve the ambiguities for IF linear combinations. The tight coupling with INS reduces the time for refixing the ambiguities after a fix is lost from 4 min to 1 min. In this scenario, a tactical grade IMU is used and GNSS reception conditions are good, demonstrating that current PPP techniques still have insufficiently long reconvergence times as far as autonomous cars are concerned.

By implementing an ultra-tightly coupled GNSS/INS integration scheme, *Li et al.* [2010] aim to increase the robustness of the localization algorithm against signal obstructions. Data from all four wheel speed sensors in conjunction with non-holonomic constraints are applied to improve the performance of the vector tracking loop, which is limited to GPS L1 C/A signals. The results show that performance in open sky scenarios is only improved slightly, but that the ultra-tightly coupling increases signal tracking performance in surroundings with signal obstructions significantly.

The book of *Bevly and Cobb* [2010] offers an overview of both vehicle state estimation and vehicle control on the basis of GNSS/INS integration, including dedicated sections about side slip angle estimation and tire parameter identification. The presented concept of online tire parameter estimation is especially interesting, because these parameters change due to road surface, weather and tire wear, limiting the usefulness of parameters that are estimated a-priori. This concept is similar to the one in *Bevly et al.* [2006] and only able to estimate lateral tire parameters, because dynamic tire radius and longitudinal tire parameters are not simultaneously observable with the available sensors. An approach similar to the one in this thesis is then taken in *Salmon and Bevly* [2014]: A tightly coupled GNSS/INS algorithm is developed and the steering wheel angle as well as wheel rotation rates from all four wheels in conjunction with Pacejka's Magic Formula are used as additional aiding. However, the GNSS receiver only tracks signals on a single frequency and all tire parameters, including r_d, are considered to be known and invariant with time.

The same types of sensors are also employed in *Steinhardt* [2014], upon which the developments in this thesis are built. Pseudoranges and range rates on GPS L1 C/A as well as the 3-D velocity vector $\mathbf{v}^b_{ew}$ from each wheel are utilized to aid the inertial navigation solution provided by a MEMS IMU. Besides the lack of multi-frequency/multi-constellation GNSS observations, the odometry preprocessing is limited to a linear tire model and no investigation of potential correlation between observations stemming from different wheels is performed.

Katriniok [2014] includes Pacejka's Magic Formula model in a similar algorithm. The main differences to *Steinhardt* [2014] are that a loosely coupled architecture is used and that a second filter exists, which inputs the results from the GNSS/INS integration as well as the odometry observations and estimates $v^b_{eb,F}$, $v^b_{eb,L}$ and $\dot{\psi}_{nb}$ as well as two "adaption states" for the scaling of longitudinal and lateral tire forces. These adaption states allow the second filter to account for uncertainties and changes in the tire parameters, e. g. due to road surface variations. Any potential correlation between observations stemming from different wheels is disregarded in this publication, as well.

Most contributions dealing with integrity mainly focus on GNSS and are therefore mentioned in Section 2.5.1. In contrast, *Hewitson and Wang* [2010] and *Tijero et al.* [2017, 2018] specifically address integrity in the context of GNSS/INS integration. The former express the EKF measurement update in terms of a Gauss-Markov model before examining its capabilities in terms of outlier detection, reliability and separability. For this purpose, simulated data are input into a tightly coupled single-frequency/multi-constellation integration filter. The results show that the inclusion of additional constellations reduces the lower bound for detectable outliers and stabilizes the test statistics for outlier detection, especially during low satellite visibility. The authors call the adaption of RAIM techniques to GNSS/INS integration algorithms *extended receiver autonomous integrity monitoring* (eRAIM).

Tijero et al. [2017, 2018] discuss the results of the ESCAPE project, funded by the European GNSS Agency. ESCAPE stands for *European Safety Critical Applications Positioning Engine* and aims at providing an integrity monitoring algorithm for autonomous vehicles from the fusion of multi-frequency/multi-constellation GNSS, PPP corrections, cameras, maps and vehicle sensors in a tightly coupled algorithm. The developed integrity monitoring technique is specifically tailored toward localization algorithms that are based on a Kalman filter. The preliminary algorithm version with GNSS and IMU as only sensors and without PPP corrections achieves a horizontal position error of 11.7 m or less in 95 % of epochs in a challenging scenario in the urban canyons of Madrid. The horizontal protection level is 45.6 m or less in 95 % of epochs. These values drop to 4.5 m for the horizontal error and 32.2 m for the horizontal protection level in open-sky conditions, which is still insufficient for the use in autonomous cars. Moreover, the targeted integrity risk for these values is 10^{-4} per epoch, which is several orders of magnitude more relaxed than the 10^{-7} per minute employed by *Pullen et al.* [2018].

Chapter 3

GNSS Preprocessing

This chapter presents the way GNSS measurements originating from the GNSS receiver are preprocessed before they are fed into the Kalman filter. The Kalman filter input consists of the observables described in Section 3.1 and their associated measurement noise, modeled according to Section 3.6. Section 3.2 depicts how observations from multiple GNSS constellations are used to increase the number of received satellites. Position accuracy is improved by exploiting the potential of civil signals on multiple carrier frequencies according to Section 3.3. Multiple constellations and frequencies introduce additional biases into the navigation algorithm. These biases need to be handled properly in order to limit their negative effect on the quality of the navigation solution. This process is depicted in Section 3.4. Another option to enhance the accuracy of the resulting navigation solution is the application of precise satellite orbits and clocks. These products are generated and published by different GNSS analysis centers and replace the information contained in the satellites' navigation data messages. Section 3.5 describes the usage of such products in this thesis.

3.1 Observables

The GNSS receiver provides four types of observations to the navigation algorithm: pseudoranges ρ, carrier phases ϕ, carrier-to-noise ratios C/N_0 and the signal reception time t^R_{rx}. The latter is identical for all tracked signals at a given measurement epoch. The time interval in between two epochs is $\tau_{g,k} = t^R_{rx,k} - t^R_{rx,k-1}$. Because the measurement epochs are determined by the receiver clock, its errors need to be considered in order to apply the observations with the correct timestamp.

The GNSS preprocessing inputs the data from the receiver and provides four types of observations to the integration filter: pseudoranges ρ^c, range rates $\dot{\rho}^c$ and their respective timestamps t^Γ_ρ and $t^\Gamma_{\dot{\rho}}$. These quantities are corrected for all known a-priori values.

The corrected pseudoranges ρ^c are computed from the measured pseudoranges $\tilde{\rho}$ by applying correction terms for receiver and satellite clock bias as well as atmospheric delays:

$$\rho^c = \tilde{\rho} - c\delta\hat{t}^-_R + c\delta\hat{t}^-_S - \delta\hat{\rho}_I - \delta\hat{\rho}_T. \tag{3.1}$$

$\delta\hat{t}^-_R$ is obtained by propagating the EKF's a-posteriori estimate forward in time, using the corrected GNSS measurement interval τ^c_g defined in (3.6):

$$\delta\hat{t}^-_{R,k} = \delta\hat{t}^+_{R,k-1} + \tau^c_{g,k}\delta\hat{\dot{t}}^+_{R,k-1}. \tag{3.2}$$

$\delta\hat{t}^-_S$ is obtained from the navigation data message or precise orbit and clock files (see Section 3.5). $\delta\hat{\rho}_I$ is estimated by the GPS Klobuchar model [*IS-GPS-200*, 2018, pp. 127-129], $\delta\hat{\rho}_T$ by the modified Hopfield model [*Hopfield*, 1972] in the formulation given in *Xu* [2007, p. 58]. Input parameters for the Klobuchar model are taken from the GPS navigation data message, input parameters for the modified Hopfield model are taken from daily weather data for the measurement area (if available) or the *Global Pressure and Temperature Model* (GPT) [*Boehm et al.*, 2007].

To get the corrected range rates $\dot{\rho}^c$, the corrected carrier phases ϕ^c have to be calculated first. ϕ^c is computed from the measured carrier phases $\tilde{\phi}$ by applying correction terms for the satellite clock bias and the atmospheric delays:

$$\phi^c = \tilde{\phi} + \frac{2\pi}{\lambda_{ca}} \left(c\delta\hat{t}_S^- + \delta\hat{\rho}_I - \delta\hat{\rho}_T \right). \tag{3.3}$$

$\dot{\rho}^c$ is the numerical time derivative of ϕ^c, converted to the range domain and corrected for the receiver clock drift:

$$\dot{\rho}_k^c = \frac{\lambda_{ca}}{2\pi} \frac{\phi_k^c - \phi_{k-1}^c}{\tau_{g,k}^c} - c\delta\hat{\dot{t}}_{R,k}^-. \tag{3.4}$$

The satellite clock drift is compensated implicitly due to the change in the satellite clock bias correction term from epoch $k-1$ to epoch k. $\delta\hat{\dot{t}}_R^-$ is obtained by propagating the EKF's a-posteriori estimate forward in time. Because $\delta\dot{t}_R$ is modeled as random walk, the propagation simplifies to:

$$\delta\hat{\dot{t}}_{R,k}^- = \delta\hat{\dot{t}}_{R,k-1}^+. \tag{3.5}$$

To account for the effect of the receiver clock drift on the measurement interval, τ_g^c has to be used instead of τ_g in (3.4):

$$\tau_{g,k}^c = t_{rx,k}^\Gamma - t_{rx,k-1}^\Gamma = \frac{\tau_{g,k}}{1 + \delta\hat{\dot{t}}_{R,k-1}^+}. \tag{3.6}$$

The corrected timestamps t_ρ^Γ and $t_{\dot{\rho}}^\Gamma$ are computed from the signal reception time t_{rx}^R by subtracting the a-priori estimate of the receiver clock bias. $t_{\dot{\rho},k}^\Gamma$ is then placed in the middle of the two epochs that have been used for the calculation of $\dot{\rho}^c$:

$$t_{\rho,k}^\Gamma = t_{rx,k}^R - \delta\hat{t}_{R,k}^- \tag{3.7}$$

$$t_{\dot{\rho},k}^\Gamma = t_{rx,k}^R - \delta\hat{t}_{R,k}^- - \frac{\tau_{g,k}^c}{2}. \tag{3.8}$$

The tightly coupled measurement models derived in Section 6.3 input the satellites' positions $\mathbf{p}_{es}^e$ and velocities $\mathbf{v}_{es}^e$. To account for the rotation of the ECEF frame during the signal travel time, these quantities are evaluated at the time of signal transmission, but resolved in the ECEF frame at the time of signal reception. For pseudorange observations at epoch k, the transmission time is $t_{tx,k}^\Gamma$ and the reception time is $t_{\rho,k}^\Gamma$. For range rate observations at epoch k, the transmission time is $(t_{tx,k}^\Gamma + t_{tx,k-1}^\Gamma)/2$ and the reception time is $t_{\dot{\rho},k}^\Gamma$.

If only single-frequency signals from a single constellation are input to the integration filter, the steps specified so far are sufficient. For multi-frequency/multi-constellation GNSS input, additional steps are necessary. These are explained in detail in the following sections.

3.2 Multi-Constellation GNSS

Galileo and GPS are chosen as the two constellations to develop a multi-constellation algorithm for several reasons: Galileo implements the same ephemeris representation as GPS [*Galileo OS-SIS-ICD*, 2016; *IS-GPS-200*, 2018]. Both are *code-division multiple access* (CDMA) systems and share two common carrier frequencies: L1/E1 at 1575.42 MHz and L5/E5a at 1176.45 MHz. All Galileo satellites and the newer generation GPS satellites broadcast civil signals on multiple frequencies. These characteristics make Galileo and GPS an ideal choice for multi-constellation GNSS in automotive positioning.

GPS currently employs 31 active satellites [*Navigation Center*, 2019]. Twelve are from block IIR or older and transmit L1 C/A as their only civil signal. Seven are from block IIR-M and transmit the civil signals L1 C/A and L2C. The remaining twelve are from block IIF and transmit the civil signals L1 C/A, L2C and L5. Galileo currently employs 22 active satellites [*European GNSS Service Center*, 2019], all of which transmit the civil signals E1, E5a and E5b.

GPS satellites broadcast the *Legacy Navigation Message* (LNAV) on L1 C/A and the *Civil Navigation Message* (CNAV) on L2C and L5. Compared to LNAV, CNAV has a more modern error correction algorithm and some additional terms for clock and ephemeris representation, allowing for more precise information about the satellites' position [*Teunissen and Montenbruck*, 2017, Section 7.4]. Galileo satellites broadcast the *Free Navigation Message* (F/NAV) on E5a and the *Integrity Navigation Message* (I/NAV) on E1 and E5b. F/NAV is intended for single-frequency E5a and dual-frequency E1/E5a users. I/NAV is intended for single-frequency E1 or E5b and dual-frequency E1/E5b users. Single-frequency E1 users may also use F/NAV data if available [*Teunissen and Montenbruck*, 2017, Section 9.2.2]. In this thesis, LNAV data for GPS and F/NAV data for Galileo are used for all signals as a backup to precise orbit information (see Section 3.5).

With the introduction of GNSS signals from multiple constellations, multiple choices for the common reference time t^{Γ} exist. GPS system time is different from Galileo system time. The difference is the *GPS to Galileo Time Offset* (GGTO) [*Hahn and Powers*, 2005]:

$$\mathrm{GGTO} = t^G - t^E, \tag{3.9}$$

where the superscript G denotes GPS and the superscript E (for **E**uropean) denotes Galileo. The GGTO is one part of the *inter-system bias* (ISB) between GPS and Galileo observations. In this thesis, an ISB between the constellations p and q is denoted by $D_{p,q}$. As GPS and Galileo are the only two constellations considered, $D_{E,G}$ is the only ISB that appears. It sums up all constellation-dependent differences in pseudorange measurements and also includes receiver-specific parts as detailed in Section 3.4. As a consequence of the ISB, observations belonging to different constellations have to be treated differently. One alternative is to consider observations from different constellations to have different receiver clock biases. Thus, two receiver clock biases are estimated by the integration filter: δt_R^G for GPS observations and δt_R^E for Galileo observations. The other alternative is to define one common receiver clock bias δt_R^{Γ} for all observations taken by a receiver at a specific epoch, independent of constellation. Nonetheless, the integration filter estimates two clock bias quantities: δt_R^{Γ} on the one hand and $(\delta t_R^{\Gamma} + D_{E,G})$ on the other hand. Both alternatives are mathematically equivalent [*Odijk and Teunissen*, 2013; *Paziewski and Wielgosz*, 2015]. In both cases, the geometry matrix **G** (2.19) is appended with a fifth column. In this thesis, the former variant is implemented. Pseudorange corrections involving the receiver clock bias apply δt_R^G to GPS pseudoranges and δt_R^E to Galileo pseudoranges. The variation of the ISB with time is assumed to be negligible, resulting in a common receiver clock drift $\delta \dot{t}_R$ for all range rate observations independent of constellation. The reference time t^{Γ} is defined to be the GPS system time t^G. The time quantities t_ρ^{Γ}, $t_{\dot{\rho}}^{\Gamma}$ and τ_g^c are identical for GPS and Galileo. This reflects the fact that the receiver takes all measurements simultaneously, independent of constellation.

3.3 Multi-Frequency GNSS

The total pseudorange error, or *user equivalent range error* (UERE), is decomposed into two parts considered statistically independent: Errors originating from space and control segment are summed up in the *signal-in-space ranging error* (SISRE), errors depending on the receiver and its environment are included in the *user equipment error* (UEE) [*Betz*, 2016, p. 140].

$$\mathrm{UERE} = \sqrt{\mathrm{SISRE}^2 + \mathrm{UEE}^2} \tag{3.10}$$

Table 3.1 lists the key contributions to the pseudorange error budget and their typical values. The range of 0.2 m to 1.0 m for the multipath error assumes that the direct signal path is present. Otherwise, the multipath error could be tens or hundreds of meters. *Betz* [2016, Section 6.1] lists the unmodeled ionospheric delay of modern single-frequency receivers (circa 2016 and later) with 5.0 m, making it the largest contributor to the UERE. In comparison, modern dual-frequency receivers provide a typical ionospheric error of 0.4 m, resulting in a UEE of 0.5 m and a UERE of 0.7 m. Hence, multi-frequency signals accessible to civil users provide the potential to drastically increase measurement accuracy. The second largest contributor to the error budget are the broadcast satellite orbits and clocks. These are addressed with the utilization of precise orbit and clock products in Section 3.5.

Modernized signals offer additional advantages over GPS L1 C/A beyond removing the ionospheric error: higher code chipping rate for improved tracking precision and more robustness against multipath, data-free pilot components allowing longer correlation times and more robust carrier tracking discriminator functions, secondary codes for reduced cross-correlation in between signals from different satellites as well as an increased signal transmission power for higher carrier-to-noise ratios [*Teunissen and Montenbruck*, 2017, Chapters 4,7,9].

Table 3.1: Contributions to the GNSS UERE [*Teunissen and Montenbruck*, 2017, p. 9]

Error Source	**Contribution 1σ (in m)**
SISRE	
Broadcast satellite orbit	0.2 - 1.0
Broadcast satellite clock	0.3 - 1.9
Broadcast group delays	0.0 - 0.2
UEE	
Unmodeled ionospheric delay	0.0 - 5.0
Unmodeled tropospheric delay	0.2
Multipath	0.2 - 1.0
Receiver noise	0.1 - 1.0
UERE	0.5 - 6.0

Multi-frequency observations reduce the ionospheric error because this error varies with the signal's carrier frequency. Three approaches are considered to model this frequency dependency when processing pseudorange measurements from multiple carrier frequencies in the integration filter:

1. Treat measurements stemming from the same satellite as a batch, i. e. assign similar variances to each pseudorange measurement from the same satellite but assume that pseudorange measurements on different frequencies, but from the same satellite, are highly correlated. Pseudoranges from different satellites are considered uncorrelated.
2. Introduce additional variables into the EKF's state vector to estimate the ionospheric delay. The simplest version of this alternative adds a single additional state representing the zenith ionospheric delay on a reference carrier frequency. The ionospheric delay's dependency on carrier frequency and elevation is represented in the EKF's pseudorange measurement model. More complex versions add one state for each satellite in view, resulting in a variable-length state vector.
3. Work with IF linear combinations of pseudoranges.

The decision on which approach is implemented is based on the following criteria:

A. Ensure high flexibility w. r. t. the GNSS receiver. The goal is that in order to work with another GNSS receiver providing observations for other signals than the previous receiver, the least possible amount of changes has to be made to the integration filter. In the best-case scenario, all changes are limited to the GNSS preprocessing and the integration filter stays the same.

B. The EKF assumes zero-mean Gaussian white measurement noise. The goal is to make the observations forwarded to the EKF satisfy these assumptions as closely as possible.

C. Keep the increase in the algorithm's complexity low.

While approach 1 takes the ionospheric delay's dependency on the carrier frequency into account, this is done implicitly via the off-diagonal elements in the measurement noise covariance matrix. Each pseudorange observation is still single-frequency and therefore corrupted by the unmodeled ionospheric delay. This violates criterion B because the remaining error after applying a ionospheric delay model is highly correlated in time and not zero-mean [*Kaplan and Hegarty*, 2017, Section 10.2.4]. Approach 2 increases the algorithm's complexity due to the additional variables in the EKF's state vector. The additional states are not observable for a single-frequency receiver, hence limiting the flexibility. Approach 3 matches all three criteria: No changes to the integration filter are necessary if the measurement noise for IF combinations is modeled properly. This is addressed in Section 3.6. After calibration of *differential code biases* (DCBs) in Section 3.4, the measurement noise of the IF pseudoranges has a mean close to zero. Because no additional entries are introduced into the state vector and only one pseudorange observation per satellite is forwarded to the integration filter, the increase in complexity is minimal. Consequently, approach 3 is implemented. This leads to two main drawbacks: Firstly, IF combinations only support two pseudorange measurements per satellite and not three or even more. Secondly, they suffer from increased code tracking noise. The first drawback is not a significant issue, as the inclusion of more than two frequencies offers diminishing improvements in terms of position accuracy when compared to the improvement achieved by including a second frequency. Furthermore, each additional frequency band increases the GNSS receiver's cost, rendering the widespread use of more than two carrier frequencies in automotive positioning applications in the near future unlikely. The effect of increased measurement noise is attenuated by the Kalman filter's inherent smoothing ability, especially if the resulting measurement noise has a mean close to zero.

To formulate IF combinations, the squared ratio of two carrier frequencies f_p and f_q is introduced:

$$\gamma_{p,q} = \left(\frac{f_p}{f_q}\right)^2 . \tag{3.11}$$

The IF pseudorange $\rho_{p/q}$ is defined as:

$$\begin{aligned} \rho_{p/q} &= \frac{\rho_q - \gamma_{p,q}\rho_p}{1-\gamma_{p,q}} \\ &= \left\|\mathbf{p}^e_{es}(t^\Gamma_{tx}) - \mathbf{p}^e_{ea}(t^\Gamma_{rx})\right\| + c\delta t_R - c\delta t_S + \delta\rho_T \\ &\quad + \frac{(\delta\rho_{M,q} + \delta\rho_{N,q} + B_q) - \gamma_{p,q}\left(\delta\rho_{M,p} + \delta\rho_{N,p} + B_p\right)}{1-\gamma_{p,q}} . \end{aligned} \tag{3.12}$$

Range, clock errors and tropospheric delay are the same as in the basic pseudorange model (2.13). The ionospheric delay has been eliminated due to its frequency dependency. However, $\rho_{p/q}$ is contaminated by multipath errors, code tracking noise and code biases from both signals. For a given receiver/antenna combination, the multipath error on a specific satellite signal depends on the motion of satellite, receiving antenna and surrounding reflective surfaces w. r. t. each other. In general, code multipath errors have non-zero mean values [*Teunissen and Montenbruck*, 2017, p. 453]. Their magnitude varies over time and is unknown to the localization algorithm. Therefore, they cannot be calibrated a-priori and have to be accounted for via the measurement noise model. The code tracking noise is modeled as zero-mean in accordance with *Groves* [2013, p. 396], causing its effect to be smoothed out over time as long as the measurement noise model reflects (3.12). Code biases on the other hand are time-invariant quantities whose a-priori calibration is possible [*Montenbruck et al.*, 2014]. They can only be neglected if just a single type of pseudorange per constellation is utilized. This single type may be a specific IF combination or a specific single-frequency pseudorange. If this is the case, the receiver-specific parts of the biases B_p and B_q are absorbed by the receiver clock bias δt_R, leaving the positioning performance unaffected. The satellite-specific parts of B_p and B_q can be considered as part of the satellite clock bias δt_S. They need to be taken into account anyway unless the same signal pair that is defined as reference by the respective constellation's control center is used to form $\rho_{p/q}$. This is impossible for civil GPS users because the reference pair consists of the encrypted signals L1 P(Y) and L2 P(Y) [*IS-GPS-200*, 2018].

Table 3.2: RINEX-Style Codes for the GNSS Signals Utilized in This Thesis

Signal Code	Constellation	Frequency Band	Tracking Mode or Channel
G1C	GPS	L1	C/A
G2X	GPS	L2	L2C-(M+L)
G5X	GPS	L5	I+Q
E1X	Galileo	E1	B+C
E5X	Galileo	E5a	I+Q
E7X	Galileo	E5b	I+Q

The results in this thesis were generated with a JAVAD Triumph-LS. This GNSS receiver provides measurements on GPS L1 C/A, L2C and L5 as well as on Galileo E1, E5a and E5b (among others). Combined tracking of data and pilot channels is employed wherever possible, so the RINEX identifiers for the pseudoranges are C1C, C2X and C5X for GPS as well as C1X, C5X and C7X for Galileo. According to the notation defined for this thesis in Section 2.2, the identifiers G1C, G2X and G5X denote the GPS signals, while the identifiers E1X, E5X and E7X denote the Galileo signals. These codes and their explanation are summarized in Table 3.2. The GNSS preprocessing inputs pseudoranges, carrier phases and carrier-to-noise ratios of these signals. It decides which observation types are forwarded to the integration filter based on the available measurements per satellite. Measurements on a specific signal are defined to be available if the respective C/N_0 is at least 30 dB-Hz. GPS satellites are defined to be available if measurements on L1 are available, Galileo satellites if measurements on E1 are available. For each GNSS measurement epoch, the GNSS preprocessing forwards one pseudorange per available satellite to the integration filter. IF combinations are preferred if they are available: $\rho_{G1C/G5X}$ or $\rho_{G1C/G2X}$ for GPS, $\rho_{E1X/E5X}$ or $\rho_{E1X/E7X}$ for Galileo. In case all three signals of a satellite are received in a specific epoch, $\rho_{G1C/G5X}$ is favored over $\rho_{G1C/G2X}$ and $\rho_{E1X/E5X}$ is favored over $\rho_{E1X/E7X}$. Higher preference is given to L5 and E5a compared to L2C and E5b for two reasons: On the one hand, the frequency separation to the L1 band is higher for L5 and E5a. This causes the amplification of multipath error and code tracking noise to be less severe for the IF combinations involving L5 and E5a compared with the IF combinations involving L2C and E5b. On the other hand, L5 and E5a share the same carrier frequency, code chipping rate and spreading code length. This causes their multipath and tracking noise characteristics to be similar to each other. From most to least favorable, the pseudorange hierarchy hence is $\rho_{G1C/G5X} > \rho_{G1C/G2X} > \rho_{G1C}$ for GPS and $\rho_{E1X/E5X} > \rho_{E1X/E7X} > \rho_{E1X}$ for Galileo. Being able to work with more than one type of pseudorange observation is more important for GPS than it is for Galileo, because all Galileo satellites broadcast the same signals while only the GPS block IIF satellites broadcast the L5 signal and only the block IIF and block IIR-M satellites broadcast the L2C signal.

Out of the three carrier phase inputs per constellation, only the ones in the L1 band (ϕ_{G1C} for GPS and ϕ_{E1X} for Galileo) are processed to form range rates. Due to the forming of time derivatives with $\tau_g^c < 1\,\mathrm{s}$, the ionospheric error is already eliminated since its correlation time is in the order of half an hour, making its change with time negligible compared to the other error sources in $\dot{\rho}^c$ [*Olynik et al.*, 2002]. As the ionospheric delay is the only error source depending on carrier frequency, the increase in accuracy gained from including carrier phase measurements on multiple frequencies is expected to be small.

3.4 Pseudorange Biases

Pseudorange observations ρ are affected by different types of biases. If these biases need to be addressed specifically or not depends on the utilized signal types and the desired accuracy. In a single-signal, single-constellation algorithm, these code biases are absorbed by the receiver and satellite clock biases. However, in any mixed-signal/mixed-constellation algorithm that aims to achieve high accuracy, the DCBs need to be addressed. Typical DCB values range from 0 ns to ±60 ns (0 m to ±18 m), so they need to be accounted for if the desired accuracy is higher than this [*Montenbruck et al.*, 2014, p. 199].

To account for code biases, the bias term B in the pseudorange model (2.13) is split into a satellite-specific part B^S and a receiver-specific part B^R:

$$B = B^S + B^R. \tag{3.13}$$

B^S and B^R depend on the navigation signal. This dependency is represented by an appropriate subscript, specifying the navigation signal in question (e. g. B^R_{E5X} is the receiver-specific code bias for the combined data and pilot tracking of the Galileo E5a signal). The absolute values of B^S and B^R do not affect positioning performance because they are absorbed by the receiver and satellite clock biases. Consequently, only the DCBs between the employed navigation signals are of interest. The DCB between the signals p and q is denoted as $d_{p,q}$:

$$d_{p,q} = B_p - B_q. \tag{3.14}$$

Analogous to the absolute bias B, the DCB is split into a satellite-specific part $d^S_{p,q}$ and a receiver-specific part $d^R_{p,q}$:

$$d_{p,q} = d^S_{p,q} + d^R_{p,q} \tag{3.15}$$

$$d^S_{p,q} = B^S_p \; - B^S_q \tag{3.16}$$

$$d^R_{p,q} = B^R_p \; - B^R_q. \tag{3.17}$$

The satellite clock errors distributed via the navigation data message or via precise orbits and clock files refer to a specific IF combination of pseudorange measurements. For GPS, P(Y) code measurements on L1 and L2 are employed [*IS-GPS-200*, 2018]. Galileo satellite clock errors refer to the IF combination of E1 and E5a if broadcast via F/NAV or to the IF combination of E1 and E5b if broadcast via I/NAV [*Galileo OS-SIS-ICD*, 2016]. The precise orbit and clock files employed in this thesis refer to P(Y) code measurements on L1/L2 for GPS and to E1/E5a code measurements for Galileo [*Deng et al.*, 2016]. Consequently, the following reference signals for the satellite clock bias are defined in this thesis:

- IF combination of P(Y) code measurements on L1/L2 for GPS: $\rho_{G1W/G2W}$.
- IF combination of combined data and pilot tracking code measurements on E1/E5a for Galileo: $\rho_{E1X/E5X}$.

For these observations, the satellite-specific code bias is therefore zero:

$$B^S_{G1W/G2W} = B^S_{E1X/E5X} = 0. \tag{3.18}$$

All other code observations have to be provided with an appropriate satellite-specific DCB. In this thesis, the satellite-specific DCBs provided by *Deutsches Zentrum für Luft- und Raumfahrt* (DLR) are used in combination with precise orbit and clock files. If the backup ephemeris data from the navigation data messages via RINEX files are employed, no additional satellite-specific DCBs are necessary because this information is provided as part of the navigation data messages: *Timing Group Delay* (TGD) and *Inter-Signal Correction Terms* (ISCs) for GPS, *Broadcast Group Delay* (BGD) for Galileo. While the satellite-specific DCBs from the navigation data messages can be used directly to get the satellite clock error for all desired navigation signals (G1C, G2X and G5X for GPS; E1X, E5X and E7X for Galileo), the values from DLR need to be transformed because only a subset of DCBs is provided. This set however is sufficient to construct any desired satellite-specific DCB [*Montenbruck and Hauschild*, 2013]. The provided subset for GPS is $d^S_{G1C,G1W}$, $d^S_{G1C,G2W}$, $d^S_{G2W,G2S}$, $d^S_{G2W,G2L}$, $d^S_{G2W,G2X}$, $d^S_{G1C,G5Q}$, $d^S_{G1C,G5X}$; the provided subset for Galileo is $d^S_{E1C,E5Q}$, $d^S_{E1X,E5X}$, $d^S_{E1C,E7Q}$, $d^S_{E1X,E7X}$, $d^S_{E1C,E8Q}$, $d^S_{E1X,E8X}$, $d^S_{E1C,E6C}$. From this subset, the relevant DCBs are constructed as follows:

$$d^S_{G1C,G1W/G2W} = \frac{d^S_{G1C,G2W} - \gamma_{G1,G2} d^S_{G1C,G1W}}{1-\gamma_{G1,G2}} \tag{3.19}$$

$$d^S_{G2X,G1W/G2W} = \frac{\gamma_{G1,G2}}{1-\gamma_{G1,G2}} \left(d^S_{G1C,G2W} - d^S_{G1C,G1W} \right) - d^S_{G2W,G2X} \tag{3.20}$$

$$d^S_{G5X,G1W/G2W} = \frac{d^S_{G1C,G2W} - \gamma_{G1,G2} d^S_{G1C,G1W}}{1-\gamma_{G1,G2}} - d^S_{G1C,G5X} \tag{3.21}$$

$$d^S_{E1X,E1X/E5X} = \frac{d^S_{E1X,E5X}}{1-\gamma_{E1,E5}} \tag{3.22}$$

$$d^S_{E5X,E1X/E5X} = \frac{\gamma_{E1,E5}}{1-\gamma_{E1,E5}} d^S_{E1X,E5X} \tag{3.23}$$

$$d^S_{E7X,E1X/E5X} = \frac{d^S_{E1X,E5X}}{1-\gamma_{E1,E5}} - d^S_{E1X,E7X}. \tag{3.24}$$

All DCBs in (3.19)-(3.24) are referenced to the respective reference IF combination for the satellite clock bias. To get the DCBs for the employed IF combinations, the IF combination of these DCBs is formed:

$$d^S_{G1C/G2X,G1W/G2W} = \frac{d^S_{G2X,G1W/G2W} - \gamma_{G1,G2} d^S_{G1C,G1W/G2W}}{1-\gamma_{G1,G2}} \tag{3.25}$$

$$d^S_{G1C/G5X,G1W/G2W} = \frac{d^S_{G5X,G1W/G2W} - \gamma_{G1,G5} d^S_{G1C,G1W/G2W}}{1-\gamma_{G1,G5}} \tag{3.26}$$

$$d^S_{E1X/E5X,E1X/E5X} = \frac{d^S_{E5X,E1X/E5X} - \gamma_{E1,E5} d^S_{E1X,E1X/E5X}}{1-\gamma_{E1,E5}} = 0 \tag{3.27}$$

$$d^S_{E1X/E7X,E1X/E5X} = \frac{d^S_{E7X,E1X/E5X} - \gamma_{E1,E7} d^S_{E1X,E1X/E5X}}{1-\gamma_{E1,E7}}. \tag{3.28}$$

In analogy to the satellite-specific part, the receiver-specific code bias of one signal per constellation is set to zero by defining a reference signal. Because the L1 band is the only frequency band in which all satellites broadcast civil signals, the following reference signals for the receiver clock bias are defined in this thesis:

- C/A code measurements on L1 for GPS: ρ_{G1C}.
- Combined data and pilot tracking code measurements on E1 for Galileo: ρ_{E1X}.

For these observations, the receiver-specific code bias is therefore zero:

$$B^R_{G1C} = B^R_{E1X} = 0. \tag{3.29}$$

All other code observations have to be provided with an appropriate receiver-specific DCB. These DCBs need to be calibrated for the employed receiver. To do this, the method described in *Montenbruck et al.* [2014] is adopted for a single receiver. The receiver is connected to a roof antenna and three consecutive data sets, each spanning 24 h with a measurement interval of 30 s, are collected. To eliminate potential impact on the DCBs caused by antenna and signal splitter, the same antenna and splitter models that are utilized to generate the results in Chapter 7 are employed and the receiver is connected to the same splitter output. Data collection was performed on *day of year* (DOY) 64, 65 and 66 of 2019. To calibrate the DCBs, the difference between two pseudorange observations to the same satellite at the same epoch, but based on different signals p and q, is formed:

$$\begin{aligned}\rho_p - \rho_q &= \delta\rho_{I,p} + \delta\rho_{M,p} + B_p + \delta\rho_{N,p} - \delta\rho_{I,q} - \delta\rho_{M,q} - B_q - \delta\rho_{N,q} \\ &= (1-\gamma_{p,q})\,\delta\rho_{I,p} + d^S_{p,q} + d^R_{p,q} + \delta\rho_{M,p} - \delta\rho_{M,q} + \delta\rho_{N,p} - \delta\rho_{N,q}.\end{aligned} \tag{3.30}$$

The geometric range from satellite to receiver antenna, the clock biases and the tropospheric delay are eliminated because they are independent of signal frequency. The ionospheric delay for a signal p is inversely proportional to its carrier frequency f_p, yielding

$$\delta\rho_{I,q} = \gamma_{p,q}\delta\rho_{I,p}. \tag{3.31}$$

Montenbruck et al. [2014] assume that multipath and code tracking noise terms for both signals have a mean of zero. Therefore, these terms are filtered out by averaging. While this assumption is valid for the tracking noise [*Groves*, 2013, p. 396], it is not generally valid for the multipath error. In order to minimize the mean multipath error's magnitude, the roof antenna is supplied with a ground plate to shield it from signal reflections stemming from below. Furthermore, the antenna is placed as high up and as far away from any surrounding reflective surfaces as possible. For the purpose of calibrating $d^R_{p,q}$, the terms $\delta\rho_{I,p}$ and $d^S_{p,q}$ need to be known. The ionospheric delay is obtained from the final *Ionosphere Map Exchange Format* (IONEX) files provided by the IGS. The satellite-specific DCBs are obtained from the files provided by DLR in combination with precise orbit and clock files and from the navigation data messages in combination with broadcast ephemeris data. An elevation cutoff angle of 20° is used in accordance with *Montenbruck et al.* [2014]. The advantage of this method for DCB calibration is its simplicity: Neither a GNSS signal simulator nor a GNSS receiver network are necessary. The calibration is performed with only one receiver and some data files provided by the IGS.

Figure 3.1 displays the progression of the receiver-specific DCBs during DOY 64. The plots on the left hand side are noisy due to the presence of multipath and code tracking noise from both respective signals. On the right hand side, these effects are removed with the help of the multipath combination $\rho_{MC,p}$, consisting of one pseudorange observation on frequency p and two carrier phase observations on frequencies p and q [*Teunissen and Montenbruck*, 2017, Section 20.2.4]:

$$\rho_{MC,p} = \rho_p - \frac{\lambda_{ca,p}}{2\pi}\phi_p + \frac{2}{1-\gamma_{p,q}}\left(\frac{\lambda_{ca,p}}{2\pi}\phi_p - \frac{\lambda_{ca,q}}{2\pi}\phi_q\right). \tag{3.32}$$

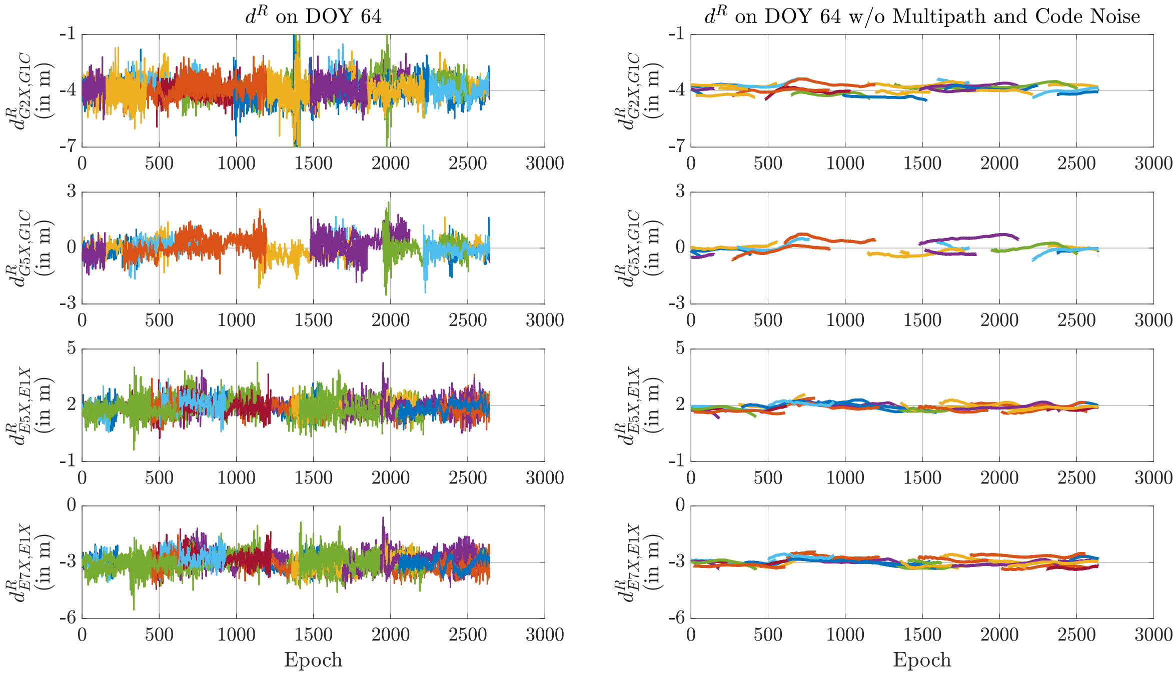

Figure 3.1: Receiver-specific DCBs on DOY 64, computed with satellite-specific DCBs from DLR. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites.

$\rho_{MC,p}$ is geometry-free and ionosphere-free. It is biased due to the carrier phase ambiguities on both frequencies. This bias is removed by subtracting the mean of $\rho_{MC,p}$ for each satellite pass. Because multipath and tracking noise for carrier phase measurements are small compared to their counterparts for code measurements, the remaining time series is dominated by code multipath and code tracking noise. Forming ρ_{MC} on both signals p and q and subtracting them from the code observations before differencing those yields with (3.30):

$$\left(\rho_p - \rho_{MC,p}\right) - \left(\rho_q - \rho_{MC,q}\right) \approx \left(1 - \gamma_{p,q}\right) \delta\rho_{I,p} + d^S_{p,q} + d^R_{p,q}. \tag{3.33}$$

As the right hand side of Figure 3.1 shows, this technique reduces the noise of the receiver-specific DCB estimation drastically. Some variation over time remains, indicating that ionospheric delay, code multipath and code tracking noise have not been eliminated completely. The graphical depiction of the progression of d^R for the remaining days looks very similar to Figure 3.1 and is included in the appendix (Section B.1). The same holds for the results of all three days, calculated with navigation message data.

The numerical evaluation of the receiver-specific DCB calibration is presented in Table 3.3 and Table 3.4. The difference between the two tables is the origin of the satellite-specific DCBs: While the values from DLR are utilized to get the results in Table 3.3, navigation message data are used for Table 3.4. The different values for d^S change the arithmetic mean of d^R by the following margins: 1.40 m for $d^R_{G2X,G1C}$, 1.55 m for $d^R_{G5X,G1C}$, 0.25 m for $d^R_{E5X,E1X}$ and 0.43 m for $d^R_{E7X,E1X}$. This disparity is caused by the dissimilar estimation strategies for d^S employed by DLR on the one hand and the respective constellation's control segment on the other hand. DLR employs a zero-mean constraint for each type of satellite-specific DCB [*Montenbruck et al.*, 2014, p. 195], while the methods employed by the control segments are not publicly documented. The magnitude of the differences between the two approaches justifies the introduction of two different sets of receiver-specific DCBs: One for the use with precise orbits and clocks and the values for d^S from DLR, one for the use

Table 3.3: Receiver-Specific DCBs, Averaged Over all Three Days, Computed With Satellite-Specific DCBs From DLR

Quantity	**Unit**	$\boldsymbol{d^R_{G2X,G1C}}$	$\boldsymbol{d^R_{G5X,G1C}}$	$\boldsymbol{d^R_{E5X,E1X}}$	$\boldsymbol{d^R_{E7X,E1X}}$
Arith. mean $\bar{d}^R$	m	−3.93	0.08	1.94	−3.07
Arith. mean $\bar{d}^R$, ρ_{MC} applied	m	−3.93	0.06	1.92	−3.07
Std. dev. σ_{d^R}	m	0.60	0.50	0.41	0.42
Std. dev. σ_{d^R}, ρ_{MC} applied	m	0.25	0.32	0.18	0.21
Std. dev. of mean $\sigma_{\bar{d}^R}$	mm	3.4	3.6	2.1	2.1
Std. dev. of mean $\sigma_{\bar{d}^R}$, ρ_{MC} applied	mm	1.4	2.4	0.9	1.1

Table 3.4: Receiver-Specific DCBs, Averaged Over all Three Days, Computed With Satellite-Specific DCBs From the Navigation Data Messages

Quantity	**Unit**	$\boldsymbol{d^R_{G2X,G1C}}$	$\boldsymbol{d^R_{G5X,G1C}}$	$\boldsymbol{d^R_{E5X,E1X}}$	$\boldsymbol{d^R_{E7X,E1X}}$
Arith. mean $\bar{d}^R$	m	−2.53	1.63	2.19	−2.63
Arith. mean $\bar{d}^R$, ρ_{MC} applied	m	−2.53	1.62	2.18	−2.63
Std. dev. σ_{d^R}	m	0.60	0.54	0.41	0.45
Std. dev. σ_{d^R}, ρ_{MC} applied	m	0.26	0.38	0.19	0.27
Std. dev. of mean $\sigma_{\bar{d}^R}$	mm	3.4	3.9	2.1	2.3
Std. dev. of mean $\sigma_{\bar{d}^R}$, ρ_{MC} applied	mm	1.5	2.8	0.9	1.4

with broadcast ephemeris and the values for d^S from the navigation data messages. The zero-mean constraint employed by DLR induces changes to all satellite-specific DCBs of a constellation each time a new satellite is deployed, causing changes to the calibration values for d^R. In order to prevent negative effects on the position accuracy, the results presented in Chapter 7 are generated with the same set of values for d^S that are used for calibration of d^R: The weekly IGS DCB solution for satellites from GPS week 2043 (March 3rd to March 9th 2019), the same week during which the calibration data for d^R were recorded. This means that no satellite-specific DCBs are available for satellites that are newly deployed in between calibration of d^R and execution of the measurements described in Chapter 7. Consequently, observations regarding such satellites must not be used in the navigation algorithm when processing the measurement data. In this particular case, the number of newly deployed satellites is zero for both GPS and Galileo. However, for applications beyond prototyping level, a way of dealing with new satellites that does not affect the calibration values for d^S and, in consequence, d^R is necessary.

Apart from the arithmetic mean, the values in Table 3.3 and Table 3.4 are in good agreement. The standard deviation of a single observation varies from 41 cm for $d^R_{E5X,E1X}$ to 60 cm for $d^R_{G2X,G1C}$ without the application of the multipath combination. The comparatively high standard deviation for $d^R_{G2X,G1C}$ is in agreement with the low code chipping rate of the GPS L2C signal (511.5 kHz compared to 10.23 MHz for GPS L5, Galileo E5a and Galileo E5b), making pseudorange measurements on GPS L2C more susceptible to multipath and causing increased code tracking noise compared to the signals with higher code chipping rate. After removing multipath error and code tracking noise via the multipath combination, the standard deviation of a single observation is reduced and now in the range of 18 cm for $d^R_{E5X,E1X}$ to 32 cm for $d^R_{G5X,G1C}$ when using DLR values for d^S and in the range of 19 cm for $d^R_{E5X,E1X}$ to 38 cm for $d^R_{G5X,G1C}$ when using navigation data message values. All satellite-specific DCBs in the navigation data messages stay constant throughout the three days, so the slightly increased standard deviation when using navigation message data cannot be attributed to a change of these values. However, the data in the IGS IONEX files are derived with a certain set of satellite-specific DCBs and the application of a zero-mean constraint is common IGS practice [*Montenbruck et al.*, 2014, p. 195], implying that the IONEX data are in agreement with the DLR values for d^S. Consequently, the remaining ionospheric delay after applying the IONEX data is slightly higher for the calibration method with navigation data message values for d^S, resulting in an increased standard deviation. The magnitude of this effect depends on the specific values for d^S. In this case, the effect is more pronounced for $d^R_{G5X,G1C}$ and $d^R_{E7X,E1X}$ (+6 cm standard deviation each) than it is for $d^R_{G2X,G1C}$ and $d^R_{E5X,E1X}$ (+1 cm standard deviation each). Assuming uncorrelated observations of a quantity q, the standard deviation $\sigma_{\bar{q}}$ of the arithmetic mean $\bar{q}$ is obtained by dividing the standard deviation σ_q of a single observation by the square root of the number of observations n:

$$\sigma_{\bar{q}} = \frac{\sigma_q}{\sqrt{n}}. \tag{3.34}$$

Application of (3.34) to the results for d^R leads to the values presented in the last two rows of Tables 3.3 and 3.4. The theoretical standard deviations of the arithmetic means are in the range of 2.1 mm to 3.9 mm without application of the multipath combination and in the range of 0.9 mm to 2.8 mm with application of the multipath combination. However, the assumption of uncorrelated observations is unsubstantiated in this case because pseudorange measurement errors are correlated in time, so these values only provide a lower bound for $\sigma_{\bar{d}^R}$.

The GNSS preprocessing outputs three different types of pseudoranges per constellation: two different IF combinations and single-frequency observations in the L1 band. The latter serve as reference signal on the receiver side, so their receiver-specific code bias is zero. Analogous to the satellite-specific DCBs in (3.25)-(3.28), the receiver-specific DCBs of the IF combinations are formed from the IF combination of their individual parts:

$$d^R_{G1C/G2X,G1C} = \frac{d^R_{G2X,G1C} - \gamma_{G1,G2} d^R_{G1C,G1C}}{1-\gamma_{G1,G2}} = \frac{d^R_{G2X,G1C}}{1-\gamma_{G1,G2}} \tag{3.35}$$

$$d^R_{G1C/G5X,G1C} = \frac{d^R_{G5X,G1C} - \gamma_{G1,G5} d^R_{G1C,G1C}}{1-\gamma_{G1,G5}} = \frac{d^R_{G5X,G1C}}{1-\gamma_{G1,G5}} \tag{3.36}$$

$$d^R_{E1X/E5X,E1X} = \frac{d^R_{E5X,E1X} - \gamma_{E1,E5} d^R_{E1X,E1X}}{1-\gamma_{E1,E5}} = \frac{d^R_{E5X,E1X}}{1-\gamma_{E1,E5}} \tag{3.37}$$

$$d^R_{E1X/E7X,E1X} = \frac{d^R_{E7X,E1X} - \gamma_{E1,E7} d^R_{E1X,E1X}}{1-\gamma_{E1,E7}} = \frac{d^R_{E7X,E1X}}{1-\gamma_{E1,E7}}. \tag{3.38}$$

With the content presented so far, all available GPS pseudoranges and their combinations can be applied consistently. The same holds for Galileo pseudoranges and their combinations. The satellite clock bias for GPS is referenced to $\rho_{G1W/G2W}$, for Galileo it is referenced to $\rho_{E1X/E5X}$. The receiver clock bias is referenced to ρ_{G1C} for GPS and to ρ_{E1X} for Galileo, respectively. This leaves a potential ISB $D_{E,G}$ between GPS pseudoranges on the one hand and Galileo pseudoranges on the other hand, as mentioned in Section 3.2. In analogy to the DCBs, it consists of a receiver-specific part D^R and a satellite-specific part D^S. Both parts are defined based on the respective reference signals for satellite and receiver clock bias. The satellite-specific part accounts for the different reference times of the satellite clock biases and is equivalent to the negative GGTO, transformed into the range domain. The receiver-specific part accounts for the different group delays of the reference signals G1C and E1X inside the receiver, caused by different signal paths from the antenna to the correlator.

$$\begin{aligned} D_{E,G} &= D^S_{E,G} + D^R_{E,G} \\ &= D^S_{E1X/E5X,G1W/G2W} + D^R_{E1X,G1C} \\ &= c(t^E - t^G) + D^R_{E1X,G1C} \\ &= -c\,\mathrm{GGTO} + D^R_{E1X,G1C} \end{aligned} \tag{3.39}$$

Treating the term $D^R_{E1X,G1C}$ as a DCB is also possible. In this case, only one of the biases (B^R_{G1C} or B^R_{E1X}) in (3.29) is set to zero and $d^R_{E1X,G1C}$ is applied instead of $D^R_{E1X,G1C}$. Both approaches are mathematically equivalent. In contrast to the DCBs, $D_{E,G}$ cannot be calibrated due to two reasons: On the one hand, $D_{E,G}$ is time-dependent because it includes the time-dependent GGTO. On the other hand, no matching pseudorange observations to the same satellite exist to calibrate $D^R_{E1X,G1C}$ because the two relevant signals belong to different constellations. Consequently, one additional parameter has to be estimated for each added constellation. In this thesis, that is accomplished by the addition of an additional receiver clock bias state δt^E_R to the state vector (see Section 3.2).

3.5 Precise Satellite Orbits and Clocks

Next to the unmodeled ionospheric delay, the errors of broadcast satellite orbits and clocks are the second largest contributor to the UERE (see Table 3.1). The magnitude of these errors is reduced by the application of precise satellite orbits and clocks. These orbit and clock products are estimated by several GNSS analysis centers and provided via the respective analysis center's server or via the IGS. Multiple types of orbit and clock products exist. Table 3.5 lists the available products for GPS in combination with their accuracy and latency. Orbit accuracy is increased by a factor of 20 when comparing broadcast ephemeris with the predicted half of the ultra-rapid orbits. Clock accuracy is only improved slightly in this case. An accuracy increase of factor 20 occurs when comparing the predicted and the observed half of the ultra-rapid clocks, demonstrating the fact that orbits can be predicted significantly better than satellite clock behavior.

As one of the requirements for the navigation algorithm designed in this thesis is to base all output values solely on information available in real time, only the predicted half of ultra-rapid products and the real time products have to be considered as a potential replacement for broadcast ephemeris. Since the IGS real time service currently only supports GPS and GLONASS, but not Galileo [*Teunissen and Montenbruck*, 2017, p. 977], the predicted half of the ultra-rapid products remains as only alternative. The GNSS analysis center *Deutsches GeoForschungsZentrum* (GFZ) offers combined ultra-rapid products covering five active constellations: GPS, GLONASS, Galileo, BeiDou and the *Quasi-Zenith Satellite System* (QZSS). Orbit and clock products are provided every three hours with a nominal latency of two hours after the last observation and with a sliding 48 h window, consisting of the previous 24 h of observations and the upcoming 24 h of predictions. The nominal epoch interval is 15 min according to *Deng et al.* [2016]. However, all products used in this thesis feature an epoch interval of 5 min. In order to apply the products, the following steps are necessary [*Kouba*, 2009]:

Table 3.5: Accuracy and Latency of GPS Satellite Orbits and Clocks [*Teunissen and Montenbruck*, 2017, p. 972]

Orbit and Clock Type	**Approx. Accuracy**	**Latency**
Broadcast	100 cm 5 ns	Real time
Ultra-rapid (predicted half)	5 cm 3 ns	Predicted
Ultra-rapid (observed half)	3 cm 150 ps	3 h to 9 h
Rapid	2.5 cm 75 ps	17 h to 41 h
Final	2 cm 75 ps	12 d to 18 d
Real time	5 cm 300 ps	25 s

1. Interpolate the satellite orbits. Based on the recommendations in *Feng and Zheng* [2005], a Lagrangian interpolation with eleven sampling points is performed. The central sampling point is the product epoch closest to the desired interpolation time.
2. Apply satellite *antenna phase center* (APC) offset. This step is necessary because the precise orbits are referenced to the satellite's CG, while the broadcast orbits are referenced to the satellite's APC [*Montenbruck et al.*, 2015, p. 324]. The APC values are taken from the IGS *Antenna Exchange Format* (ANTEX) file published in GPS week 2061. Only phase center offsets are applied, no phase center variations. The phase center offset for each satellite is computed for the reference IF combination of the satellite clock bias ($\rho_{G1W/G2W}$ for GPS, $\rho_{E1X/E5X}$ for Galileo).
3. Interpolate the satellite clocks. Because the clock data are not as smooth as the orbit data, a Lagrangian interpolation with only three sampling points is performed, as recommended in *Zumberge and Gendt* [2001]. The central sampling point is the product epoch closest to the desired interpolation time.
4. Apply relativistic clock correction term $\delta t_{S,rel}$ [*Groves*, 2013, p. 390]:

   $$\delta t_{S,rel} = -2 \frac{\mathbf{p}_{es}^{e\,\intercal} \mathbf{v}_{es}^{e}}{c^2}. \tag{3.40}$$

   The satellite's position $\mathbf{p}_{es}^{e}$ is already known from step 1. The satellite's velocity $\mathbf{v}_{es}^{e}$ is computed via numerical differentiation of $\mathbf{p}_{es}^{e}$ in between two epochs spaced ±0.1 s around the desired interpolation time.

In order to achieve the highest accuracy possible, the product file that was published as close as possible prior to the start of a measurement scenario is utilized. During a specific measurement scenario, the product file is kept fixed to avoid possible discontinuities, even if a more recent one becomes available.

Timing information distributed via navigation data messages generally refers to the respective constellation's system time. In contrast, all information in precise product files refers to the same reference time. In case of the GFZ ultra-rapid products, this reference time is the GPS system time. Consequently, all satellite clock biases are referenced to the same time scale when using precise orbits and clocks. This influences the ISB apparent to the navigation algorithm, because the satellite-specific part of (3.39) is eliminated. On the other hand, another term is added to the ISB apparent to the navigation algorithm due to the process of aligning the satellite clock biases to a common reference time performed by the GNSS analysis center. Different strategies to accomplish this exist [*Steigenberger et al.*, 2015, Section 2.3]. The most common ones are to either fix $D_{E,G}$ of a selected reference station to zero or to apply a zero-mean constraint for the ensemble of all station ISBs. In either way, an additional term depending on the station network used to generate the precise orbit files appears in (3.39). GFZ estimates one ISB between GPS and Galileo per station per day [*Uhlemann et al.*, 2014]. This additional term is constant for each product file, but changes from file to file. In terms of the navigation algorithm's accuracy, this change is absorbed by the degree of freedom provided by the two receiver clock bias parameters δt_R^G and δt_R^E and therefore does not impact position accuracy negatively.

3.6 Measurement Noise

In addition to the GNSS observables, the integration filter inputs their corresponding measurement noise covariance matrix $\mathbf{R}$. The two types of GNSS observables (pseudorange ρ^c and range rates $\dot{\rho}^c$) are considered to be uncorrelated to each other, so two distinct matrices $\mathbf{R}_{\rho^c}$ and $\mathbf{R}_{\dot{\rho}^c}$ are output by the GNSS preprocessing. In addition, observations to different satellites are considered to be uncorrelated to each other, so both $\mathbf{R}_{\rho^c}$ and $\mathbf{R}_{\dot{\rho}^c}$ are diagonal matrices. The non-zero elements of $\mathbf{R}_{\rho^c}$ are computed as the sum of the individual error sources in the pseudorange measurement model (2.13), as suggested by *Salos et al.* [2010]:

$$\sigma_{\rho^c}^2 = \sigma_{\rho^c,S}^2 + \sigma_{\rho^c,I}^2 + \sigma_{\rho^c,T}^2 + \sigma_{\rho^c,M}^2 + \sigma_{\rho^c,N}^2, \tag{3.41}$$

where the terms on the right hand side are the pseudorange errors stemming from satellite position and clock bias ($\sigma_{\rho^c,S}^2$), unmodeled ionospheric delay ($\sigma_{\rho^c,I}^2$), unmodeled tropospheric delay ($\sigma_{\rho^c,T}^2$), multipath ($\sigma_{\rho^c,M}^2$) and code tracking noise ($\sigma_{\rho^c,N}^2$). Since the range rate measurement model is a lot simpler, the corresponding error model only consists of the single term $\sigma_{\dot{\rho}^c}^2$ which accounts for all measurement errors combined. *Salos et al.* [2010] apply overbounding techniques according to *DeCleene* [2000] in order to obtain the standard deviation of a zero-mean normal distribution whose *cumulative distribution function* (CDF) is always greater than the CDF of the empirical data for negative errors and always smaller than the CDF of the empirical data for positive errors. This parametrization assures that the error model accounts for the tails of the empirical data's distribution function, which are typically much thicker than the ones of a normal distribution. Overbounding techniques are common practice for integrity monitoring, since they lead to very conservative error estimates. However, they are not ideal for the purpose of measurement noise parametrization in this thesis, because such conservative error models are not suited for outlier detection. Moreover, they do not represent nominal GNSS performance, therefore impeding the optimal weighting of GNSS, IMU and odometry data relative to one another in the integration filter. Instead, an error magnitude is assigned to each of the six individual error terms (five for ρ^c, one for $\dot{\rho}^c$) according to a model described in *Prieto-Cerdeira et al.* [2014]:

$$\sigma_q(x) = a_q + b_q \, \mathrm{e}^{-\frac{x}{x_q}}. \tag{3.42}$$

The subscript q denotes any of the six error terms. x is a quantity that the error depends on (e. g. elevation, carrier-to-noise ratio, etc.). The constants a_q, b_q and x_q are fitted for each error term based on measurement data. To do so, GNSS observations from 16 data sets are used: twelve static data sets with a total of approx. 200 000 epochs and four kinematic data sets with a total of approx. 210 000 epochs. The static data sets were collected with a roof antenna and demonstrate good GNSS reception conditions. Their measurement intervals range from 0.2 s to 30 s. The kinematic data sets were collected with measurement intervals of 0.1 s or 0.2 s and demonstrate GNSS reception conditions varying from good over poor to unavailable. The driven trajectories feature similar characteristics to the routes driven to collect the data for the results presented in Chapter 7 and contain among others open sky, rural roads, forests, inner city with multi-story buildings and tunnels. The results for each of the error terms are presented in the following sections. Unless noted otherwise, an elevation mask of 5° is applied. The same cutoff angle is used for all the results in Chapter 7.

3.6.1 Satellite Position and Clock Bias

Due to the inaccuracy of each satellite's position and clock bias data, the corresponding terms in the pseudorange measurement model (2.13) cannot be eliminated completely. Both error terms are indistinguishable from the receiver's point of view, so they are summed up into one error term $\delta\tilde{\rho}^c_S$ according to *Perea et al.* [2016, p. 1728]:

$$\delta\tilde{\rho}^c_S = {\mathbf{u}^e_{as}}^{\mathsf{T}} \delta\tilde{\mathbf{p}}^e_{es} - \delta\left(\widetilde{c\delta t_S}\right), \tag{3.43}$$

where $\delta\tilde{\mathbf{p}}^e_{es}$ denotes the error in satellite position and $\delta\left(\widetilde{c\delta t_S}\right)$ denotes the error in satellite clock bias. To evaluate (3.43), both the predicted half of the ultra-rapid precise orbits and the broadcast orbits are compared to a set of final precise orbits. The unit vector $\mathbf{u}^e_{as}$ is computed based on the antenna's position as estimated by reference equipment, consisting of a geodetic GNSS receiver for RTK positioning and a navigation grade IMU (see Chapter 7 for details). The IMU is only necessary for the kinematic data sets.

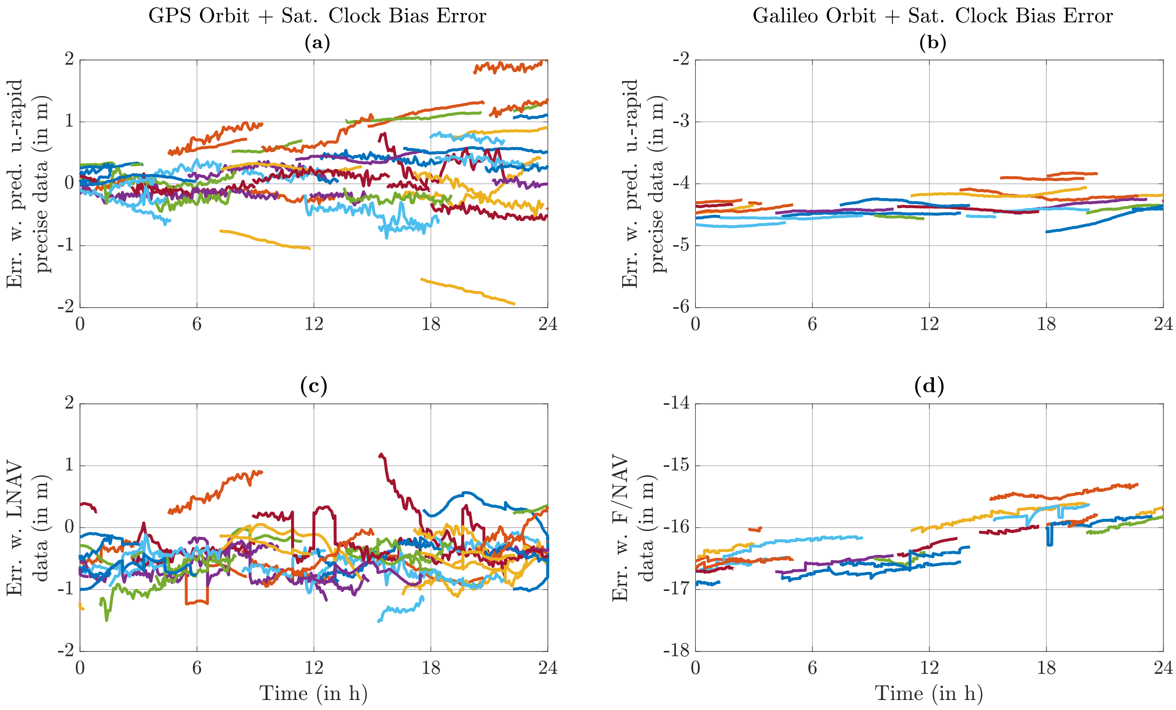

Figure 3.2: Satellite position and clock bias error for GPS and Galileo satellites. **(a)**,**(b)** Predicted half of ultra-rapid precise data. **(c)**,**(d)** Navigation message data. Different colors indicate different satellites.

The results from one of the static data sets are depicted in Figure 3.2. The set's duration is 24 h, the measurement interval is 30 s. Both the ultra-rapid precise (Figure 3.2a) and the broadcast orbits (Figure 3.2c) are close to zero-mean for GPS. This is not the case for

Galileo, where the ultra-rapid orbits (Figure 3.2b) have a mean error of approx. −4.5 m, while the broadcast orbits (Figure 3.2d) have a mean error of approx. −16 m, demonstrating the ISBs occurring in the precise and broadcast orbits. This mean error will be absorbed by the integration filter's two states for the receiver clock bias and not influence positioning performance negatively. Throughout the data set's duration, only one set of ultra-rapid precise orbits is employed, while the broadcast data are updated as recommended by the respective constellation's *interface control document* (ICD): every two hours for GPS, every ten minutes for Galileo. This update process is responsible for the discontinuities in the errors of the broadcast data. It also prevents the errors of the broadcast data from growing over time, because actual measurement data are incorporated in the predictions with every new update. In contrast, the errors of the predicted ultra-rapid orbits grow over time. The main contributor to this growth is the clock bias error, because the satellites' positions can be predicted significantly better than the satellite clock behavior. Galileo satellites feature passive hydrogen masers as their most stable on-board clock. The performance of the GPS satellite clocks depends on the satellite generation, with the rubidium atomic clocks of the block IIF satellites achieving similar stability levels as Galileo's hydrogen masers [*Teunissen and Montenbruck*, 2017, Section 5.3.6]. Previous generations employ less stable clocks, whose behavior is more difficult to predict. Consequently, clock prediction quality for the whole constellation is better for Galileo than it is for GPS. This is demonstrated by the fact that the error difference of the two worst GPS satellites grows to approx. 4 m after 24 h, while the corresponding value for Galileo is only 1 m. As described in Section 3.5, the ultra-rapid file that was published as close as possible prior to the start of a measurement scenario is utilized. Therefore, the measurement noise covariance is only estimated based on ultra-rapid data no older than six hours (files are provided every three hours with a nominal latency of two hours after the last observation). For the satellite position and clock bias error, no quantity that this error depends on is apparent other than time. With the reduction to a maximum prediction age of six hours, this dependency is minimized as well. That's why the measurement noise model (3.42) is reduced to a single constant term for this type or error:

$$\sigma_{\rho^c,S} = a_{\rho^c,S}. \tag{3.44}$$

Based on all evaluated data, $a_{\rho^c,S}$ takes values of 0.5 m for GPS satellites (for both ultra-rapid and navigation message data). For Galileo satellites, a value of 0.15 m is obtained for ultra-rapid products and a value of 0.25 m for navigation message data.

3.6.2 Unmodeled Ionospheric Delay

To determine the magnitude of the unmodeled ionospheric delay, $\delta\rho_I$ is determined in three different ways. Values obtained from final IONEX files provided by the IGS serve as reference to evaluate the two methods utilized in the GNSS preprocessing: the Klobuchar model for single-frequency pseudoranges and the formation of IF combinations for multi-frequency observations. In order to get smoother data, the multipath combination is applied to the multi-frequency observations. The corresponding DCB needs to be removed, too. Consequently, the ionospheric delay on signal p as estimated via multi-frequency observations is:

$$\delta\hat{\rho}_{I,p} = \frac{(\rho_p - \rho_{MC,p}) - (\rho_q - \rho_{MC,q}) - d_{p,q}}{1 - \gamma_{p,q}}. \tag{3.45}$$

All three values for $\delta\rho_I$ are calculated for the L1 band. The difference between the reference from IONEX and the two methods employed by the GNSS preprocessing as a function of the elevation θ is depicted in Figure 3.3. The fitted error magnitude function (explained in detail in the next paragraph) is shown in the same plot as the error for single-frequency pseudoranges to visualize the relation between the raw data and the model fit. For the IF combinations, only $\rho_{E1X/E5X}$ is displayed since it features the largest number of observations. The plots for the other IF combinations look very similar and appear in the appendix (Section B.2). Due to the large amount of observations, the individual data points are unrecognizable. The dependency of the remaining ionospheric delay on the elevation after applying the Klobuchar model is clearly visible. The remaining ionospheric delay after applying the IF combination is significantly smaller: While the RMS of the remaining error is 2.5 m for the Klobuchar model, it is only 0.5 m for the IF combination. Because this value of 0.5 m incorporates contributions from the remaining multipath and code noise, the uncertainty of the DCBs and the errors in the IONEX reference file, the term $\sigma_{\rho^c,I}$ is set to zero for all IF combinations. The subsumed contributions of the mentioned error sources are accounted for by the other terms of (3.41).

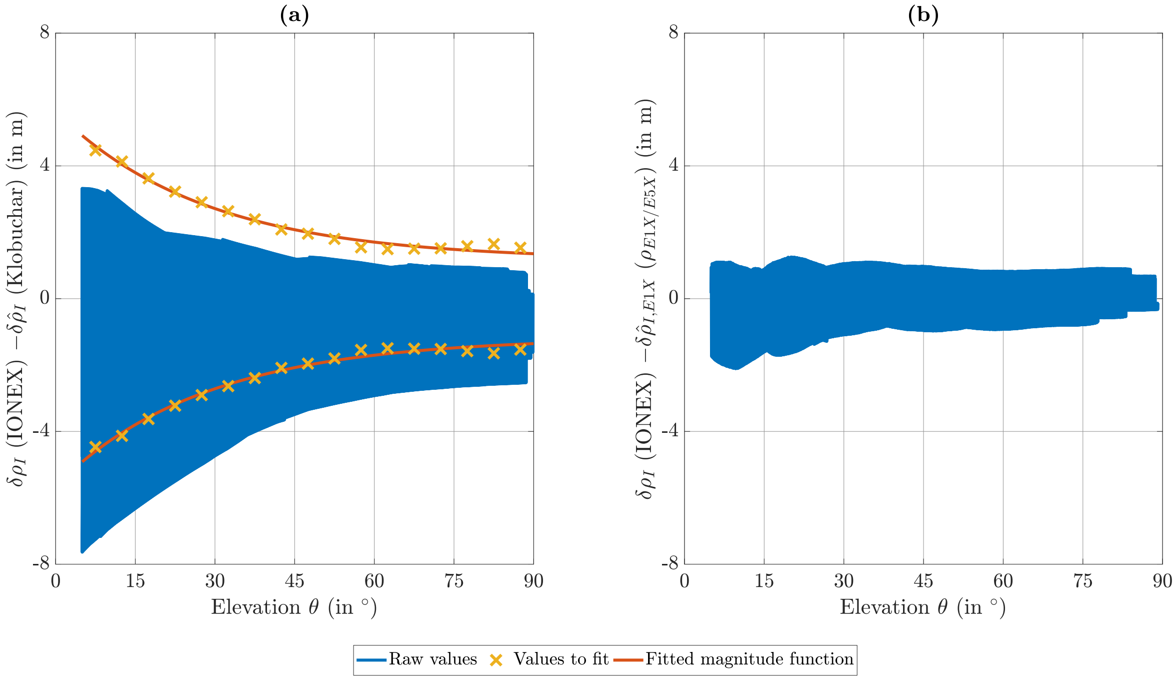

Figure 3.3: Difference between reference value from IONEX and estimated value for ionospheric delay in the L1 band. **(a)** Estimated value obtained via Klobuchar model. **(b)** Estimated value obtained from multi-frequency pseudoranges on E1X and E5X.

To account for the unmodeled ionospheric delay of single-frequency pseudoranges, a function according to (3.42) is fitted. To do so, the elevation range from 5° to 90° is split into 17 bins of 5° each. In each bin, the following three empirical quantities of the remaining ionospheric delay are calculated: standard deviation, RMS and 68.3 % quantile. The latter one is the percentage of sample points in the $\pm 1\sigma$ interval around the mean, if the sample's underlying distribution is the normal distribution. It is computed based on the error's absolute value. This process is done to check how well the remaining error corresponds to the Kalman filter's assumptions of normally distributed zero-mean errors. If the remaining error fulfills the assumptions, all quantities are identical for each bin. As expected, this is not the case. The RMS is approx. twice as large as the standard deviation, indicating a non-zero mean. The 68.3 % quantile is approx. 10 % larger than the RMS in each bin, indicating that fewer sample points are close to the mean than is to be expected from a normal distribution. In order to get a conservative estimate of the measurement noise, the largest of the three quantities in each bin (i. e. the 68.3 % quantile) is used as data basis to fit the error magnitude function (3.46) for $\sigma_{\rho^c,I}$. The data points and the fitted function (as well as the negative of them) are displayed in Figure 3.3a. The numerical representation of $\sigma_{\rho^c,I}$ is:

$$\sigma_{\rho^c,I}\left(\theta\right) = 1.17\,\mathrm{m} + 4.47\,\mathrm{m}\,\mathrm{e}^{-\frac{\theta}{28.2^\circ}}. \tag{3.46}$$

3.6.3 Unmodeled Tropospheric Delay

The magnitude of the unmodeled tropospheric delay is determined similarly to the one of the unmodeled ionospheric delay. The reference data are tropospheric estimates from an IGS station in Frankfurt am Main (station identifier FFMJ, approx. 25 km away from the roof antenna used to collect the static data for the results in this section). These estimates stem from files in the *Solution Independent Exchange Format* (SINEX) troposphere format. The reference is compared to two different versions of the modified Hopfield model: the first one inputs temperature and air pressure values from the GPT and assumes the relative humidity to be 50 %, the second one inputs measured weather data for temperature, air pressure and relative humidity. These measurements are performed at the beginning of each scenario. The input values for the modified Hopfield model are then kept constant during the entire scenario.

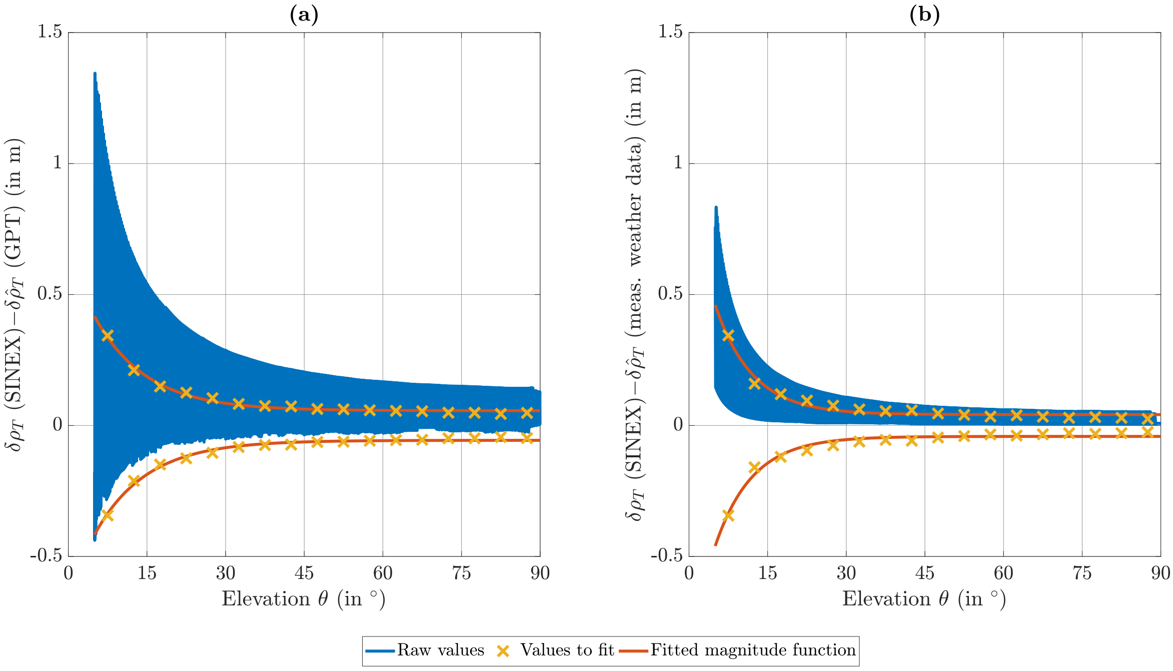

Figure 3.4: Difference between reference value from SINEX and estimated value for tropospheric delay. **(a)** Estimated value obtained via modified Hopfield model with GPT weather data. **(b)** Estimated value obtained from modified Hopfield model with measured weather data.

The difference between reference and model data as a function of the elevation θ is depicted in Figure 3.4 for both of the two model versions. Again, the fitted error magnitude functions (as well as their negatives) are shown in the same plot as the remaining error to visualize the relation between the raw data and the model fit. Compared to the ionospheric delay, the tropospheric delay's variation with time is smaller. Consequently, the model matches the reference data better and the unmodeled delay is smaller. As expected, the model using measured weather data as input performs better than the model using GPT weather data. Fitting the error magnitude function (3.42) for the unmodeled tropospheric delay is done in the same way as for the unmodeled ionospheric delay. The elevation range from 5° to 90° is split into 17 bins of 5° each. For GPT weather data, all three empirical quantities (standard deviation, RMS, 68.3 % quantile) agree well with each other (max. deviation $< 10\,\text{cm}/25\,\%$). For measured weather data, the standard deviation is approx. half as large as the other two quantities, indicating a non-zero mean. Because the RMS is the largest of the three empirical quantities in each bin for both model types, it is chosen as data basis to fit the error magnitude function in order to get a conservative estimate of the actual error.

The numerical representation of $\sigma_{\rho^c,T}$ is:

$$\sigma_{\rho^c,T}\left(\theta\right) = 5.7\,\mathrm{cm} + 59.8\,\mathrm{cm}\;\mathrm{e}^{-\frac{\theta}{9.8^\circ}} \tag{3.47}$$

for the modified Hopfield model with GPT input data and

$$\sigma_{\rho^c,T}\left(\theta\right) = 4.2\,\mathrm{cm} + 84.8\,\mathrm{cm}\;\mathrm{e}^{-\frac{\theta}{7.1^\circ}} \tag{3.48}$$

for the modified Hopfield model with measured input data.

3.6.4 Code Multipath and Code Tracking Noise

The magnitude of code multipath $\delta\rho_M$ and code tracking noise $\delta\rho_N$ is estimated based on the multipath combination ρ_{MC} (3.32). ρ_{MC} is an estimate for the sum of $\delta\rho_M$ and $\delta\rho_N$, leveled to be zero-mean. Because the extent of multipath in a GNSS pseudorange measurement mainly depends on the reception conditions, the spreading code and the receiver's code tracking loop properties, this analysis is carried out for each signal individually, once for static data and once for kinematic data. The error magnitude function for the multipath error is again fitted as a function of θ in 17 bins of 5° each, spanning from 5° to 90°.

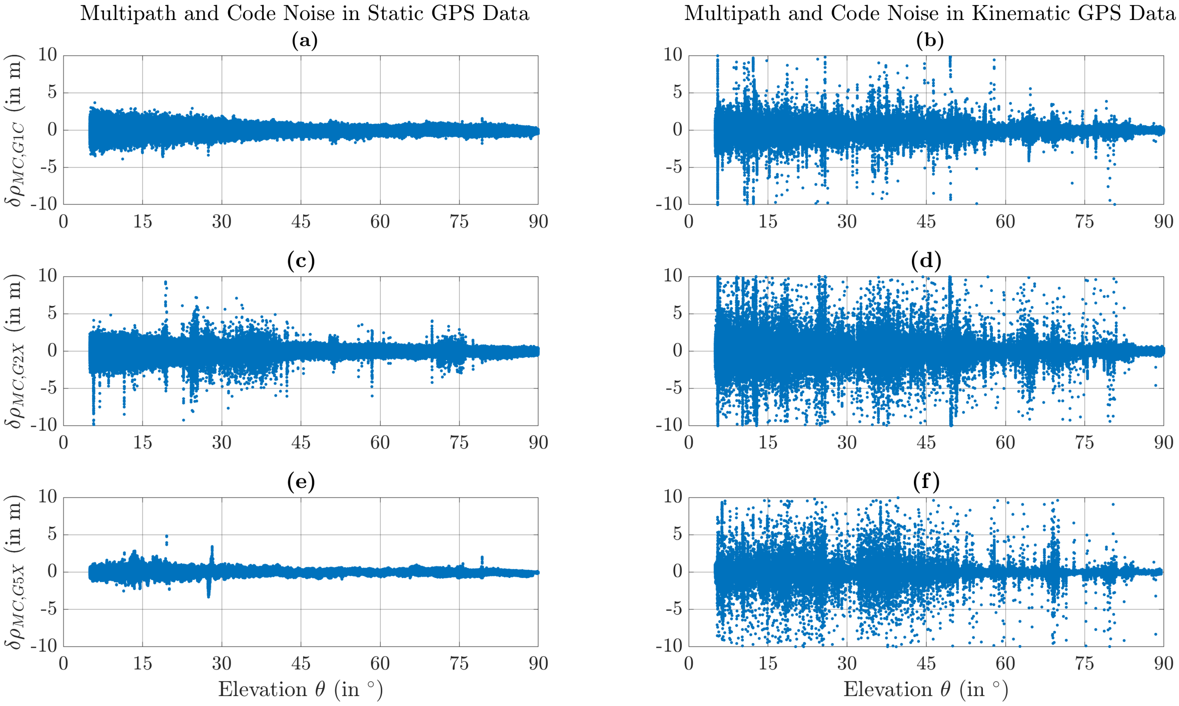

Figure 3.5: Code multipath and code tracking noise for GPS, estimated via the multipath combination. **(a)**,**(c)**,**(e)** Static data for G1C, G2X and G5X. **(b)**,**(d)**,**(f)** Kinematic data for G1C, G2X and G5X.

The results for GPS as a function of θ are displayed in Figure 3.5. The equivalent figure for Galileo appears in the appendix (Section B.2). Due to the identical code chipping rates, the plot for $\rho_{MC,E1X}$ is similar to the one for $\rho_{MC,G1C}$, while the plots for $\rho_{MC,E5X}$ and $\rho_{MC,E7X}$ are similar to the one for $\rho_{MC,G5X}$. The static data displayed in Figures 3.5a, c, e were recorded in an environment associated with little multipath (direct signal path always present, no surfaces with high reflection coefficients nearby). Under these circumstances, the expected multipath mainly depends on the signal's code chipping rate f_{co} (higher chipping rate equals less multipath), its modulation (BPSK is more prone to multipath than BOC) and the early-to-late spacing of the receiver's tracking loop (narrower spacing equals less multipath) [*Teunissen and Montenbruck*, 2017, Section 15.5]. While all three tracked GPS signals contain BPSK modulation and details about the receiver's tracking loops are unknown, the effect of f_{co} is visible for the static data in Figures 3.5a, c, e: ρ_{G5X} with the highest chipping rate of 10.23 MHz has the lowest multipath error, ρ_{G1C} with the intermediate chipping rate of 1.023 MHz has a medium multipath error and ρ_{G2X} with the lowest chipping rate of 511.5 kHz has the largest multipath error. The difference between G5X and G1C is more pronounced than the difference between G1C and G2X, mirroring the larger difference in chipping rate (factor of 10 for G5X and G1C, factor of 2 for G1C and G2X).

The effect of the reception conditions on the multipath error is apparent when comparing the results from kinematic data sets in Figures 3.5b, d, f to the ones from static data sets. While the effect of f_{co} is still present, the multipath error in the kinematic sets is much larger. Statistical analysis shows that this increase affects the wider quantiles more than narrower ones: The 68.3 % quantiles of static and kinematic data are nearly identical over the whole elevation range for all six signals (e. g. 0.27 m for the bin ranging from 5° to 10° for static data on G5X compared to 0.30 m for kinematic data). The 95.5 % quantiles (percentage of sample points in the $\pm 2\sigma$ interval around the mean for normally distributed data) are approx. twice as large for kinematic data as they are for static data (e. g. 0.59 m static compared to 1.19 m kinematic in the 5°-10° bin for G5X). The 99.7 % quantiles ($\pm 3\sigma$ interval for normally distributed data) for kinematic data suffer from large outliers, making them infeasible as data points to fit the magnitude function (e. g. 0.98 m static compared to 8.17 m kinematic in the 5°-10° bin for G5X).

Fitting the magnitude functions for $\sigma_{\rho^c,M}$ and $\sigma_{\rho^c,N}$ requires some additional steps compared to fitting the ones for $\sigma_{\rho^c,I}$ and $\sigma_{\rho^c,T}$ for two main reasons:

- Separating code tracking noise from code multipath is difficult. To do so, the code tracking noise is only fitted based on observations expected to have no significant multipath error. To isolate these observations, two criteria are applied: $\theta \geq 60°$ and $|\rho_{MC}| < 3\,\text{m}$. The function for $\sigma_{\rho^c,N}$ is then fitted only based on observations that meet both criteria.
- ρ_{MC} varies significantly from static to kinematic data. As the intended application is kinematic and features varying GNSS reception conditions, fitting $\sigma_{\rho^c,M}$ based on static data leads to overly optimistic results. The quantiles for the kinematic data show that the multipath error is far from being normally distributed and contains a high amount of large outliers. This however is unproblematic: One aspect of the measurement noise model is to provide a basis for reliable outlier detection and elimination in the integration filter (see Chapter 6). To identify pseudoranges corrupted with large multipath errors, $\sigma_{\rho^c,M}$ has to describe the nominal amount of multipath present in kinematic data.

The magnitude functions for $\sigma_{\rho^c,M}$ and $\sigma_{\rho^c,N}$ are therefore fitted in the following manner:

1. $\sigma_{\rho^c,N}$ is fitted as a function of C/N_0 for each signal individually based on all pseudorange observations with $\theta \geq 60°$ and $|\rho_{MC}| < 3\,\text{m}$. The dependency of $\sigma_{\rho^c,N}$ on C/N_0 is discussed in detail in e. g. *Kaplan and Hegarty* [2017, Section 8.9]. Since pseudoranges with $C/N_0 < 30\,\text{dB-Hz}$ are discarded anyway, a corresponding offset is inserted into (3.42):

$$\sigma_{\rho^c,N}(C/N_0) = a_{\rho^c,N} + b_{\rho^c,N}\,\mathrm{e}^{-\frac{C/N_0 - 30\,\text{dB-Hz}}{x_{\rho^c,N}}}. \tag{3.49}$$

While mathematically equivalent to the formulation without the offset, this stabilizes the fitting process because $a_{\rho^c,N}$ and $b_{\rho^c,N}$ now have a similar order of magnitude. As the maximum C/N_0 in all the data sets is 60 dB-Hz, the range from 30 dB-Hz to 60 dB-Hz is divided into 15 bins of 2 dB-Hz each. Statistical analysis shows that all aforementioned empirical quantities agree well with each other in each bin, indicating that the remaining errors are close to being zero-mean normally distributed, so the 68.3 % quantile of each bin is chosen as value to fit.

2. $\sigma_{\rho^c,M}$ is fitted as a function of θ for each signal individually based on all pseudorange observations from kinematic data sets. In order to get a representation of the nominal error to facilitate outlier detection, the 68.3 % quantile of each bin is chosen as value to fit the error magnitude function (3.42). After the fitting process, the constant term $a_{\rho^c,M}$ is removed from the fitted function because it corresponds to the constant noise floor present in all observations. This effect has already been parametrized via the code tracking noise function $\sigma_{\rho^c,N}$.

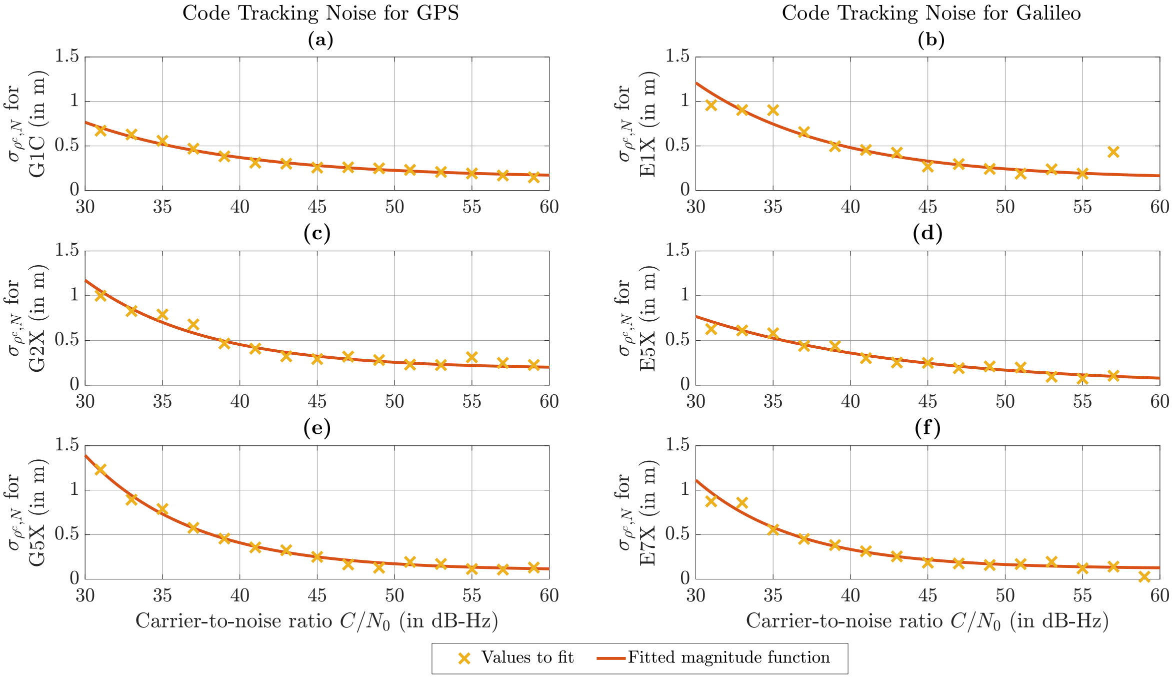

Figure 3.6: Fitted standard deviation of code tracking noise. **(a)**,**(c)**,**(e)** GPS signals G1C, G2X and G5X. **(b)**,**(d)**,**(f)** Galileo signals E1X, E5X and E7X.

Table 3.6: Fitted Parameters of Code Tracking Noise Magnitude Function

Signal	$a_{\rho^c,N}$ **(in cm)**	$b_{\rho^c,N}$ **(in cm)**	$x_{\rho^c,N}$ **(in dB-Hz)**
G1C	14.1	62.4	10.0
G2X	18.0	99.3	7.8
G5X	9.8	129.4	7.0
E1X	12.7	108.3	8.9
E5X	0.0	76.8	13.2
E7X	11.7	99.7	6.6

Figure 3.6 depicts the fitted standard deviation of the code tracking noise as a function of C/N_0 for all six employed signals. The corresponding numerical values are presented in Table 3.6. Theoretical analysis predicts a lower tracking noise for signals with a higher chipping rate [*Groves*, 2013, p. 397]. This does not correspond to the fitted functions. $\sigma_{\rho^c,N}$ for G1C and E5X are the lowest of all six signals throughout the fitting interval, while $\sigma_{\rho^c,N}$ for G2X and G5X are the highest. However, the signals with higher chipping rate also exhibit higher average C/N_0 values (e. g. 54.2 dB-Hz for E7X compared to 50.1 dB-Hz for G2X), so

their RMS tracking noise error over all values to fit is in fact the lowest: 16.5 cm to 18.7 cm for the three signals with 10.23 MHz chipping rate, 23.5 cm to 26.0 cm for the two signals with 1.023 MHz chipping rate and 29.2 cm for the G2X signal with 511.5 kHz chipping rate.

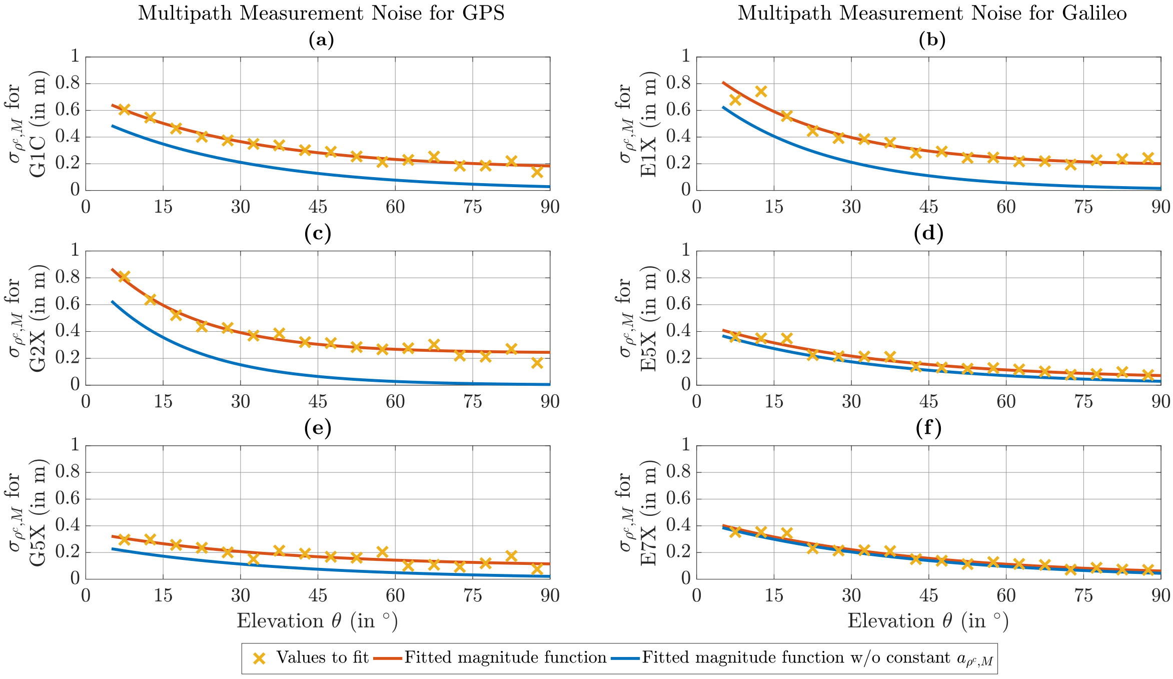

Figure 3.7: Fitted standard deviation of code multipath measurement noise, based on kinematic data. **(a)**,**(c)**,**(e)** GPS signals G1C, G2X and G5X. **(b)**,**(d)**,**(f)** Galileo signals E1X, E5X and E7X.

Table 3.7: Fitted Parameters of Code Multipath Error Magnitude Function. $a_{\rho^c,M}$ is Removed After the Fitting Process Because the Effect of the Constant Noise Floor is Accounted for via the Code Tracking Noise Function $\sigma_{\rho^c,N}$.

Signal	$\boldsymbol{a_{\rho^c,M}}$ **(in cm)**	$\boldsymbol{b_{\rho^c,M}}$ **(in cm)**	$\boldsymbol{x_{\rho^c,M}}$ **(in °)**
G1C	15.5	57.4	29.9
G2X	24.0	83.3	17.6
G5X	9.4	26.2	35.8
E1X	18.5	78.0	23.0
E5X	4.3	42.7	33.5
E7X	1.7	43.8	39.3

Figure 3.7 presents the fitted standard deviation of the code multipath error as a function of θ for all six signals, while Table 3.7 lists the numerical values. The expected dependency of the multipath error on the chipping rate is clearly visible: G5X, E5X and E7X have the same f_{co} and very similar multipath characteristics. Due to its BOC modulation, E1X is expected to suffer from less multipath error than the BPSK-modulated G1C, although both share a common chipping rate. This expectation is not backed up by the experimental data, which show better multipath characteristics on G1C. G2X with the lowest chipping rate has the highest multipath error, but the difference to E1X is small.

The presented results for $\sigma_{\rho^c,M}$ and $\sigma_{\rho^c,N}$ are applicable to single-frequency pseudoranges. When forming IF combinations from two pseudoranges on different frequencies, multipath and code tracking errors from both involved signals have to be taken into account. Their contributions are scaled depending on the two involved frequencies:

$$\sigma^2_{\rho^c_{p/q},M} = \left(\frac{1}{1-\gamma_{p,q}}\right)^2 \sigma^2_{\rho^c_p,M} + \left(\frac{\gamma_{p,q}}{1-\gamma_{p,q}}\right)^2 \sigma^2_{\rho^c_q,M}. \tag{3.50}$$

This equation holds for tracking noise errors in the same way as it does for multipath errors by simply replacing the subscript M with N in all three standard deviations. The numerical values for the scaling factors of the individual variances range from 1.59 to 6.48, demonstrating why IF combinations are associated with increased noise. Due to the zero-mean characteristics of both ρ_M and ρ_N, the increased noise is filtered out by the EKF over time.

3.6.5 Total Pseudorange Error

To sum up the results presented so far and provide a clear overview of the achievable GNSS positioning performance, the total pseudorange measurement error $\delta\tilde{\rho}$ is evaluated in this section. Only the static data sets are taken into account to exclude large multipath errors and possible influences due to the varying accuracy of the receiving antenna's reference position in kinematic data. The reference value for the pseudoranges is taken to be the true range from satellite antenna to receiver antenna. The satellite's position is computed based on final precise satellite orbit and clock data. The receiver antenna's position is known with an accuracy of better than 5 cm. All effects that are known a-priori or from the available pseudoranges (δt_R, δt_S, $\delta\rho_I$, $\delta\rho_T$, $d_{p,q}$) are compensated in the same manner as they are for processing in the integration filter when forming the corrected pseudoranges ρ^c.

Figures 3.8 and 3.9 display the CDF and the relative frequency of the total errors for five different Galileo pseudoranges, respectively: single-frequency observations on E1X, E5X and E7X as well as IF combinations on E1X/E5X and E1X/E7X. Galileo was chosen because it offers the largest amount of multi-frequency observations. The corresponding plots for GPS look similar and appear in the appendix (Section B.2). The same holds for the total pseudorange errors of GPS and Galileo when broadcast satellite orbits and clocks are used instead of ultra-rapid precise data. The error of the IF combinations in Figures 3.8d, e and 3.9d, e appears to be approximately normally distributed with mean and median close to zero. In comparison, the single-frequency observations all have a mean and a median significantly smaller than zero, indicating the effect of the unmodeled ionospheric delay. Their relative frequencies also feature a lower maximum and wider slopes. These findings are quantified by the numerical values in Table 3.8. The means of all four IF combinations utilized in the integration filter differ less than 1σ from zero. For single-frequency pseudoranges, the opposite is true. Because the magnitude of the unmodeled ionospheric delay varies from one data set to another, the standard deviations of the IF combinations are smaller than the ones for the single-frequency pseudoranges, too. All this leads to an RMS of 1.11 m to 1.44 m for the errors of the IF combinations compared to RMS values of 3.00 m to 5.44 m for the errors of the single-frequency observations. These results underline the advantages of working with IF combinations as far as accuracy is concerned. Additionally, the small means of the IF combinations in Table 3.8 serve as a verification for the DCB calibration performed in Section 3.4.

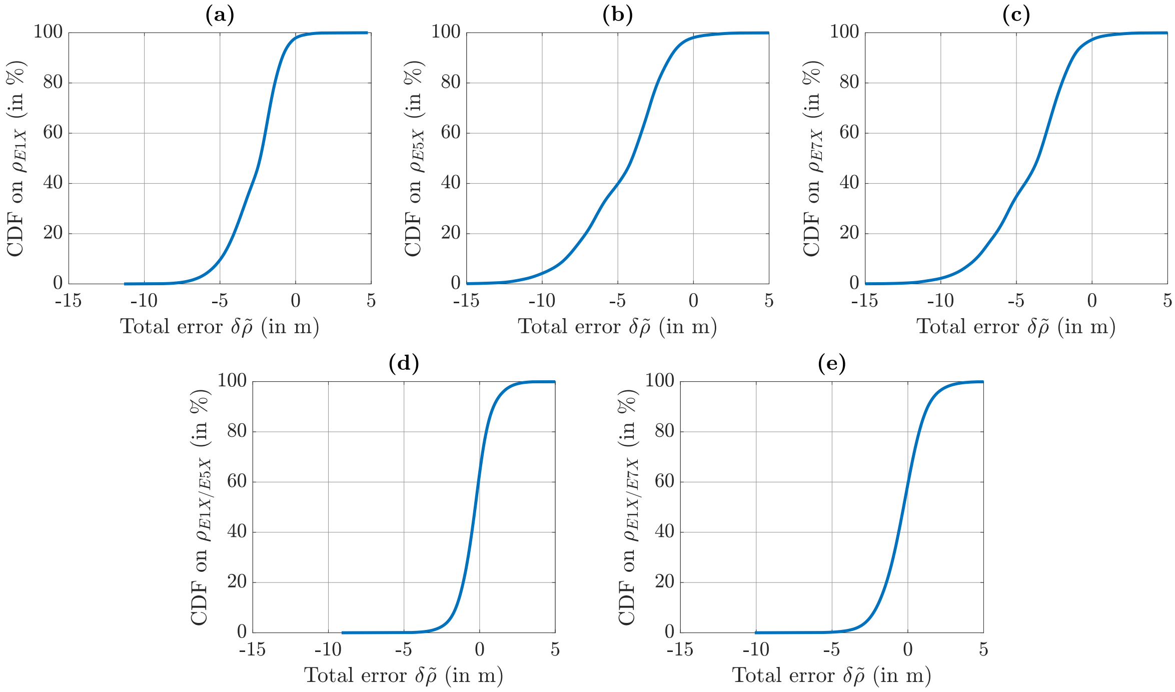

Figure 3.8: CDF of total pseudorange error for Galileo with predicted half of ultra-rapid precise satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on E1X, E5X and E7X. **(d)**,**(e)** IF combinations on E1X/E5X and E1X/E7X.

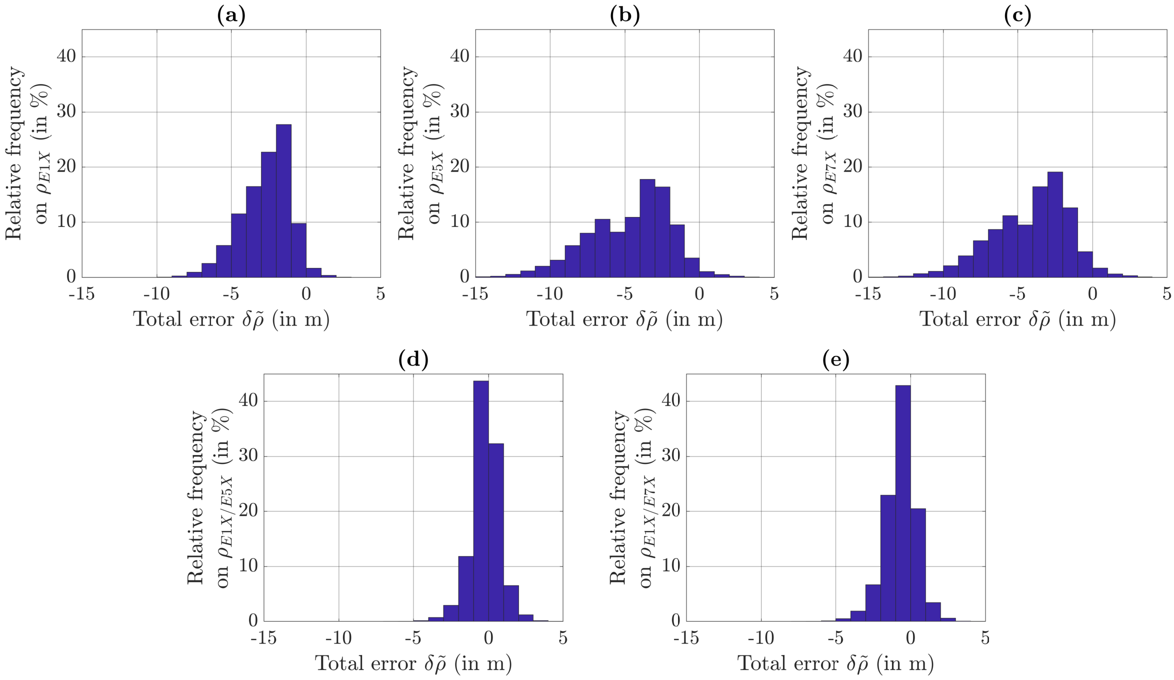

Figure 3.9: Relative frequency of total pseudorange error for Galileo with predicted half of ultra-rapid precise satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on E1X, E5X and E7X. **(d)**,**(e)** IF combinations on E1X/E5X and E1X/E7X.

Table 3.8: Total Pseudorange Error With Predicted Half of Ultra-Rapid Precise Satellite Orbits and Clocks, Corrected for all Applicable DCBs. Only Static Data to Exclude Large Multipath Errors

Signal	**Mean (in m)**	**Standard Deviation (in m)**	**RMS (in m)**
G1C	−2.55	1.58	3.00
G2X	−4.04	2.38	4.69
G5X	−4.53	2.56	5.20
G1C/G2X	−0.33	1.41	1.44
G1C/G5X	−0.31	1.06	1.11
E1X	−2.70	1.64	3.16
E5X	−4.13	2.60	4.88
E7X	−4.67	2.79	5.44
E1X/E5X	−0.66	1.08	1.26
E1X/E7X	−0.66	1.08	1.26

3.6.6 Range Rates

The magnitude of the range rate error $\delta\tilde{\dot{\rho}}$ is fitted based on the difference between a reference value - the true line-of-sight velocity between satellite and receiver antenna based on final precise orbit and clock data - and the corrected range rates $\dot{\rho}^c$. In analogy to the total pseudorange error, all effects that are known a-priori ($\delta\dot{t}_R$, $\delta\dot{t}_S$) are compensated in the same manner as they are for processing in the integration filter.

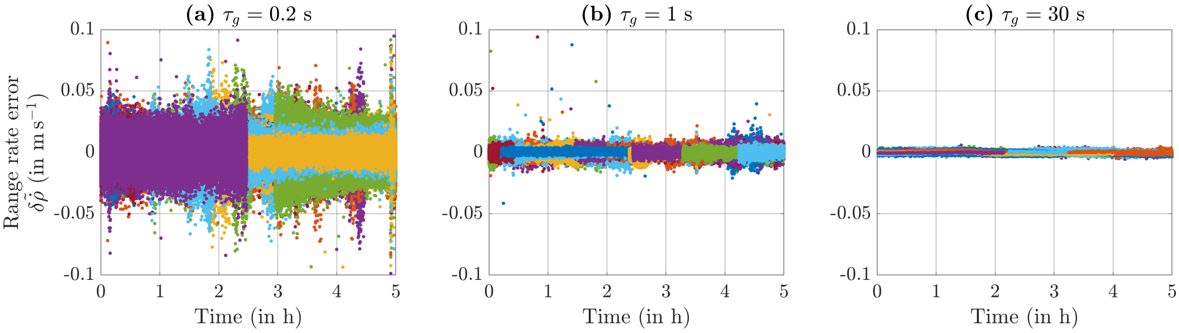

Figure 3.10: Range rate error in static data for varying GNSS measurement intervals τ_g for both GPS and Galileo. Different colors indicate different satellites.

Figure 3.10 depicts the influence of the GNSS measurement interval τ_g on the range rate error $\delta\tilde{\dot{\rho}}$ in static data. Because range rates are the time derivative of carrier phase measurements, their accuracy increases with increasing measurement intervals, as long as the carrier phase's accuracy remains independent of time. Assuming uncorrelated input data and no cycle slips, the variance of $\dot{\rho}^c$ if obtained from (3.4) via variance propagation is:

$$\sigma^2_{\dot{\rho}^c_k} = 2\left(\frac{\lambda_{ca}}{2\pi\tau^c_{g,k}}\right)^2\sigma^2_{\phi} + \left(\frac{\lambda_{ca}}{2\pi}\frac{\phi^c_k - \phi^c_{k-1}}{{\tau^c_{g,k}}^2}\right)^2\sigma^2_{\tau^c_{g,k}} + c^2\sigma^2_{\delta\hat{\dot{t}}^-_{R,k}}\,. \tag{3.51}$$

Due to the high accuracy of GNSS-based time measurement, the second term on the right hand side of (3.51) can be neglected. As $\delta\hat{\dot{t}}^-_{R,k}$ is estimated based on range rate observations, its variance decreases with increasing τ_g, too. This holds as long as the assumption of a constant receiver clock drift throughout the measurement interval is true. Consequently, $\sigma^2_{\tilde{\rho}^c_k}$ decreases with the square of τ_g for small values of τ_g. This is verified by the results in Figures 3.10a, b: The product $\sigma^2_{\tilde{\rho}^c_k} {\tau^c_{g,k}}^2$ is approx. $3\,\mathrm{mm}^2$ for $\tau_g = 0.2\,\mathrm{s}$ and approx. $5\,\mathrm{mm}^2$ for $\tau_g = 1\,\mathrm{s}$. For larger values of τ_g, the variation of $\delta \dot{t}_R$ during the measurement interval plays a significant role. Therefore, the product $\sigma^2_{\tilde{\rho}^c_k} {\tau^c_{g,k}}^2$ increases for larger measurement intervals and is approx. $2000\,\mathrm{mm}^2$ for $\tau_g = 30\,\mathrm{s}$.

While τ_g is the dominant factor for the range rates' accuracy in static data, its influence in kinematic data sets gets outmatched by two additional error sources: On the one hand, the vehicle's velocity might change during τ_g. This is reflected better in the reference solution with its data rate of 100 Hz than in the range rates with their data rate of 5 Hz to 10 Hz. On the other hand, the kinematic data sets feature varying GNSS reception conditions, resulting in loss of tracking lock, a reduced C/N_0 and ultimately cycle slips.

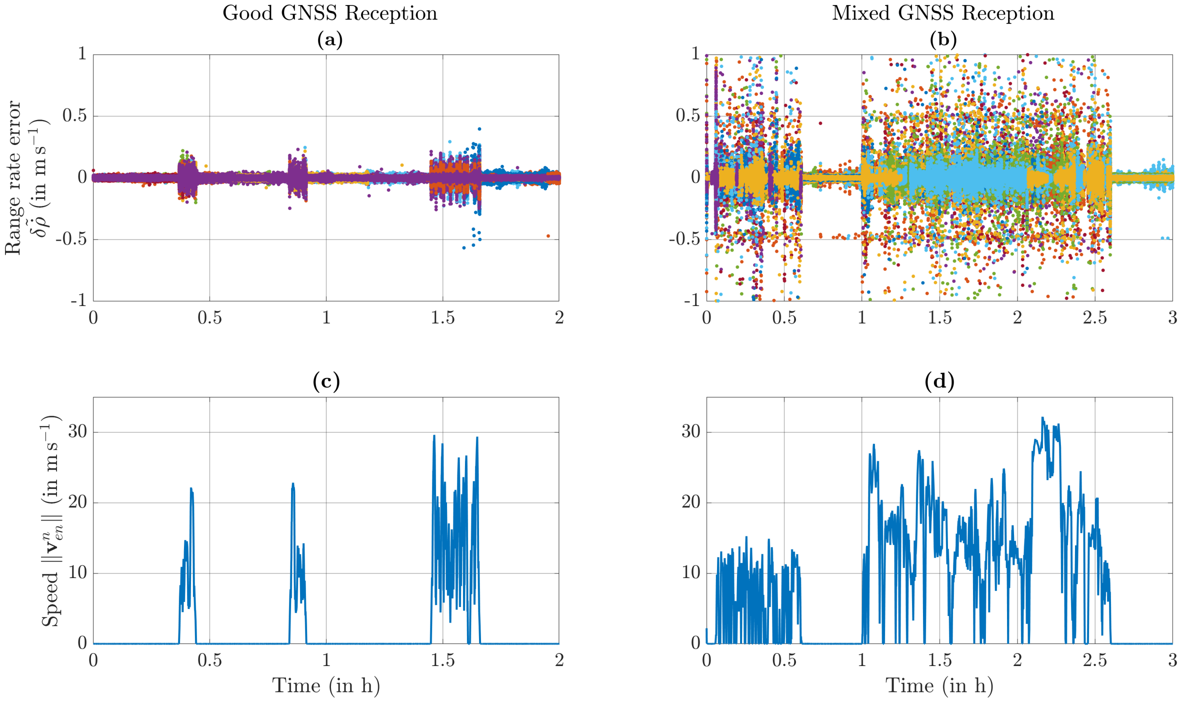

Figure 3.11: Range rate error and vehicle speed in kinematic data for varying GNSS reception conditions for both GPS and Galileo. GNSS measurement interval $\tau_g = 0.2\,\mathrm{s}$, different colors in **(a)**,**(b)** indicate different satellites.

These effects are visible in Figure 3.11. It depicts the range rate error $\delta\tilde{\dot{\rho}}$ and the vehicle speed $\|\mathbf{v}^n_{en}\|$ as a function of time in two different kinematic data sets: one with good GNSS reception conditions (Figures 3.11a, c) and one with mixed GNSS reception conditions (Figures 3.11b, d). τ_g is 0.2 s in both sets. The scale of the vertical axis is reduced by a factor of ten in comparison to Figure 3.10 due to the increased magnitude of $\delta\tilde{\dot{\rho}}$. Figure 3.11a visualizes the increased range rate error due to the vehicle's movement. During standstill, $\delta\tilde{\dot{\rho}}$ is on the same level as in Figure 3.10a. As soon as the vehicle starts moving, the magnitude of $\delta\tilde{\dot{\rho}}$ increases. The value of $\delta\tilde{\dot{\rho}}$ depends on the change of $\mathbf{v}^n_{en}$, so the range rate error is especially large during accelerating, braking and cornering maneuvers. Figure 3.11b illustrates the second additional error source in kinematic data: cycle slips due to poor GNSS reception conditions. Even during the standstill phase from 0.6 h to 1 h, peaks of approx. $\pm 0.5\,\mathrm{m\,s^{-1}}$ are visible. These correspond to carrier phase errors of approx. $\pm 10\,\mathrm{cm}$ or $\pm\frac{\lambda_{ca}}{2}$ for $\tau_g = 0.2\,\mathrm{s}$. Such half-cycle slips can occur when tracking a data-modulated signal (e. g. G1C) with a tracking loop insensitive to errors of ±180° [*Groves*, 2013, p. 446]. Once the vehicle starts moving, cycle slips occur frequently, resulting in range rate errors of up to $6\,\mathrm{m\,s^{-1}}$. However, this effect is exaggerated in Figure 3.11b due to the amount of observations. The overall RMS of $\delta\tilde{\dot{\rho}}$ is approx. $7\,\mathrm{cm\,s^{-1}}$ and only $5\,\%$ of range rate observations have an absolute error of more than that. In comparison, the overall RMS for the data set in Figure 3.11a is approx. $1.5\,\mathrm{cm\,s^{-1}}$ and $1\,\%$ of range rate observations have an absolute error of more than $7\,\mathrm{cm\,s^{-1}}$.

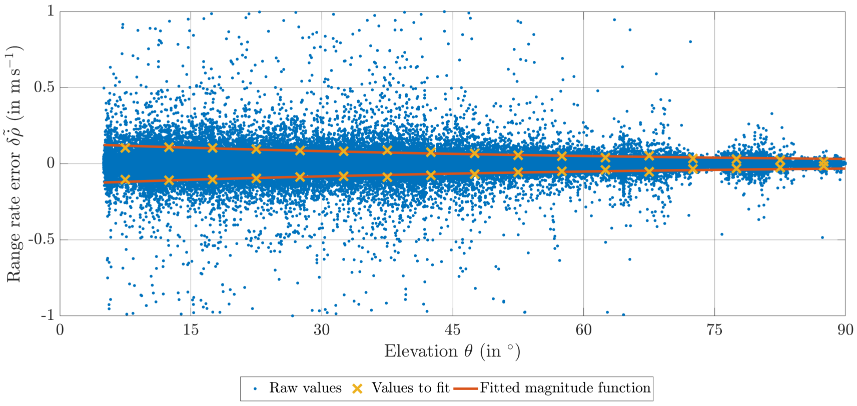

Figure 3.12: Range rate error in kinematic data and its fitted standard deviation.

Figure 3.12 shows $\delta\tilde{\dot{\rho}}$ from all kinematic data sets as a function of θ. In analogy to the unmodeled atmospheric delays, the fitted error magnitude function and its negative are depicted in the same plot as the actual error. Fitting the error magnitude function is also done in the same way with 17 bins of 5° each. The calculation of standard deviation, RMS and $68.3\,\%$ quantile reveals that while the first two are almost identical (indicating a mean close to zero), the latter one is roughly three times smaller. This was to be expected because the errors induced by cycle slips influence standard deviation and RMS, but not the $68.3\,\%$ quantile due to their infrequency. The largest RMS occurs in the second bin with $10.9\,\mathrm{cm\,s^{-1}}$. This is still small enough to enable the detection of half-cycle slips on the 3σ level for GNSS measurement rates of $5\,\mathrm{Hz}$ or higher. Consequently, the RMS is chosen in order to get an conservative estimate of the actual error while still enabling detection and elimination of outliers. The data points and the fitted function (as well as its negative) are displayed in Figure 3.12. The numerical representation of $\sigma_{\dot{\rho}^c}$ is:

$$\sigma_{\dot{\rho}^c}(\theta) = 13.4\,\mathrm{cm\,s^{-1}}\;\mathrm{e}^{-\frac{\theta}{62.4^\circ}}\,. \tag{3.52}$$

The constant term $a_{\dot{\rho}^c}$ is neglected because it is estimated by the fitting algorithm as $a_{\dot{\rho}^c} = 1.7\cdot 10^{-11}\,\mathrm{m\,s^{-1}}$.

Chapter 4

Odometry Preprocessing

This chapter details how odometry measurements received from the vehicle's built-in sensors are preprocessed before they are forwarded to the integration filter. The forwarded data consist of the observables described in Section 4.1 and their associated measurement noise, modeled as laid out in Section 4.3. A central part of the odometry preprocessing is the estimation of longitudinal and lateral tire slip, both of which are then compensated in order to obtain vehicle velocities w. r. t. the ECEF frame as accurately as possible. Slip estimation is performed with different models of varying complexity, as detailed in Section 4.2.

4.1 Observables

The vehicle's built-in ESC distributes several types of messages via CAN bus. Out of these messages, two types of measurements are utilized in the navigation algorithm: wheel rotation rate ω_w for each of the four wheels and steering wheel angle δ_{sw}. The wheel rotation rates are favored over the likewise available wheel ticks, i. e. the number of wheel encoder increments that have moved past the detector since the last measurement epoch. This is done because the wheel ticks suffer from large quantization errors due to the encoder's limited angular resolution [*Bevly and Cobb*, 2010, Section 4.4.2]. While this angular resolution is the same for the rotation rates, they are computed internally by the ESC with its built-in clock, enabling a much higher time resolution than the interval between two ESC messages of the same type being transmitted via CAN bus, which is 10 ms for the particular vehicle utilized to collect measurement data in this thesis. As a result, the rotation rates offer a resolution of approx. $8.2 \cdot 10^{-3}\,\mathrm{rad\,s^{-1}}$ instead of approx. $6.5\,\mathrm{rad\,s^{-1}}$ for the wheel ticks, as determined from measurement data. The resolution for the wheel rotation rate observations is obtained from the wheel speed resolution of $0.01\,\mathrm{km\,h^{-1}}$ in the respective CAN message, divided by a dynamic tire radius r_d of 34 cm. This value for r_d represents the arithmetic mean of the values at the four wheels, obtained via reference equipment as detailed in Section 4.2. The resolution for the wheel tick observations is obtained from the encoder's resolution of 96 increments per full wheel revolution, divided by the time interval of 10 ms between two messages containing wheel tick data being transmitted via CAN bus.

In order to apply the odometry observations correctly within the integration filter, they need to have a timestamp t_o^G, given in GPS system time t^G. Because t^G is unknown to the ESC, the values for t_o^G are supplied by one of the on-board GNSS receivers. Due to their location in different CAN messages, ω_w and δ_{sw} are not transmitted simultaneously. This is countered by interpolating the time series for δ_{sw} with the query points of the timestamps for ω_w. Afterwards, all five odometry measurements in epoch k have the same timestamp $t_{o,k}^G$.

The odometry preprocessing inputs these data and calculates the velocity vector $\mathbf{v}_{ew}^{b}$ for each wheel:

$$\mathbf{v}_{ew}^{b} = \begin{pmatrix} v_{ew,F}^{b} \\ v_{ew,L}^{b} \\ v_{ew,U}^{b} \end{pmatrix}. \tag{4.1}$$

The entry $v_{ew,U}^{b}$ is only included to keep the notation consistent with the definitions from Section 2.1 and to allow for the usage of operations reserved for three-dimensional space (e. g. the cross product). $v_{ew,U}^{b}$ cannot be measured by the odometry sensors and is not processed in the integration filter. For computation of $\mathbf{v}_{ew}^{b}$, each wheel's velocity in its own wheel coordinate frame w is determined based on its rotation rate ω_w, the a-priori estimate of the corresponding dynamic tire radius $\hat{r}_d^-$ and the estimated slip values $\hat{\lambda}_x$ and $\hat{\alpha}$ by rearranging the slip definitions (2.25) and (2.26). In order to prevent the distinction of cases in the definition of longitudinal slip (2.25) from appearing whenever λ_x is mentioned, an auxiliary quantity κ_x, called longitudinal tire slip correction factor, is introduced:

$$\kappa_x = \begin{cases} \dfrac{1}{1+\lambda_x} & \text{for } \omega_w r_d \leq v_{ew,x}^{w} \quad \text{(braking)} \\ 1-\lambda_x & \text{for } \omega_w r_d > v_{ew,x}^{w} \quad \text{(acceleration).} \end{cases} \tag{4.2}$$

With the help of κ_x, each wheel's velocity in its own wheel coordinate frame is expressed as:

$$\mathbf{v}_{ew}^{w} = \begin{pmatrix} v_{ew,x}^{w} \\ v_{ew,y}^{w} \\ v_{ew,z}^{w} \end{pmatrix} = \begin{pmatrix} 1 \\ -\tan\hat{\alpha} \\ 0 \end{pmatrix} \omega_w \hat{r}_d^- \hat{\kappa}_x. \tag{4.3}$$

Different models for slip estimation are detailed in Section 4.2. Because r_d is modeled as random walk, the propagation simplifies to:

$$\hat{r}_{d,k}^{-} = \hat{r}_{d,k-1}^{+}. \tag{4.4}$$

For the unsteered rear wheels, $\mathbf{v}_{ew}^{b}$ and $\mathbf{v}_{ew}^{w}$ are identical. For the steered front wheels, the two rotation matrices $\mathbf{C}_b^{fl}$ for front left and $\mathbf{C}_b^{fr}$ for front right are obtained from the steering wheel angle δ_{sw}. This requires the individual steering angles δ_{fl} and δ_{fr}, which are expressed in terms of the mean steering angle δ_f and the difference steering angle $\Delta\delta_f$ at the front axle [*Harrer and Pfeffer*, 2017, Section 4.2]:

$$\delta_f = \frac{\delta_{fl}+\delta_{fr}}{2} \qquad \Delta\delta_f = \delta_{fl}-\delta_{fr} \tag{4.5}$$

$$\delta_{fl} = \delta_f + \frac{1}{2}\Delta\delta_f \qquad \delta_{fr} = \delta_f - \frac{1}{2}\Delta\delta_f. \tag{4.6}$$

δ_f and $\Delta\delta_f$ are fitted as third order polynomials of δ_{sw}:

$$\delta_f\left(\delta_{sw}\right) = \sum_{i=0}^{3} c_{\delta_f,i}\delta_{sw}^{i} \qquad \Delta\delta_f\left(\delta_{sw}\right) = \sum_{i=0}^{3} c_{\Delta\delta_f,i}\delta_{sw}^{i}. \tag{4.7}$$

Symmetrical steering is assumed, i. e. the left wheel does the same for $\delta_{sw} > 0$ as the right wheel does for $\delta_{sw} < 0$, but with opposite sign:

$$\delta_{fl}\left(\delta_{sw}\right) = -\delta_{fr}\left(-\delta_{sw}\right). \tag{4.8}$$

Imposing this condition on (4.7) reduces the number of polynomial coefficients to be fitted by a factor of two. As a result, δ_f becomes an odd function of δ_{sw}, while $\Delta\delta_f$ becomes an even function of δ_{sw}:

$$\delta_f\left(\delta_{sw}\right) = c_{\delta_f,1}\delta_{sw} + c_{\delta_f,3}\delta_{sw}^{3} \qquad \Delta\delta_f\left(\delta_{sw}\right) = c_{\Delta\delta_f,0} + c_{\Delta\delta_f,2}\delta_{sw}^{2}. \tag{4.9}$$

The four remaining coefficients are fitted based on measurement data obtained during standstill with the front wheels placed on turntables. δ_{sw} is known from the built-in sensor, δ_{fl} and δ_{fr} are measured by the turntables. Figure 4.1 shows the measured values and the fitted polynomials. The sample points are not spaced evenly, but densified around $\delta_{sw} = 0$ because small steering wheel angles are more common during ordinary driving scenarios than large ones.

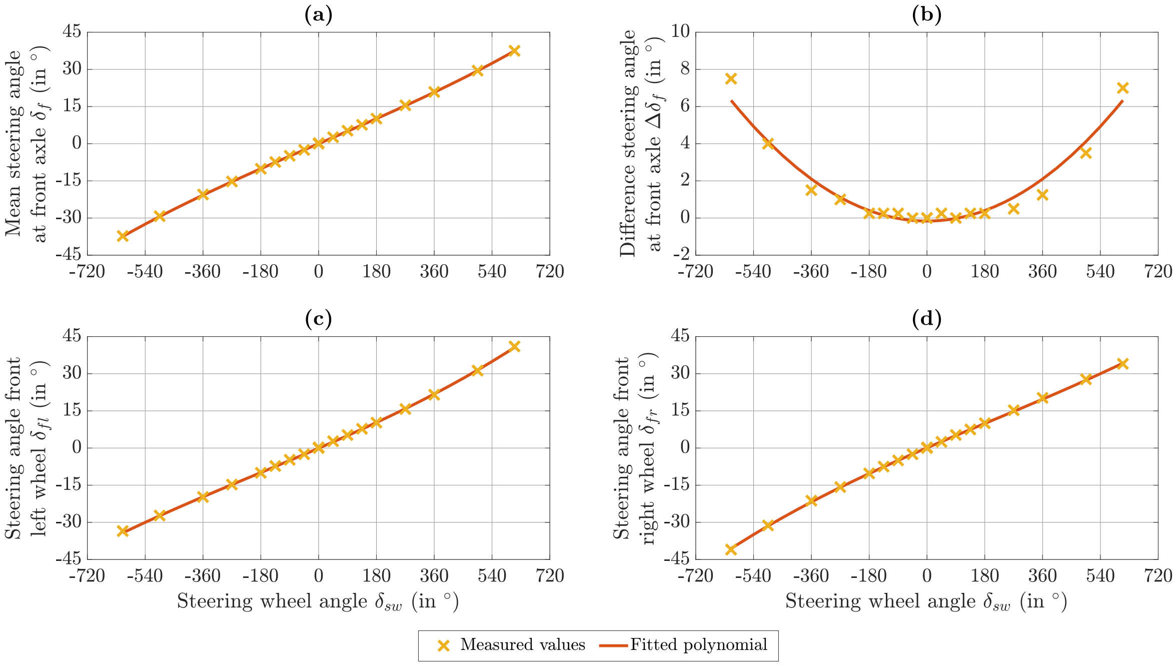

Figure 4.1: Steering angles as a function of the steering wheel angle. **(a)** Mean angle at front axle. **(b)** Difference angle at front axle (scale of vertical axis differs from the other three subfigures). **(c)** Front left wheel. **(d)** Front right wheel.

4.2 Tire Slip Estimation

Estimating the tires' longitudinal and lateral slip is an integral aspect of odometry preprocessing, because both types of slip need to be compensated for accurate estimation of the vehicle's velocity w. r. t. the Earth. The degree of accuracy enhancement generated by proper slip estimation depends on the driving scenario's acceleration profile. Driving with moderate horizontal acceleration corresponds to low tire slip levels. Consequently, neglect of tire slip does not decrease velocity estimation accuracy as severe as it does for more dynamic driving scenarios with large horizontal accelerations and high levels of tire slip. Two different tire models for slip estimation are described and parametrized in this section. The section starts with all aspects that are common to both presented slip estimation methods and the measurement data recorded for model parametrization. Each model's individual aspects as well as the parametrization results are detailed in the corresponding subsections. Section 4.2.3 lays out a way to estimate the tires' side slip based on gyroscope measurements instead of accelerometer measurements to reduce the influence of vehicle vibrations on the side slip estimate.

Common to all examined methods is the application of a single-track model. Hence, all quantities related to slip estimation (α, λ_x, κ_x, μ_x, μ_y, $\mathbf{F}^w_{ew}$) may differ from front to rear axle, but are identical for left and right wheels. The position $\mathbf{p}^b_{bc}$ of the vehicle's CG is modeled as known and invariant, because the employed measurement equipment is unable to detect changes in $\mathbf{p}^b_{bc}$ during driving. The CG is assumed to be located above the line connecting the single-track model's two wheels, with height h_{CG} above the road surface. Wheel load transfer between front and rear axle due to longitudinal acceleration is therefore included in tire slip estimation. In contrast, wheel load transfer between left and right wheels is not included in tire slip estimation, since this requires a more advanced modeling concept than the single-track model. All calculations are performed with specific forces $\mathbf{f}_{pq}$, obtained by dividing a force $\mathbf{F}_{pq}$ by the vehicle mass m_v:

$$\mathbf{f}_{pq} = \frac{\mathbf{F}_{pq}}{m_v}. \tag{4.10}$$

Because formulation of kinetic equations is simplest for a rigid body's CG [*Gross et al.*, 2019, Section 3.4], the corrected IMU measurements are transferred to the vehicle's CG:

$$\mathbf{f}^b_{ic} = \mathbf{f}^b_{ib} + \boldsymbol{\omega}^b_{ib} \times \left(\boldsymbol{\omega}^b_{ib} \times \mathbf{p}^b_{bc}\right) + \dot{\boldsymbol{\omega}}^b_{ib} \times \mathbf{p}^b_{bc}. \tag{4.11}$$

$\mathbf{f}^b_{ic}$ is the specific force acting on the CG w. r. t. the ECI frame. $\mathbf{f}^b_{ib}$ and $\boldsymbol{\omega}^b_{ib}$ are the IMU measurements, corrected for the estimated IMU errors. $\dot{\boldsymbol{\omega}}^b_{ib}$ is assumed to be negligible. In the context of tire slip estimation, forces and kinematic quantities of the vehicle w. r. t. the road are of interest. Due to the magnitude of these kinematic quantities (e. g. yaw rates of up to $40\,^\circ\,\mathrm{s}^{-1}$), the motion between the ECEF frame and the ECI frame can be neglected, yielding $\mathbf{f}^b_{ic} = \mathbf{f}^b_{ec}$ and $\boldsymbol{\omega}^b_{ic} = \boldsymbol{\omega}^b_{ec}$. Therefore, wheel load estimation is performed based on the components of $\mathbf{f}^b_{ec}$, known from (4.11):

$$f^b_{ef,U} = f^b_{ec,U}\frac{l_r}{l} - f^b_{ec,F}\frac{h_{CG}}{l} \tag{4.12}$$

$$f^b_{er,U} = f^b_{ec,U}\frac{l_f}{l} + f^b_{ec,F}\frac{h_{CG}}{l}, \tag{4.13}$$

where $f^b_{ef,U}$ and $f^b_{er,U}$ are the specific wheel loads for the single-track model's front and rear wheel, respectively. For computation of the lateral specific forces, steady-state cornering ($\ddot{\psi}_{nb} = 0$) is assumed, yielding:

$$f^b_{ef,L} = f^b_{ec,L}\frac{l_r}{l} \qquad\qquad f^b_{er,L} = f^b_{ec,L}\frac{l_f}{l}. \tag{4.14}$$

The longitudinal components $f^b_{ef,F}$ and $f^b_{er,F}$ complete the vectors $\mathbf{f}^b_{ef}$ and $\mathbf{f}^b_{er}$. A distinction of cases needs to be made because the particular vehicle utilized to collect measurement data in this thesis brakes with all four wheels, but accelerates only with the front ones:

$$f^b_{ef,F} = \begin{cases} f^b_{ec,F} & \text{for } f^b_{ec,F} \geq 0 \\ f^b_{ec,F}\frac{f^b_{ef,U}}{f^b_{ec,U}} & \text{for } f^b_{ec,F} < 0 \end{cases} \tag{4.15}$$

$$f^b_{er,F} = \begin{cases} 0 & \text{for } f^b_{ec,F} \geq 0 \\ f^b_{ec,F}\frac{f^b_{er,U}}{f^b_{ec,U}} & \text{for } f^b_{ec,F} < 0. \end{cases} \tag{4.16}$$

(4.15) and (4.16) include the additional assumption that the distribution of brake force between the single-track model's front and rear wheel happens according to the wheel load:

$$\frac{f^b_{ef,F}}{f^b_{er,F}} = \frac{f^b_{ef,U}}{f^b_{er,U}} \quad \text{for } f^b_{ec,F} < 0. \tag{4.17}$$

For $\delta_f = 0$, this is denoted as ideal brake force distribution and equivalent to the equality of the longitudinal friction coefficients at both wheels [*Pischinger and Seiffert*, 2016, p. 752]:

$$\mu_{x,f} = \mu_{x,r} = \frac{f^b_{ec,F}}{f^b_{ec,U}} \quad \text{for } f^b_{ec,F} < 0. \tag{4.18}$$

However, typical built-in brake force distributions feature a constant ratio between $f^b_{ef,F}$ and $f^b_{er,F}$, independent of $f^b_{ec,F}$ [*Pischinger and Seiffert*, 2016, p. 753], so assuming an ideal brake force distribution causes errors in the estimation of longitudinal tire slip. It is done nonetheless because the vehicle's actual brake force distribution is unknown. The introduced error is minimized by applying the same assumption for the parametrization of the tire slip model as for the slip estimation algorithm, leading to some degree of error cancellation.

Because tire slip estimation happens in the wheel coordinate frame, $\mathbf{f}^b_{ef}$ needs to be rotated accordingly to get $\mathbf{f}^f_{ef}$. The rotation matrix $\mathbf{C}^f_b$ for the single-track model's front wheel is obtained from δ_f. Since the rear wheel is unsteered, no rotation is necessary and $\mathbf{f}^b_{er} = \mathbf{f}^r_{er}$. Longitudinal and lateral friction coefficients for both of the single-track model's wheels are computed from the components of $\mathbf{f}^f_{ef}$ and $\mathbf{f}^r_{er}$:

$$\mu_{x,f} = \frac{f^f_{ef,x}}{f^f_{ef,z}} \qquad \mu_{x,r} = \frac{f^r_{er,x}}{f^r_{er,z}} \tag{4.19}$$

$$\mu_{y,f} = \frac{f^f_{ef,y}}{f^f_{ef,z}} \qquad \mu_{y,r} = \frac{f^r_{er,y}}{f^r_{er,z}}. \tag{4.20}$$

With the computation of these friction coefficients, the steps common to both employed tire slip estimation models are concluded. Details about the different models and their parametrization results appear in the respective subsections.

Measurement data for model parametrization are recorded in three sets:

1. Driving in a straight line with constant velocity on a horizontal road for estimation of each wheel's dynamic tire radius r_d. This is performed in both directions to compensate for potential systematic effects (e. g. varying drag due to wind) and with speeds ranging from $5\,\mathrm{km\,h^{-1}}$ to $100\,\mathrm{km\,h^{-1}}$.

2. Braking and accelerating in a straight line on a horizontal road with varying absolute acceleration values for estimation of the tire parameters for longitudinal slip. The maximum deceleration is reached when the vehicle's *anti-lock braking system* (ABS) engages, the maximum acceleration is limited by engine power.

3. Driving in a circle with constant velocity on a horizontal road to estimate the tire parameters for lateral slip. This is performed both clockwise and counter-clockwise and repeated with speeds ranging from $5\,\mathrm{km\,h^{-1}}$ to $35\,\mathrm{km\,h^{-1}}$. The circle's radius is kept constant at $15\,\mathrm{m}$, yielding lateral accelerations of up to $\pm 6.3\,\mathrm{m\,s^{-2}}$.

All sets are recorded with good GNSS conditions on a dry road. Reference values for $\mathbf{v}^b_{eb}$, $\mathbf{f}^b_{ib}$ and $\boldsymbol{\omega}^b_{ib}$ are obtained by integrating observations from a geodetic GNSS receiver for RTK positioning with a navigation grade IMU (see Chapter 7 for details). These reference values are used to parametrize the tire models. This parametrization is performed with a Gauss-Helmert model. The input data's variance is obtained from the precision of the reference values via variance propagation. As tire slip estimation is performed within the odometry preprocessing based on measurements from a MEMS IMU, observations from this MEMS IMU are recorded simultaneously and utilized for measurement noise parametrization in Section 4.3.

4.2.1 Linear Tire Model

The linear tire model assumes proportionality between μ_x and λ_x for longitudinal slip and between μ_y and α for lateral slip, with the proportionality constants c_λ and c_α according to (2.27). In general, both slip stiffnesses may vary from front to rear axle, yielding a total of four unknowns for parametrization of the model. With the parametrized model, tire slip estimation happens by combining (2.27), (4.19) and (4.20):

$$\hat{\lambda}_{x,f} = \frac{\mu_{x,f}}{c_{\lambda,f}} \qquad \hat{\lambda}_{x,r} = \frac{\mu_{x,r}}{c_{\lambda,r}} \tag{4.21}$$

$$\hat{\alpha}_f = \frac{\mu_{y,f}}{c_{\alpha,f}} \qquad \hat{\alpha}_r = \frac{\mu_{y,r}}{c_{\alpha,r}}. \tag{4.22}$$

The numerical values of the stiffnesses depend on the individual tire. The stiffness of a given tire typically increases with tire wear due to the reclining tread depth. In contrast, weather and road surface do not influence a tire's slip stiffness, but the maximum possible friction coefficient and the slip values for which this maximum friction coefficient is reached [*Pacejka*, 2006, Chapter 1]. Consequently, the individual tire and its surrounding conditions (e. g. road surface, weather, tire wear) as well as the desired accuracy determine the linear tire model's application boundary.

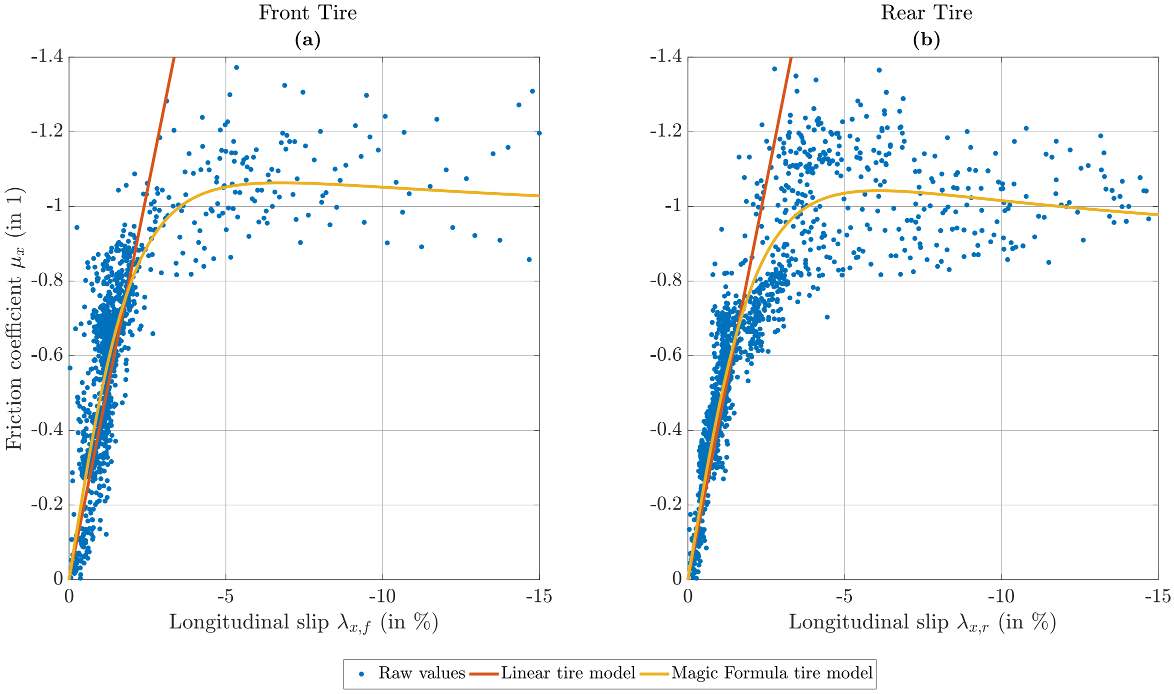

Figure 4.2: Fitted longitudinal tire slip models. **(a)** Front tire. **(b)** Rear tire.

Figures 4.2 and 4.3 depict the data points for fitting the tire models and the results of the fitting process. Both the linear and the Magic Formula tire model are shown in the same plot in order to compare them directly with each other. The Magic Formula and its comparison with the linear tire model are detailed in Section 4.2.2. The displayed values for the longitudinal slip λ_x are computed from the reference equipment's estimate of $\mathbf{v}_{eb}^b$, the wheel rotation rates from the CAN bus and the reference values for r_d, which are determined in a preceding step. The depicted side slip angles α_f and α_r are obtained based on the reference equipment's estimates of $\mathbf{v}_{eb}^b$ and $\boldsymbol{\omega}_{eb}^b$ as well as the steering angle δ_f, which in turn is computed from the steering wheel angle transmitted via the CAN bus with the help of (4.9). The displayed values for the friction coefficients μ_x and μ_y are obtained from the reference equipment's estimates of $\mathbf{f}_{ib}^b$ and $\boldsymbol{\omega}_{ib}^b$ by applying (4.11)-(4.20).

In order to decide whether usage of the linear tire model is appropriate, an application boundary needs to be set for it. This is done by finding the largest value of the friction coefficient $|\mu_{lin}|$ for which the relative error as defined by (4.23) stays below 5 % for all of the

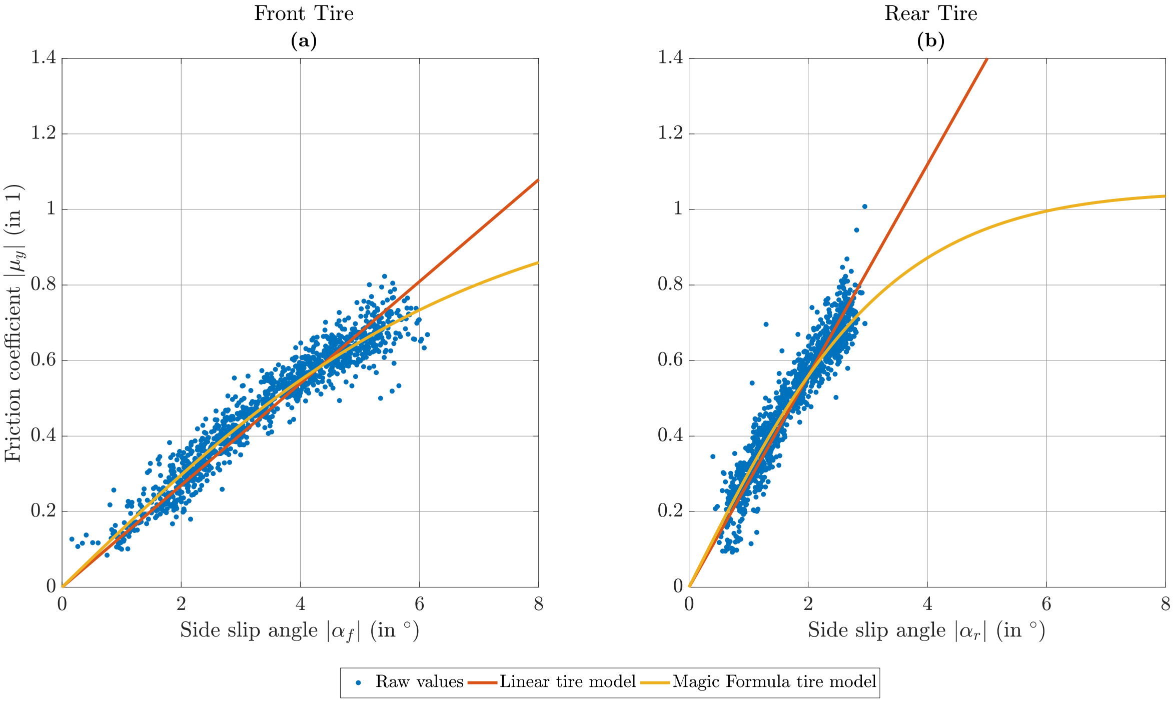

Figure 4.3: Fitted lateral tire slip models. **(a)** Front tire. **(b)** Rear tire.

four quantities $\mu_{x,f}$, $\mu_{x,r}$, $\mu_{y,f}$ and $\mu_{y,r}$:

$$\left|\frac{\delta\tilde{\mu}}{\mu_{MF}}\right| = \left|\frac{\mu_{lin} - \mu_{MF}}{\mu_{MF}}\right|, \tag{4.23}$$

where the subscripts lin and MF denote linear and Magic Formula tire model, respectively. Based on the measurement data, a limit of $|\mu_{lin}| \leq 0.7$ is identified, with $\mu_{y,r}$ being the critical of the four quantities $\mu_{x,f}$, $\mu_{x,r}$, $\mu_{y,f}$ and $\mu_{y,r}$. To assure that the combination of longitudinal and lateral slip does not exceed this level, $\sqrt{\mu_x^2 + \mu_y^2} \leq 0.7$ is set as application boundary for the linear tire model. If the estimated horizontal friction coefficient at one of the single-track model's tires exceeds this limit, measurements from the corresponding axle are not forwarded to the integration filter in that epoch. In case the vehicle's ABS or ESC engages, no odometry measurements from either axle are utilized in the EKF, because each of these indicators signalizes that the linear model's application limit is exceeded. Hence, only data points satisfying these limits are used for fitting the tire stiffnesses. Figure 4.2 depicts the longitudinal tire characteristics. Only braking is shown because the particular vehicle utilized for this thesis features an undriven rear axle. Also, higher absolute slip values are reached during braking than during acceleration. The longitudinal slip stiffness is similar for both of the single-track model's tires: $c_{\lambda,f} = 41.8$ and $c_{\lambda,r} = 42.5$, with a standard deviation of 0.9 in both cases. Figure 4.3 displays the lateral tire characteristics. Absolute values of α and μ_y are shown since tire behavior is assumed to be identical for left and right turns. In contrast to the longitudinal data in Figure 4.2, the measurement data only cover friction coefficients up to $|\mu| \approx 0.8$, so neither tire leaves its linear range by a large margin. Both tires experience similar friction coefficients due to the assumption of steady-state cornering (4.14) and the fact that longitudinal acceleration is close to zero during this measurement scenario. The front tire's side slip angle is larger than the rear tire's, resulting in a lower stiffness at the front axle: $c_{\alpha,f} = 7.7\,\mathrm{rad}^{-1}$ and $c_{\alpha,r} = 16.0\,\mathrm{rad}^{-1}$. The reason for this is that the steering elasticity reduces the effective side slip stiffness at the front axle, so the observable value of $c_{\alpha,f}$ is lower than the one for $c_{\alpha,r}$ [*Harrer and Pfeffer*, 2017, Section 5.2.2]. The obtained standard deviations are $\sigma_{c_{\alpha,f}} = 0.08\,\mathrm{rad}^{-1}$ and $\sigma_{c_{\alpha,r}} = 0.18\,\mathrm{rad}^{-1}$.

4.2.2 Magic Formula Tire Model

Pacejka's Magic Formula (2.28) is an empirical tire model for describing longitudinal and lateral tire forces as functions of λ_x and α, respectively. Its basic form, which is employed in this thesis, does not cover the occurrence of combined longitudinal and lateral slip simultaneously. In addition, it only applies to steady-state tire characteristics. Extensions of the basic model for combined slip and instationary tire behavior exist, but require extended model parametrization [*Pacejka*, 2006]. These extensions are therefore not covered in this thesis, because the aim is to keep the effort necessary for model parametrization as low as possible. Simplification of the Magic Formula is possible, e. g. by setting the curvature factor $E_{\lambda/\alpha}$ to zero [*Schramm et al.*, 2013, p. 237], yielding:

$$\begin{aligned} \mu_x &= D_\lambda \sin\left\{C_\lambda \arctan\left[B_\lambda \lambda_x\right]\right\} + G_\lambda \\ \mu_y &= D_\alpha \sin\left\{C_\alpha \arctan\left[\,B_\alpha \alpha\,\right]\right\} + G_\alpha. \end{aligned} \tag{4.24}$$

This approach is applied here due to the following reasons:

- Reduced parametrization effort.
- $E_{\lambda/\alpha}$ mainly influences the shape of the curves of $\mu_x(\lambda_x)/\mu_y(\alpha)$ close to the maximum friction coefficient and for even higher slip values [*Pacejka*, 2006, Section 4.3]. These high slip values correspond to high absolute horizontal accelerations, which rarely occur during ordinary driving scenarios.
- In contrast to the version with non-zero curvature factor, the simplified Magic Formula (4.24) can easily be solved algebraically for λ_x and α. This is important for both slip estimation and variance propagation (see Section 4.3).

In general, all model parameters in (4.24) may vary from front to rear axle, yielding a total of 16 unknowns for parametrization of the model. The numerical values of these unknowns depend on the individual tire. Among others, they change with tire wear, tire pressure, weather and road surface. Consequently, extensive calibration is required to parametrize the model for a variety of conditions. In order to keep the parametrization effort low, the 16 unknowns are only estimated for one set of conditions in this thesis. The results in Chapter 7 verify that even this reduced parametrization effort leads to improved accuracy in comparison with the linear tire model.

Table 4.1 shows the numerical results of the parameter adjustment, while Figures 4.2 and 4.3 visualize these results. The vertical shift $G_{\lambda/\alpha}$ is not estimated, because an adjustment process including this parameter indicates that it does not deviate from zero significantly in any of the four cases. Since the maximum friction coefficient in lateral direction $\mu_{y,M}$ is not reached during the corresponding measurement set, it is assumed to be identical to $\mu_{x,M}$, yielding $D_\alpha = D_\lambda$. As a consequence of assuming an ideal brake force distribution, $\mu_{x,M}$ is almost identical for front and rear. The Magic Formula model displays good agreement with the linear model up to the latter's validity limit of $|\mu| = 0.7$. The maximum absolute values of the differences in estimated slip are: less than 0.18 % for $\hat{\lambda}_{x,f}$, less than 0.11 % for $\hat{\lambda}_{x,r}$, less than 0.4° for $\hat{\alpha}_f$ and less than 0.25° for $\hat{\alpha}_r$.

The parameters' standard deviations in Table 4.1 reveal that the adjustment is more precise at the front tire than at the rear one. That's because the rear wheel load is smaller than the front wheel load. As a result, $\sigma_{\mu_{x,r}}$ is larger than $\sigma_{\mu_{x,f}}$, because both are obtained via variance propagation from the precision of $\mathbf{f}^b_{eb}$ as measured by the reference equipment. The static wheel load at the rear wheel is 69 % of the static front wheel load, while the rear wheel load during braking with $\mu_x = -1$ is less than 20 % of the front wheel load.

Table 4.1: Estimated Parameters and Their Standard Deviations for Magic Formula Tire Model

Parameter	**Unit**	$\boldsymbol{\lambda_{x,f}}$	$\boldsymbol{\lambda_{x,r}}$	$\boldsymbol{\alpha_f}$	$\boldsymbol{\alpha_r}$
$B_{\lambda/\alpha}$	1 or rad^{-1}	38.3 ± 1.2	34.5 ± 1.4	6.36 ± 0.35	12.3 ± 0.7
$C_{\lambda/\alpha}$	1	1.31 ± 0.02	1.39 ± 0.02	1.30 ± 0.06	1.39 ± 0.07
$D_{\lambda/\alpha}$	1	1.06 ± 0.01	1.04 ± 0.01	1.06 ± 0.01	1.04 ± 0.01
$G_{\lambda/\alpha}$	1	0	0	0	0

With the parametrized model, tire slip estimation is performed by combining (4.19), (4.20) and (4.24):

$$\hat{\lambda}_{x,f} = \frac{1}{B_{\lambda,f}} \tan \frac{\arcsin \frac{\mu_{x,f}}{D_{\lambda,f}}}{C_{\lambda,f}} \qquad \hat{\lambda}_{x,r} = \frac{1}{B_{\lambda,r}} \tan \frac{\arcsin \frac{\mu_{x,r}}{D_{\lambda,r}}}{C_{\lambda,r}} \tag{4.25}$$

$$\hat{\alpha}_f = \frac{1}{B_{\alpha,f}} \tan \frac{\arcsin \frac{\mu_{y,f}}{D_{\alpha,f}}}{C_{\alpha,f}} \qquad \hat{\alpha}_r = \frac{1}{B_{\alpha,r}} \tan \frac{\arcsin \frac{\mu_{y,r}}{D_{\alpha,r}}}{C_{\alpha,r}}. \tag{4.26}$$

Special care needs to be taken for large absolute horizontal accelerations: If a tire approaches its maximum friction coefficient, the input for the arcsine function may reach an absolute value larger than one (e. g. due to IMU measurement noise), which is outside the arcsine function's domain. In addition, the restriction of the arcsine function's output to the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$ limits the estimated slip values to $|\hat{\lambda}_x| \leq \lambda_{x,M}$ and $|\hat{\alpha}| \leq \alpha_M$. $\lambda_{x,M}$ and α_M are the slip values for which the maximum possible friction coefficients $\mu_{x,M}$ and $\mu_{y,M}$ are reached. This leads to gross estimation errors in case the actual slip value is outside this interval (i. e. $|\lambda_x| > \lambda_{x,M}$ or $|\alpha| > \alpha_M$). Due to the vehicle's ABS and ESC, these conditions only persist for a very short period of time, if at all. Since the engagement of ABS or ESC indicates that at least one of the tires is operating close to or beyond $\lambda_{x,M}/\alpha_M$, no odometry measurements from either axle are utilized in the integration filter if one of the systems takes action. To overcome the aforementioned restrictions for epochs in which neither system is engaged, the following distinction of cases is made:

1. If $\sqrt{\mu_x^2 + \mu_y^2} > 1.4$ at one of the single-track model's tires, measurements from the corresponding axle are not forwarded to the integration filter in that epoch. This limit doubles the application range compared to the linear tire model.

2. If $\sqrt{\mu_x^2 + \mu_y^2} \leq 1.4$, but $|\mu| > \mu_M - 0.02$ in longitudinal and/or lateral direction, the respective friction coefficient is set to $\pm(\mu_M - 0.02)$. This ensures that the input for the arcsine function stays within the interval $(-1, 1)$. The margin of 0.02 limits the magnitude of the corresponding partial derivative $\frac{\partial \lambda_x}{\partial \mu_x}$ or $\frac{\partial \alpha}{\partial \mu_y}$, both of which approach $\pm\infty$ for $\mu \to \pm\mu_M$.

3. Otherwise, no additional steps are necessary in order to apply (4.25) and (4.26).

4.2.3 Side Slip Estimation Based on Gyroscope Measurements

Computation of the lateral specific forces requires the lateral component of $\mathbf{f}_{ec}^b$. Due to the vibration induced by the vehicle's engine and by road unevenness, $f_{ec,L}^b$ as derived from the IMU's acceleration measurements according to (4.11) is very noisy. A way to avoid this noise is to derive $f_{ec,L}^b$ from the IMU's vertical rotation rate measurement $\omega_{ib,U}^b$ instead. Under the assumption that Earth rotation rate $\boldsymbol{\omega}_{ie}^b$ and transport rate $\boldsymbol{\omega}_{en}^b$ are small in comparison to $\boldsymbol{\omega}^b_{ib}$ and that roll and pitch angle can be neglected, $\omega^b_{ib,U}$ equals the yaw rate $\dot{\psi}_{nb}$. This enables computation of $f^b_{ec,L}$ based on $\omega^b_{ib,U}$ and the CG's horizontal velocity [*Harrer and Pfeffer*, 2017, p. 99]:

$$\begin{aligned} f^b_{ec,L} = {} & \dot{\psi}_{nb}\sqrt{{v^b_{ec,F}}^2 + {v^b_{ec,L}}^2} \\ \approx {} & \omega^b_{ib,U}\sqrt{{v^b_{ec,F}}^2 + {v^b_{ec,L}}^2}, \end{aligned} \tag{4.27}$$

which includes the additional assumption of steady-state cornering ($\ddot{\psi}_{nb} = \dot{\beta} = 0$). To get the CG's horizontal velocity, the longitudinal velocity at the single-track model's rear wheel is obtained as the mean of the longitudinal velocity from both rear wheels:

$$v^b_{er,F} = \frac{\hat{\kappa}_{x,r}}{2}\left(\omega_{w,rl}\hat{r}^-_{d,rl} + \omega_{w,rr}\hat{r}^-_{d,rr}\right). \tag{4.28}$$

$\hat{\kappa}_{x,r}$ is estimated based on $f^b_{er,F}$, which is already known. The radii of curvature for the single-track model's rear wheel R_r and the CG R_{CG} are obtained via geometric relations concerning the vehicle's instantaneous center of movement [*Harrer and Pfeffer*, 2017, p. 93]:

$$R_r = \left|\frac{l\cos(\delta_f - \alpha_f)}{\sin(\delta_f - \alpha_f + \alpha_r)}\right| \tag{4.29}$$

$$R_{CG} = \sqrt{R_r^2 + l_r^2 - |2R_r l_r \sin\alpha_r|}. \tag{4.30}$$

The absolute values are taken in both cases to assure that left and right turns are treated equally. $f^b_{ec,L}$ gets the correct sign from the yaw rate $\dot{\psi}_{nb}$ in (4.27). With R_r and R_{CG}, the lateral specific force is computed by combining (4.27) and the definition of the vehicle side slip angle (4.31):

$$\cos\beta = \frac{v^b_{ec,F}}{\sqrt{{v^b_{ec,F}}^2 + {v^b_{ec,L}}^2}} \tag{4.31}$$

$$\begin{aligned} f^b_{ec,L} &= \frac{\dot{\psi}_{nb}v^b_{ec,F}}{\cos\beta} \\ &= \frac{\dot{\psi}_{nb}v^b_{er,F}}{\cos\beta} \\ &= \frac{\dot{\psi}_{nb}v^b_{er,F}}{\cos\alpha_r}\frac{R_{CG}}{R_r}, \end{aligned} \tag{4.32}$$

making use of the fact that the longitudinal velocity is the same for all points on the line connecting the single-track model's two wheels, so $v^b_{ec,F} = v^b_{er,F}$. The connection between β, α_r, R_r and R_{CG} is once again based on geometric relations. Equations (4.29)-(4.32) contain the tires' side slip angles α_f and α_r. These are unknown prior to tire slip estimation. To overcome this, iteration is performed. Initial values for α_f and α_r are obtained based on $f^b_{ec,L}$ as derived from the IMU's acceleration measurements according to (4.11). Iteration is considered successful once the absolute change between two iteration steps is less than 0.01° for both α_f and α_r. If the iteration does not converge, odometry measurements from this epoch are discarded. This only happens in less than 0.1 % of all examined epochs. After successful iteration, $f^b_{ec,L}$ is distributed between front and rear tire based on the assumption of steady-state cornering according to (4.14). The computation of $f^b_{ec,L}$ based on $\omega^b_{ib,U}$ instead of $f^b_{ib,L}$ increases the lateral velocity's accuracy drastically: The RMS of the measurement error $\delta\tilde{v}^b_{ew,L}$ drops from $18\,\mathrm{cm\,s^{-1}}$ to $4\,\mathrm{cm\,s^{-1}}$ for each of the front wheels and from $9\,\mathrm{cm\,s^{-1}}$ to $2\,\mathrm{cm\,s^{-1}}$ for each of the rear wheels.

However, when the lateral velocities derived from $\omega^b_{ib,U}$ are input into the integration filter instead of the ones derived from $f^b_{ib,L}$, the overall estimation accuracy decreases. This only occurs during longer periods without GNSS measurement updates, e. g. in tunnels. Due to a faster yaw angle error growth of the variant that uses $\omega^b_{ib,U}$, the lateral position error in tunnels grows more quickly. When GNSS measurements are available, the two variants perform almost identical since the localization algorithm has an additional source of information available to stabilize the yaw estimation. A possible explanation for the degraded performance of the variant with side slip angle estimation based on $\omega^b_{ib,U}$ is the fact that while the IMU's measurement of $f^b_{ib,L}$ is noisy, its error is close to zero-mean and its time correlation is small. These properties allow the EKF to filter out the noise. When side slip estimation is performed based on $\omega^b_{ib,U}$, the side slip angle at the front axle exhibits a bias (see Figure 4.9). Although this bias is small, its existence violates the Kalman filter assumption of zero-mean measurement noise, leading to suboptimal estimation accuracy. Another important aspect that might cause degraded integration filter performance is that both the a-priori yaw estimate and the odometry observations rely on the same input value when side slip estimation is performed based on $\omega^b_{ib,U}$. This leads to a correlation between the EKF's state vector and the odometry measurement vector, making it more difficult to correct the yaw angle, which is derived from IMU measurements, with odometry observations. Consequently, the results presented in Chapter 7 rely on the lateral velocities derived from accelerometer measurements. The potential of the method outlined in this section is demonstrated in Figures 4.9 and 4.12, which depict the accuracy of side slip angle and lateral velocity estimation on the basis of $\omega^b_{ib,U}$.

4.3 Measurement Noise

The integration filter inputs the odometry observables together with their measurement noise covariance matrix $\mathbf{R}_o$. In contrast to the matrices $\mathbf{R}_{\rho^c}$ and $\mathbf{R}_{\dot{\rho}^c}$ supplied by GNSS preprocessing, $\mathbf{R}_o$ is not a diagonal matrix, because the correlation between the odometry observations cannot be neglected. One reason for this is the fact that the eight entries of the odometry measurement vector (formed by one two-dimensional velocity vector for each wheel) stem from just five odometry sensors: four for wheel rotation rate and one for the steering wheel angle. Another reason for correlation between the wheels is the utilization of the single-track model: The estimated tire slip is identical for both wheels of each axle, which is not true in general. Since multiple different quantities (e. g. the side slip angles of the front left and the front right tire) are estimated as a single value (e. g. the side slip angle of the single-track model's front tire), the resulting estimation errors of these quantities are correlated. To account for this correlation, the computation of $\mathbf{R}_o$ is performed in two steps, which are detailed in Sections 4.3.1 and 4.3.2, respectively: First, the measurement noise covariance matrix of each individual wheel is calculated. In the second step, these individual matrices are assembled in form of a block-diagonal matrix and the correlation between the wheels is accounted for by including non-zero off-diagonal elements.

4.3.1 Noise Determination for Individual Wheels

The covariance matrix $\mathbf{\Sigma}_{\mathbf{v}^b_{ew}}$ of each individual wheel is obtained via variance propagation from the inputs for the calculation of $\mathbf{v}^b_{ew}$:

$$\mathbf{v}^b_{ew} = \mathbf{C}^{w\top}_b \begin{pmatrix} 1 \\ -\tan\hat{\alpha} \\ 0 \end{pmatrix} \omega_w \hat{r}^-_d \hat{\kappa}_x. \tag{4.33}$$

$\mathbf{C}_b^w$ is assumed to be error-free since δ_{fl} and δ_{fr} are known with high accuracy. The variance of $\hat{r}_d^-$ does not need to be included in $\mathbf{R}_o$ because r_d is contained in the integration filter's state vector. Its variance will therefore be accounted for via the filter's measurement model described in Section 6.3. The three remaining inputs $\hat{\alpha}$, $\hat{\kappa}_x$ and ω_w are assumed to be uncorrelated. Any neglected correlations between these inputs that influence the off-diagonal elements of $\mathbf{R}_o$ are accounted for later on when the correlation between the wheels is included (see Section 4.3.2). The variances of $\hat{\alpha}$ and $\hat{\lambda}_x$ are computed via variance propagation of (4.21)-(4.22) for the linear tire model and via variance propagation of (4.25)-(4.26) for the Magic Formula tire model. This requires the covariance matrices of the tire parameters ($c_{\lambda/\alpha}$ for the linear model; $B_{\lambda/\alpha}$, $C_{\lambda/\alpha}$ and $D_{\lambda/\alpha}$ for the Magic Formula model), which are known from the parameter adjustment process in Section 4.2. It also requires the variance of the friction coefficients from (4.19)-(4.20), which is assessed based on the IMU's measurement characteristics. To perform variance propagation, the partial derivatives of $\hat{\alpha}$ and $\hat{\lambda}_x$ w. r. t. the input values (tire parameters and friction coefficients) are necessary. Since the curvature factor $E_{\lambda/\alpha}$ is set to zero, these partial derivatives can be obtained analytically even for the Magic Formula model. After the variance of $\hat{\lambda}_x$ has been obtained, the variance of $\hat{\kappa}_x$ is computed via variance propagation of (4.2). $\sigma^2_{\omega_w}$ as the last required input for the covariance propagation is parametrized by comparing a reference value for ω_w with the actual measurement.

This parametrization is performed by driving in a straight line with constant velocity on a horizontal road with speeds ranging from $30\,\mathrm{km\,h^{-1}}$ to $90\,\mathrm{km\,h^{-1}}$. The road is part of Griesheim airfield, a former airfield used by the TU Darmstadt for research concerning automotive engineering as well as other tasks. The road's surface consists of tarmac. Because the dynamic tire radii r_d are known with a precision of $\sigma_{r_d} \leq 0.1\,\mathrm{mm}$ from the data recorded for tire model parametrization in Section 4.2 and the reference equipment estimates the vehicle's speed with a standard deviation of less than $2\,\mathrm{mm\,s^{-1}}$, the reference value for ω_w has a standard deviation in the range of $0.01\,\mathrm{rad\,s^{-1}}$ to $0.025\,\mathrm{rad\,s^{-1}}$ for this speed interval. As depicted in Figure 4.4, the empirical standard deviation of the measurement error $\delta\tilde{\omega}_w$ increases with $|\omega_w|$. Based on these results, σ_{ω_w} is fitted as a linear function of $|\omega_w|$. To prevent unrealistically low values for the measurement noise covariance at low speeds, a minimum value of $0.15\,\mathrm{rad\,s^{-1}}$ is introduced for σ_{ω_w}.

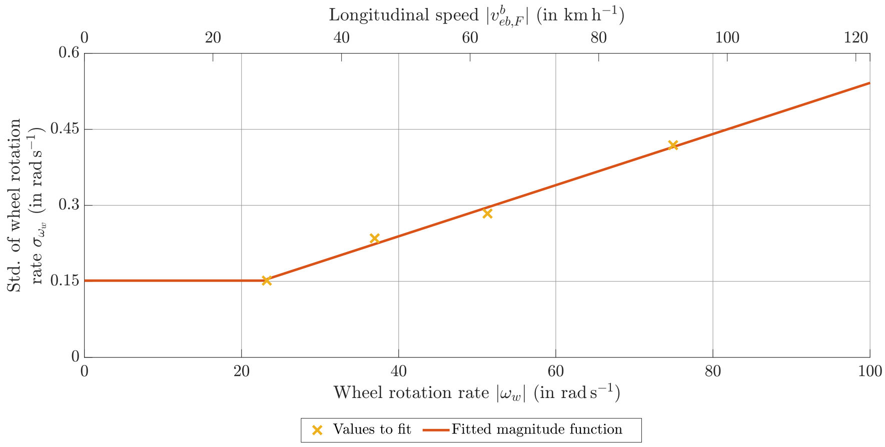

Figure 4.4: Fitted standard deviation of wheel rotation rate. The additional horizontal axis at the top is based on the arithmetic mean of the four dynamic tire radii r_d and is included for visualization purposes only.

The accuracy of tire slip estimation is visualized in Figures 4.5-4.9. Identical to Figures 4.2 and 4.3, the reference values for the longitudinal slip λ_x are computed from the reference equipment's estimate of $\mathbf{v}_{eb}^b$, the wheel rotation rates from the CAN bus and the reference values for r_d. The reference values for the side slip angle α are obtained based on the reference equipment's estimates of $\mathbf{v}_{eb}^b$ and $\boldsymbol{\omega}_{eb}^b$ as well as the steering angle δ_f, which in turn is determined from the steering wheel angle transmitted via CAN bus with the help of (4.9). In contrast to Figures 4.2 and 4.3, the friction coefficients μ_x and μ_y for all the data in Figures 4.5-4.9 are computed based on the MEMS IMU's measurements of $\mathbf{f}_{ib}^b$ and $\boldsymbol{\omega}_{ib}^b$, since these measurements are available to the localization algorithm during normal operation, while the reference IMU's observations are not. Consequently, the reference values in Figures 4.5-4.9 appear to be very noisy, because their slip value on the vertical axis has the reference equipment's accuracy, while their value for the friction coefficient on the horizontal axis is considerably less accurate. This behavior is desired, since the aim of the procedure is to parametrize the slip estimation errors occurring during normal operation. In all five Figures 4.5-4.9, the reference slip value computed in this manner is displayed in the same plot as the value calculated with the respective tire model. To show the expected estimation precision, the tire model's estimate of the slip's standard deviation is included in form of the 1σ confidence interval. A more detailed analysis reveals that the major part of the tire slip estimation error stems from an incorrectly estimated friction coefficient, which in turn is a result of the MEMS IMU's measurement errors due to vehicle vibration and sensor noise. Only a minor part of the slip estimation error stems from incorrect tire parameters.

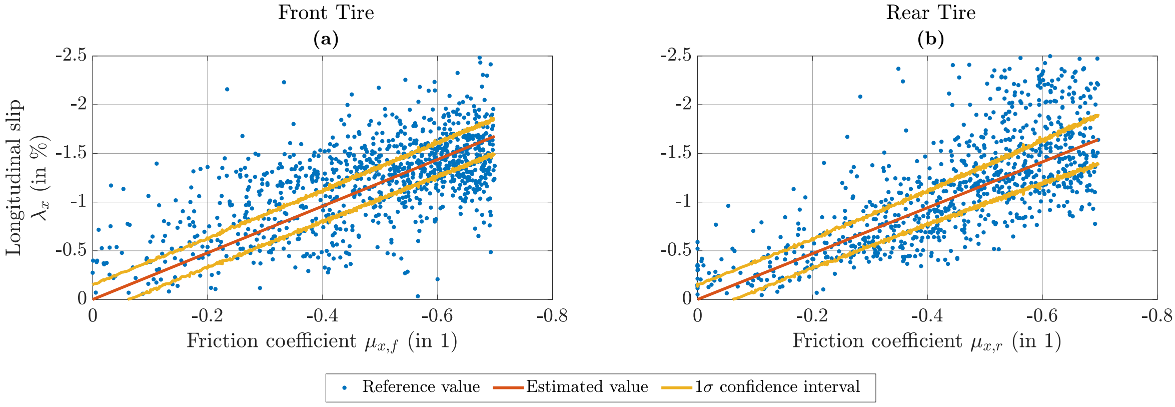

Figure 4.5: Longitudinal slip estimation accuracy with linear tire model. **(a)** Front tire. **(b)** Rear tire.

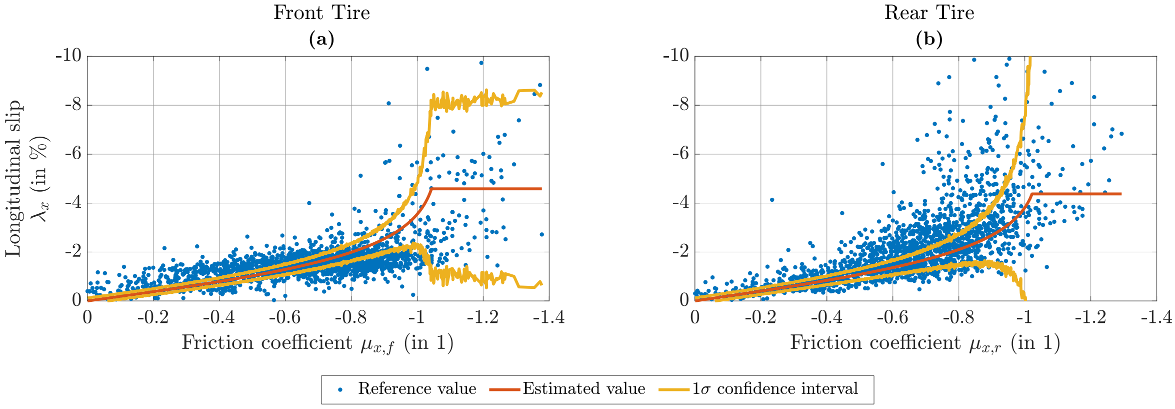

Figure 4.6: Longitudinal slip estimation accuracy with Magic Formula tire model. **(a)** Front tire. **(b)** Rear tire.

Figure 4.5 displays the accuracy of longitudinal slip estimation with the linear tire model, Figure 4.6 with the Magic Formula tire model. The data basis for the two figures is the same as for the adjustment of the longitudinal tire slip parameters, because these data cover the whole range of longitudinal tire slip up to ABS engagement. The shown range is selected based on each model's validity range. Therefore, no values for $|\mu_x| > 0.7$ exist in Figure 4.5. For both tire models, the slip's estimated standard deviation increases with $|\mu_x|$. However, this increase is hardly noticeable in Figure 4.5 due to the linear tire model's limited application range. In Figure 4.6, the bounded domain of the arcsine function in (4.25) becomes visible: $|\hat{\lambda}_{x,f}|$ and $|\hat{\lambda}_{x,r}|$ are restricted to a maximum of 4.6 % and 4.4 %, respectively. They do not increase any more for $|\mu_x| > \mu_{x,M} - 0.02$.

Examination of the stochastic model reveals that the reference value falls within the 1σ confidence interval in 35 % of the epochs for the front tire with the linear model. With the Magic Formula model, this value rises to 42 %. While both of these values are too low, they are based on data consisting solely of strong braking maneuvers. For more moderate longitudinal slip values, they are in fact higher. Compared to the linear tire model, the Magic Formula broadens the application range while simultaneously increasing the correctness of precision estimation. This effect also occurs at the rear tire: The reference value falls within the 1σ confidence interval estimated by the linear tire model in 40 % of the epochs, while this percentage rises to 49 % with the Magic Formula model.

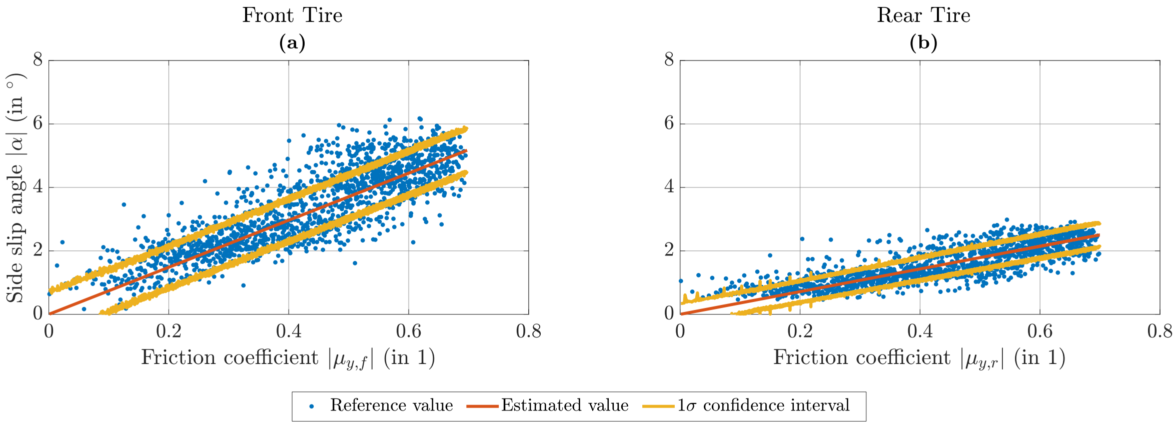

Figure 4.7: Side slip angle estimation accuracy with linear tire model, based on accelerometer measurements. **(a)** Front tire. **(b)** Rear tire.

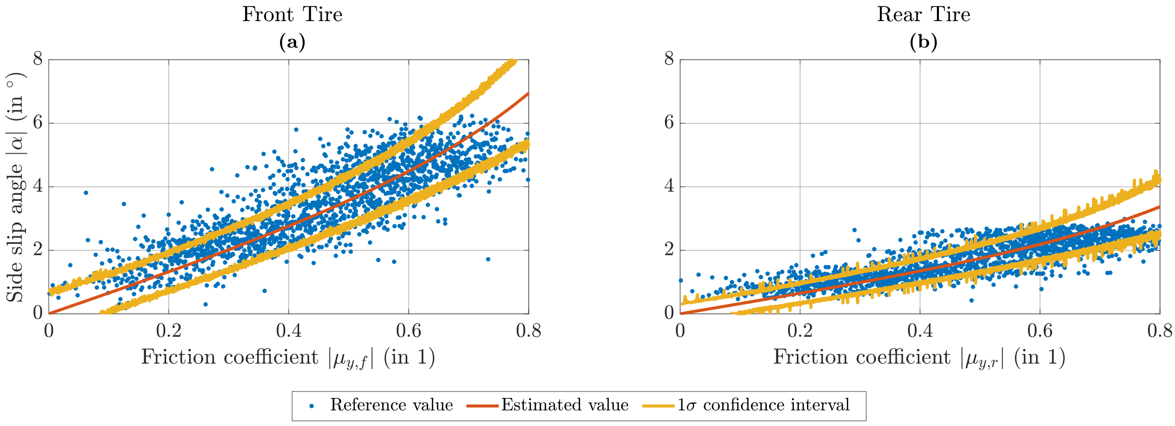

Figure 4.8: Side slip angle estimation accuracy with Magic Formula tire model, based on accelerometer measurements. **(a)** Front tire. **(b)** Rear tire.

Figure 4.7 displays the accuracy of side slip angle estimation with the linear tire model, Figure 4.8 with the Magic Formula tire model. μ_y is obtained from the MEMS IMU's accelerometer measurements in both cases. The data basis for the two figures is the same as for the adjustment of the side slip tire parameters, because these data cover a wide range of lateral acceleration up to $\pm 6.3\,\mathrm{m\,s^{-2}}$. In contrast to the plots for longitudinal slip, the scale of the axes is identical for both tire models, since the measurement data contain only very few points outside the linear model's validity range of $|\mu| \leq 0.7$. For both tire models, the slip's estimated standard deviation increases with $|\mu_y|$. In case of the linear model, $\sigma_\alpha \approx \frac{\sigma_{\mu_y}}{c_\alpha}$ because σ_{c_α} is very low. While σ_{μ_y} increases with $|\mu_y|$, the increase in σ_α is hardly visible in Figure 4.7.

The reference value falls within the 1σ confidence interval in 67 % of the epochs for the front tire and 68 % of the epochs for the rear tire with the linear model. With the Magic Formula model, these values rise to 69 % at the front and 70 % at the rear. Just as for longitudinal slip estimation, the Magic Formula broadens the application range while simultaneously increasing the correctness of precision estimation compared to the linear model. The improvements achieved by the Magic Formula model are smaller than they are for longitudinal slip because the measurement data do not contain many values outside the tires' linear range.

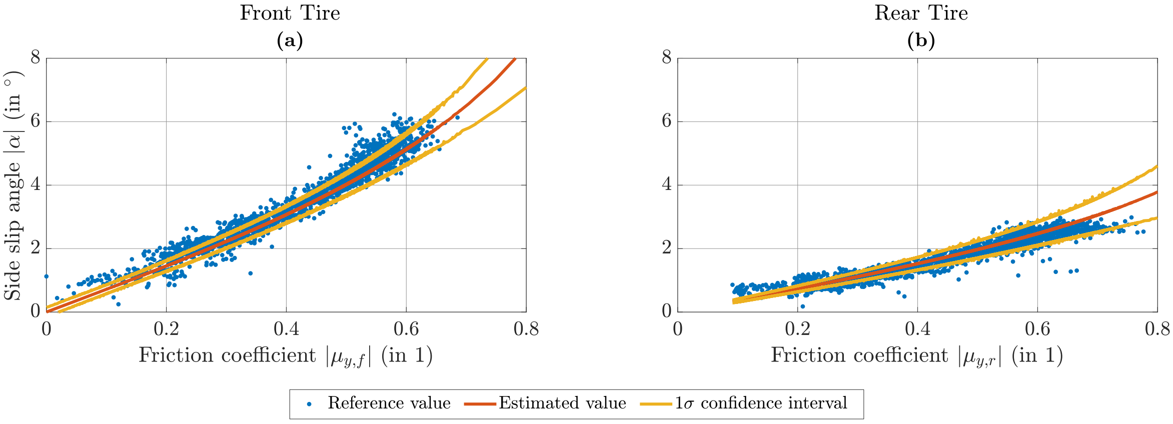

Figure 4.9: Side slip angle estimation accuracy with Magic Formula tire model, based on gyroscope measurements. **(a)** Front tire. **(b)** Rear tire.

The accuracy of side slip angle estimation is improved when μ_y is obtained from gyroscope measurements instead of accelerometer measurements. This is depicted in Figure 4.9. It is based on the same data set and the same tire model as Figure 4.8. In contrast to Figure 4.8, the tire model's parameters are fitted based on the reference IMU's measurements of the vertical rotation rate $\omega^b_{ib,U}$ instead of the lateral specific force $f^b_{ec,L}$ derived from accelerometer measurements according to (4.11). The equivalent plot for the linear tire model appears in the appendix (Section B.3). The RMS of the side slip angle estimation error decreases from 0.87° to 0.35° at the front tire and from 0.62° to 0.23° at the rear tire when gyroscope data are used instead of accelerometer data. However, Figure 4.9a indicates a bias in the estimation of α_f: The reference value is higher than the estimated value in 74 % of epochs. The arithmetic mean of the estimation error is −0.17° while the error's empirical standard deviation is 0.30°. Because the data set consists of driving in a circle with constant velocity, the assumption of steady-state cornering is valid and cannot be the reason for this bias. A possible explanation is an unmodeled scale factor error in the MEMS IMU whose measurements are used to calculate $\hat{\alpha}$, resulting in too low values for $|\dot{\psi}_{nb}|$ and subsequently $|f^b_{ec,L}|$ as well as $|\mu_y|$.

4.3.2 Correlation in Between the Wheels

With the estimated values for longitudinal and side slip, the 3-D observation vector $\tilde{\mathbf{z}}_o = \mathbf{v}_{ew}^b$ for each wheel is computed, resulting in an overall observation vector with twelve entries. The corresponding 12×12 covariance matrix has block-diagonal form, consisting of four 3×3 submatrices. Before these data are forwarded to the integration filter, the correlation between the wheels is included to ensure proper processing of the observations in the EKF. These correlations are too large to be neglected because slip estimation is performed with a single-track model, yielding the same slip estimate for both wheels of each axle. The correlation coefficients are parametrized by forming the difference between the 12×1 observation vector and a reference solution, obtained from the integration of a geodetic GNSS receiver for RTK positioning and a navigation grade IMU. This measurement error allows for computation of $\boldsymbol{\Xi}$, the matrix of correlation coefficients. $\boldsymbol{\Xi}$ concerns the correlation between the measurement errors of the different velocity components at the four wheels within the same epoch, any potential temporal correlations are neglected. Since these empirical quantities depend on the evaluated measurement data, they are averaged over a number of different scenarios to obtain representative results. The correlations for the vertical components $\delta\tilde{v}_{ew,U}^b$ are not important as these entries are not taken into account by the EKF anyway. The empirical correlation values between longitudinal measurement error on the one hand and lateral measurement error on the other hand are close to zero and therefore neglected. The values for longitudinal and lateral velocity measurement error are:

$$\boldsymbol{\Xi}_{\delta\tilde{\mathbf{v}}_{ew,F}^b} \approx \begin{pmatrix} 1 & 0.5 & 0.4 & 0.2 \\ 0.5 & 1 & 0.2 & 0.4 \\ 0.4 & 0.2 & 1 & 0.5 \\ 0.2 & 0.4 & 0.5 & 1 \end{pmatrix} \qquad \boldsymbol{\Xi}_{\delta\tilde{\mathbf{v}}_{ew,L}^b} \approx \begin{pmatrix} 1 & 0.98 & 0.3 & 0.3 \\ 0.98 & 1 & 0.3 & 0.3 \\ 0.3 & 0.3 & 1 & 0.99 \\ 0.3 & 0.3 & 0.99 & 1 \end{pmatrix}, \tag{4.34}$$

where $\boldsymbol{\Xi}_{\delta\tilde{\mathbf{v}}_{ew,F}^b}$ contains the correlation coefficients for the errors of the four longitudinal wheel velocities and $\boldsymbol{\Xi}_{\delta\tilde{\mathbf{v}}_{ew,L}^b}$ contains the correlation coefficients for the errors of the four lateral wheel velocities. Both are in the order front left, front right, rear left, rear right. With the exception of the entries for the lateral correlation in between two wheels of the same axle (0.98 and 0.99), all values in (4.34) are rounded to one significant digit. The high correlation in lateral direction along each axle indicates that the two entries contain almost identical information. It stems from the fact that lateral velocity estimation uses the same side slip angle for both wheels of each axle. Inclusion of such high correlation coefficients into the measurement covariance matrix $\mathbf{R}_o$ causes this matrix to be close to singular, because some rows now contain almost the same information. To avoid this, the arithmetic mean of the two lateral velocity estimates of each axle is formed and only two lateral velocities, $v_{ef,L}^b$ and $v_{er,L}^b$, are processed in the integration filter. After removal of the four vertical velocity components, the odometry observation vector $\tilde{\mathbf{z}}_{o,6D}$ contains six entries:

$$\tilde{\mathbf{z}}_{o,6D} = \begin{pmatrix} v_{efl,F}^b & v_{efr,F}^b & v_{erl,F}^b & v_{err,F}^b & v_{ef,L}^b & v_{er,L}^b \end{pmatrix}^{\mathsf{T}}. \tag{4.35}$$

The correlation coefficients of $\tilde{\mathbf{z}}_{o,6D}$ are obtained by combining $\boldsymbol{\Xi}_{\delta\tilde{\mathbf{v}}_{ew,F}^b}$ with the correlation coefficient in between the lateral velocity errors at both axles:

$$\boldsymbol{\Xi}_{\delta\tilde{\mathbf{z}}_{o,6D}} \approx \begin{pmatrix} 1 & 0.5 & 0.4 & 0.2 & 0 & 0 \\ 0.5 & 1 & 0.2 & 0.4 & 0 & 0 \\ 0.4 & 0.2 & 1 & 0.5 & 0 & 0 \\ 0.2 & 0.4 & 0.5 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0.3 \\ 0 & 0 & 0 & 0 & 0.3 & 1 \end{pmatrix}. \tag{4.36}$$

The accuracy of velocity estimation is depicted in Figures 4.10-4.12. Each of the figures shows the velocity error and the 1σ confidence interval estimated by odometry preprocessing as a function of the wheel rotation rate. The data set spans a duration of approx. 2 h and contains various acceleration profiles, including the maneuvers used for parametrization of the tire models.

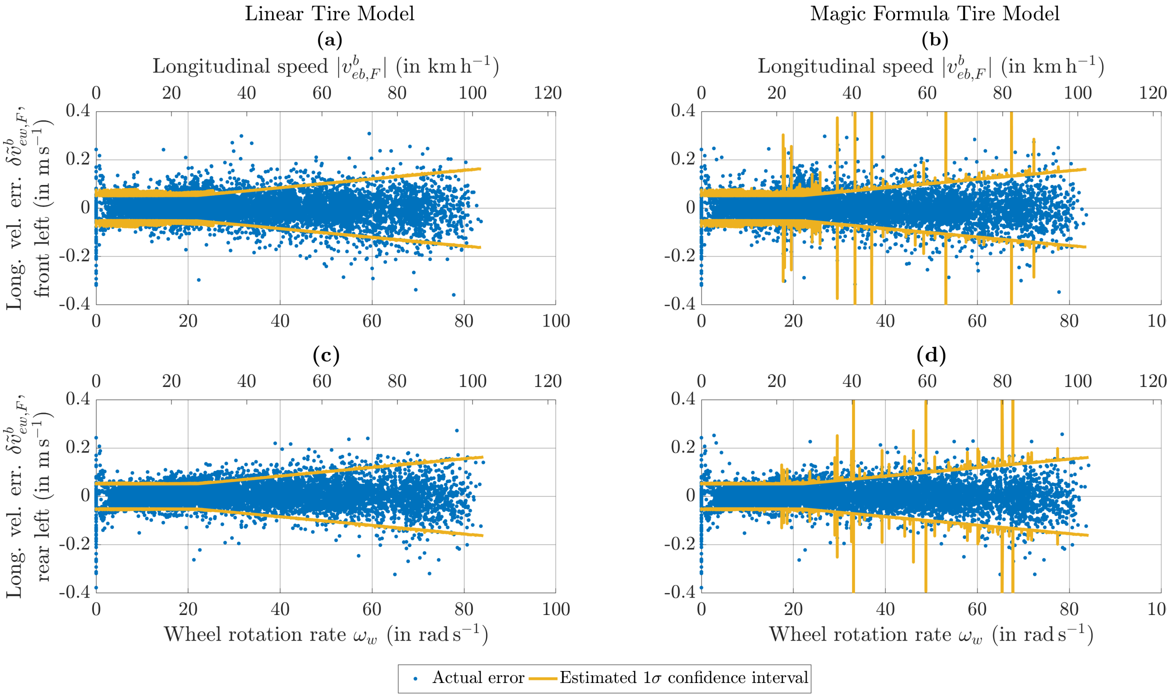

Figure 4.10: Longitudinal odometry velocity error, front and rear left wheel. The additional horizontal axis at the top is based on the arithmetic mean of the four dynamic tire radii r_d and is included for visualization purposes only. **(a)**,**(c)** Linear tire model. **(b)**,**(d)** Magic Formula tire model.

Figure 4.10 displays the longitudinal velocity error for the left wheels. The plot for the right wheels is very similar and appears in the appendix (Section B.3). The estimation error shows the same behavior for both front and rear left wheel because longitudinal slip estimation for both axles is performed on the basis of longitudinal acceleration measurements from the IMU, which are noisy due to the vehicle's vibrations. For the actual error, no difference between the linear and the Magic Formula tire model is apparent since the presented data mainly feature horizontal accelerations within the tires' linear range. The error's RMS is 4 cm s^{-1} in all four subfigures. The 1σ confidence interval $[-\sigma_{v^w_{ew,F}}, \sigma_{v^w_{ew,F}}]$ increases with ω_w since σ_{ω_w} does so, too. For values of $|\mu_x|$ up to 0.5, σ_{ω_w} is the major influence on $\sigma_{v^w_{ew,F}}$. Under these conditions, the estimated standard deviation ranges from 5 cm s^{-1} at standstill to 20 cm s^{-1} for $\omega_w = 100$ rad s^{-1} (approx. 120 km h^{-1}).

The difference between the tire models is exemplified by the peaks in the estimated confidence interval that appear in Figures 4.10b, d, but do not appear in Figures 4.10a, c. They stem from epochs with large absolute values of μ_x. These are outside the linear tire model's application range, therefore they are discarded in Figures 4.10a, c. In contrast, the Magic Formula model is able to perform slip estimation in these epochs. Since large values of $|\mu_x|$ lead to increased uncertainty in $\hat{\lambda}_x$ and therefore $\hat{\kappa}_x$, the estimated confidence interval of the velocity measurement is comparatively high. The stochastic model is set up to deliver a rather pessimistic estimate of the velocity precision. This leads to the fact that 93 % to 96 % of the errors fall within the 1σ confidence interval in each of the four subfigures of 4.10. The reason for this is that due to the odometry's high data rate of 100 Hz, the measurement noise is correlated over many measurement updates, violating the EKF's assumption of white noise. Consequently, the integration filter cannot smooth out odometry measurement errors as quickly as it expects. The increased estimated measurement noise models this behavior and prevents the EKF from obtaining unrealistically low values for the error covariance matrix $\mathbf{P}$.

Figure 4.11 depicts the lateral velocity error at the front and rear axle when side slip estimation is performed based on accelerometer measurements. The error increases with ω_w due to two reasons: On the one hand, the absolute error in ω_w grows with the vehicle speed. On the other hand, the vehicle's vibration due to road unevenness increases, too. The latter effect is the dominant one in the lateral velocity error growth. This is apparent in Figure 4.12, which presents the results of the same data set, but with side slip estimation on the basis of gyroscope measurements: While the error's RMS is $18\,\mathrm{cm\,s^{-1}}$ at the front axle and $9\,\mathrm{cm\,s^{-1}}$ at the rear axle in Figure 4.11, it is only $4\,\mathrm{cm\,s^{-1}}$ at the front axle and $2\,\mathrm{cm\,s^{-1}}$ at the rear axle in Figure 4.12. These values are independent of the employed tire model since the measurement data mainly contain lateral acceleration values inside the tires' linear range.

When performing variance propagation with the inputs $\sigma^2_{\hat{\alpha}}$, $\sigma^2_{\hat{\kappa}_x}$ and $\sigma^2_{\omega_w}$ in order to obtain the variance estimate for the lateral velocity of each wheel, all partial derivatives become zero for $\hat{\alpha} = \hat{\lambda}_x = \omega_w = 0$, leading to a variance estimate of zero for $v^w_{ew,L}$. This causes numerical problems within the integration filter because $\mathbf{R}_o$ becomes singular. To avoid these issues, a constant term is added to the propagated variance. Since the lateral velocity at the rear axle is smaller than the one at the front axle for small vehicle speeds, this constant term is set to a higher value at the front than at the rear ($25\,\mathrm{cm\,s^{-1}}$ at front and $10\,\mathrm{cm\,s^{-1}}$ at rear when side slip is estimated based on accelerometer measurements, $10\,\mathrm{cm\,s^{-1}}$ at front and $5\,\mathrm{cm\,s^{-1}}$ at rear when side slip is estimated based on gyroscope measurements). This is visible in Figures 4.11 and 4.12. The estimated 1σ confidence interval increases with ω_w, reaching values of $50\,\mathrm{cm\,s^{-1}}$ at the front and $25\,\mathrm{cm\,s^{-1}}$ at the rear axle for $\omega_w = 100\,\mathrm{rad\,s^{-1}}$ (approx. $120\,\mathrm{km\,h^{-1}}$) when side slip is estimated based on accelerometer measurements. The Magic Formula tire model exhibits peaks in the propagated variance for large values of $|\mu_y|$ in the same way it does for large values of $|\mu_x|$ for the longitudinal velocity. These values are outside of the linear tire model's application range, hence they do not appear in the corresponding subfigures. Independent of tire model and axle, the lateral velocity error is inside the estimated confidence interval in 93 % of the epochs when side slip estimation is performed based on accelerometer measurements and in 98 % of the epochs when gyroscope measurements are used.

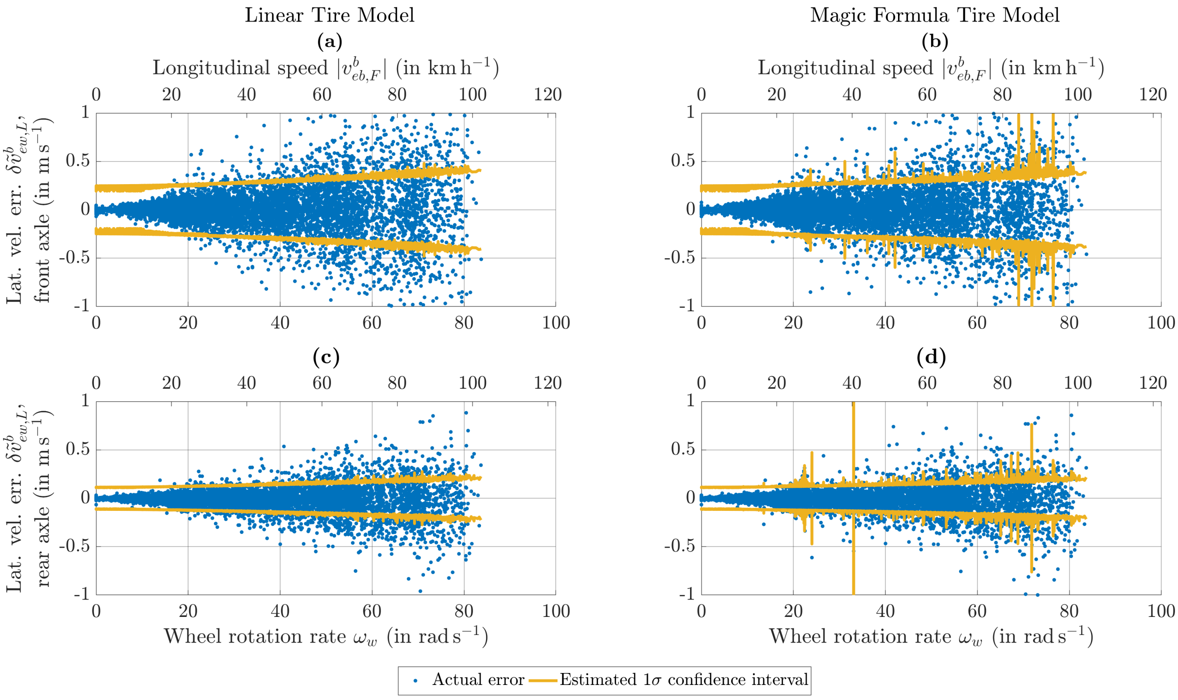

Figure 4.11: Lateral odometry velocity error when side slip is estimated based on accelerometer measurements, front and rear axle. The additional horizontal axis at the top is based on the arithmetic mean of the four dynamic tire radii r_d and is included for visualization purposes only. **(a)**,**(c)** Linear tire model. **(b)**,**(d)** Magic Formula tire model.

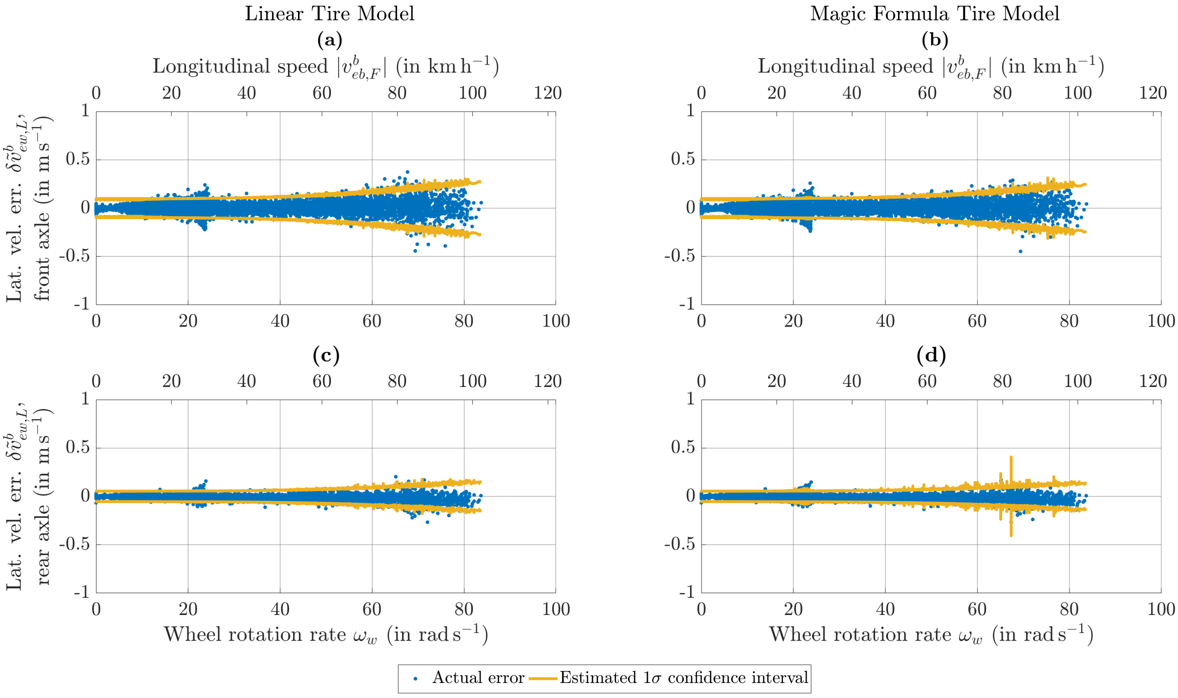

Figure 4.12: Lateral odometry velocity error when side slip is estimated based on gyroscope measurements, front and rear axle. The additional horizontal axis at the top is based on the arithmetic mean of the four dynamic tire radii r_d and is included for visualization purposes only. **(a)**,**(c)** Linear tire model. **(b)**,**(d)** Magic Formula tire model.

Chapter 5

Quality Assessment

This chapter deals with different techniques and metrics which are useful for assessing the quality of data within a localization algorithm. Section 5.1 lists multiple categories of performance metrics for localization algorithms in general and specifies various metrics within these categories. The capability to perform quality assessment internally, i. e. by applying techniques that are possible in real time and without employing external reference data, is subject of Section 5.2. The main goal of this internal quality assessment is to obtain a realistic estimate of the current estimation error. Section 5.3 completes the chapter by defining criteria to choose a subset of metrics for assessing the quality of the particular localization algorithm designed within this thesis and detailing the metrics that match these criteria. Because evaluation of said metrics requires accurate external reference data, this process is called external quality assessment.

5.1 Performance Metrics for Localization Algorithms

This section introduces four categories of performance metrics for localization algorithms: accuracy, integrity, availability and continuity. These four categories are widely employed in order to get a complete picture of an algorithm's overall performance, e. g. in *Peyret* [2013], *Pullen* [2008] and *COST Action TU1302* [2017]. The subsequent definitions of metrics from the four categories are mainly based on these three references.

The dynamic tire radius r_d plays a special role among the quantities estimated within the localization algorithm, because it undergoes a short transient phase in the beginning of each test scenario and then remains largely constant throughout the remainder of the scenario. Therefore, the conventional metrics from the four mentioned categories are not suited to assess the algorithm's estimation performance as far as r_d is concerned. Section 5.1.5 is thus dedicated to performance metrics from control engineering which are able to accomplish that task.

5.1.1 Accuracy

The accuracy of any estimate output by a localization algorithm is assessed based on the estimation error $\delta\hat{\mathbf{y}}$, which is the difference between an estimated quantity $\hat{\mathbf{y}}$ and its true counterpart $\check{\mathbf{y}}$:

$$\delta\hat{\mathbf{y}} = \hat{\mathbf{y}} - \check{\mathbf{y}}. \tag{5.1}$$

The vector $\mathbf{y}$ acts as a placeholder in this formulation and may refer to any scalar or vector quantity that appears within a localization algorithm. Because the true value $\check{\mathbf{y}}$ is usually unknown, it is replaced by a reference value obtained from sources of information with higher accuracy than $\hat{\mathbf{y}}$. One way to characterize the estimation error is its CDF, which is most useful for scalar estimation errors, since visualization and interpretation of the CDF is complicated for non-scalar quantities.

A variety of metrics focuses on the estimation error and its distribution. The most common ones include:

- The error's arithmetic mean, which judges the unbiasedness of the estimation process. It is not as robust against outliers as the median, which is why the difference between arithmetic mean and median is an indicator for the presence of outliers or an otherwise skewed distribution.
- The empirical covariance matrix $\mathbf{\Sigma}_{\delta\hat{\mathbf{y}}}$ of the estimation error, which describes the error's variation around its arithmetic mean. The error's empirical standard deviation $\sigma_{\delta\hat{\mathbf{y}}}$ is formed by taking the square root of the elements on the main diagonal of $\mathbf{\Sigma}_{\delta\hat{\mathbf{y}}}$. Both $\mathbf{\Sigma}_{\delta\hat{\mathbf{y}}}$ and $\sigma_{\delta\hat{\mathbf{y}}}$ quantify estimation precision, not estimation accuracy.
- Accuracy assessment is accomplished by the error's RMS, which quantifies the variation of $\hat{\mathbf{y}}$ around the reference value.
- In order to extract individual points from a scalar error's CDF, error quantiles are employed. These are typically based on the error's absolute value and define the value $|\delta\hat{y}|_{p_q}$ which $|\delta\hat{y}|$ does not exceed in a percentage p_q of epochs:

  $$\mathrm{CDF}\left(|\delta\hat{y}|_{p_q}\right) = p_q. \tag{5.2}$$

  Typical values for p_q are 50 %, 95 % and 99 %. In the context of localization algorithms for automotive positioning, 3-D vectors of quantities such as position and velocity are often transformed into scalars by splitting them up into their horizontal and their vertical part.

5.1.2 Integrity

Integrity characterizes the amount of trust that can be placed in a quantity within a localization algorithm. In order to assess integrity, some estimate concerning the magnitude of the estimation error has to be computed by the localization algorithm. One way to quantify integrity is to use the three parameters alert limit, time to alert and integrity risk [*Pullen*, 2008, p. 21]:

- The alert limit defines the magnitude of error that cannot be exceeded without posing a safety risk.
- The time to alert is defined as the maximum acceptable time between exceeding the alert limit and issuing a warning about it.
- The probability of exceeding the alert limit without issuing a warning within the time to alert is called the integrity risk.

Whenever no dedicated alert limit is specified, integrity can be quantified with the concept of a protection level. The protection level is output by the localization algorithm in addition to the estimated value $\hat{\mathbf{y}}$ and provides an upper bound for the estimated error. In this case, the integrity risk is the probability that the actual error exceeds the computed protection level [*Peyret*, 2013, p. 6].

Since the localization algorithm designed in the context of this thesis does not provide any protection level, integrity assessment can only be performed based on the covariance matrices output by various parts of the algorithm. The goal of this integrity assessment is to define a metric that quantifies the agreement between the estimated covariance and the actual estimation error. Because covariance matrices specify precision, but integrity concerns accuracy, a good agreement can only be reached in case of an unbiased estimation. In order to quantify this agreement, a test quantity $\varepsilon_{\delta\hat{\mathbf{y}}}$ is computed from $\delta\hat{\mathbf{y}}$ and its estimated covariance $\hat{\mathbf{\Sigma}}_{\delta\hat{\mathbf{y}}}$ at each epoch k. For normally distributed errors $\delta\hat{\mathbf{y}}$, this test quantity is chi-squared distributed with degrees of freedom equal to the dimension of $\delta\hat{\mathbf{y}}$ [*Niemeier*, 2008, Section 3.2]:

$$\varepsilon_{\delta\hat{\mathbf{y}},k} = \delta\hat{\mathbf{y}}_k^{\mathsf{T}} \hat{\mathbf{\Sigma}}_{\delta\hat{\mathbf{y}},k}^{-1} \delta\hat{\mathbf{y}}_k \tag{5.3}$$

$$\varepsilon_{\delta\hat{\mathbf{y}},k} \sim \chi^2_{\dim(\delta\hat{\mathbf{y}})}. \tag{5.4}$$

Assuming no correlation between $\hat{\mathbf{y}}$ and $\check{\mathbf{y}}$, $\hat{\mathbf{\Sigma}}_{\delta\hat{\mathbf{y}}}$ is the sum of the respective estimated covariance matrices:

$$\hat{\mathbf{\Sigma}}_{\delta\hat{\mathbf{y}}} = \hat{\mathbf{\Sigma}}_{\hat{\mathbf{y}}} + \hat{\mathbf{\Sigma}}_{\check{\mathbf{y}}}. \tag{5.5}$$

The agreement between estimated and actual error is quantified by specifying a percentage p_ε and computing the corresponding value $\xi_{\delta\hat{\mathbf{y}},p_\varepsilon}$ from the CDF of a chi-squared distribution $\chi^2_{\dim(\delta\hat{\mathbf{y}})}$:

$$\mathrm{CDF}_{\chi^2_{\dim(\delta\hat{\mathbf{y}})}}\left(\xi_{\delta\hat{\mathbf{y}},p_\varepsilon}\right) = p_\varepsilon. \tag{5.6}$$

Afterwards, it is checked how often the test quantity $\varepsilon_{\delta\hat{\mathbf{y}},k}$ is smaller than $\xi_{\delta\hat{\mathbf{y}},p_\varepsilon}$ within the examined time interval. The percentage of epochs for which this check is passed is equal to the specified percentage p_ε in case all of the following statements are true:

- $\hat{\mathbf{y}}$ is an unbiased estimate of $\check{\mathbf{y}}$.
- $\delta\hat{\mathbf{y}}$ is distributed normally.
- $\hat{\mathbf{\Sigma}}_{\delta\hat{\mathbf{y}}}$ describes the actual error $\delta\hat{\mathbf{y}}$ adequately.

The described technique constitutes a very basic level of integrity assessment. While a high degree of correlation between the actual and estimated errors is a necessity for algorithms with good performance in terms of integrity, it is not sufficient for safety-critical systems. More advanced levels of integrity assessment aim to overbound the actual error, identify potential failures in the input data and alert the user about possibly unsafe operating conditions.

5.1.3 Availability

Any quantity within a localization algorithm is defined to be available at a certain epoch if this quantity is provided with the required accuracy and integrity at that epoch [*Peyret*, 2013, p. 6]. The related metric is the percentage of epochs during a test scenario for which a specific quantity is available. Unavailability occurs when the actual error is larger than the specified tolerable maximum error, when the algorithm flags its output as unusable or when the algorithm is unable to output any estimate at all. Due to the fact that no tolerable maximum error is specified for any quantity within the localization algorithm designed in this thesis and that this algorithm does not provide any usability flags, unavailability only occurs if no output is provided at all. This case does not occur during any of the test scenarios described in Chapter 7, therefore availability is not considered to be one of the important metrics to demonstrate the algorithm's performance.

5.1.4 Continuity

Continuity concerns a localization algorithm's capability to output quantities continuously over a defined time interval. These quantities have to fulfill specified accuracy and integrity requirements. Continuity is quantified in terms of the continuity risk, which states the probability that the algorithm will stop providing a quantity within the required limits during a time interval, given that the output is within the required limits at the beginning of that time interval [*Pullen*, 2008, p. 22]. Due to the nature of this definition, the continuity risk is usually assessed in post-processing. If it is required in real time, the continuity risk can be provided on the basis of a large set of previous measurements. By comparing the current situation with similar ones that have occurred within this set of previous measurements, the current continuity risk may be derived. In analogy to availability, continuity is not addressed for the algorithm developed in this thesis as there are no specified requirements for accuracy and integrity.

5.1.5 Control Engineering Metrics for Closed-Loop Systems

Performance metrics for localization algorithms generally deal with estimated quantities $\hat{\mathbf{y}}$ whose true value $\check{\mathbf{y}}$ changes significantly over time, e. g. attitude, velocity and position. This is not the case for the dynamic tire radius, whose change over the course of a test scenario due to factors like tire wear, temperature or wheel load change is very small. In a typical test scenario, the estimates of all four tire radii are initialized with the same, constant value. This value is obtained from the CAN bus and does not change from one scenario to the next. Consequently, the behavior of $\hat{\mathbf{r}}_d$ during the transient phase after initialization is very important for the localization algorithm's performance. This type of behavior is also found in the command step response of a closed-loop system in the field of control engineering.

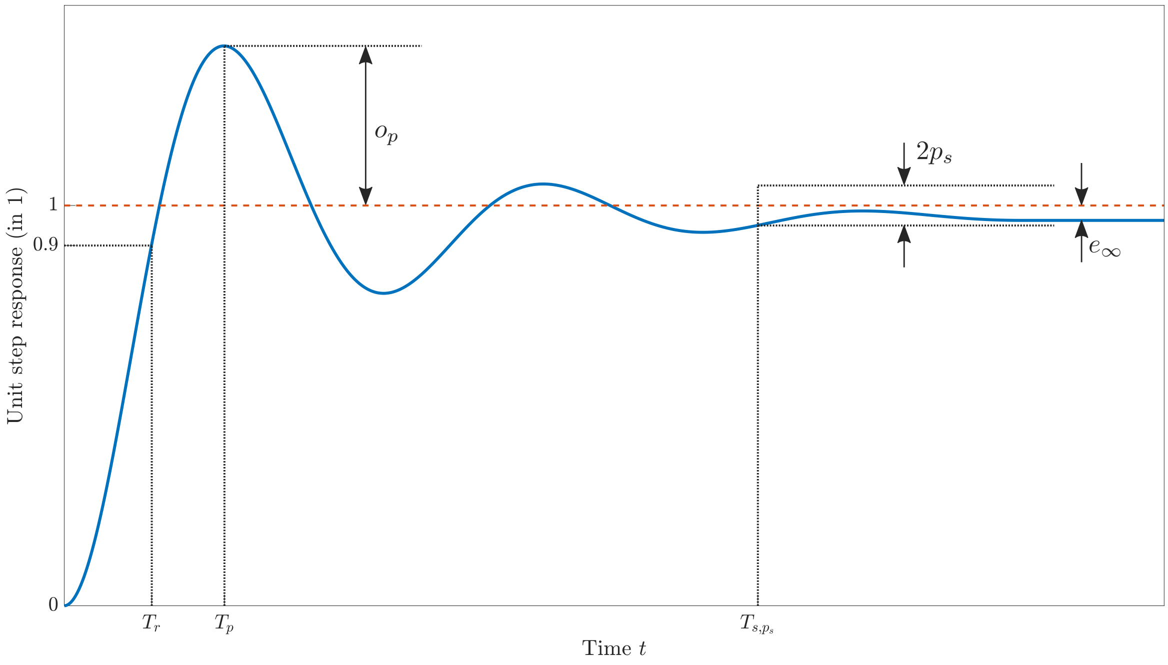

Figure 5.1: Command step response of a closed-loop system with associated performance metrics [*Lunze*, 2016, p. 355].

A typical command step response of a linear system is depicted in Figure 5.1. The figure is created in analogy to *Lunze* [2016, Section 7.1], where also the performance metrics defined in this paragraph stem from. After a unit step is input into the system at $t = 0$, the rise time T_r measures the time until the system's output reaches 0.9 for the first time. The peak overshoot o_p occurs at the peak time T_p. The settling time T_{s,p_s} is defined as the point in time after which the command step response stays inside the interval $[1 - p_s, 1 + p_s]$ indefinitely. Typical values for p_s are $2\,\%$, $5\,\%$ and $10\,\%$ [*Lunze*, 2016, p. 488]. The final performance metric depicted in Figure 5.1 is the steady-state error e_∞, which is the difference between one and the limit of the system's response for $t \to \infty$.

5.2 Internal Quality Assessment

Internal quality assessment refers to all techniques regarding quality assessment that are possible in real time and only utilize information which is present within the localization algorithm. This means in particular that only data that have been generated in previous or the current epoch can be employed, but no future data. It also means that no external reference data are available. Consequently, all consistency checks have to be performed by comparing quantities within the localization algorithm with each other. The goal of this section is to outline a method for quantifying quality of data within the integration algorithm in all processing steps, i. e. from raw sensor data to the the integration filter's output vector $\hat{\mathbf{x}}^{+}$. In this context, the term quality of data stands for accuracy whenever accuracy assessment is possible with internal data alone. Whenever that is not the case, the term quality of data stands for precision.

Quality assessment for the input and output data of the GNSS and odometry preprocessing modules has already been performed in Chapters 3 and 4. While this assessment was carried out with the help of external reference data, the obtained parametrization for the measurement noise only requires data from within the localization algorithm, e. g. satellite elevation or wheel rotation rate. This enables internal quality assessment for these in- and outputs. The IMU's data quality is specified in its data sheet. The values supplied by the manufacturer are verified via laboratory tests. The integration filter's system model and the corresponding system noise (see Section 6.2) are responsible for computing an estimate of the a-priori navigation solution's quality. Since no external information is utilized in this process, it qualifies as a form of internal quality assessment. Therefore, internal quality assessment methods are implemented for all quantities within the localization algorithm, with the exception of the a-posteriori integrated solution $\hat{\mathbf{x}}^{+}$. While the integration filter provides the a-posteriori error covariance matrix $\mathbf{P}^{+}$, this matrix is solely based on all of the EKF's input covariance matrices. Consequently, mismatches between these input covariance matrices and the actual measurement inputs do not affect $\mathbf{P}^{+}$, unless these mismatches are large enough to be detected via outlier detection (see Section 6.4). That's why the rest of this section is dedicated to outlining a method for checking the agreement between the integration filter's measurement inputs and their associated data quality.

The method for internal assessment of the data quality within the integration filter relies on two concepts from parameter adjustment theory: a-posteriori variance factors and redundancy components. In order to apply these techniques, the Kalman filter's measurement update is expressed in terms of a Gauss-Markov model:

$$\mathbf{l} \quad = \quad \mathbf{A}\,\mathbf{d}\hat{\mathbf{x}} \quad - \quad \upsilon \tag{5.7}$$

$$\begin{pmatrix} \hat{\mathbf{x}}^{-} \\ \tilde{\mathbf{z}} \end{pmatrix} = \begin{pmatrix} \mathbf{I} \\ \mathbf{H} \end{pmatrix} \hat{\mathbf{x}}^{+} - \begin{pmatrix} \hat{\mathbf{x}}^{+} - \hat{\mathbf{x}}^{-} \\ \delta\mathbf{z}^{+} \end{pmatrix}. \tag{5.8}$$

The vector on the left hand side is the observation vector of the Gauss-Markov model, $\mathbf{A}$ is its design matrix, $\mathbf{d}\hat{\mathbf{x}}$ is the vector of unknown parameters and υ is the improvement vector. The observation covariance matrix $\boldsymbol{\Sigma}_{\mathbf{l}}$, expressed in Kalman filter terms, is:

$$\boldsymbol{\Sigma}_{\mathbf{l}} = \begin{pmatrix} \mathbf{P}^{-} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix}. \tag{5.9}$$

The a-posteriori variance factor $\hat{\sigma}_0^2$ characterizes the agreement between the improvements and the stochastic model [*Niemeier*, 2008, Section 4.5.3]:

$$\hat{\sigma}_0^2 = \frac{\upsilon^{\mathsf{T}} \boldsymbol{\Sigma}_{\mathbf{l}}^{-1} \upsilon}{\dim(\mathbf{l}) - \dim(\mathbf{d}\hat{\mathbf{x}})}. \tag{5.10}$$

This formulation with the weight matrix in the numerator equal to the inverse of the observation covariance matrix implies an a-priori variance factor of one. The denominator in (5.10) is the redundancy of the adjustment process. For the Kalman filter equivalent, it always equals the dimension of $\tilde{\mathbf{z}}$, since $\hat{\mathbf{x}}^{-}$ and $\hat{\mathbf{x}}^{+}$ have the same dimension. If both functional and stochastic model are correct and the observation vector does not contain outliers or systematic errors, $\hat{\sigma}_0^2$ is close to one. Values larger than one imply the presence of outliers/systematic errors or an overly optimistic stochastic model, values smaller than one an overly pessimistic stochastic model.

While the true parameter vector within a classical Gauss-Markov is assumed to be constant in time, the true state vector of a Kalman filter changes with time according to its state function. The Kalman filter works recursively and only processes the current observations in conjunction with the current a-priori state vector in each epoch in order to obtain an optimal estimate of the current state. Therefore, the redundancy in each measurement update is small compared to classical parameter adjustment. The three types of EKF measurement updates that are performed within the designed algorithm input pseudoranges, range rates and odometry observations, respectively. If no outliers are detected, the redundancy for the GNSS measurement updates equals the number of received satellites (i. e. less than 20 even under perfect reception conditions), while the redundancy for the odometry measurement update is six. The number of unknowns is equal to the number of elements within the state vector, which is 22 in this case. This low level of redundancy causes the a-posteriori variance factor to fluctuate from one measurement epoch to another, with the resulting value largely depending on the short-term measurement noise. Figure 5.2 depicts this behavior for the three types of measurement updates in a test scenario. This makes it difficult to obtain useful information about the actual quality of the EKF's stochastic model from the time series of $\hat{\sigma}_0^2$.

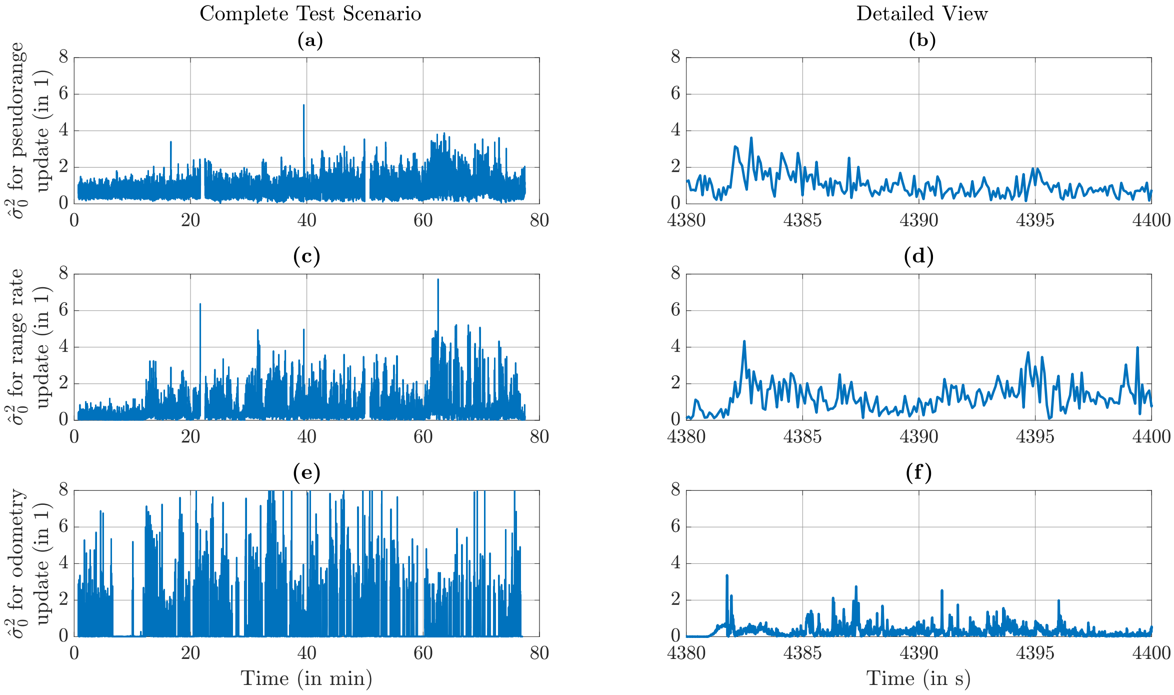

Figure 5.2: A-posteriori variance factor for all three types of measurement updates during a test scenario through the inner city of Darmstadt. **(a)**,**(c)**,**(e)** Complete test scenario. **(b)**,**(d)**,**(f)** Detailed view.

Another problem is the fact that all elements of the improvement vector $\mathfrak{v}$ are merged into a single quantity by (5.10). During any of the EKF's measurement updates, the individual entries of the state vector are not observable equally well. For instance, pseudorange observations mainly influence the position and receiver clock bias states, but have very little impact on the dynamic tire radius. This is quantified by the associated redundancy components r_c, which are the diagonal elements of the matrix $\boldsymbol{\Sigma}_\mathfrak{v}\boldsymbol{\Sigma}_\mathbf{l}^{-1}$. The sum of all individual redundancy components equals the overall redundancy $\dim(\mathbf{l}) - \dim(\mathbf{d}\hat{\mathbf{x}})$. It is equivalent to the trace of $\boldsymbol{\Sigma}_\mathfrak{v}\boldsymbol{\Sigma}_\mathbf{l}^{-1}$ [*Niemeier*, 2008, Section 8.3.2]:

$$\begin{aligned} \operatorname{tr}\left(\boldsymbol{\Sigma}_\mathfrak{v}\boldsymbol{\Sigma}_\mathbf{l}^{-1}\right) &= \operatorname{tr}\left(\mathbf{I} - \mathbf{A}\left(\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}_\mathbf{l}^{-1}\mathbf{A}\right)^{-1}\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}_\mathbf{l}^{-1}\right) \\ &= \sum_{j=1}^{\dim(\mathbf{l})} r_{c,j} \\ &= \dim(\mathbf{l}) - \dim(\mathbf{d}\hat{\mathbf{x}}). \end{aligned} \tag{5.11}$$

With the help of these redundancy components, individual a-posteriori variance factors can be computed for each of the observation vector's elements, or subsets of those. These individual a-posteriori variance factors are obtained by only using the elements of $\mathfrak{v}$ and $\boldsymbol{\Sigma}_\mathbf{l}^{-1}$ corresponding to the desired subset when evaluating the numerator in (5.10) while replacing the denominator with the sum of the redundancy components of the desired subset's elements. This enables the internal assessment of the agreement between the current state estimate and the measurement vector on the one hand and their associated covariance matrices on the other hand.

In order to perform this quality assessment, all three types of measurement updates are merged into one Gauss-Markov model. Due to the differing measurement intervals (100 ms for GNSS vs. 10 ms for odometry), multiple odometry updates occur in between two GNSS updates. Under nominal conditions, ten odometry updates are performed in the same time span as one pseudorange and one range rate update, yielding the following model:

$$\underbrace{\begin{pmatrix} \hat{\mathbf{x}}^- \\ \tilde{\mathbf{z}}_{\rho^c} \\ \tilde{\mathbf{z}}_{\dot{\rho}^c} \\ \tilde{\mathbf{z}}_{o,1} \\ \vdots \\ \tilde{\mathbf{z}}_{o,10} \end{pmatrix}}_{\mathbf{l}^*} = \underbrace{\begin{pmatrix} \mathbf{I} \\ \mathbf{H}_{\rho^c} \\ \mathbf{H}_{\dot{\rho}^c} \\ \mathbf{H}_{o,1} \\ \vdots \\ \mathbf{H}_{o,10} \end{pmatrix}}_{\mathbf{A}^*} \hat{\mathbf{x}}^+ - \underbrace{\begin{pmatrix} \hat{\mathbf{x}}^+ - \hat{\mathbf{x}}^- \\ \delta\mathbf{z}^+_{\rho^c} \\ \delta\mathbf{z}^+_{\dot{\rho}^c} \\ \delta\mathbf{z}^+_{o,1} \\ \vdots \\ \delta\mathbf{z}^+_{o,10} \end{pmatrix}}_{\mathfrak{v}^*}. \tag{5.12}$$

The observation covariance matrix has block-diagonal form:

$$\boldsymbol{\Sigma}_{\mathbf{l}^*} = \begin{pmatrix} \mathbf{P}^- & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{\rho^c} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_{\dot{\rho}^c} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{R}_{o,1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R}_{o,10} \end{pmatrix}. \tag{5.13}$$

In practice, the actual measurement intervals of both the GNSS receiver and the odometry sensors vary. To account for this, two consecutive pseudorange updates are taken to mark the beginning and end of a particular merging interval. $\hat{\mathbf{x}}^-$ in (5.12) is the a-priori estimate of the total state right before the first pseudorange update, $\hat{\mathbf{x}}^+$ is the last a-posteriori estimate before the second pseudorange update. All measurement matrices $\mathbf{H}$ and measurement noise covariance matrices $\mathbf{R}$ occurring in the meantime are stored in memory to be able to form $\mathbf{A}^*$ and $\boldsymbol{\Sigma}_{\mathbf{l}^*}$. $\mathbf{l}^*$ and $\boldsymbol{\upsilon}^*$ are not needed, because only the redundancy components and a-posteriori variance factors of the elements in the state vector $\hat{\mathbf{x}}^-$ are of interest. The measurement residuals $\delta\mathbf{z}^+$ of all updates with the exception of the last one do not refer to the last a-posteriori estimate $\hat{\mathbf{x}}^+$ within the particular merging interval, anyway.

When no GNSS measurements are available, e. g. in a tunnel, the time span in between two consecutive pseudorange updates is much larger than the nominal 100 ms. To avoid having to store many odometry update matrices $\mathbf{H}_o$ and $\mathbf{R}_o$ in memory, the nominal GNSS measurement interval is used as merging interval if no pseudorange update occurred in the last 120 ms. In this case, the entries corresponding to the GNSS updates are omitted when forming $\mathbf{A}^*$ and $\boldsymbol{\Sigma}_{\mathbf{l}^*}$. This overall process yields time series for the improvements $\hat{\mathbf{x}}^+ - \hat{\mathbf{x}}^-$, the design matrix $\mathbf{A}^*$ and the weight matrix $\boldsymbol{\Sigma}_{\mathbf{l}^*}^{-1}$, enabling the computation of redundancy components $r_{c,j}$ and a-posteriori variance factors $\hat{\sigma}_{0,j}^2$ for arbitrary subsets j of the elements in $\hat{\mathbf{x}}^-$.

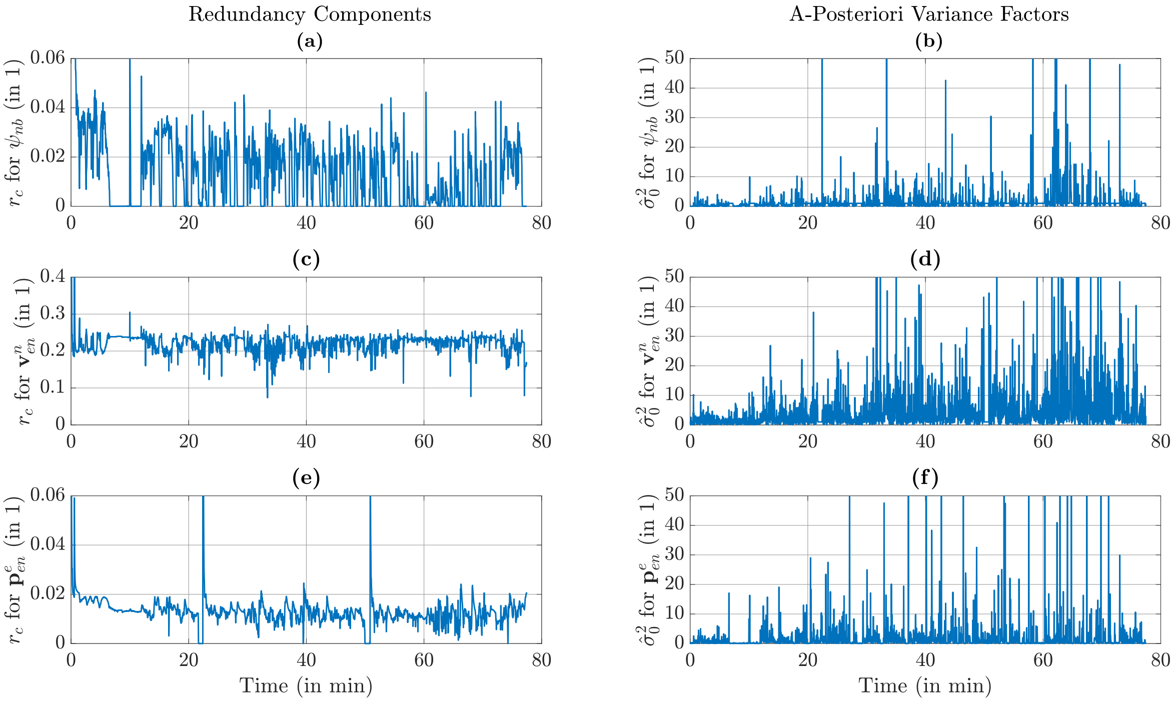

Figure 5.3: Redundancy components and a-posteriori variance factors for yaw angle, velocity and position during a test scenario through the inner city of Darmstadt. **(a)**,**(c)**,**(e)** Redundancy components. **(b)**,**(d)**,**(f)** A-posteriori variance factors.

Figure 5.3 displays r_c and $\hat{\sigma}_0^2$ for the yaw angle ψ_{nb}, the 3-D velocity vector $\mathbf{v}_{en}^n$ and the 3-D position vector $\mathbf{p}_{en}^e$ during the same test scenario whose results are depicted in Figure 5.2. The redundancy component for the vectors is obtained by summing up the three redundancy components of its elements. The three quantities yaw, velocity and position are chosen because they are the most important elements of the EKF's total state vector $\mathbf{x}$ in regard to automotive localization performance. The values for r_c reflect the current degree of observability for each quantity. Because velocity is measured by both GNSS and odometry, its redundancy component is the largest, varying between 0.1 and 0.3 most of the time. The redundancy components for ψ_{nb} and $\mathbf{p}_{en}^e$ are considerably smaller and rarely exceed 0.05. The yaw angle is only observable when the vehicle is moving, so r_c drops to zero during standstill in Figure 5.3a, e. g. from 6 min to 9 min or from 10 min to 12 min. Position is only observable through pseudoranges, so r_c drops to zero in Figure 5.3e during the two tunnels that occur within the test scenario around minute 22 and minute 50. Because the estimated position variance grows during the tunnels, the pseudorange measurements right after the tunnel have higher weights relative to the a-priori position estimate $\hat{\mathbf{p}}^e_{en}$ as they have during periods of unobstructed sky view, causing peaks in the redundancy component for $\mathbf{p}^e_{en}$ of up to 0.4. With the exception of the receiver clock error states, all other elements in $\hat{\mathbf{x}}^-$ have redundancy components of $\ll 1 \cdot 10^{-3}$, indicating that they are only very weakly observable through a single set of merged measurements. As far as the a-posteriori variance factors are concerned, the plots in Figure 5.3 still show a high degree of fluctuation. During the first 15 min, GNSS reception conditions are good. All plots for $\hat{\sigma}^2_0$ display smaller values and less variation than for the remaining test scenario, during which GNSS reception conditions are impaired due to driving through urban surroundings. $\hat{\sigma}^2_0$ for $\hat{\mathbf{p}}^e_{en}$ in Figure 5.3f is especially low during two time spans of standstill from 6 min to 9 min and from 10 min to 12 min. The maximum values of $\hat{\sigma}^2_0$ exceed $1 \cdot 10^4$, caused by redundancy components very close to zero.

In order to avoid these large peaks, a lower limit of $1 \cdot 10^{-5}$ for r_c is introduced. For all values below this, the measurements do not contain enough information to assess the agreement between the actual and the estimated error in $\hat{\mathbf{x}}^+$, making this form of internal quality assessment impossible. Consequently, $\hat{\sigma}^2_0$ is set to one when r_c is less than $1 \cdot 10^{-5}$. To limit the influence of short-term measurement noise on $\hat{\sigma}^2_0$, a moving average is formed for each individual a-posteriori variance factor. The aim of this smoothing process is to reduce the rapid fluctuation of the a-posteriori variance factors, but still allow for quick adaption to changing conditions. These condition changes might be due to the shape of the driven trajectory, e. g. turning into a corner at the end of a straight road segment, or they might be due to varying GNSS reception conditions, e. g. satellites that dis- and later reappear when the vehicle is driving past a building or passing under an overpass. *Khanafseh et al.* [2018] examine the temporal correlation of code and carrier multipath. Correlation times in kinematic data sets are much lower than in static data sets and range from 1.8 s to 96 s. In order to capture the effects that code multipath with such short correlation times has on the integrated navigation solution, the moving average of each a-posteriori variance factor is computed over a 2 s window. This short window is already able to smooth out much of the fluctuation in $\hat{\sigma}^2_0$, as shown subsequently.

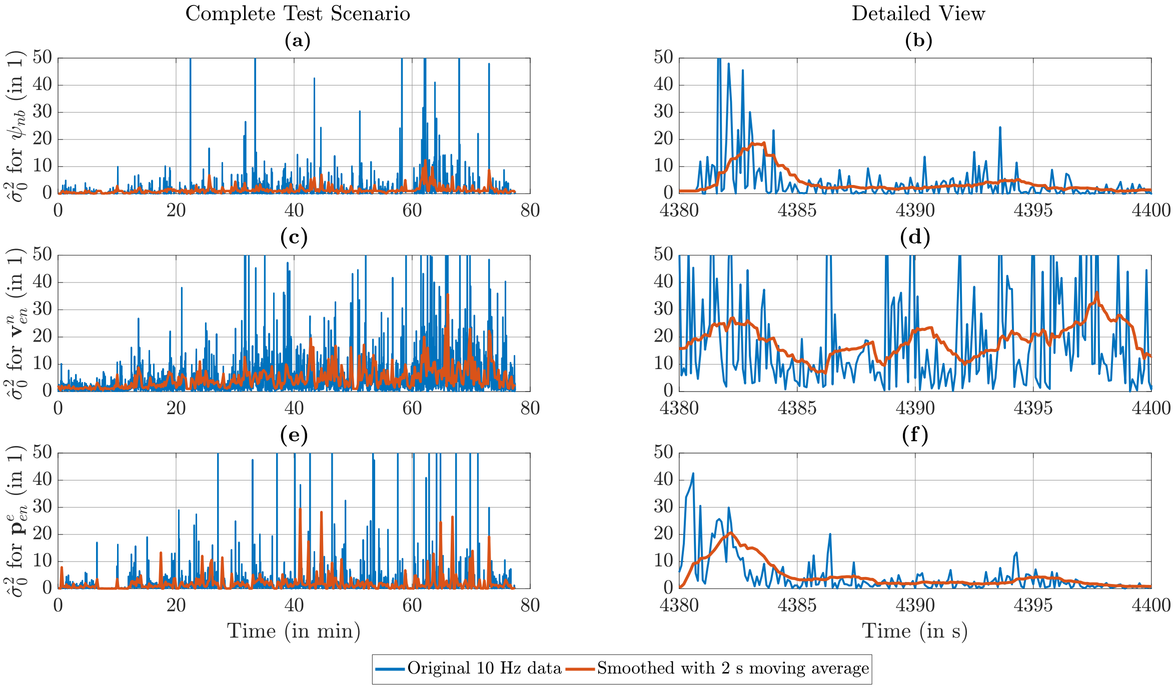

Figure 5.4: Original and smoothed a-posteriori variance factors for yaw angle, velocity and position during a test scenario through the inner city of Darmstadt. **(a)**,**(c)**,**(e)** Complete test scenario. **(b)**,**(d)**,**(f)** Detailed view.

The impact of the moving average smoother is depicted in Figure 5.4. In the overview of the complete test scenario in Figures 5.4a, c, e, the peaks of the original data are reduced drastically by the smoothing process. While all three a-posteriori variance factors exceed 100 in multiple epochs in the original data, none of the smoothed values is larger than 40 in any epoch. The smoothing effect becomes even more apparent in the detailed view in Figures 5.4b, d, f: All three time series of the original data are noisy, the values of $\hat{\sigma}_0^2$ in between two consecutive epochs often differ by a factor of more than 20. This behavior does not allow for an assessment of the current agreement between actual and estimated errors in the integrated navigation solution. These jumps are filtered out by the moving average, which exhibits a smooth progression while still being able to adapt to conditions that vary on a time scale of a few seconds, e. g. road curvature or code multipath.

The smoothed a-posteriori variance factors are used to scale the estimated variance of the integrated navigation solution. While the a-posteriori values $\hat{\mathbf{x}}^+$ and $\mathbf{P}^+$ remain unchanged for processing in the EKF in the next epoch, an additional covariance matrix is formed for each subset of $\hat{\mathbf{x}}^+$ for which an a-posteriori variance factor has been computed. This output covariance matrix is obtained by multiplying the submatrix of $\mathbf{P}^+$ concerning the respective subset with the corresponding value of $\hat{\sigma}_0^2$. A lower limit of one is applied to $\hat{\sigma}_0^2$ in this process in order to avoid reducing the entries in $\mathbf{P}^+$ even further. The estimated variance is typically too low anyway due to the EFK's assumption of measurement noise that is uncorrelated in time, an assumption which is not true for real measurement data.

5.3 External Quality Assessment

External quality assessment refers to all techniques that judge the quality of quantities provided by the localization algorithm by comparing these quantities to external reference data. Such techniques are employed to obtain information about the algorithm's performance and as a means to evaluate how well the internal quality assessment works. This section defines criteria to select suitable metrics for external quality assessment from the catalog of metrics detailed in Section 5.1 before listing the chosen metrics for the integrated navigation solution.

Selection criteria for performance metrics to be used for external quality assessment are:

1. The metrics are suited to quantify the deficits of the existing approaches as outlined in Chapter 1 as well as the potential improvements that are made by the localization algorithm developed in this thesis.
2. The metrics address quantities that are relevant for the localization algorithm's application in the automotive field.
3. The metrics have to be evaluable, i. e. a reference solution with sufficient accuracy has to be available.

Because the most important contribution of this thesis is the integration of multi-frequency pseudoranges from multiple GNSS constellations, which addresses the elimination of the position error put forth by the ionospheric delay, position accuracy is the most important metric according to criterion 1. The odometry preprocessing is designed to improve estimation quality for horizontal velocity and the yaw angle, so accuracy of these quantities has to be evaluated, too. Criterion 2 implies that more weight is to be put on the horizontal errors of position and velocity, since their vertical components only play a minor role for automotive applications as long as all wheels remain in contact with the road. Reference values for attitude, velocity, position and the dynamic tire radii exist. This is not the case for the remaining elements of the EKF's state vector, which are IMU biases and GNSS receiver clock errors (see Section 6.2). This is unproblematic as the accuracy of these quantities is not of interest according to criterion 2, anyway.

According to the criteria defined in the previous paragraph, the following metrics are selected for external quality assessment of the localization algorithm's integrated navigation solution:

- Position error $\delta\hat{\mathbf{p}}_{en}^{n}$: RMS for each of the three components, 95 % quantile of the horizontal error $|\delta\hat{p}_{en,hor}^{n}|_{95\,\%}$ and the vertical error $|\delta\hat{p}_{en,U}^{n}|_{95\,\%}$, percentage of epochs for which horizontal and vertical error are inside their respective 95 % confidence interval: $p\Big(\varepsilon_{\delta\hat{p}_{en,hor}^{n}} \leq \xi_{\delta\hat{p}_{en,hor}^{n},95\,\%}\Big)$ and $p\Big(\varepsilon_{\delta\hat{p}_{en,U}^{n}} \leq \xi_{\delta\hat{p}_{en,U}^{n},95\,\%}\Big)$. Because the position error is the single most important metric, the complete CDFs of its horizontal and vertical components are selected, as well.

- Velocity error $\delta\hat{\mathbf{v}}_{eb}^{b}$: RMS for each of the three components, 95 % quantile of the horizontal error $|\delta\hat{v}_{eb,hor}^{b}|_{95\,\%}$, percentage of epochs for which the horizontal error is inside its 95 % confidence interval: $p\Big(\varepsilon_{\delta\hat{v}_{eb,hor}^{b}} \leq \xi_{\delta\hat{v}_{eb,hor}^{b},95\,\%}\Big)$.

- Attitude error $\delta\hat{\boldsymbol{\psi}}_{nb}$: Yaw error RMS, 95 % quantile of the yaw error $|\delta\hat{\psi}_{nb}|_{95\,\%}$, percentage of epochs for which the yaw error is inside its 95 % confidence interval: $p\Big(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\Big)$.

- Dynamic tire radius error $\delta\hat{\mathbf{r}}_{d}$: Settling time $T_{s,10\,\%}$.

As detailed in Section 5.1, all of these metrics concern accuracy and integrity, because availability and continuity cannot be evaluated in a meaningful way without defining specific performance requirements. The velocity error is evaluated in the body frame because correct velocity representation in this frame is more important in automotive engineering than the representation in the navigation frame. The yaw error is picked as only component of the attitude error since roll and pitch angles as well as their estimation errors remain close to zero during usual driving conditions. The settling time is picked as most important metric for $\delta\hat{\mathbf{r}}_{d}$ as it combines the need for fast response with the requirement of a low steady-state error. The ±10 % margin is computed from the difference between reference and initial values for each of the four tire radii.

Chapter 6

Integration Filter

This chapter describes how measurements from GNSS receiver, IMU and odometry sensors are fused together to form the integrated navigation solution. It begins with an overview of the localization algorithm's architecture in Section 6.1. Afterwards, the integration filter's system model is presented in Section 6.2. That section includes information about the EKF's state vector and its initialization as well as details about covariance prediction and tuning of the stochastic model, in particular the system noise covariance matrix $\mathbf{Q}$. Besides the system model, the measurement models for GNSS and odometry described in Section 6.3 are the most important part of the EKF. In order to limit the effect of potentially faulty observations on the integrated navigation solution, additional steps are implemented within the measurement updates to detect and eliminate outliers. These steps focus on measurement innovation $\delta\mathbf{z}^-$ and measurement residual $\delta\mathbf{z}^+$. They are presented in Section 6.4.

6.1 Overall Architecture

The localization algorithm's overall architecture is depicted in Figure 6.1. On the left hand side are the sensors that provide raw measurement data to the preprocessing blocks: pseudoranges ρ and carrier phase measurements ϕ from the GNSS receiver, specific forces $\mathbf{f}_{ib}^b$ and rotation rates $\boldsymbol{\omega}_{ib}^b$ from the IMU as well as wheel rotation rates ω_w and the steering wheel angle δ_{sw} from the odometry sensors. These data are preprocessed before they are forwarded to the integration filter. As detailed in Chapter 3, the GNSS preprocessing outputs corrected pseudoranges ρ^c and range rates $\dot{\rho}^c$ as well as a-priori estimates of receiver clock bias $c\hat{\delta t}_R^-$ and drift $c\hat{\delta \dot{t}}_R^-$. In return, it inputs the a-posteriori estimates of receiver clock bias and drift. The odometry preprocessing provides wheel velocities $\mathbf{v}_{ew}^b$ and receives the a-posteriori estimates of the dynamic tire radii $\hat{\mathbf{r}}_d^+$ in return as presented in Chapter 4. The two remaining blocks in Figure 6.1 are the strapdown algorithm and the integration filter. Both are described in this section, while additional details about the integration filter are given in Sections 6.2 to 6.4.

The strapdown algorithm functions as the IMU preprocessing. It inputs the IMU data and computes a-priori estimates of the direction-cosine matrix $\hat{\mathbf{C}}_b^{n,-}$, the velocity $\hat{\mathbf{v}}_{en}^{n,-}$ and the position $\hat{\mathbf{p}}_{en}^{e,-}$. These estimates are forwarded to the integration filter which feeds back the corresponding a-posteriori estimates as well as the a-posteriori estimates of the accelerometer biases $\mathbf{b}_a$ and the gyroscope biases $\mathbf{b}_\omega$. The bias definition is:

$$\tilde{\mathbf{a}}_{ib}^b = \breve{\mathbf{a}}_{ib}^b + \mathbf{b}_a + \mathbf{w}_a \tag{6.1}$$

$$\tilde{\boldsymbol{\omega}}_{ib}^b = \breve{\boldsymbol{\omega}}_{ib}^b + \mathbf{b}_\omega + \mathbf{w}_\omega, \tag{6.2}$$

where ($\tilde{\mathbf{a}}_{ib}^b$,$\tilde{\boldsymbol{\omega}}_{ib}^b$) are the values as measured by the IMU, ($\breve{\mathbf{a}}_{ib}^b$,$\breve{\boldsymbol{\omega}}_{ib}^b$) are the true values and ($\mathbf{w}_a$,$\mathbf{w}_\omega$) are measurement noise terms.

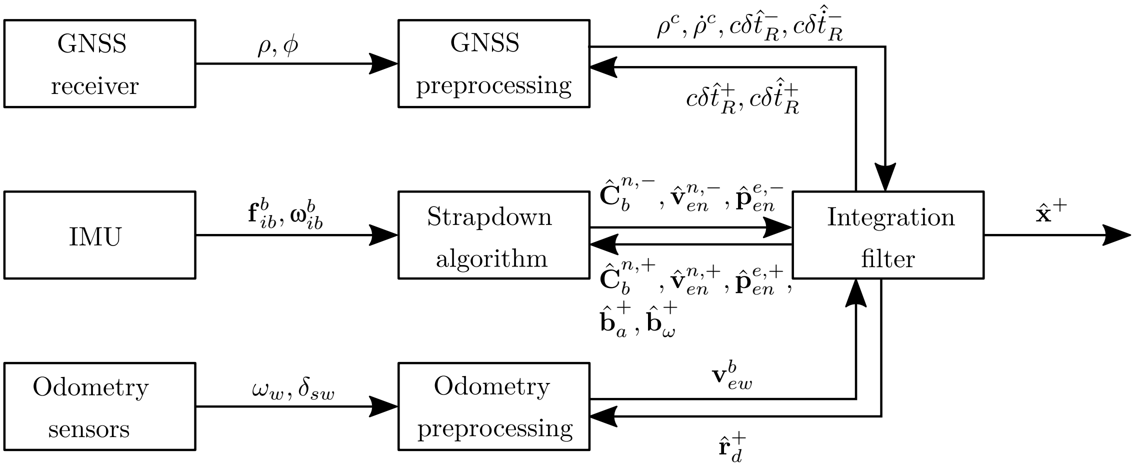
GNSS receiver
ρ, ϕ
GNSS preprocessing
$\rho^c, \dot{\rho}^c, c\delta\hat{t}_R^-, c\delta\hat{\dot{t}}_R^-$
$c\delta\hat{t}_R^+, c\delta\hat{\dot{t}}_R^+$
IMU
$\mathbf{f}_{ib}^b, \boldsymbol{\omega}_{ib}^b$
Strapdown algorithm
$\hat{\mathbf{C}}_b^{n,-}, \hat{\mathbf{v}}_{en}^{n,-}, \hat{\mathbf{p}}_{en}^{e,-}$
Integration filter
$\hat{\mathbf{x}}^+$
$\hat{\mathbf{C}}_b^{n,+}, \hat{\mathbf{v}}_{en}^{n,+}, \hat{\mathbf{p}}_{en}^{e,+},$
$\hat{\mathbf{b}}_a^+, \hat{\mathbf{b}}_\omega^+$
Odometry sensors
ω_w, δ_{sw}
Odometry preprocessing
$\mathbf{v}_{ew}^b$
$\hat{\mathbf{r}}_d^+$

Figure 6.1: Overall localization algorithm architecture. Measurement timestamps and covariance matrices omitted to increase readability (Repetition of Figure 1.1 with additional information).

A part of the localization algorithm that is not depicted explicitly in Figure 6.1 is the timing module. Its task is to check whether new sensor data are available and to trigger execution of the relevant algorithm parts. Whenever new GNSS measurement are detected, the GNSS preprocessing is triggered and its output data are forwarded to the integration filter. The same procedure is applied to the odometry preprocessing each time new odometry observations are available. The integration filter itself is triggered synchronously with the strapdown algorithm whenever new IMU measurements are recognized. When these IMU measurements are the only new sensor data, the integration filter solely performs its prediction step. If new data from GNSS and/or odometry preprocessing are available, the corresponding measurement updates are triggered, too. After the integration filter has performed all requested steps, it outputs the a-posteriori estimate of the total state vector $\hat{\mathbf{x}}^+$. It is identical with the a-priori estimate $\hat{\mathbf{x}}^-$ in case no measurement updates were triggered.

The strapdown algorithm's main task is the computation of the a-priori navigation solution, consisting of $\hat{\mathbf{C}}_b^{n,-}$, $\hat{\mathbf{v}}_{en}^{n,-}$ and $\hat{\mathbf{p}}_{en}^{e,-}$. It is performed in the navigation frame formulation according to *Groves* [2013, Section 5.4]. The first step is the attitude update:

$$\hat{\mathbf{C}}_{b,k}^{n,-} = \hat{\mathbf{C}}_{b,k-1}^{n,+} \mathrm{e}^{\boldsymbol{\Omega}_{nb}^b \tau_{i,k}}, \tag{6.3}$$

where $\boldsymbol{\Omega}_{nb}^b$ is the skew-symmetric matrix of the rotation rate vector $\boldsymbol{\omega}_{nb}^b$ and $\tau_{i,k} = t_{i,k} - t_{i,k-1}$ is the current IMU measurement interval. $\boldsymbol{\omega}_{nb}^b$ is obtained from $\tilde{\boldsymbol{\omega}}_{ib}^b$ by applying corrections for the gyroscope bias $\mathbf{b}_\omega$, the Earth rotation rate $\boldsymbol{\omega}_{ie}^b$ and the transport rate $\boldsymbol{\omega}_{en}^b$. For the evaluation of (6.3), the matrix exponential function has to be used. The next step is the velocity update. To account for the attitude change during $\tau_{i,k}$, the mean of $\hat{\mathbf{C}}_{b,k-1}^{n,+}$ and $\hat{\mathbf{C}}_{b,k}^{n,-}$ is deployed:

$$\hat{\mathbf{v}}_{en,k}^{n,-} = \hat{\mathbf{v}}_{en,k-1}^{n,+} + \tau_{i,k} \left[\frac{\hat{\mathbf{C}}_{b,k-1}^{n,+} + \hat{\mathbf{C}}_{b,k}^{n,-}}{2} \mathbf{a}_{ib}^b - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \hat{\mathbf{v}}_{en,k-1}^{n,+} - \mathbf{g}_{ib}^n \right], \tag{6.4}$$

where $\mathbf{a}_{ib}^b$ is the IMU's acceleration measurement, corrected for the accelerometer bias $\mathbf{b}_a$, $(2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \hat{\mathbf{v}}_{en,k-1}^{n,+}$ is the Coriolis correction and $\mathbf{g}_{ib}^n$ is the gravity correction. The strapdown algorithm's last step is the position update. In analogy to the velocity update, the mean of a-posteriori values from the previous epoch and already updated a-priori values from the current epoch is deployed to account for the change of velocity and position during $\tau_{i,k}$:

$$\begin{aligned}
h_{e,k}^{-} &= h_{e,k-1}^{+} + \frac{\tau_{i,k}}{2}\left[v_{en,U,k-1}^{n,+} + v_{en,U,k}^{n,-}\right] \\
\varphi_{e,k}^{-} &= \varphi_{e,k-1}^{+} + \frac{\tau_{i,k}}{2}\left[\frac{v_{en,N,k-1}^{n,+}}{R_N + h_{e,k-1}^{+}} + \frac{v_{en,N,k}^{n,-}}{R_N + h_{e,k}^{-}}\right] \\
\lambda_{e,k}^{-} &= \lambda_{e,k-1}^{+} + \frac{\tau_{i,k}}{2}\left[\frac{v_{en,E,k-1}^{n,+}}{\left(R_E + h_{e,k-1}^{+}\right)\cos\varphi_{e,k-1}^{+}} + \frac{v_{en,E,k}^{n,-}}{\left(R_E + h_{e,k}^{-}\right)\cos\varphi_{e,k}^{-}}\right],
\end{aligned} \tag{6.5}$$

where R_N is the reference ellipsoid's radius of curvature in north-south direction (meridian radius of curvature) and R_E is the reference ellipsoid's radius of curvature in east-west direction (transverse radius of curvature). They are computed before the position update based on the a-posteriori position estimate from the previous epoch $\hat{\mathbf{p}}_{en,k-1}^{e,+}$.

The integration filter is capable of performing three types of measurement updates: GNSS pseudorange update, GNSS range rate update and odometry update. These three are considered to be independent of each other because their corresponding measurement noise covariance matrices $\mathbf{R}_{\rho^c}$, $\mathbf{R}_{\dot{\rho}^c}$ and $\mathbf{R}_o$ are uncorrelated. This fact makes it possible to process the different types of observations sequentially in case two or more different measurement updates are requested in the same epoch [*Groves*, 2013, Section 3.2.7]. To do so, the first measurement update is performed with an a-priori error state estimate of $\delta\hat{\mathbf{x}}_k^- = \mathbf{0}$ and the error covariance matrix $\mathbf{P}_k^-$. The a-posteriori values $\delta\hat{\mathbf{x}}_k^+$ and $\mathbf{P}_k^+$ from this first measurement update are then used as a-priori values for the second measurement update, and so on. Hence, the only apparent distinction between the first and the following updates is that the input error state estimate is non-zero for all updates except the first one. For linear measurement functions $\mathbf{h}$, the sequential update is equivalent to the batch update. Whether the difference between sequential and batch update can be neglected for nonlinear measurement functions depends on the characteristics of $\mathbf{h}$ and the magnitude of $\delta\hat{\mathbf{x}}_k^-$. It can be neglected if the linear approximation is still valid, i. e. if the following hold:

$$\mathbf{h}(\hat{\mathbf{x}}_k^- + \delta\hat{\mathbf{x}}_k^-) \approx \mathbf{h}(\hat{\mathbf{x}}_k^-) + \left.\frac{\partial\mathbf{h}}{\partial\mathbf{x}}\right|_{\hat{\mathbf{x}}_k^-}\delta\hat{\mathbf{x}}_k^- \tag{6.6}$$

$$\left.\frac{\partial\mathbf{h}}{\partial\mathbf{x}}\right|_{\hat{\mathbf{x}}_k^-} \approx \left.\frac{\partial\mathbf{h}}{\partial\mathbf{x}}\right|_{\hat{\mathbf{x}}_k^- + \delta\hat{\mathbf{x}}_k^-}. \tag{6.7}$$

For all types of measurement updates discussed in this thesis, it is assumed that the error state estimates $\delta\hat{\mathbf{x}}_k^-$ are always small enough for (6.6) and (6.7) to be true.

After all requested measurement updates are completed, a final error state estimate $\delta\hat{\mathbf{x}}_k^+$ results. It is utilized to update the total state estimate $\hat{\mathbf{x}}_k^-$:

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- - \delta\hat{\mathbf{x}}_k^+. \tag{6.8}$$

The negative sign is due to the definition of the error state vector:

$$\breve{\mathbf{x}} = \hat{\mathbf{x}} - \delta\mathbf{x}, \tag{6.9}$$

rearranging (2.29) to see that the estimated error state vector needs to be subtracted from the current total state estimate in order to obtain an estimate that is closer to the true value. For the state vector of the localization algorithm as defined in Section 6.2, (6.8) cannot be applied directly. The reason for this is that the representation of attitude and position differs from strapdown algorithm to error state vector, requiring more complex updates of the a-priori navigation solution. The reason for the changing representation and the resulting update methods are described in Section 6.2. For the state vector's other elements, (6.8) is applied directly.

6.2 System Model

The integrations filter's state vector $\mathbf{x}$ consists of 22 states. The first nine states comprise the navigation solution: attitude $\boldsymbol{\psi}_{nb}$, velocity $\mathbf{v}^n_{en}$ and position $\mathbf{p}^e_{en}$. The next six states are the IMU errors: gyroscope bias $\mathbf{b}_\omega$ and accelerometer bias $\mathbf{b}_a$. They are followed by three GNSS clock errors: receiver clock bias for GPS $c\delta t^G_R$, receiver clock drift $c\delta\dot{t}_R$ and receiver clock bias for Galileo $c\delta t^E_R$. The last four entries are the dynamic tire radii, assembled into one vector $\mathbf{r}_d$:

$$\mathbf{x} = \begin{pmatrix} \boldsymbol{\psi}_{nb} \\ \mathbf{v}^n_{en} \\ \mathbf{p}^e_{en} \\ \mathbf{b}_\omega \\ \mathbf{b}_a \\ c\delta t^G_R \\ c\delta\dot{t}_R \\ c\delta t^E_R \\ \mathbf{r}_d \end{pmatrix} . \tag{6.10}$$

With the exception of $\mathbf{p}^e_{en}$, the error state vector $\delta\mathbf{x}$ is the difference between $\hat{\mathbf{x}}$ and $\check{\mathbf{x}}$ as defined in (2.29). The error state vector's entries for the position are changed due to numerical reasons: While $\mathbf{p}^e_{en}$ consists of longitude λ_e, latitude φ_e and height h_e in the units radian, radian and meter, the error state entry $\delta\mathbf{p}^n_{en}$ is resolved in the navigation frame in the unit meter for all three components. This improves conditioning of the error covariance matrix $\mathbf{P}$ and leads to entries of similar magnitude in $\delta\mathbf{x}$. To update the total states, the a-posteriori estimate $\delta\hat{\mathbf{p}}^{n,+}_{en}$ is converted back to the desired units [*Wendel*, 2007, p. 214]:

$$\hat{\lambda}^+_e = \hat{\lambda}^-_e - \frac{\delta\hat{r}^{n,+}_{en,E}}{\left(R_E + h^-_e\right)\cos\varphi^-_e} \tag{6.11}$$

$$\hat{\varphi}^+_e = \hat{\varphi}^-_e - \frac{\delta\hat{r}^{n,+}_{en,N}}{R_N + h^-_e} \tag{6.12}$$

$$\hat{h}^+_e = \hat{h}^-_e - \delta\hat{r}^{n,+}_{en,U} \tag{6.13}$$

While the attitude is represented via Euler angles in the state vector, it is represented via direction-cosine matrix in the strapdown algorithm. In order to update $\hat{\mathbf{C}}^{n,-}_b$, the a-posteriori estimate $\delta\hat{\boldsymbol{\psi}}^+_{nb}$ is converted into the direction-cosine matrix $\delta\hat{\mathbf{C}}^{n,+}_b$ [*Groves*, 2013, p. 563]:

$$\hat{\mathbf{C}}^{n,+}_b = \delta\hat{\mathbf{C}}^{n,+^{\mathsf{T}}}_b \hat{\mathbf{C}}^{n,-}_b . \tag{6.14}$$

The conversion between $\delta\hat{\boldsymbol{\psi}}^+_{nb}$ and $\delta\hat{\mathbf{C}}^{n,+}_b$ happens analogous to the conversion between $\boldsymbol{\psi}_{nb}$ and $\mathbf{C}^n_b$. It is nonlinear in general, but may be linearized if the small angle approximation applies. Both the linear and the nonlinear conversion are detailed in *Groves* [2013, pp. 38-39]. For the results presented in Chapter 7, the nonlinear conversion routine is applied regardless of the magnitude of $\delta\hat{\boldsymbol{\psi}}^+_{nb}$.

Covariance prediction is performed based on the state vector's time derivative as described in (2.31)-(2.34). The time derivatives for the first 15 states are essentially the same as in *Groves* [2013, Section 14.2.4]. The differences are:

- While *Groves* [2013] formulates the equations in the north-east-down version of the n-frame, the east-north-up version is employed in this thesis.
- *Groves* [2013] resolves the position error state in the e-frame with the units radian, radian and meter.

- *Groves* [2013] employs a different gravity model, so the partial derivative of $\dot{\mathbf{v}}^e_{en}$ w. r. t. the position is slightly different.
- Due to a different treatment of the IMU biases, all partial derivatives w. r. t. either $\mathbf{b}_\omega$ or $\mathbf{b}_a$ receive the opposite sign.

The resulting system model for the first 15 states is detailed in the appendix (Section C.1). The time derivative for the three GNSS clock states and the four dynamic tire radii is:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} c\delta t_R^G \\ c\delta\dot{t}_R \\ c\delta t_R^E \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}\begin{pmatrix} c\delta t_R^G \\ c\delta\dot{t}_R \\ c\delta t_R^E \end{pmatrix} \tag{6.15}$$

$$\dot{\mathbf{r}}_d = \mathbf{0}, \tag{6.16}$$

where only the deterministic part is given.

6.2.1 Initialization

Before the localization algorithm's regular routine of prediction and measurement updates can take place, initialization of the total state $\hat{\mathbf{x}}$ and the error covariance matrix $\mathbf{P}$ is required. This cannot happen in a single epoch because while most states are initialized during standstill, the yaw angle ψ_{nb} can only be initialized when the vehicle is moving. Initialization of ψ_{nb} during standstill is impossible with the given sensor configuration since the IMU's gyroscopes are not accurate enough to perform gyrocompassing and the GNSS receiver is only equipped with a single antenna. In the following, the initialization process is described step-by-step as it is performed within the localization algorithm. All errors are assumed to be uncorrelated unless stated otherwise.

1. The IMU biases are initialized as zero with a variance according to the IMU's data sheet ($\sigma_{b_\omega} = 0.2\,{}^\circ\,\mathrm{s}^{-1}$, $\sigma_{b_a} = 0.03\,\mathrm{m\,s}^{-2}$).
2. The tire radii are initialized with a value received from the CAN bus. This value corresponds to an average of all permissible tire sizes, therefore it is the same for all four tires and does not change from one measurement scenario to another. The standard deviation is set to $3\sigma_{r_d} = 5\,\mathrm{mm}$, approximating the tire's wear over its lifetime.
3. Roll and pitch are estimated based on accelerometer readings [*Groves*, 2013, p. 198]:

   $$\eta_{nb} = \operatorname{arctan}_2\left(f^b_{ib,L}, f^b_{ib,U}\right) \qquad \vartheta_{nb} = \arctan\frac{-f^b_{ib,F}}{\sqrt{{f^b_{ib,L}}^2 + {f^b_{ib,U}}^2}}. \tag{6.17}$$

   Their standard deviation is set to $\sigma_{\eta_{nb}} = \sigma_{\vartheta_{nb}} = 1^\circ$ based on the IMU's measurement quality during standstill, evaluated against a navigation grade IMU.
4. Antenna position and receiver clock biases are obtained via a single-epoch GNSS solution. Their full error covariance matrix is known due to the adjustment process, so their correlation is included in $\mathbf{P}$. The total state vector contains the IMU position $\mathbf{p}^e_{en}$, which differs from the antenna position by the lever arm $\mathbf{p}^b_{ba}$. Because the yaw angle is still unknown, the lever arm's orientation w. r. t. the Earth is unknown, as well. To account for this additional uncertainty of $\mathbf{p}^e_{en}$, the horizontal position variance components in $\mathbf{P}$ are increased depending on the lever arm's horizontal length.

5. Initialization of velocity and receiver clock drift via GNSS requires at least four range rates, which are computed from two consecutive GNSS measurement epochs. It also requires a known antenna position since the unit vectors $\mathbf{u}^n_{as}$ are needed. Therefore, it happens after position initialization. In analogy to the latter, the full error covariance matrix of $\mathbf{v}^n_{en}$ and $c\delta\dot{t}_R$ is known, so it is incorporated into $\mathbf{P}$.

6. The yaw angle is set to zero, enabling the start of strapdown algorithm and EKF covariance prediction. GNSS measurement updates are activated as well, since their observables are resolved in the n-frame. $\sigma_{\psi_{nb}}$ is set to 60°, reflecting a maximum yaw angle error of $3\sigma = 180°$. The variance is reset to that value before every EKF prediction step until yaw angle initialization is completed. This is done to counter an unrealistic decrease of $\sigma_{\psi_{nb}}$ during the GNSS measurement updates, which assume small errors. This assumption is not true for the yaw angle in general until it has been initialized correctly in the next step. This violation of the linearization conditions may lead to filter divergence if disregarded. With the high value of $\sigma_{\psi_{nb}}$, high measurement innovations will not affect the other states too much.

7. Once the vehicle's speed is sufficiently high and it is traveling in a straight line, the yaw angle is initialized by computing the angle between the x^n-axis and $\mathbf{v}^n_{ea}$. The former points east, the latter is obtained via GNSS as in step 5 [*Groves*, 2013, p. 225]:

$$\psi_{nb} = \arctan_2\left(v^n_{ea,N}, v^n_{ea,E}\right). \tag{6.18}$$

   (6.18) is only valid if the vehicle side slip angle β is zero. To check whether this condition holds, $\mathbf{v}^b_{eb}$ is calculated by forming the arithmetic mean of the four individual wheel velocities $\mathbf{v}^b_{ew}$. To enable yaw angle initialization, all of the following conditions must be met: $\left|v^b_{eb,L}\right| \leq 0.2\,\mathrm{m\,s^{-1}}$, $\left|v^b_{eb,F}\right| \geq 5\,\mathrm{m\,s^{-1}}$, $\sqrt{{v^n_{ea,E}}^2 + {v^n_{ea,N}}^2} \geq 5\,\mathrm{m\,s^{-1}}$ and $\left|\omega^b_{ib,U}\right| \leq 1\,{}^\circ\,\mathrm{s^{-1}}$. $\omega^b_{ib,U}$ is measured by the IMU. The standard deviation of ψ_{nb} computed in this manner is obtained by comparing it to a reference solution. For the sensor configuration employed in this thesis, $\sigma_{\psi_{nb}}$ is set to 2° for the initial yaw value based on results from numerous test scenarios. This completes the initialization process. The odometry measurement update is activated subsequently because the integration filter is able to calculate a valid estimate of $\mathbf{v}^b_{eb}$ now.

6.2.2 System Noise

Just as the measurement noise covariance matrices $\mathbf{R}$, the system noise covariance matrix $\mathbf{Q}$ plays an important role for the integration filter's performance. It describes how the estimation errors evolve over time. The relative magnitude of $\mathbf{Q}$ and $\mathbf{R}$ determines the weighting of the various sensor outputs relative to each other. When integrating an INS with other sensors as is performed in this thesis, $\mathbf{Q}$ contains among other things information about the IMU's measurement quality. Based on *Groves* [2013, Section 14.2.6], the continuous-time system noise covariance matrix $\check{\mathbf{Q}}$ is obtained from the *power spectral densities* (PSDs) of the relevant sensor output, denoted as S. For conversion to discrete time, $\check{\mathbf{Q}}$ is multiplied by the state propagation interval $\tau_{s,k}$:

$$\mathbf{Q}_k = \check{\mathbf{Q}}\tau_{s,k}. \tag{6.19}$$

For the integration filter presented in this thesis, $\tau_{s,k}$ is identical to the IMU measurement interval $\tau_{i,k}$ because strapdown algorithm and covariance propagation run with the same update rate. All elements of $\check{\mathbf{Q}}$ are specified subsequently. The notation $\check{\mathbf{Q}}(\mathbf{y})$ is used to denote the covariance of the continuous-time system noise $\check{\mathbf{w}}$ for the elements $\mathbf{y}$ in $\mathbf{x}$. Unless noted otherwise, all off-diagonal elements are zero.

- $\check{\mathbf{Q}}(\psi_{nb}) = S_{rg}\mathbf{I}$, with the PSD of the gyroscopes' random noise S_{rg}. According to the IMU's data sheet, $S_{rg} \leq 6.9 \cdot 10^{-8}\,\mathrm{rad}^2\,\mathrm{s}^{-1}$. This value is multiplied by two to account for additional error sources, e. g. due to vibration in the car.
- $\check{\mathbf{Q}}(\mathbf{v}_{en}^{n}) = S_{ra}\mathbf{I}$, with the PSD of the accelerometers' random noise S_{ra}. According to the IMU's data sheet, $S_{ra} \leq 2.2 \cdot 10^{-6}\,\mathrm{m}^2\mathrm{s}^{-3}$. This value is multiplied by two to account for additional error sources, e. g. due to vibration in the car.
- $\check{\mathbf{Q}}(\mathbf{p}_{en}^{e}) = \mathbf{0}$.
- $\check{\mathbf{Q}}(\mathbf{b}_{\omega}) = \frac{\sigma_{bg}^2}{\tau_{bg}}\mathbf{I}$ with the standard deviation of the gyroscopes' in-run bias variation σ_{bg} and its correlation time τ_{bg}. According to the IMU's data sheet, $\sigma_{bg} = 10\,°\,\mathrm{h}^{-1}$. τ_{bg} is estimated from static measurement data to be approx. 200 s. The resulting value of $\frac{\sigma_{bg}^2}{\tau_{bg}} \approx 1.2 \cdot 10^{-11}\,\mathrm{rad}^2\mathrm{s}^{-3}$ is multiplied by two to account for additional error sources.
- $\check{\mathbf{Q}}(\mathbf{b}_{a}) = \frac{\sigma_{ba}^2}{\tau_{ba}}\mathbf{I}$ with the standard deviation of the accelerometers' in-run bias variation σ_{ba} and its correlation time τ_{ba}. According to the IMU's data sheet, $\sigma_{ba} = 4 \cdot 10^{-4}\,\mathrm{m}\,\mathrm{s}^{-2}$. τ_{ba} is estimated from static measurement data to be approx. 125 s. The resulting value of $\frac{\sigma_{ba}^2}{\tau_{ba}} \approx 1.3 \cdot 10^{-9}\,\mathrm{m}^2\mathrm{s}^{-5}$ is multiplied by two to account for additional error sources. The obtained values for τ_{bg} and τ_{ba} are plausible, since in-run bias variations typically change over periods of the order of a minute [*Groves*, 2013, p. 152].
- $\check{\mathbf{Q}}(c\delta t_r^G) = \check{\mathbf{Q}}(c\delta t_r^E) = S_{c\phi}$, with the receiver clock phase drift PSD $S_{c\phi}$. It is estimated from kinematic data with good GNSS reception conditions via [*Groves*, 2013, p. 418]:

  $$S_{c\phi} = \frac{\sigma^2\left(c\delta t_{R,k} - c\delta t_{R,k-1} - c\delta \dot{t}_{R,k-1}\tau_g\right)}{\tau_g}, \tag{6.20}$$

  where σ^2 denotes an operator to obtain the variance of the term inside the parentheses. The resulting numerical value is $S_{c\phi} = 5 \cdot 10^{-4}\,\mathrm{m}^2\,\mathrm{s}^{-1}$.
- $\check{\mathbf{Q}}(c\delta \dot{t}_r) = S_{cf}$, with the receiver clock frequency drift PSD S_{cf}. It is estimated from kinematic data with good GNSS reception conditions via [*Groves*, 2013, p. 418]:

  $$S_{cf} = \frac{\sigma^2\left(c\delta \dot{t}_{R,k} - c\delta \dot{t}_{R,k-1}\right)}{\tau_g}, \tag{6.21}$$

  resulting in a numerical value of $S_{cf} = 0.01\,\mathrm{m}^2\mathrm{s}^{-3}$.
- $\check{\mathbf{Q}}(\mathbf{r}_d)$ is the only part of $\check{\mathbf{Q}}$ which includes non-zero off-diagonal elements, because tire wear and tire temperature variation as the main causes for radius variation are highly correlated among all four wheels [*Groves*, 2013, p. 674]. The correlation coefficient is estimated as 0.25, so the respective part of the system noise covariance matrix becomes:

  $$\check{\mathbf{Q}}(\mathbf{r}_d) = \begin{pmatrix} 1 & 0.25 & 0.25 & 0.25 \\ 0.25 & 1 & 0.25 & 0.25 \\ 0.25 & 0.25 & 1 & 0.25 \\ 0.25 & 0.25 & 0.25 & 1 \end{pmatrix} S_{r_d}, \tag{6.22}$$

  with the PSD of tire radius variation S_{r_d}. This PSD is parametrized based on the assumption that the change in r_d does not exceed 0.2 mm per day, with a confidence interval of 3σ:

  $$S_{r_d} = \frac{\left(\frac{0.2\,\mathrm{mm}}{3}\right)^2}{24\,\mathrm{h}} \approx 5.1 \cdot 10^{-14}\,\mathrm{m}^2\,\mathrm{s}^{-1}. \tag{6.23}$$

  Assuming an average speed of $50\,\mathrm{km}\,\mathrm{h}^{-1}$, a change of 0.2 mm per day equals a change of 5 mm per 30 000 km, which corresponds to a tire's overall wear over its lifetime.

6.3 Measurement Models

The GNSS and odometry preprocessing modules send their respective observables and the corresponding covariance matrices to the integration filter. The filter then utilizes these data to update the a-priori state estimate $\hat{\mathbf{x}}$. For this task, the EKF's measurement vector $\mathbf{z}$ and the measurement matrix $\mathbf{H}$ are necessary:

$$\mathbf{z} = \hat{\mathbf{z}} - \tilde{\mathbf{z}} = \mathbf{h}(\hat{\mathbf{x}}^-) - \tilde{\mathbf{z}} \qquad\qquad \mathbf{H} = \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}^-}, \tag{6.24}$$

repeating (2.39)-(2.41). Each of these two quantities is required three times: ($\mathbf{z}_{\rho^c}$,$\mathbf{H}_{\rho^c}$) for pseudoranges, ($\mathbf{z}_{\dot{\rho}^c}$,$\mathbf{H}_{\dot{\rho}^c}$) for range rates and ($\mathbf{z}_o$,$\mathbf{H}_o$) for odometry. They are detailed in this order in the subsequent subsections. Afterwards, the issue of time synchronization in between the different types of observables is detailed as well as the approach that this thesis takes to deal with this issue.

6.3.1 Pseudorange Update

For the pseudorange measurement update, the total measurement vector $\tilde{\mathbf{z}}_{\rho^c}$ consists of the corrected pseudoranges ρ^c, while the predicted measurement vector $\hat{\mathbf{z}}_{\rho^c}$ consists of the estimated geometric range between satellite and receiver antenna. An entry z_{ρ^c} of the measurement vector in error state formulation $\mathbf{z}_{\rho^c}$ is formed as:

$$\begin{aligned} z_{\rho^c} &= \hat{z}_{\rho^c} - \tilde{z}_{\rho^c} \\ &= \|\mathbf{p}^e_{es} - \hat{\mathbf{p}}^e_{ea}\| - \left(\rho - c\delta\hat{t}_R + c\delta\hat{t}_S - \delta\hat{\rho}_I - \delta\hat{\rho}_T\right) \\ &= \left(\|\mathbf{p}^e_{es} - \hat{\mathbf{p}}^e_{ea}\| + c\delta\hat{t}_R\right) - \left(\rho + c\delta\hat{t}_S - \delta\hat{\rho}_I - \delta\hat{\rho}_T\right) \\ &= \left(\left\|\mathbf{p}^e_{es} - \left[\hat{\mathbf{p}}^e_{en} + \mathbf{C}^e_n \hat{\mathbf{C}}^n_b \mathbf{p}^b_{ba}\right]\right\| + c\delta\hat{t}_R\right) - \left(\rho + c\delta\hat{t}_S - \delta\hat{\rho}_I - \delta\hat{\rho}_T\right). \end{aligned} \tag{6.25}$$

The first set of parentheses subsumes all terms that depend on $\hat{\mathbf{x}}$, while the terms in the second set of parentheses do not. $\delta\hat{t}_R$ refers to $\delta\hat{t}^G_R$ for GPS satellites and to $\delta\hat{t}^E_R$ for Galileo satellites. The only non-zero columns in the measurement matrix $\mathbf{H}_{\rho^c}$ are the partial derivatives w. r. t. attitude, position and receiver clock bias:

$$\mathbf{H}_{\rho^c} = \begin{pmatrix} \frac{\partial \mathbf{h}_{\rho^c}}{\partial \boldsymbol{\psi}_{nb}} & \mathbf{0} & \frac{\partial \mathbf{h}_{\rho^c}}{\partial \mathbf{p}^n_{en}} & \mathbf{0} & \mathbf{0} & \frac{\partial \mathbf{h}_{\rho^c}}{\partial \left(c\delta t^G_R\right)} & \mathbf{0} & \frac{\partial \mathbf{h}_{\rho^c}}{\partial \left(c\delta t^E_R\right)} & \mathbf{0} \end{pmatrix}. \tag{6.26}$$

Wendel [2007, pp. 212-213] gives the resulting partial derivatives for a single-constellation setup. For a multi-constellation setup, they are identical except that the entry for the receiver clock bias is either zero or one, depending on which constellation the pseudoranges stem from:

$$\frac{\partial z_{\rho^c}}{\partial \boldsymbol{\psi}_{nb}} = \mathbf{u}^{n\,\mathsf{T}}_{as} \left[\hat{\mathbf{C}}^n_b \mathbf{p}^b_{ba} \times\right] \tag{6.27}$$

$$\frac{\partial z_{\rho^c}}{\partial \mathbf{p}^n_{en}} = -\mathbf{u}^{n\,\mathsf{T}}_{as} \tag{6.28}$$

$$\frac{\partial z_{\rho^c}}{\partial \left(c\delta t^G_R\right)} = \begin{cases} 1 & \text{for GPS satellites} \\ 0 & \text{for Galileo satellites} \end{cases} \tag{6.29}$$

$$\frac{\partial z_{\rho^c}}{\partial \left(c\delta t^E_R\right)} = \begin{cases} 0 & \text{for GPS satellites} \\ 1 & \text{for Galileo satellites.} \end{cases} \tag{6.30}$$

6.3.2 Range Rate Update

For the range rate measurement update, the total measurement vector $\tilde{\mathbf{z}}_{\dot\rho^c}$ consists of the corrected range rates $\dot\rho^c$, while the predicted measurement vector $\hat{\mathbf{z}}_{\dot\rho^c}$ consists of the estimated line-of-sight velocity between satellite and receiver antenna. An entry $z_{\dot\rho^c}$ of the measurement vector in error state formulation $\mathbf{z}_{\dot\rho^c}$ is formed as:

$$\begin{aligned} z_{\dot\rho^c} &= \hat z_{\dot\rho^c} - \tilde z_{\dot\rho^c} \\ &= \mathbf{u}_{as}^{n\,\mathsf{T}} \left[\mathbf{v}_{es}^n - \hat{\mathbf{v}}_{ea}^n\right] - \left[\frac{\lambda_{ca}}{2\pi}\frac{\phi_k^c - \phi_{k-1}^c}{\tau_{g,k}^c} - c\delta\hat{\dot t}_R\right] \\ &= \left(\mathbf{u}_{as}^{n\,\mathsf{T}} \left[\mathbf{v}_{es}^n - \left\{\hat{\mathbf{v}}_{en}^n + \hat{\mathbf{C}}_b^n\left(\hat{\boldsymbol{\omega}}_{eb}^b \times \mathbf{p}_{ba}^b\right)\right\}\right] + c\delta\hat{\dot t}_R\right) - \frac{\lambda_{ca}}{2\pi}\frac{\phi_k^c - \phi_{k-1}^c}{\tau_{g,k}^c}. \end{aligned} \tag{6.31}$$

The outer parentheses subsume all terms that depend on $\hat{\mathbf{x}}$, while the last term does not (the dependency of τ_g^c on $c\delta\hat{\dot t}_R$ is neglected). Due to the magnitude of $\boldsymbol{\omega}_{eb}^b$ (up to $40\,^\circ\,\mathrm{s}^{-1}$) in comparison to the Earth rotation rate, $\boldsymbol{\omega}_{ie}$ can be neglected and $\boldsymbol{\omega}_{eb}^b \approx \boldsymbol{\omega}_{ib}^b$. The estimate of $\boldsymbol{\omega}_{ib}^b$ depends on the estimated gyroscope bias:

$$\hat{\boldsymbol{\omega}}_{eb}^b \approx \hat{\boldsymbol{\omega}}_{ib}^b = \tilde{\boldsymbol{\omega}}_{ib}^b - \hat{\mathbf{b}}_\omega. \tag{6.32}$$

The only non-zero columns in the measurement matrix $\mathbf{H}_{\dot\rho^c}$ are therefore the partial derivatives w. r. t. attitude, velocity, gyroscope bias and receiver clock drift:

$$\mathbf{H}_{\dot\rho^c} = \begin{pmatrix} \frac{\partial \mathbf{h}_{\dot\rho^c}}{\partial \boldsymbol{\psi}_{nb}} & \frac{\partial \mathbf{h}_{\dot\rho^c}}{\partial \mathbf{v}_{en}^n} & \mathbf{0} & \frac{\partial \mathbf{h}_{\dot\rho^c}}{\partial \mathbf{b}_\omega} & \mathbf{0} & \mathbf{0} & \frac{\partial \mathbf{h}_{\dot\rho^c}}{\partial (c\delta\dot t_R)} & \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{6.33}$$

Because the multi-constellation setup only differs from the single-constellation setup in terms of receiver clock bias, but not in terms of receiver clock drift, the partial derivatives can be taken directly from *Wendel* [2007, p. 213]:

$$\frac{\partial h_{\dot\rho^c}}{\partial \boldsymbol{\psi}_{nb}} = \mathbf{u}_{as}^{n\,\mathsf{T}} \left[\hat{\mathbf{C}}_b^n \hat{\boldsymbol{\Omega}}_{eb}^b \mathbf{p}_{ba}^b \times\right] \tag{6.34}$$

$$\frac{\partial h_{\dot\rho^c}}{\partial \mathbf{v}_{en}^n} = -\mathbf{u}_{as}^{n\,\mathsf{T}} \tag{6.35}$$

$$\frac{\partial h_{\dot\rho^c}}{\partial \mathbf{b}_\omega} = -\mathbf{u}_{as}^{n\,\mathsf{T}} \hat{\mathbf{C}}_b^n \left[\mathbf{p}_{ba}^b \times\right] \tag{6.36}$$

$$\frac{\partial h_{\dot\rho^c}}{\partial \left(c\delta\dot t_R\right)} = 1. \tag{6.37}$$

Wendel [2007, p. 213] uses $\hat{\boldsymbol{\Omega}}_{ib}^b$ instead of $\hat{\boldsymbol{\Omega}}_{eb}^b$ in (6.34). Based on the aforementioned neglect of $\boldsymbol{\omega}_{ie}$, these two are identical.

6.3.3 Odometry Update

For the odometry measurement update, all derivations are first performed for one of the four wheels. Three-dimensional velocity vectors are employed to allow the usage of operations reserved for three-dimensional space (e. g. the cross product). All calculations happen analogously for the other three wheels. Before forwarding the assembled 12×1 vector to the EKF's measurement update, it is reduced to 6 × 1 by eliminating the vertical component for each wheel and forming the mean of the lateral components on each axle. The same holds for the measurement matrix, which is reduced from 12×22 to 6×22.

The total measurement vector for a single wheel $\tilde{\mathbf{z}}_o$ consists of the three-dimensional velocity vector $\mathbf{v}_{ew}^b$ output by the odometry preprocessing, while the predicted measurement vector $\hat{\mathbf{z}}_o$ is formed by transferring $\hat{\mathbf{v}}_{en}^n$ to the wheel and resolving it in the b-frame.

The measurement vector in error state formulation for a single wheel $\mathbf{z}_o$ is formed as:

$$\begin{aligned}\mathbf{z}_o &= \hat{\mathbf{z}}_o - \tilde{\mathbf{z}}_o \\ &= \quad \hat{\mathbf{v}}_{eb}^b \quad + \quad \hat{\boldsymbol{\omega}}_{eb}^b \times \mathbf{p}_{bw}^b - \mathbf{C}_b^{w\mathsf{T}} \begin{pmatrix} 1 \\ -\tan\hat{\alpha} \\ 0 \end{pmatrix} \omega_w \hat{r}_d \hat{\kappa}_x \\ &= \hat{\mathbf{C}}_b^{n\mathsf{T}} \hat{\mathbf{v}}_{en}^n + \hat{\boldsymbol{\omega}}_{eb}^b \times \mathbf{p}_{bw}^b - \mathbf{C}_b^{w\mathsf{T}} \begin{pmatrix} 1 \\ -\tan\hat{\alpha} \\ 0 \end{pmatrix} \omega_w \hat{r}_d \hat{\kappa}_x .\end{aligned} \tag{6.38}$$

All three terms depend on $\hat{\mathbf{x}}$. In analogy to the range rate update, $\boldsymbol{\omega}_{ie}$ is neglected and (6.32) holds, allowing to calculate the dependency of $\mathbf{z}_o$ on the gyroscope bias error. The other non-zero columns in the measurement matrix $\mathbf{H}_o$ are the partial derivatives w. r. t. attitude, velocity and dynamic tire radius error:

$$\mathbf{H}_o = \begin{pmatrix} \frac{\partial \mathbf{h}_o}{\partial \boldsymbol{\psi}_{nb}} & \frac{\partial \mathbf{h}_o}{\partial \mathbf{v}_{en}^n} & \mathbf{0} & \frac{\partial \mathbf{h}_o}{\partial \mathbf{b}_\omega} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\partial \mathbf{h}_o}{\partial \mathbf{r}_d} \end{pmatrix}. \tag{6.39}$$

These partial derivatives evaluate to:

$$\frac{\partial \mathbf{h}_o}{\partial \boldsymbol{\psi}_{nb}} = \hat{\mathbf{C}}_b^{n\mathsf{T}} \left[\hat{\mathbf{v}}_{en}^n \times\right] \tag{6.40}$$

$$\frac{\partial \mathbf{h}_o}{\partial \mathbf{v}_{en}^n} = \hat{\mathbf{C}}_b^{n\mathsf{T}} \tag{6.41}$$

$$\frac{\partial \mathbf{h}_o}{\partial \mathbf{b}_\omega} = \left[\mathbf{p}_{bw}^b \times\right] \tag{6.42}$$

$$\frac{\partial \mathbf{h}_o}{\partial \mathbf{r}_d} = \begin{pmatrix} \mathbf{C}_b^{w\mathsf{T}} \begin{pmatrix} -1 \\ \tan\hat{\alpha} \\ 0 \end{pmatrix} \omega_w \hat{\kappa}_x & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{6.43}$$

The non-zero entries of $\frac{\partial \mathbf{h}_o}{\partial \mathbf{r}_d}$ are shifted to the appropriate column, depending on which wheel the measurement vector $\mathbf{z}_o$ refers to.

6.3.4 Delayed Availability of Measurement Data

The three sensor types whose data are input by the localization algorithm are not synchronized, i. e. they employ different measurement intervals and take their first measurement at different times. This means that the timestamps of a total measurement vector $\tilde{\mathbf{z}}$ and its predicted counterpart $\hat{\mathbf{z}}$ do not coincide in general. To solve this issue, the approach outlined in *Steinhardt* [2014, Section 4.8] is applied. All elements necessary to compute $\hat{\mathbf{z}}$ and $\mathbf{H}$ for the three types of measurement updates as well as the corresponding IMU measurement timestamp t_i are kept in memory for a certain period of time. As a result, estimated a-priori values in the time interval $[t_{i,k-n_t}, t_{i,k}]$ are available to form $\hat{\mathbf{z}}$ and $\mathbf{H}$ after strapdown computations at time $t_{i,k}$ have been completed. $n_t + 1$ is the number of epochs to be stored. Once a new set of IMU measurements has been processed in the strapdown algorithm, the stored data with the oldest timestamp get deleted. According to Sections 6.3.1-6.3.3, the following elements have to be stored in order to compute $\hat{\mathbf{z}}$ and $\mathbf{H}$: $\hat{\mathbf{p}}_{en}^e$, $\hat{\mathbf{C}}_b^n$, $\hat{\mathbf{v}}_{en}^n$ and $\hat{\boldsymbol{\omega}}_{eb}^b$. When a new set of pseudorange, range rate or odometry measurements $\tilde{\mathbf{z}}$ with timestamp t_m is received, these stored quantities are interpolated linearly in between the two stored timestamps that are closest to t_m. If $t_m \notin [t_{i,k-n_t}, t_{i,k}]$, the measurements are discarded. As long as the errors due to linear interpolation of $\hat{\mathbf{p}}_{en}^e$, $\hat{\mathbf{C}}_b^n$, $\hat{\mathbf{v}}_{en}^n$ and $\hat{\boldsymbol{\omega}}_{eb}^b$ are acceptable, this approach produces the correct values for $\hat{\mathbf{z}}$ and $\mathbf{H}$ at the measurement time t_m. The only quantity relevant to the measurement update that is not synchronized is the a-priori error covariance matrix $\mathbf{P}^-$. The underlying assumption is that n_t is selected in such a way that the change of $\mathbf{P}^-$ during the interval $[t_{i,k-n_t}, t_{i,k}]$ is small enough to be neglected. Due to the nominal data rates of the employed sensors (IMU: 200 Hz, odometry: 100 Hz, GNSS receiver: 10 Hz), keeping $n_t + 1 = 51$ epochs in memory is sufficient to interpolate measurements within the interval $[t_{i,k} - 0.25\,\mathrm{s}, t_{i,k}]$, while the maximum nominal measurement interval is $\tau_g = 0.1\,\mathrm{s}$. Examination of the change in $\mathbf{P}$ over time verifies that the aforementioned assumption holds for such small time intervals. The only exception is the filter's transient phase after initialization, during which $\mathbf{P}$ decreases quickly. During this decrease, utilization of the value for $\mathbf{P}_k^-$ from the current epoch $t_{i,k}$ to perform a measurement update with the timestamp $t_m < t_{i,k}$ means that $\mathbf{P}^-$ is smaller than it should be. Consequently, more weight is put on the a-priori solution $\hat{\mathbf{x}}^-$ and less weight is put on the measurement vector $\tilde{\mathbf{z}}$ than it would be done if the correct value for $\mathbf{P}^-$ at the epoch t_m was used instead. As a result, the estimated error state $\delta\hat{\mathbf{x}}^+$ is smaller and the transient phase takes slightly longer than they would otherwise. This behavior is unproblematic because it cannot cause the filter to become unstable, which could be the case if the error state estimates were excessively large.

Once all measurement updates in an epoch are completed, the resulting error state $\delta\hat{\mathbf{x}}^+$ is applied to all values stored in memory. This is necessary to avoid correcting the same error in $\hat{\mathbf{x}}$ multiple times, which would result in an overcorrection and lead to a possibly larger error with the opposite sign.

6.4 Outlier Detection

Because all sensors may produce erroneous measurements, the integrated navigation solution may become corrupted if these errors are not detected. An erroneous measurement in this sense means any observation whose error is much larger than what is to be expected based on its measurement noise covariance. These erroneous measurements are also called outliers. Two types of techniques to deal with such outliers are implemented in the localization algorithm developed in this thesis. Both techniques aim to detect potential outliers and prevent their incorporation in the integrated navigation solution. This type of technique is known as *fault detection and isolation* [*Groves*, 2013, p. 701]. The first technique is innovation monitoring, the second one residual monitoring. They are similar to each other, but innovation monitoring happens prior to the integration filter's measurement update while residual monitoring is performed afterwards.

Both monitoring techniques are not applied during initialization as well as during the transient phase between initialization and normal filter operation. For these periods of time, large innovations and residuals are expected because the integration filter has not converged yet. To account for the transient phase, both monitoring techniques are disabled for an additional 5 s after the whole initialization process (see Section 6.2.1) has been completed.

6.4.1 Innovation Monitoring

The basic principle of innovation monitoring is to compare the EKF's innovation $\delta\mathbf{z}^-$ with its covariance matrix $\mathbf{\Sigma}_{\delta\mathbf{z}^-}$:

$$\delta\mathbf{z}^- = \tilde{\mathbf{z}} - \mathbf{h}(\hat{\mathbf{x}}^-) \tag{6.44}$$

$$\mathbf{\Sigma}_{\delta\mathbf{z}^-} = \mathbf{H}\mathbf{P}^-\mathbf{H}^\mathsf{T} + \mathbf{R}. \tag{6.45}$$

According to (6.44), $\delta\mathbf{z}^-$ is the negative of the EKF's measurement vector $\mathbf{z}$. Due to the sequential processing of measurement updates described in Section 6.1, it is possible that the a-priori error state estimate $\delta\hat{\mathbf{x}}^-$ is non-zero for any measurement update except the first one.

This has to be taken into account when computing the innovation. To derive the necessary modifications, the EKF's measurement update (2.36) is repeated in (6.46) and compared to the one of a total state Kalman filter with linear measurement function (6.47):

$$\delta\hat{\mathbf{x}}^{+} = \delta\hat{\mathbf{x}}^{-} + \mathbf{K}\left(\mathbf{z} - \mathbf{H}\delta\hat{\mathbf{x}}^{-}\right) \tag{6.46}$$

$$\hat{\mathbf{x}}^{+} = \hat{\mathbf{x}}^{-} + \mathbf{K}\left(\tilde{\mathbf{z}} - \mathbf{H}\hat{\mathbf{x}}^{-}\right). \tag{6.47}$$

The term inside the parentheses in (6.47) is the total state Kalman filter's innovation. Its equivalent for the error state EKF is the term inside the parentheses in (6.46). For $\delta\hat{\mathbf{x}}^{-} = \mathbf{0}$, the two have the same absolute value, but the opposite sign. This seeming contradiction is resolved by the fact that $\mathbf{H}$ and therefore $\mathbf{K}$ have an opposite sign as well in the error state EKF when compared to a total state Kalman filter. For innovation monitoring, the sign of the innovation does not matter because all errors are assumed to be zero-mean. In order to obtain an equivalent definition for the innovation if $\delta\hat{\mathbf{x}}^{-} \neq \mathbf{0}$, the negative of the term inside the parentheses in (6.46) is defined as innovation for the sequential measurement update:

$$\delta\mathbf{z}^{-} = \tilde{\mathbf{z}} - \mathbf{h}(\hat{\mathbf{x}}^{-}) + \mathbf{H}\delta\hat{\mathbf{x}}^{-}. \tag{6.48}$$

In order to obtain the correct covariance matrix for the innovation in the sequential measurement update, $\mathbf{P}^{-}$ needs to be replaced with the covariance matrix of $\delta\hat{\mathbf{x}}^{-}$ in (6.45):

$$\mathbf{\Sigma}_{\delta\mathbf{z}^{-}} = \mathbf{H}\mathbf{\Sigma}_{\delta\hat{\mathbf{x}}^{-}}\mathbf{H}^{\mathsf{T}} + \mathbf{R}. \tag{6.49}$$

For the first measurement update in each epoch, $\mathbf{P}^{-}$ and $\mathbf{\Sigma}_{\delta\hat{\mathbf{x}}^{-}}$ are identical.

Before outlier detection, $\delta\mathbf{z}^{-}$ is normalized, i. e. each element δz_j^{-} is divided by its standard deviation to form $\delta z_j^{-,*}$:

$$\delta z_j^{-,*} = \frac{\delta z_j^{-}}{\sigma_{\delta z_j^{-}}}. \tag{6.50}$$

$\sigma_{\delta z_j^{-}}$ is obtained by taking the square root of the corresponding entry on the main diagonal of $\mathbf{\Sigma}_{\delta\hat{\mathbf{x}}^{-}}$. All individual $\delta z_j^{-,*}$ are then assembled in the normalized innovation vector $\delta\mathbf{z}^{-,*}$. Each element of $\delta\mathbf{z}^{-,*}$ is standard normally distributed if all Kalman filter assumptions hold [*Groves*, 2013, p. 707]. Outlier detection is now performed on the 3σ level: If any of the elements in $\delta\mathbf{z}^{-,*}$ has an absolute value larger than three, the corresponding rows and columns are removed from $\delta\mathbf{z}^{-}$ and $\mathbf{\Sigma}_{\delta\mathbf{z}^{-}}$. The detected outliers are also removed from $\mathbf{H}$, $\mathbf{R}$ and $\mathbf{z}$. The EKF's measurement update is then performed with the remaining elements.

6.4.2 Residual Monitoring

Residual monitoring is performed in a similar manner to innovation monitoring. Instead of the innovation $\delta\mathbf{z}^{-}$, the measurement residual $\delta\mathbf{z}^{+}$ and its covariance matrix $\mathbf{\Sigma}_{\delta\mathbf{z}^{+}}$ are compared with each other:

$$\delta\mathbf{z}^{+} = \tilde{\mathbf{z}} - \mathbf{h}(\hat{\mathbf{x}}^{+}) \tag{6.51}$$

$$\mathbf{\Sigma}_{\delta\mathbf{z}^{+}} = \mathbf{R} - \mathbf{H}\mathbf{P}^{+}\mathbf{H}^{\mathsf{T}}. \tag{6.52}$$

The derivation of (6.52) is given in the appendix (Section C.2). Before performing an EKF measurement update, $\tilde{\mathbf{z}}$ and $\mathbf{R}$ are supplied by the preprocessing module. $\mathbf{z}$ and $\mathbf{H}$ are then computed from $\hat{\mathbf{x}}^{-}$ and $\tilde{\mathbf{z}}$. The measurement update produces $\delta\hat{\mathbf{x}}^{+}$ and $\mathbf{P}^{+}$. If $\delta\hat{\mathbf{x}}^{+}$ is sufficiently small to allow linearization, $\delta\mathbf{z}^{+}$ can be calculated as follows:

$$\begin{aligned}\delta\mathbf{z}^{+} &= \tilde{\mathbf{z}} - \mathbf{h}(\hat{\mathbf{x}}^{-} - \delta\hat{\mathbf{x}}^{+}) \approx \tilde{\mathbf{z}} - \left[\mathbf{h}(\hat{\mathbf{x}}^{-}) - \mathbf{H}\delta\hat{\mathbf{x}}^{+}\right]\\ &= \mathbf{H}\delta\hat{\mathbf{x}}^{+} - \left[\mathbf{h}(\hat{\mathbf{x}}^{-}) - \tilde{\mathbf{z}}\right] = \mathbf{H}\delta\hat{\mathbf{x}}^{+} - \mathbf{z}.\end{aligned} \tag{6.53}$$

Otherwise, the a-posteriori total state $\hat{\mathbf{x}}^+$ has to be computed first and (6.51) has to be applied. For the results presented in Chapter 7, the nonlinear calculation routine is applied regardless of the magnitude of $\delta\hat{\mathbf{x}}^+$. Outlier detection is performed in a similar way for residual filtering as it is for innovation filtering. Because each entry in the residual vector depends on all elements of $\mathbf{z}$, the elimination of outliers happens sequentially, i. e. only one outlier is removed at a time:

1. $\delta\mathbf{z}^+$ is normalized by dividing it by its standard deviation to form $\delta\mathbf{z}^{+,*}$:
$$\delta z_j^{+,*} = \frac{\delta z_j^+}{\sigma_{\delta z_j^+}}. \tag{6.54}$$
2. If any of the elements in $\delta\mathbf{z}^{+,*}$ has an absolute value larger than three, the rows and columns corresponding to the largest absolute value are removed from $\mathbf{z}$, $\mathbf{R}$ and $\mathbf{H}$. The EKF's measurement update is repeated with the shortened measurement vector and another set of $\delta\mathbf{z}^+$ and $\mathbf{\Sigma}_{\delta\mathbf{z}^+}$ is obtained.

This process is repeated until no element in $\delta\mathbf{z}^{+,*}$ has an absolute value larger than three or all elements have been removed.

In contrast to innovation filtering, no additional precautions due to the sequential processing of measurement updates need to be taken: After each measurement update, $\delta\hat{\mathbf{x}}^+$ contains the total amount of change that has to be applied to $\hat{\mathbf{x}}^-$ due to all measurement updates in this epoch so far combined. $\mathbf{P}^+$ describes the estimated remaining error in $\hat{\mathbf{x}}^+$ after all these measurement updates. If $|\delta\mathbf{z}^+|$ is now larger than what is to be expected based on $\mathbf{\Sigma}_{\delta\mathbf{z}^+}$, one of the measurement updates must have contained an outlier. Because all previous measurement updates have passed the residual monitoring, they did not contain any outliers. Therefore, the outlier must stem from the current measurement update.

The residual covariance matrix $\mathbf{\Sigma}_{\delta\mathbf{z}^+}$ is formed as difference of two matrices according to (6.52). $\mathbf{R}$ represents the precision of the measurements, $\mathbf{H}\mathbf{P}^+\mathbf{H}^\mathsf{T}$ represents the precision of the integrated solution. During normal operation, the integrated solution is typically much more precise than the measurements, yielding:

$$\mathbf{H}\mathbf{P}^+\mathbf{H}^\mathsf{T} \ll \mathbf{R} \tag{6.55}$$
$$\mathbf{\Sigma}_{\delta\mathbf{z}^+} \approx \mathbf{R}. \tag{6.56}$$

The comparison of the magnitude of two matrices in (6.55) is to be understood as a comparison of their singular values. There are however cases when $\mathbf{P}^+$ is so large that (6.55) does not hold. This stems from large uncertainties in the a-priori total state $\hat{\mathbf{x}}^-$. If these uncertainties get so large that the measurement covariance matrix $\mathbf{R}$ is way smaller than $\mathbf{H}\mathbf{P}^-\mathbf{H}^\mathsf{T}$, the a-posteriori state depends almost entirely on the total measurement vector $\tilde{\mathbf{z}}$. As a consequence, $\mathbf{R}$ and $\mathbf{H}\mathbf{P}^+\mathbf{H}^\mathsf{T}$ have a similar order of magnitude and their difference $\mathbf{\Sigma}_{\delta\mathbf{z}^+}$ is close to zero, indicating that there is not enough redundancy to decide whether outliers are present or not. This most commonly happens when the first pseudorange measurements are received after driving through a tunnel. The a-priori position is almost exclusively based on integrated IMU measurements and odometry corrections, while the a-priori receiver clock biases have been predicted based on the most recent estimate of the receiver clock drift. This leads to large uncertainties in these states, causing $\mathbf{\Sigma}_{\delta\mathbf{z}^+}$ to be conditioned badly. As a result, the normalization of $\delta\mathbf{z}^+$ might lead to misleading results. To avoid this, residual monitoring is omitted and the error state $\delta\hat{\mathbf{x}}^+$ is forwarded directly when $\mathbf{\Sigma}_{\delta\mathbf{z}^+}$ is conditioned badly.

Chapter 7

Results

The results from the integrated processing of GNSS, IMU and odometry data are presented in this chapter. These results stem from a variety of test scenarios, which are detailed in Section 7.1. That section also describes the equipment used to collect input data for the algorithm under test as well as the reference solution. The input data are then processed by the developed localization algorithm in various configuration, before the generated integrated solution is compared with the reference solution. This comparison is done with the help of the metrics defined in Section 5.3. The obtained results for accuracy are examined in Section 7.2, the ones concerning integrity in Section 7.3.

7.1 Test Scenarios

The localization algorithm's performance is evaluated in various test scenarios. These scenarios are chosen to reflect a broad range of operating conditions for all sensors. The main focus lies on covering diverse GNSS reception environments, since this is the variable that experiences the largest spread. To achieve these aims, four types of data sets are recorded:

1. Griesheim airfield: This former airfield is used by the TU Darmstadt for research concerning automotive engineering as well as other tasks. It features unobstructed sky view and therefore ideal GNSS reception. The test scenario is a 2 km long circuit on the airfield that is lapped five times: twice counter-clockwise, then twice clockwise and again counter-clockwise for a total of 10 km. In order to compare the linear tire model with the Magic Formula model, the test scenario is performed in two different fashions: On the one hand in a moderate driving style, with longitudinal and lateral accelerations of up to $\pm 3\,\mathrm{m\,s^{-2}}$, resulting in a total time of 11 min. On the other hand in a dynamic driving style, with longitudinal accelerations between $-10\,\mathrm{m\,s^{-2}}$ and $+5\,\mathrm{m\,s^{-2}}$, lateral accelerations of up to $\pm 6\,\mathrm{m\,s^{-2}}$ and a resulting time of 8 min.

2. Odenwald: This data set is 150 km long and lasts 3 h to 4 h. It features rural roads, forests and highways as well as towns and villages of different sizes. Especially challenging for the localization algorithm are the two tunnels in this data set: The first one is called Saukopftunnel. It has a length of 2.7 km and a transit time of 140 s. The second one is the Lohbergtunnel with a length of 1.1 km and a transit time of 75 s.

3. Darmstadt: This test scenario leads through the inner city of Darmstadt. It has a length of 35 km and a duration of 75 min to 95 min. To allow for proper initialization of the localization algorithm and the reference equipment, it starts and ends outside of the city center in areas with good GNSS reception. Challenging parts for the localization algorithm include narrow streets with multi-story buildings on both sides and two tunnels, each 550 m long and with a transit time of 50 s. The first tunnel contains two 90° turns, the second tunnel contains one 90° turn.

4. Frankfurt: This is the most challenging data set in terms of GNSS reception conditions, featuring the inner city of Frankfurt with urban canyons formed by skyscrapers. Just as set 3, this set starts and ends outside of the city center to facilitate initialization of the algorithm under test as well as the reference solution. In addition to the urban canyons, the scenario includes four different tunnels with lengths ranging from 300 m to 600 m and transit times ranging from 30 s to 45 s. Three of these tunnels are within the inner city. The data set's total length is 70 km, its duration between 95 min and 110 min.

Each of these test scenarios is repeated three times to get an extensive data basis and test the repeatability of the algorithm's performance. The only exception is scenario 2, which is only repeated twice due to its length. Additional details about the data sets and the encountered GNSS reception conditions appear in Appendix A.

Input data for the algorithm under test stem from the following sources: GNSS observations with a rate of 10 Hz from a JAVAD Triumph-LS, IMU measurements with a rate of 200 Hz from an Xsens MTi-G-700 and odometry measurements with a rate of 100 Hz from the vehicle's built-in sensors. Reference data are collected with a geodetic GNSS receiver operating at 5 Hz and a navigation grade ring laser gyroscope IMU (iMAR iNAV-RQH-1003) operating at 300 Hz. These reference data are processed together with GNSS observations from a base station in Novatel's Waypoint - Inertial Explorer software, utilizing the possibility to process the data both forward and backward in time and combine the obtained solutions. Afterwards, the solution is transferred to the origin of the navigation and body frames, i. e. the position of the Xsens MTi-G-700, by applying the appropriate lever arm.

7.2 Accuracy

All metrics concerning accuracy that were selected in Section 5.3 are evaluated in this section. Due to the large amount of test scenarios, only a subset of scenarios is featured here. The results for the remaining scenarios are listed in the appendix (Section B.4). This section starts by examining the reference solution's quality to verify that this quality is sufficient for proper evaluation of the selected metrics. Afterwards, the improvements brought forth by the developed GNSS and odometry processing methods are assessed. The main improvements through multi-frequency/multi-constellation GNSS occur in terms of position accuracy, whereas the main improvements through the novelties in odometry processing occur in terms of velocity accuracy in the body frame and yaw angle accuracy. Consequently, only the accuracy metrics concerning the respective quantities are presented in the corresponding subsections.

7.2.1 Reference Solution Quality

The reference equipment is able to provide $\check{\mathbf{v}}^b_{eb}$ and $\check{\psi}_{nb}$ with sufficiently high precision to qualify as valid reference in all data sets. This is demonstrated by the 95 % quantiles of the corresponding estimated standard deviations: Both $|\hat{\sigma}_{\check{v}^b_{eb,hor}}|_{95\,\%}$ and $|\hat{\sigma}_{\check{v}^b_{eb,U}}|_{95\,\%}$ are below $1.5\,\mathrm{cm\,s^{-1}}$ in all data sets, while $|\hat{\sigma}_{\check{\psi}_{nb}}|_{95\,\%}$ does not exceed 0.01° in any data set. Since the 95 % quantile of the horizontal velocity error $|\delta\hat{v}^b_{eb,hor}|_{95\,\%}$ reaches values as low as $4\,\mathrm{cm\,s^{-1}}$, any further improvement regarding velocity accuracy requires a better reference solution. This is not the case for the yaw angle, because its 95 % error quantile $|\delta\hat{\psi}_{nb}|_{95\,\%}$ does not sink below 0.15° in any data set. However, for the quantification of further improvements to yaw angle accuracy, the misalignment between reference IMU and the vehicle needs to be estimated, because the current mechanical setup does not allow for a sufficient repeat accuracy after the equipment is dis- and remounted.

The position accuracy of the reference solution varies among the data sets because it is mainly influenced by the GNSS reception conditions. In order to verify that the reference solution's quality is sufficient, the 95 % quantiles of its estimated standard deviation for the horizontal and vertical position error, $|\hat{\sigma}_{\breve{p}^n_{en,hor}}|_{95\,\%}$ and $|\hat{\sigma}_{\breve{p}^n_{en,U}}|_{95\,\%}$, are provided alongside the quantiles for the horizontal and vertical position error in the upcoming tables. $|\hat{\sigma}_{\breve{p}^n_{en,hor}}|_{95\,\%}$ ranges from 0.4 cm at Griesheim airfield to 23 cm in the inner city of Frankfurt, while $|\hat{\sigma}_{\breve{p}^n_{en,U}}|_{95\,\%}$ spans the interval [0.5 cm, 15 cm]. Moreover, the CDFs of $\hat{\sigma}_{\breve{p}^n_{en,hor}}$ and $\hat{\sigma}_{\breve{p}^n_{en,U}}$ are plotted together with the CDFs of $\delta\hat{p}^n_{en,hor}$ and $|\delta\hat{p}^n_{en,U}|$, respectively. From these steps, it is clear that the reference equipment is able to provide $\breve{\mathbf{p}}^e_{en}$ with sufficient quality to qualify as a valid reference, as well.

7.2.2 Improvements Through Multi-Frequency/Multi-Constellation GNSS

The inclusion of multi-frequency/multi-constellation GNSS data into the localization algorithm mainly benefits position accuracy. To demonstrate this, the results achieved by this advanced algorithm configuration are compared with the results achieved by a basic single-frequency/single-constellation configuration that inputs GNSS observations on GPS L1 C/A only. The advanced configuration uses all aspects described in Chapter 3, including the predicted half of ultra-rapid precise satellite orbits and clocks. The basic configuration uses navigation message data instead. Apart from these restrictions, the input data are identical, i. e. they stem from the exact same sensors during the exact same test run. Both configurations employ the linear tire model for slip estimation, because utilization of Pacejka's Magic Formula model in mass-market applications is unlikely due to the limited validity of its parameters under varying road and weather conditions. While the slip stiffnesses of the linear tire model are impacted by tire wear, tire pressure and wheel load (among other things) as well, they do not depend on the road surface and therefore vary slower with time than the parameters of the Magic Formula model. Comparison of the navigation solution that is obtained when the linear tire model is applied with the navigation solution that is obtained when the Magic Formula tire model is applied shows almost identical position accuracy, anyway. This holds for all collected data sets, which is why the numerical results of this comparison are omitted. One of the runs from every test scenario is presented in detail in this section, the results from the other runs appear in the appendix (Section B.4.1). Only metrics concerning position accuracy are shown, because velocity, yaw angle and dynamic tire radius accuracy remain almost unaltered in both GNSS processing configurations.

Figures 7.1-7.5 display the CDF of the horizontal and vertical position error for one run from each test scenario, respectively. Each figure features the results from the multi-frequency/multi-constellation algorithm in comparison with the results from the single-frequency GPS L1 C/A algorithm. The standard deviation of the reference, as estimated by the Inertial Explorer software, is included in each plot to verify that the reference is sufficiently more accurate than the algorithms under test. The 50 %, 95 % and 99 % quantiles of the horizontal and vertical position error for both algorithms are indicated in each figure to facilitate comparison of these important performance metrics. Two different sets of scales are employed for the horizontal axes: The first set is featured in Figures 7.1 and 7.2, which depict the results obtained at Griesheim airfield. The second set is featured in Figures 7.3-7.5, which depict the results obtained in the Odenwald, in Darmstadt and in Frankfurt, respectively. The reason for the scale adjustment is the varying range of the displayed values, which changes considerably from the first group to the second one due to the dissimilar GNSS reception conditions in the respective scenarios. The numerical values for the position accuracy metrics selected in Section 5.3 are listed in Table 7.1.

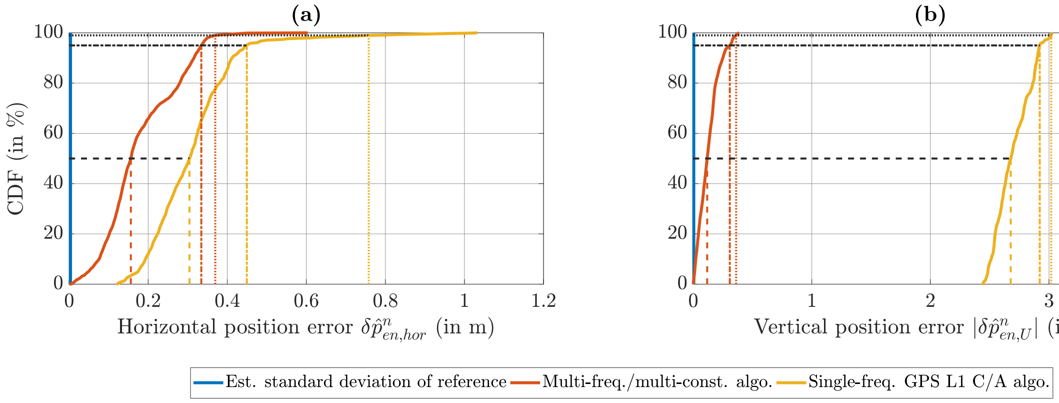

Figure 7.1: Position error CDF for Griesheim airfield data set, moderate driving style. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

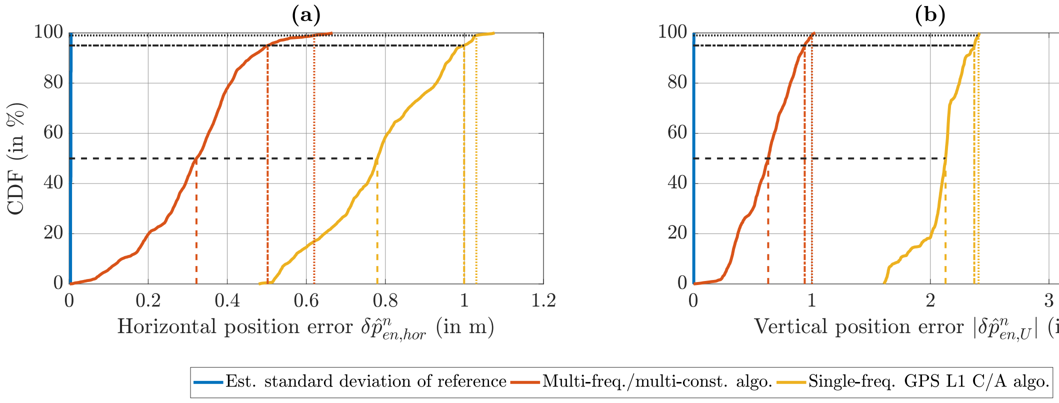

Figure 7.2: Position error CDF for Griesheim airfield data set, dynamic driving style. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

Figure 7.1 depicts the position errors for Griesheim airfield in moderate driving style, Figure 7.2 in dynamic driving style. The advanced algorithm configuration outperforms the basic configuration in both scenarios. While the quantiles for the horizontal position error are roughly halved through the inclusion of multi-frequency/multi-constellation GNSS, improvement in the vertical position error is even larger. The single-frequency algorithm exhibits a vertical bias of more than 2 m in both scenarios, presumably caused by the remaining ionospheric error after applying the Klobuchar model. As a consequence, the largest vertical error of the advanced configuration is smaller than the smallest vertical error of the basic configuration. For the moderate driving style, the minimum error of the single-frequency algorithm exceeds the maximum error of the multi-frequency algorithm by more than 2 m, which corresponds to a factor of more than five. The data displayed in Figures 7.1 and 7.2 represent the level of performance both algorithm versions are capable of under ideal GNSS reception conditions. Because the position accuracy of the single-frequency algorithm depends on the accuracy of the Klobuchar model for the specific ionospheric conditions, the performance achievable with the basic configuration is substantially more time-dependent than the performance achievable with the advanced configuration.

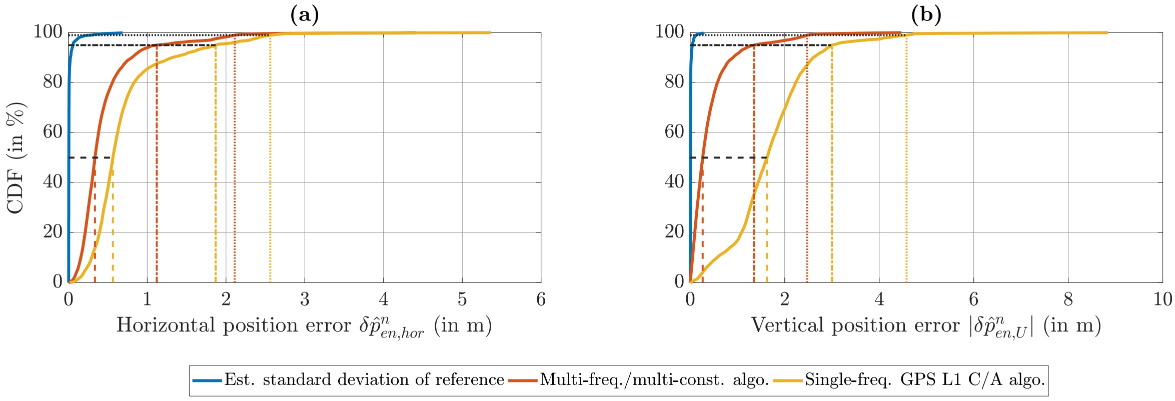

Figure 7.3: Position error CDF for Odenwald data set. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

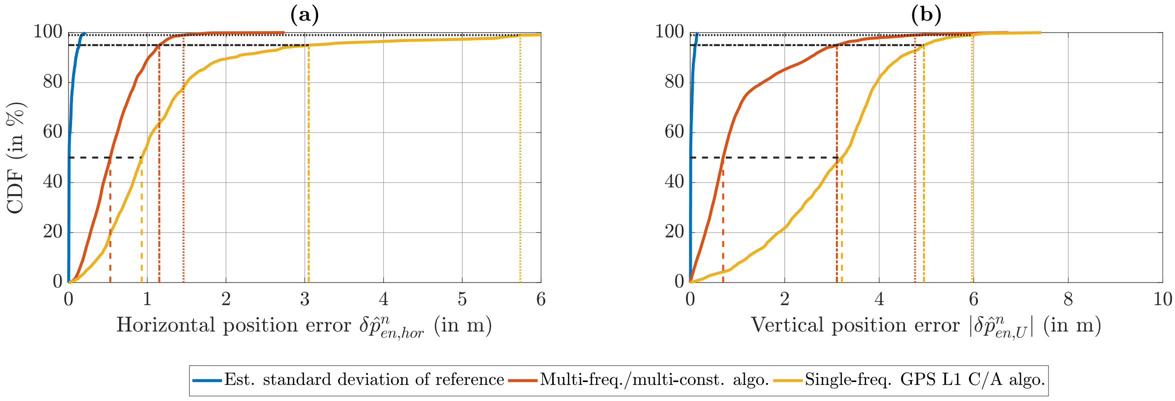

Figure 7.4: Position error CDF for Darmstadt data set. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

Figure 7.3 displays the position errors in the Odenwald data set. The declining GNSS reception conditions in comparison with the Griesheim airfield data sets show in the region between 80 % to 100 % of the CDFs. This concurs with the expectation, because most of the data set consists of roads with nearly unobstructed sky view, while the remaining areas feature dense forests, multi-story buildings and tunnels. These conditions are responsible for the bend in the CDFs above the 80 % mark and the almost horizontal portion of the curves when approaching 100 %. The multi-frequency/multi-constellation algorithm achieves better performance in horizontal and vertical direction throughout the scenario than the basic algorithm configuration, with the vertical direction showing the largest improvements.

For the scenario in the inner city of Darmstadt, which is shown in Figure 7.4, the largest improvements occur at the upper end of the horizontal error's CDF. The basic algorithm is more susceptible to large errors in pseudorange measurements from a few satellites, because the number of available satellites is lower than in the multi-constellation configuration. During stretches within urban canyons, multipath and NLOS errors cannot be identified as clearly as they can in the advanced algorithm, which can also rely on Galileo signals to stabilize position estimation. This leads to a reduction of $|\delta\hat{p}^n_{en,hor}|_{95\,\%}$ by 62 % (from 3.05 m to 1.15 m), an effect that is even more pronounced at the 99 % level, where the reduction is 75 % (from 5.74 m to 1.46 m). In contrast to the horizontal error, the major benefits of the advanced configuration in terms of vertical position accuracy occur in the lower region of the CDF, because the single-frequency algorithm's vertical position estimate is biased due to the remaining ionospheric error after application of the Klobuchar model.

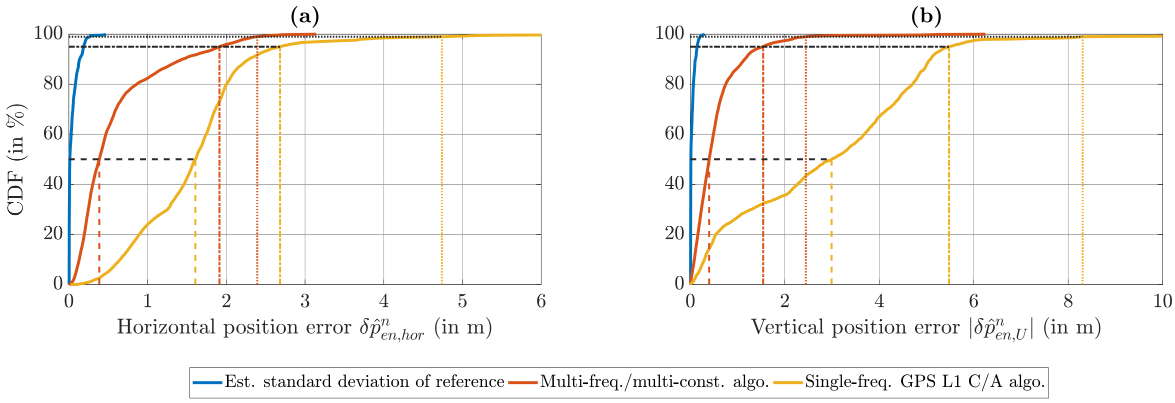

Figure 7.5: Position error CDF for Frankfurt data set. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

The position error exhibits similar characteristics in the most challenging of the test scenarios, Frankfurt inner city (depicted in Figure 7.5), as it does in the Darmstadt scenario. In this case, even the multi-frequency/multi-constellation algorithm starts to struggle because the surrounding buildings are even higher and urban canyons occur more frequently. The advanced algorithm's performance is not as good as it is in the Darmstadt scenario for horizontal position accuracy, while it is even better than in the Darmstadt scenario for vertical accuracy. The advanced algorithm still outperforms the basic configuration in all areas of the CDF, especially in vertical direction. Given the severely impaired GNSS reception conditions with long stretches spent inside urban canyons and tunnels, the overall position accuracy of the multi-frequency/multi-constellation algorithm is remarkable and shows the performance that the developed localization algorithm is capable of, due to the optimal collaboration of data from different sensors in the integration filter.

The numerical results from the presented scenarios in terms of position accuracy are summarized in Table 7.1. The multi-frequency/multi-constellation configuration achieves higher position accuracy than the single-frequency GPS L1 C/A configuration in each scenario for all evaluated metrics. The improvement is most pronounced in vertical direction. Under ideal conditions, the designed algorithm comes close to achieving the 95 % horizontal error quantile of 30 cm, which is the current target for localization algorithms in autonomous cars [*Humphreys et al.*, 2018, p. 459]. This marks the limit of what is possible with real time pseudorange positioning techniques in this sensor setup. The advanced algorithm's performance decreases under impaired GNSS conditions, but the increased number of available observations due to the multi-frequency/multi-constellation capability makes the advanced configuration more robust in these scenarios than the basic algorithm, which is limited to GPS L1 C/A observations only.

Table 7.1: Position Accuracy Metrics, Largest Improvement for Each Data Set in **Bold**. Metrics for Each Scenario: Position Error RMS (1^{st} Row), 95 % Quantile of Horizontal and Vertical Position Error (2^{nd} Row), 95 % Quantile of Estimated Reference Standard Deviation in Horizontal and Vertical Direction (3^{rd} Row)

Metric	**Unit**	**Multi-Frequency/ Multi-Constellation**	**Single-Frequency GPS L1 C/A only**
Griesheim airfield data set, moderate driving style			
RMS of $\delta\hat{\mathbf{p}}_{en}^{n}$	m	$[0.11; 0.17; 0.15]$	$[0.16; 0.29; 2.70]$
$\lvert\delta\hat{p}_{en,hor}^{n}\rvert_{95\,\%}\big/\lvert\delta\hat{p}_{en,U}^{n}\rvert_{95\,\%}$	m	0.33/**0.31**	0.45/**2.92**
$\lvert\hat{\sigma}_{\breve{p}_{en,hor}^{n}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}_{en,U}^{n}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Griesheim airfield data set, dynamic driving style			
RMS of $\delta\hat{\mathbf{p}}_{en}^{n}$	m	$[0.17; 0.29; 0.65]$	$[0.66; 0.44; 2.10]$
$\lvert\delta\hat{p}_{en,hor}^{n}\rvert_{95\,\%}\big/\lvert\delta\hat{p}_{en,U}^{n}\rvert_{95\,\%}$	m	0.50/**0.94**	1.00/**2.37**
$\lvert\hat{\sigma}_{\breve{p}_{en,hor}^{n}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}_{en,U}^{n}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Odenwald data set			
RMS of $\delta\hat{\mathbf{p}}_{en}^{n}$	m	$[0.34; 0.49; 0.65]$	$[0.39; 0.77; 1.89]$
$\lvert\delta\hat{p}_{en,hor}^{n}\rvert_{95\,\%}\big/\lvert\delta\hat{p}_{en,U}^{n}\rvert_{95\,\%}$	m	1.12/**1.35**	1.87/**3.00**
$\lvert\hat{\sigma}_{\breve{p}_{en,hor}^{n}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}_{en,U}^{n}}\rvert_{95\,\%}$	m	0.06/0.05	
Darmstadt data set			
RMS of $\delta\hat{\mathbf{p}}_{en}^{n}$	m	$[0.40; 0.53; 1.43]$	$[1.11; 1.12; 3.25]$
$\lvert\delta\hat{p}_{en,hor}^{n}\rvert_{95\,\%}\big/\lvert\delta\hat{p}_{en,U}^{n}\rvert_{95\,\%}$	m	**1.15**/3.10	**3.05**/4.95
$\lvert\hat{\sigma}_{\breve{p}_{en,hor}^{n}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}_{en,U}^{n}}\rvert_{95\,\%}$	m	0.14/0.11	
Frankfurt data set			
RMS of $\delta\hat{\mathbf{p}}_{en}^{n}$	m	$[0.39; 0.71; 0.77]$	$[1.12; 1.38; 3.53]$
$\lvert\delta\hat{p}_{en,hor}^{n}\rvert_{95\,\%}\big/\lvert\delta\hat{p}_{en,U}^{n}\rvert_{95\,\%}$	m	1.91/**1.54**	2.68/**5.48**
$\lvert\hat{\sigma}_{\breve{p}_{en,hor}^{n}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}_{en,U}^{n}}\rvert_{95\,\%}$	m	0.19/0.15	

7.2.3 Improvements Through Enhanced Odometry Processing

Processing odometry observations in the integration filter mainly serves two tasks: On the one hand, these observations offer an additional source of information about the vehicle's movement in the body frame, thereby increasing the estimation quality of $\mathbf{v}_{eb}^{b}$ and ψ_{nb}. On the other hand, they stabilize the whole estimation process in case of severely impaired GNSS conditions, e. g. in urban canyons or tunnels, thereby reducing the rate of error growth in all quantities of the integrated navigation solution, including position. While the former effect is visible continuously in every data set, the latter only occurs in surroundings with very limited sky view. At all other times, position accuracy is nearly unaffected by odometry measurements, because enough pseudorange observations are available to correct the errors in the a-priori position.

This section examines both of these odometry capabilities. The effects on estimation accuracy for velocity, yaw angle and dynamic tire radius provided by different versions of odometry processing are examined in the first part. One of the runs from every test scenario is featured here, while the others appear in the appendix (Sections B.4.2-B.4.4). The runs presented in this section are identical to the ones presented in Section 7.2.2. This section's second part deals with the positive effect that odometry observations have on position accuracy during stretches with severely obstructed satellite visibility. Throughout this whole section, the multi-frequency/multi-constellation configuration is employed for GNSS processing.

7.2.3.1 Velocity, Yaw Angle and Dynamic Tire Radius

Tables 7.2-7.4 list the evaluated accuracy metric concerning velocity, yaw angle and dynamic tire radius in all test scenarios, respectively. Four odometry configurations are compared with each other in each table. Ordered by increasing complexity, these configurations are:

- Odometry measurement update deactivated completely.
- Odometry preprocessing with neglected correlation in between the wheels. All correlation coefficients between the horizontal velocity components of the wheels are assumed to be zero. This makes it unnecessary to summarize the lateral velocities at the four wheel contact patches into one measurement for each axle. Therefore the odometry measurement vector contains eight elements in this configuration, namely longitudinal and lateral velocity from each of the four wheels. The linear tire model is employed for slip estimation. This configuration is included in the evaluation to emphasize the benefits of non-zero correlation coefficients.
- Odometry preprocessing with linear tire model as detailed in Chapter 4. The odometry measurement vector contains six elements, correlations are taken into account.
- Odometry preprocessing with Pacejka's Magic Formula tire model as detailed in Chapter 4. Apart from the tire model, this configuration is identical to the previous one.

Table 7.2 presents the selected velocity accuracy metrics. The largest improvement in each scenario occurs when the odometry update is activated, no matter what configuration is applied for odometry processing. The RMS of the lateral velocity error is reduced the most, e. g. from $34.3\,\mathrm{cm\,s^{-1}}$ to $3.2\,\mathrm{cm\,s^{-1}}$ when using the odometry model with uncorrelated measurement noise at Griesheim airfield in moderate driving style. As detailed below, this increased lateral velocity accuracy mainly stems from the fact that yaw angle observability is improved when odometry measurements are included. Since reducing the lateral error also reduces the horizontal error, $|\delta\hat{v}_{eb,hor}^{b}|_{95\,\%}$ drops as well when the odometry measurement update is activated. Under ideal GNSS reception conditions, the RMS in longitudinal direction grows slightly when odometry measurements are included. The opposite effect occurs when

Table 7.2: Velocity Accuracy Metrics. Metrics for Each Scenario: Velocity Error RMS (1st Row), 95 % Quantile of Horizontal Velocity Error (2nd Row)

Metric	Unit	Magic Formula Tire Model	Linear Tire Model	Uncorrelated Meas. Noise	Odometry Turned off
Griesheim airfield data set, moderate driving style					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.4; 2.0; 3.5]	[1.5; 2.2; 3.5]	[1.7; 3.2; 3.4]	[1.2; 34.3; 3.8]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.5	5.0	6.5	59.3
Griesheim airfield data set, dynamic driving style					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[2.8; 3.8; 6.0]	[2.9; 4.2; 6.3]	[3.3; 8.0; 7.5]	[1.4; 35.7; 5.3]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	8.5	9.1	14.9	52.1
Odenwald data set					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.7; 1.7; 3.9]	[1.7; 2.0; 3.9]	[1.9; 2.5; 3.7]	[34.2; 45.5; 4.8]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.3	4.8	6.0	53.8
Darmstadt data set					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.8; 1.8; 3.0]	[1.8; 2.0; 3.0]	[1.7; 2.1; 3.2]	[6.9; 21.9; 3.6]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	5.1	5.4	5.6	47.2
Frankfurt data set					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.8; 1.9; 3.2]	[1.8; 2.1; 3.2]	[1.7; 2.8; 3.1]	[3.2; 31.6; 4.4]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.9	5.2	6.3	66.3

GNSS reception conditions are impaired. The largest reduction happens in the Odenwald data set, where the longitudinal RMS drops from $34.2\,\mathrm{cm\,s^{-1}}$ to $\leq 1.9\,\mathrm{cm\,s^{-1}}$ when odometry updates are turned on. The unusually large RMS without odometry stems from the two long tunnels in this data set, during which the longitudinal velocity error of the solution with deactivated odometry grows to $9\,\mathrm{m\,s^{-1}}$. None of the other odometry configurations exhibits a longitudinal error of more than $10\,\mathrm{cm\,s^{-1}}$ in these tunnels.

Taking the correlation in between the wheels into account increases velocity accuracy even further. While lateral RMS and $\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$ are reduced in all scenarios, the longitudinal RMS decreases in three scenarios and increases in the other two. The biggest improvements occur for the dynamic scenario at Griesheim airfield with a high level of horizontal acceleration. Comparison of linear and Magic Formula tire model reveals that while the errors with the Magic Formula configuration are smaller for 13 out of 20 metrics and equal for the remaining seven, the improvements are small in comparison with the ones achieved through incorporation of the correlations. For all three configurations with activated odometry measurement updates, the difference in velocity accuracy brought forth by varying GNSS conditions is insignificant. The only scenario exhibiting a higher error level is the one with the dynamic driving style. This demonstrates the odometry sensors' ability to provide velocity observations with the same quality that GNSS is able to under ideal reception conditions, as long as the vehicle's horizontal accelerations remain on a moderate level. Overall, the integrated localization algorithm with GNSS, INS and odometry data achieves a high level of velocity accuracy, independent of the surrounding conditions.

Table 7.3: Yaw Angle Accuracy Metrics. Metrics for Each Scenario: Yaw Angle Error RMS (1^{st} Row), 95 % Quantile of Yaw Angle Error (2^{nd} Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Griesheim airfield data set, moderate driving style					
RMS of $\delta\hat{\psi}_{nb}$	°	0.07	0.09	0.10	1.14
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.17	0.20	0.22	1.66
Griesheim airfield data set, dynamic driving style					
RMS of $\delta\hat{\psi}_{nb}$	°	0.14	0.14	0.19	0.98
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.27	0.33	0.36	1.16
Odenwald data set					
RMS of $\delta\hat{\psi}_{nb}$	°	0.09	0.10	0.10	0.87
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.19	0.21	0.21	1.18
Darmstadt data set					
RMS of $\delta\hat{\psi}_{nb}$	°	0.22	0.23	0.22	0.99
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.47	0.48	0.48	1.36
Frankfurt data set					
RMS of $\delta\hat{\psi}_{nb}$	°	0.19	0.20	0.20	1.23
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.40	0.41	0.40	1.74

The metrics for yaw angle accuracy in Table 7.3 show similar results as the ones for velocity accuracy. Once again, the largest improvements are made when activating the odometry update. Both the incorporation of correlation into the measurement noise covariance matrix and the Magic Formula tire model offer an additional increase in accuracy, although neither of these increases is particularly large. The biggest difference in between these three odometry configurations occurs for the dynamic scenario at Griesheim airfield, demonstrating the benefits of the Magic Formula's wider application range in terms of horizontal acceleration when compared with the linear tire model. In contrast to velocity, yaw angle accuracy deteriorates when GNSS reception conditions are impaired. Driving moderately at Griesheim airfield results in the best performance, while the scenarios in the inner cities of Darmstadt and Frankfurt show the worst results. The reason for this behavior is that since velocity accuracy is assessed in the body frame, data from IMU and odometry are largely responsible for those results, while GNSS observations only play a minor role. Accurate yaw angle estimation on the other hand requires observations with high quality in both body and navigation frame. Poor GNSS reception conditions impede carrier tracking in the GNSS receiver, yielding increased tracking noise and more cycle slips, which leads to larger range rate errors. Due to signal obstructions, fewer range rate observations are available anyway. Consequently, estimation of the vehicle's velocity w. r. t. the n-frame is less accurate, which in turn results in decreased yaw angle accuracy.

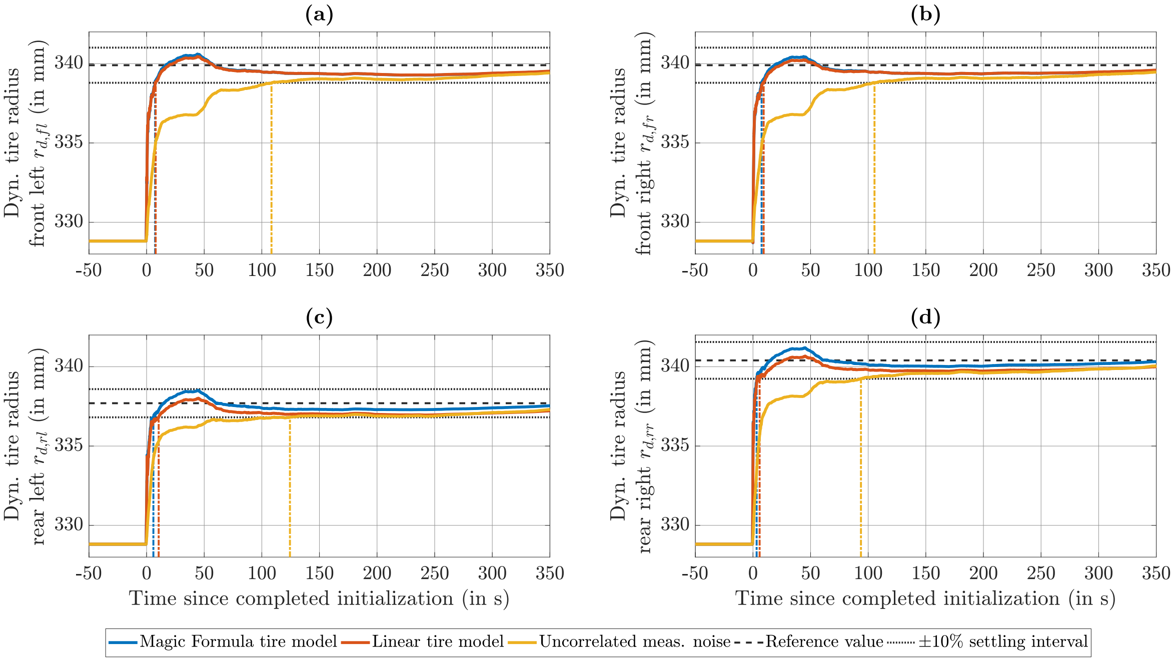

Figure 7.6: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Frankfurt data set. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

The final metric selected in Section 5.3 is $T_{s,10\,\%}$, the 10 % settling time for the dynamic tire radius. This metric quantifies how quickly the dynamic tire radii go through their transient phase and reach steady-state after initialization. One exemplary transient phase is depicted in Figure 7.6. The estimated values of r_d are shown for all four wheels and the three different odometry configurations with activated odometry update. While the initialization value is taken from the CAN bus and is identical for all tires and all scenarios, the reference value reflects each tire's true dynamic radius and differs from wheel to wheel. The ±10 % settling interval is computed from the difference between reference and initial value for each wheel. Once the vehicle has started moving and initialization is completed, all dynamic tire radii leave their initialization values and move toward the reference. When compared among the wheels, the curves for each of the configurations resemble each other closely. Moreover, the curves for Magic Formula and linear tire model with non-zero correlation are very similar to one another. Both configurations reach the lower limit of the settling interval within 10 s or less before they overshoot slightly. In the depicted scenario, the overshoot stays within the ±10 % settling interval, which is true for 13 out of the 14 total runs. In the one run in which the overshoot exceeds the settling interval (depicted in the appendix, Figure B.42), it stays below ±15 % and $T_{s,10\,\%}$ is achieved after less than 40 s for both configurations experiencing overshoot. In Figure 7.6, the configuration with uncorrelated measurement noise takes 93 s to 124 s before reaching the settling interval at the four wheels, but does not overshoot. No difference is apparent in between the three odometry configurations as far as steady-state error is concerned. Given that the difference between reference value and initialization value ranges from 8 mm to 12 mm, the initialization value of 1.67 mm for the tire radii's standard deviation appears too low. However, since all dynamic tire radii reach their settling interval after just 10 s with non-zero correlation, increasing the initial standard deviation is unnecessary. Moreover, it leads to more overshooting and worse performance overall. Due

Table 7.4: Dynamic Tire Radius Accuracy Metrics. Only Metric is 10 % Settling Time

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Griesheim airfield data set, moderate driving style					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	3.4	3.3	338	N/A
Griesheim airfield data set, dynamic driving style					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	2.7	2.7	164	N/A
Odenwald data set					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	128	164	173	N/A
Darmstadt data set					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	10	95	128	N/A
Frankfurt data set					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	7.5	10	124	N/A

to the odometry's short measurement interval and its low measurement noise, the estimated standard deviation drops quickly anyway. During the depicted time interval, it sinks from its initialization value of 1.67 mm down to 0.02 mm.

Numerical values of $T_{s,10\,\%}$ from all test scenarios are listed in Table 7.4. Since all four tire radii exhibit different settling times in general, the greatest value of the four is presented. With deactivated odometry update, no estimation of $\mathbf{r}_d$ is possible, so the respective entries are N/A. For the three configurations with activated odometry update, the tire radii's settling time shows the largest difference in between the configurations of all evaluated metrics. While settling takes at least 124 s with uncorrelated odometry measurement noise, it is accomplished in 10 s or less in seven out of ten scenarios when correlations are taken into account. The tire radii's settling time is also the metric whose results vary the most for a given configuration, both in between different test scenarios and in between different runs of the same test scenario. The reason for these large variations requires further investigation.

7.2.3.2 Position During Impaired Satellite Availability

Position accuracy is dictated by pseudorange observations. As long as a sufficient number of pseudorange measurements is available, the effect of odometry observations on the estimated position is negligible. Even deactivating the odometry update completely does not worsen position accuracy in this case. However, this changes during stretches with severely impaired satellite availability, e. g. urban canyons or tunnels. Under these circumstances, odometry observations are able to stabilize the position estimate by limiting how fast the estimates for attitude and velocity drift away from the correct values. The odometry configuration does not play a deciding role for stabilizing position estimation, all three versions described in Section 7.2.3.1 yield almost identical results. This positive effect that odometry observations have on position accuracy is demonstrated subsequently by examining position error growth in two exemplary tunnels. Two odometry configurations are compared with each other: Linear tire model with non-zero correlations on the one hand and deactivated odometry update on the other hand.

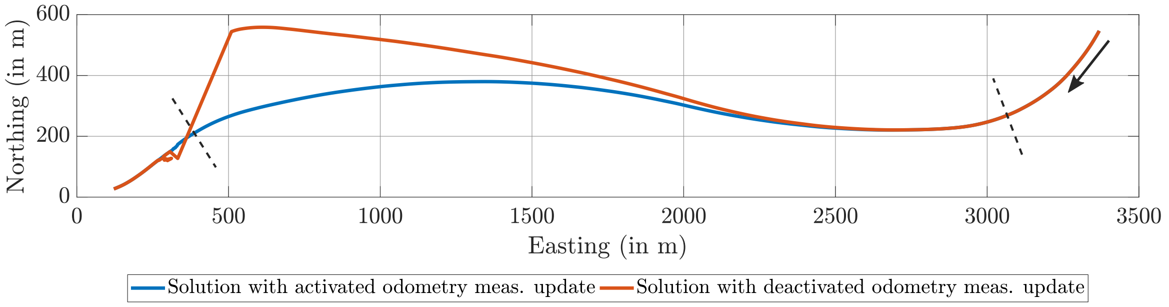

Figure 7.7: Horizontal projection of position solution during tunnel in Odenwald data set, with and without odometry aiding. Beginning and end of tunnel as well as driving direction are indicated. On this scale, the reference solution is indistinguishable from the solution with activated odometry update and hence omitted.

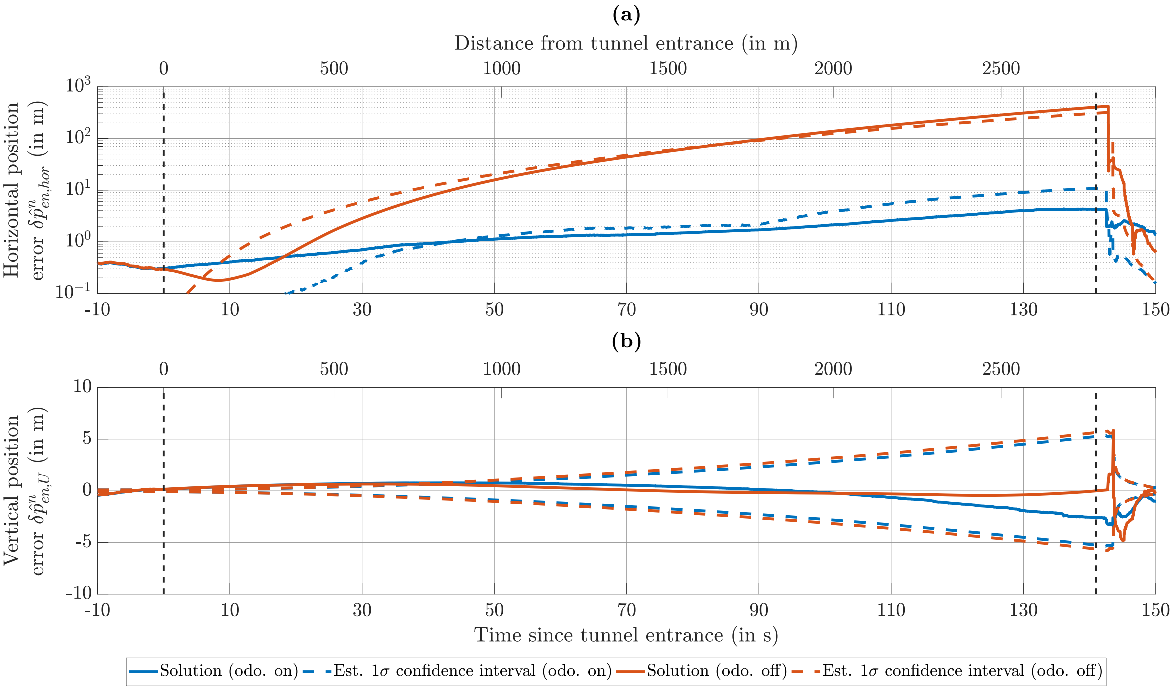

Figure 7.8: Position error during tunnel in Odenwald data set, with and without odometry aiding. Beginning and end of tunnel are indicated. **(a)** Horizontal error and its estimated 1σ confidence interval over time. **(b)** Vertical error and its estimated 1σ confidence interval over time.

Results from the first tunnel are presented in Figures 7.7 and 7.8. With a length of 2.7 km and a transit time of 140 s, this tunnel from the Odenwald data set is the longest one encountered in any of the test scenarios. As shown in Figure 7.7, the tunnel runs in east-west direction and features an elongated double bend. Since the horizontal position error $\delta\hat{p}^n_{en,hor}$ with activated odometry measurement update stays below 5 m within the depicted time interval, it cannot be distinguished from the reference solution in this figure. Therefore, the reference solution is omitted. The solution with deactivated odometry measurement update is solely based on IMU dead-reckoning once the vehicle enters the tunnel. $\delta\hat{p}^n_{en,hor}$ grows to more than 400 m during the tunnel as the IMU's measurement errors are integrated over time.

Once the first pseudorange observations are available after leaving the tunnel, the position estimate jumps toward the correct solution. The first five GNSS measurement updates after the tunnel yield jumps of 450 m, 30 m, 15 m, 10 m and 2 m, respectively.

The error's progression over time is shown in Figure 7.8. In addition to the horizontal and vertical error of the two odometry configurations, their estimated 1σ confidence intervals are plotted, as well. Due to the position aiding provided by the odometry, the horizontal error's estimated standard deviation $\hat{\sigma}_{\delta p^n_{en,hor}}$ does not exceed 11 m when the odometry update is activated, while it grows to more than 300 m without odometry aiding. No significant difference between the two odometry configurations is apparent for the vertical position error, because odometry does not provide information about the vehicle's vertical velocity. Since the vertical velocity is much smaller than the horizontal one, $\hat{\sigma}_{\delta p^n_{en,U}}$ grows slower than $\hat{\sigma}_{\delta p^n_{en,hor}}$ and never reaches more than 6 m.

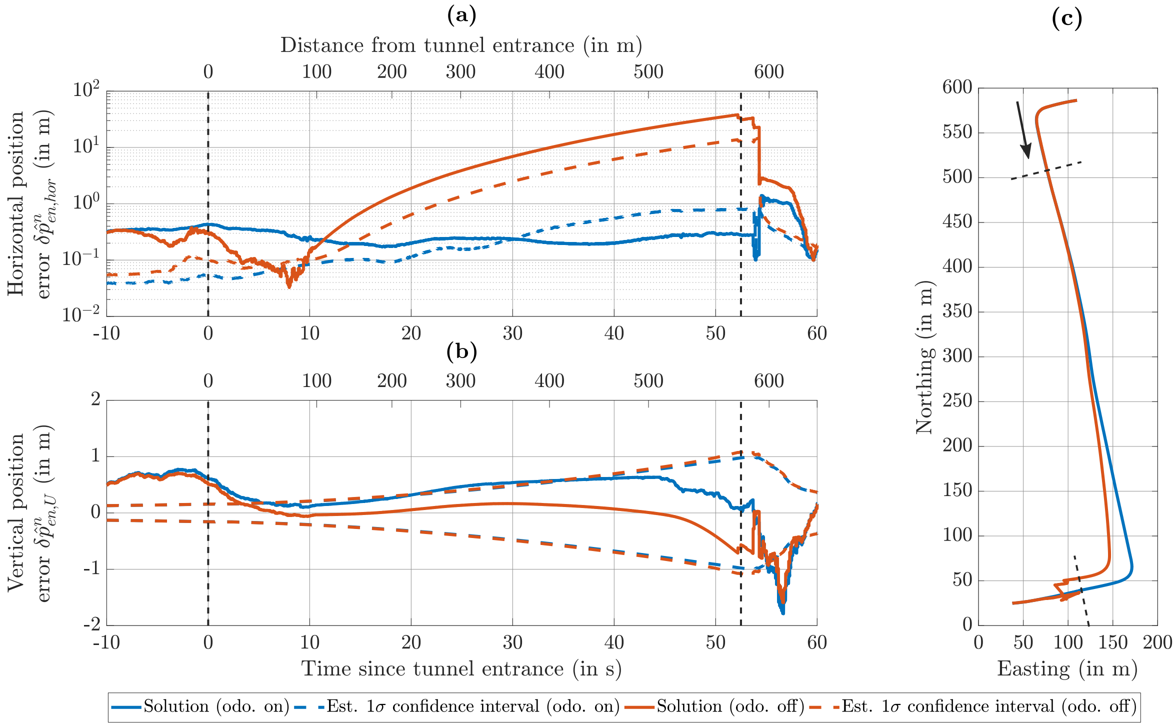

Figure 7.9: Position error during tunnel in Darmstadt data set, with and without odometry aiding. Beginning and end of tunnel are indicated in all subfigures. **(a)** Horizontal error and its estimated 1σ confidence interval over time. **(b)** Vertical error and its estimated 1σ confidence interval over time. **(c)** Horizontal projection of position solution. The arrow indicates the driving direction. On this scale, the reference solution is indistinguishable from the solution with activated odometry update and hence omitted.

Figure 7.9 displays the same plots for the second tunnel, which is located in the inner city of Darmstadt, has a length of 550 m and a transit time of 50 s. A 90° right turn occurs after 450 m, otherwise the tunnel is straight. The reference solution cannot be distinguished from the solution with activated odometry update in Figure 7.9c, so the reference solution is omitted. While $\delta\hat{p}^n_{en,hor}$ grows to 38 m without odometry aiding, it does not exceed 1.5 m with activated odometry update. The corresponding values for the estimated standard deviation are 15 m without and 0.9 m with odometry aiding, respectively. Once the first pseudorange measurements are received after the tunnel, jumps of up to 22 m occur if odometry aiding is turned off, whereas no GNSS update causes a position correction of more than 0.8 m with activated odometry update. As far as the vertical position error and its estimated standard deviation are concerned, both odometry configurations perform equally well.

7.3 Integrity

This section assesses the localization algorithm's performance in terms of integrity and to what extent the a-posteriori variance factors are able to increase this performance. Integrity is evaluated for position, velocity and yaw angle by applying the metrics selected in Section 5.3. These metrics quantify the agreement between actual and estimated error by computing the percentage of epochs for which the actual error is inside its estimated 95 % confidence interval. The numerical values for position, velocity and yaw angle integrity are listed in Table 7.5, Table 7.6 and Table 7.7, respectively. One run of each scenario is presented in every table, the results from the remaining runs appear in the appendix (Sections B.4.5-B.4.7). The runs featured in this section are identical to the ones presented in Sections 7.2.2 and 7.2.3. All results in this section stem from the algorithm configuration utilizing multi-frequency/multi-constellation GNSS and the linear tire model with non-zero measurement noise correlation. The metrics are evaluated once with the applied a-posteriori variance factors and once without them in order to demonstrate the integrity improvements brought forth by this technique of internal quality assessment.

Table 7.5: Position Integrity Metrics. Metrics for Each Scenario: Percentage of Epochs for Which Horizontal Error (1st Row) and Vertical Error (2nd Row) are Inside Their Respective 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Griesheim airfield data set, moderate driving style			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	18.4	14.4
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	63.7	56.8
Griesheim airfield data set, dynamic driving style			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	10.6	3.2
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	6.8	0.8
Odenwald data set			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	8.4	6.5
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	44.8	40.3
Darmstadt data set			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	6.5	3.4
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	27.3	19.3
Frankfurt data set			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	11.8	8.2
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	35.0	28.2

Table 7.5 shows the evaluated position integrity metrics. None of the percentages reaches its target level of 95 %, with values ranging from 3.2 % to 18.4 % in horizontal and from 0.8 % to 63.7 % in vertical direction. The generally low level of agreement between actual and estimated error demonstrates the limitations of any approach that is based on estimated standard deviations or covariance matrices: While these estimated quantities describe the error's precision, the computed error w. r. t. the reference solution describes accuracy. Because pseudoranges are the algorithm's only source of absolute position information, the integrated navigation solution is biased due to all remaining biases in the pseudoranges. These biases might stem from errors in satellite position and clock bias data, from unmodeled atmospheric delays or from multipath errors. The EKF does not account for any remaining biases that are present in its measurement input, which leads to overly optimistic values in the output covariance matrix. With the help of the a-posteriori variance factor $\hat{\sigma}_0^2$, the problem is mitigated slightly. The improvements range from 1.9 to 8.0 percentage points. Since all a-posteriori variance factors are computed from the Gauss-Markov model's residuals, biases that change slowly or not at all with time cannot be detected via $\hat{\sigma}_0^2$ either. Therefore, $\hat{\sigma}_0^2$ is not able to raise the estimated covariance high enough to account for the time correlation and remaining biases in the pseudorange errors. The localization algorithm displays an especially poor performance in terms of integrity for the vertical position error in the Griesheim airfield data set with dynamic driving style. Even with applied a-posteriori variance factor, the actual error is only inside the estimated 95 % confidence interval in 6.8 % of epochs. This is particularly interesting because the test scenario at Griesheim airfield with moderate driving style was recorded less than two hours earlier and the algorithm outputs a 95 % confidence interval that includes the actual vertical error in 63.7 % of epochs in that scenario. A possible explanation for this behavior would be that the algorithm obtains an initial position estimate outside of the confidence interval in the run with dynamic driving style and is unable to correct this estimate within the short duration of the test scenario (approx. 8 min). However, this is not the case, as is demonstrated by Figure 7.10.

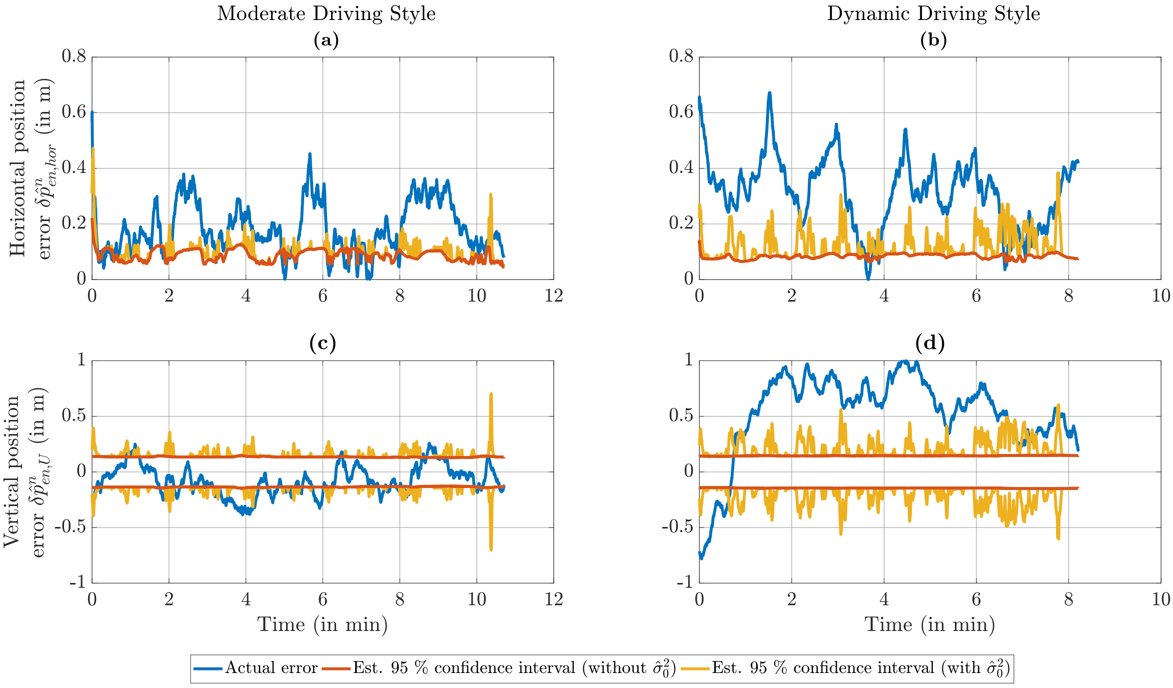

Figure 7.10: Position error and its estimated 95 % confidence interval, Griesheim airfield data set, with and without a-posteriori variance factor. **(a)** Horizontal error, moderate driving style. **(b)** Horizontal error, dynamic driving style. **(c)** Vertical error, moderate driving style. **(d)** Vertical error, dynamic driving style.

Figure 7.10 depicts the position error together with its estimated 95 % confidence interval for two test scenarios at Griesheim airfield. Figures 7.10a, c stem from the run with moderate driving style, Figures 7.10b, d from the run with dynamic driving style. The estimated confidence interval without application of the a-posteriori variance factor is very similar in both cases, reaching values around 10 cm for the horizontal and around 15 cm for the vertical error. These values would describe the actual error correctly if all pseudorange measurement noise components were uncorrelated in time, zero-mean and normally distributed, which they are not. Due to the higher velocity and acceleration in the run with dynamic driving style, position prediction is not as accurate as it is in the run with moderate driving style, yielding larger position residuals and consequently a higher a-posteriori variance factor. The actual errors are lower in the run with moderate driving style. While this is true for both horizontal and vertical error, the effect is more pronounced in vertical direction. The initial vertical position estimate in Figure 7.10d is 0.75 m too low and outside the estimated 95 % confidence interval. However, it gets corrected and matches the reference value after 45 s before it leaves the confidence interval on the other side and remains too high for the remaining run. The single-epoch GNSS position solutions exhibit a bias of −0.67 m and an empirical standard deviation of 0.59 m in vertical direction over the whole duration of the scenario. This bias is close to the initial vertical position error, but the time series of single-epoch position solutions suggests that the error should stay negative throughout the run. Why this is not the case and the error switches its sign and gets too large shortly after initialization is not clear and requires further investigation. The issue is not caused by the odometry update because the error's progression does not change when the odometry update is deactivated. A possible explanation is that the higher vibrations caused by the increased vehicle speed and acceleration increase the IMU's unmodeled errors to a level that is not reflected by the system noise covariance matrix **Q**. MEMS IMUs are known to be particularly vulnerable to these types of errors [*Groves*, 2013, p. 159].

Table 7.6: Velocity Integrity Metrics. Only Metric is Percentage of Epochs for Which the Horizontal Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Griesheim airfield data set, moderate driving style			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	51.8	46.8
Griesheim airfield data set, dynamic driving style			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	20.1	15.6
Odenwald data set			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	75.0	53.7
Darmstadt data set			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	80.4	52.3
Frankfurt data set			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	71.0	46.6

The evaluated velocity integrity metrics are listed in Table 7.6. The resulting values are generally higher than for position and lie between 15.6 % and 80.4 %. Application of the a-posteriori variance factor yields larger gains in this case, with improvements ranging from 4.5 to 28.1 percentage points. The reason for the enhanced overall performance in comparison with position integrity is that both sources of velocity information - GNSS range rates and odometry - do not exhibit any significant biases. Moreover, their error correlation time is much shorter than the one of pseudorange observations. Because two independent sources of velocity information exist, they are able to correct each other's errors, yielding a better agreement between actual and estimated velocity error. Just as for position integrity, the poorest performance is obtained in the Griesheim airfield scenario with dynamic driving style. While the horizontal velocity error is inside the estimated 95 % confidence interval in 46.6 % to 53.7 % of epochs for the other four test scenarios without applying the a-posteriori variance factor, this in only the case in 15.6 % of epochs for the Griesheim airfield scenario with dynamic driving style. Application of the a-posteriori variance factor does not increase the estimated covariance enough to achieve a level of integrity that is similar to the other scenarios, indicating that the reason for the unusually poor performance in terms of integrity stems from the high level of horizontal acceleration. This hypothesis is examined further in the subsequent paragraph.

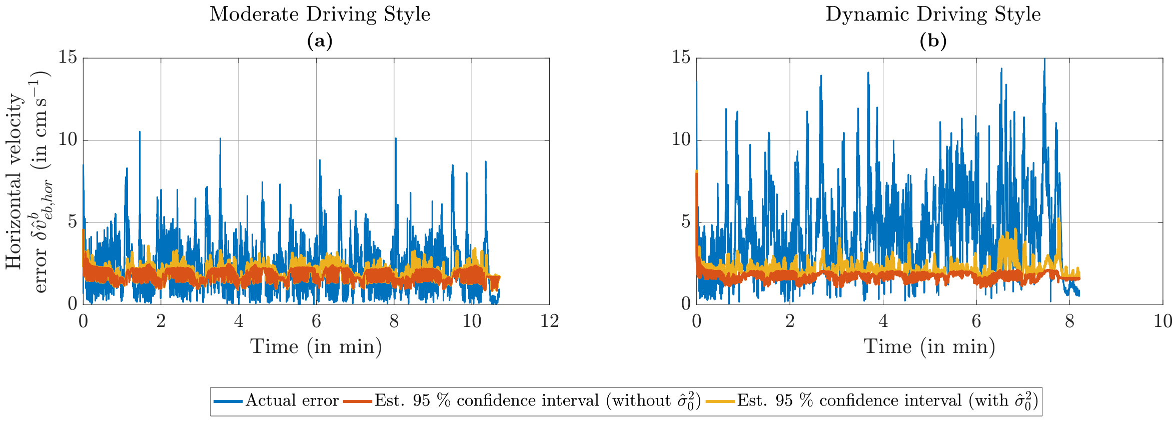

Figure 7.11: Horizontal velocity error and its estimated 95 % confidence interval, Griesheim airfield data set, with and without a-posteriori variance factor. **(a)** Moderate driving style. **(b)** Dynamic driving style.

Figure 7.11 depicts the horizontal velocity error $\delta\hat{v}^b_{eb,hor}$ together with its estimated 95 % confidence interval both test scenarios at Griesheim airfield. It is evident that velocity estimation suffers from more short-term noise than position estimation. This is due to the vehicle's vibrations that are observed by the IMU. The confidence level without $\hat{\sigma}_0^2$ is almost identical for both driving styles, varying between $1\,\mathrm{cm\,s^{-1}}$ to $2.5\,\mathrm{cm\,s^{-1}}$ once the transient phase after completed initialization is over. Similarly to the behavior for position estimation in Figure 7.10, the higher accelerations in the run with dynamic driving style cause velocity estimation to be less accurate than in the run with moderate driving style, leading to larger corrections from the measurement updates and therefore a higher a-posteriori variance factor. Moreover, the GNSS range rates are less accurate when the vehicle is subject to high accelerations, as detailed in Section 3.6.6. The reason for the decreased accuracy is that numerical differentiation is performed to compute the range rates from carrier phase observations. Consequently, the velocity obtained from range rates is an average of the velocity within the previous GNSS measurement interval of $\tau_g = 0.1\,\mathrm{s}$. This average cannot account for accelerations that are not constant within τ_g. Because the range rate's measurement noise is parametrized as a function of satellite elevation θ, higher vehicle dynamics are not reflected properly in the measurement noise covariance matrix $\mathbf{R}_{\dot{\rho}_c}$, causing a discrepancy between actual and estimated errors.

While the odometry measurement noise grows both with wheel rotation rate as well as with horizontal acceleration, the actual odometry measurement error grows simultaneously, as examined in Section 4.3. Due to the high level of horizontal acceleration in the dynamic driving style and the increased IMU noise caused by the larger vibrations of the vehicle at higher speeds, the horizontal friction coefficient is outside the linear tire model's validity limit of $|\mu| = 0.7$ in 7 % of epochs in the scenario with dynamic driving style, compared to less than 1 % of epochs in the scenario with moderate driving style. Consequently, a lot of odometry observations are disregarded by the odometry preprocessing module and not forwarded to the integration filter in the dynamic scenario, yielding the increased velocity error that is visible in Figure 7.11b. The a-posteriori variance factor is unable to account for the magnitude of this increase, resulting in the comparatively poor performance in terms of velocity integrity exhibit by the localization algorithm in the scenario with dynamic driving style.

Table 7.7: Yaw Angle Integrity Metrics. Only Metric is Percentage of Epochs for Which the Yaw Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Griesheim airfield data set, moderate driving style			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	75.5	75.1
Griesheim airfield data set, dynamic driving style			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	54.4	44.6
Odenwald data set			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	76.3	73.8
Darmstadt data set			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	67.2	59.1
Frankfurt data set			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	71.8	66.8

The final integrity metric selected in Section 5.3 concerns the yaw angle. Its numerical results are presented in Table 7.7. The yaw angle achieves the best performance in terms of integrity of all quantities selected for integrity assessment, with the actual error inside the 95 % confidence interval in 44.6 % to 76.3 % of epochs. Attitude is the only quantity of the EKF's state vector that has non-zero entries in the measurement matrices of all three types of measurement updates, indicating that all types of measurement updates contribute directly to attitude correction and not only through the attitude's correlation with the other states. However, the yaw angle is only directly observable from the combination of range rate and odometry update when the vehicle is moving, as verified by the progression of its redundancy component in Section 5.2. Both range rates and odometry yield velocity estimates without any significant biases and with short error correlation times. Consequently, their noise comes close to the EKF's model of zero-mean Gaussian white measurement noise. Therefore, the integration filter is able to filter out the noise almost as well as the EKF's model assumes, leading to an estimated yaw angle covariance that is not as overly optimistic as the one for position.

The a-posteriori variance factor yields comparatively small improvements between 0.4 and 9.8 percentage points. While all types of measurement updates have non-zero entries for attitude in their measurement matrices, none of them provides explicit information about the yaw angle. Hence, the entries in the measurement matrices concerning the yaw angle are smaller than the ones concerning the states for which the respective measurement update provides explicit information, e. g. the entries concerning position in case of the pseudorange update or the entries concerning velocity in case of the range rate update. Therefore, the yaw angle corrections are small compared to the corrections for states for which explicit information is provided by the measurement updates, leading to smaller residuals and a smaller a-posteriori variance factor. The poorest performance is once again achieved in the Griesheim airfield scenario with dynamic driving style, analogous to position and velocity integrity.

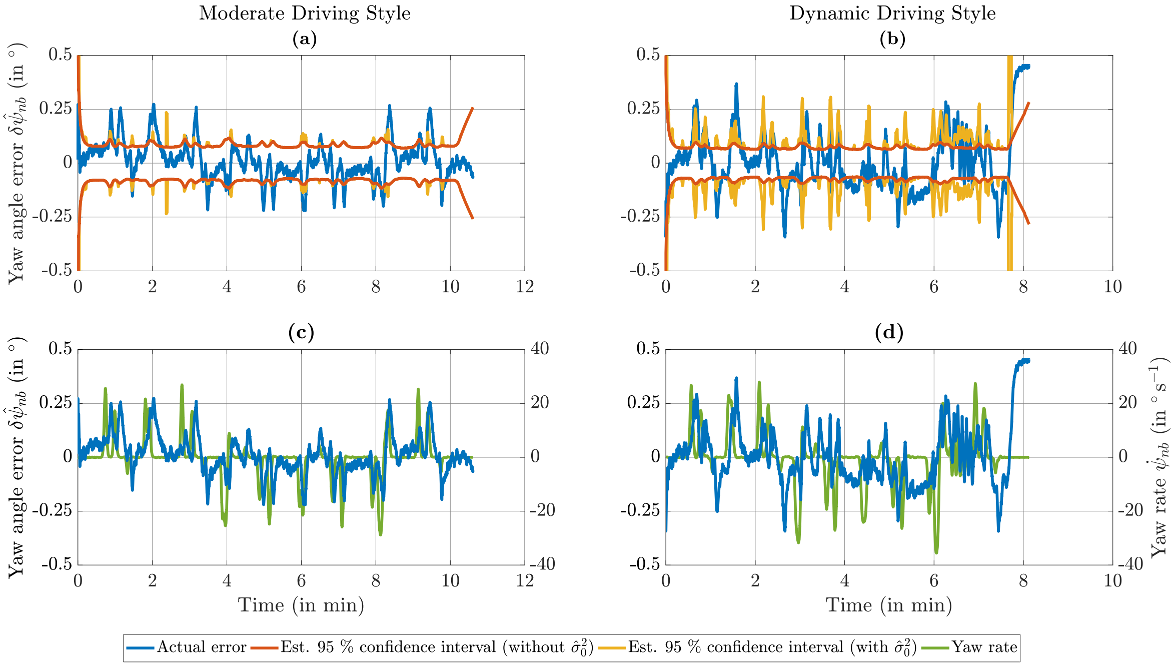

Figure 7.12: Yaw angle error, its estimated 95 % confidence interval and yaw rate, Griesheim airfield data set, with and without a-posteriori variance factor. **(a)**,**(c)** Moderate driving style. **(b)**,**(d)** Dynamic driving style.

Yaw angle error and its estimated confidence interval for two scenarios at Griesheim airfield are presented in Figures 7.12a, b. In comparison with the velocity error, which is overlaid by a lot of short-terms noise stemming from the vehicle's vibrations, both the yaw error as well as its confidence interval are much smoother. The reason for this phenomenon is that while road unevenness causes significant linear vibrations along all three body axes when the vehicle is moving, no significant rotary vibrations are induced, especially around the vertical axis. Once the vehicle starts moving and the transient phase is over, the 95 % confidence interval without $\hat{\sigma}_0^2$ stays between 0.08° and 0.12° until the car slows down to a standstill at the end of the run in Figure 7.12a, causing the confidence level to increase. Due to the higher speed in the dynamic driving style, yaw angle observability even better and the 95 % confidence interval without $\hat{\sigma}_0^2$ is slightly lower, varying from 0.06° to 0.10° in between transient phase and final braking maneuver. As demonstrated in Figures 7.12c, d, the yaw angle error is correlated with the yaw rate $\dot{\psi}_{nb}$. As long as the vehicle is traveling in a straight line, $\delta\hat{\psi}_{nb}$ remains close to zero. Left turns cause a positive yaw rate and a positive yaw angle error, right turns cause negative values for both quantities. The reason for this behavior is that the non-zero vehicle side slip angle during cornering makes distinction between direction of travel and orientation of the body frame's x-axis necessary, resulting in decreased yaw angle accuracy. Moreover, the directions of the velocity vectors $\mathbf{v}_{ea}^{n}$ at the GNSS antenna on the one hand and $\mathbf{v}_{en}^{n}$ at the IMU on the other hand are unequal during cornering, impeding yaw angle estimation even further. The combination of these effects leads to increased discrepancies between prediction and measurement, resulting in larger yaw angle residuals and a higher a-posteriori variance factor. This correlation between yaw angle error and $\hat{\sigma}_0^2$ is especially evident in the run featuring dynamic driving style, an observation that is backed up by the fact that the application of $\hat{\sigma}_0^2$ increases the percentage of epochs for which the yaw error is inside the 95 % confidence interval by 9.8 percentage points in this case. This is the greatest improvement among all test scenarios in Table 7.7.

Evaluation of the entirety of integrity metrics proves that the chosen metrics are able to assess the quality of the output covariance matrix. This assessment reveals the EKF's deficiencies when measurement and/or system noise deviate from the idealized model of zero-mean Gaussian white noise. Because the noise is not filtered out as quickly as the model assumes, the estimated covariance matrix is overly optimistic. The larger the discrepancy between model and reality gets, the greater the mismatch between actual and estimated error becomes. In this thesis, the pseudorange errors display the longest correlation times and largest biases of all noise sources, causing position to be the quantity with the poorest performance in terms of integrity. While the a-posteriori variance factors are able to increase integrity for all examined quantities, this increase is not sufficient to reach an estimated 95 % confidence level that covers the actual error in 95 % of epochs in any of the evaluated test scenarios. A possible approach to fix this problem is to account for the errors' time correlation, e. g. by raising the noise covariance dependent on the respective error's correlation time or by adding prefilters to whiten the noise. However, an increased measurement noise covariance hampers outlier detection, possibly leading to worse accuracy.

While the technique of a-posteriori variance factors has proven to be insufficient to yield a level of integrity that is adequate for safety-critical systems, the redundancy components obtained in order to compute $\hat{\sigma}_0^2$ are suitable to assess how well any subset of elements from the EKF's state vector is observable based on the measurement input at a given time, as shown in Section 5.2. This is an important aspect of state vector observability, although a thorough observability analysis needs to take the IMU data and the system model into account, as well. With the depicted technique of merging pseudorange, range rate and odometry updates into one Gauss-Markov model to form redundancy components, observability is quantified purely on an epoch-by-epoch basis, leaving out the system dynamics and the EKF's ability to observe states by inferring from the progression of measurements over longer periods of time.

Chapter 8

Conclusion and Outlook

This final chapter consists of two eponymous sections. In the first one, the results of the thesis are summarized and analyzed in the context of the aims defined in the introduction. Based on this analysis, conclusions are drawn about which elements of the developed localization algorithm already achieve satisfactory performance and which elements still need to be improved. The second section gives an outlook on potential next steps to enhance the algorithm's performance within the limits of the chosen approach as well as possible expansions beyond these limits.

8.1 Conclusion

This thesis aims to make additional GNSS constellations and frequencies beyond GPS L1 usable for automotive applications by including them in an integrated GNSS/INS localization algorithm. While the main advantages are increased satellite availability and the elimination of the ionospheric error, modern GNSS signals offer additional benefits like reduced susceptibility to multipath errors, higher signal transmission power and faster signal acquisition via pilot components free of navigation data. In order to achieve a high degree of flexibility w. r. t. the available pseudoranges, the GNSS preprocessing is able to work with different types of IF combinations and single-frequency pseudoranges simultaneously. This capability is especially important for GPS processing because the currently broadcast signals vary from satellite to satellite due to the ongoing stepwise modernization of the GPS constellation. Since each signal is affected by different biases, their simultaneous processing requires the consideration of DCBs. While satellite-specific DCBs are obtained from external sources, receiver-specific DCBs are calibrated a-priori for the employed receiver. To account for the signal biases arising from the multi-constellation approach, an additional receiver clock bias entry is included in the integration filter's state vector. Satellite orbit and clock data are obtained from the predicted half of ultra-rapid precise products published by GFZ instead of the navigation messages, which serve as backup. This preserves the algorithm's real time capability while reducing the satellite position and clock bias error, which is the largest remaining error after the ionospheric delay has been eliminated through multi-frequency observations. A very important part of the developed GNSS preprocessing module is the measurement noise model. Due to the inherently different error characteristics of IF combinations on the one hand and single-frequency pseudoranges on the other hand, a sophisticated measurement noise model is necessary to enable optimal weighting of all types of input data in the integration filter.

With the help of selected metrics for position accuracy, the improvements brought forth by multi-frequency/multi-constellation GNSS in comparison with conventional single-frequency GPS L1 C/A integration are quantified. The algorithm developed in this thesis achieves consistently higher accuracy than the single-frequency algorithm throughout all recorded test scenarios. These test scenarios feature varying GNSS conditions, ranging from unobstructed sky view at an airfield to severely impaired reception inside urban canyons. The largest advancements include the reduction of the 95 % horizontal error quantile from 3.05 m to 1.15 m for a scenario in the inner city of Darmstadt and the reduction of the 95 % vertical error quantile from 2.92 m to 0.31 m for a scenario at Griesheim airfield.

In addition to data from GNSS receiver and IMU, observations from the vehicle's built-in odometry sensors are input by the integration filter. Wheel rotation rates from all four wheels as well as the steering wheel angle are preprocessed to obtain the horizontal velocity vector at each of the four wheel contact patches. During odometry preprocessing, longitudinal and lateral tire slip are compensated with the help of tire models with varying complexity in conjunction with a single-track model. Due to the fact that all tire slip estimates are based on the same IMU acceleration measurements, the velocity errors are correlated in between the wheels. Correlation in lateral direction among the wheels of each axle is particularly severe, so these four measurements are merged to form two new observations, each representing the lateral velocity at one of the axles. Moreover, non-zero off-diagonal elements are included in the noise covariance matrix to account for the correlation in the entries of the resulting 6×1 odometry measurement vector. The improvements brought forth by enhanced odometry processing are twofold: One the one hand, lateral velocity and yaw angle accuracy increase because an additional source of information about the vehicle's movement in the body frame is available. On the other hand, position accuracy is improved in surroundings with severely impaired GNSS reception, e. g. urban canyons or tunnels. These improvements are quantified with the help of selected metrics, revealing that the inclusion of correlation coefficients in between the wheels reduces lateral velocity and yaw angle error noticeably. In contrast, switching from a basic linear tire model to a more advanced Magic Formula tire model yields only minor improvements. Hence, the Magic Formula's increased parametrization effort is not justified. Moreover, its parameters depend on road surface and weather, thereby limiting its generality.

For safety-critical systems, performance metrics beyond accuracy are required. A necessity to achieve satisfactory integrity performance is the availability of an error estimate that adequately models the actual error. This thesis examines if the concept of a-posteriori variance factors is able to derive such an adequate error estimate by monitoring the agreement between the integration filter's measurement inputs and their attributed covariance matrices. The results demonstrate that this is not the case, as the actual error falls within the estimated 95 % confidence interval in only 6.5 % to 80.4 % of epochs, even after applying the a-posteriori variance factors. Integrity performance is especially poor for position due to the fact that Kalman filters tend to deliver overly optimistic error estimates if the input errors are biased and/or highly correlated in time, both of which are true for pseudorange observations. Although the utilization of IF combinations eliminates the bias stemming from the unmodeled ionospheric delay, the remaining errors account for enough bias and time correlation to yield a biased position estimate in the integrated navigation solution. Consequently, describing the solution's precision via its covariance matrix cannot adequately monitor its accuracy, leading to poor integrity performance.

8.2 Outlook

While the work carried out in the context of this thesis increases the localization algorithm's performance significantly, room for improvement and areas requiring further investigation still exist. Enhancing the algorithm's performance may be carried out within the limits of the approach chosen in this thesis (i. e. without applying carrier phase positioning methods or other techniques requiring a steady stream of external input data) or outside of these limits.

Possible expansions beyond the limits of this thesis' approach are manifold, so only a selection is mentioned here. Positioning techniques based on carrier phase observations, such as RTK and PPP, offer position accuracy on the centimeter level. However, additional research is necessary to improve their reliability in surroundings with impaired GNSS reception conditions, especially for kinematic applications. Vector tracking schemes or deep GNSS/INS integration are potential solutions on the way to an algorithm that minimizes the likelihood of losing ambiguity fixes while simultaneously reducing the time for refixing the ambiguities in case the fix is lost nonetheless. Another option for satisfying the accuracy requirements of autonomously driving cars is the inclusion of data from additional sensors (e. g. radar, cameras, lidar) into the integration filter.

Several enhancements within the limits set for this thesis exist, as well. An obvious one is the inclusion of GNSS observations from constellations beyond GPS and Galileo, i. e. GLONASS and BeiDou, thereby raising satellite availability even more. The current design does not utilize all data provided by the receiver. Only two pseudoranges and one carrier phase measurement per satellite are processed in each epoch, neglecting one pseudorange and two carrier phase measurements in epochs for which the receiver tracks signals of a satellite on three frequencies. These neglected observations could be used for purposes like cycle slip detection or outlier elimination.

As far as odometry processing is concerned, the single-track model with linear tire characteristics has proven to be sufficient for dynamics that usually occur in everyday road traffic. However, the odometry preprocessing module does currently not have the capability to account for variations in the tire slip stiffnesses, which have to be parametrized a-priori. Because these stiffnesses change over time, e. g. due to tire wear and variations in tire pressure, a method for estimating them during the localization algorithm's normal operation is required in order to maintain the current performance level. The odometry module's lateral velocity observations are very noisy due to the vibrations induced by road unevenness. Estimating the side slip angles based on gyroscope measurements avoids this problem and reduces the error's RMS by 75 %, but decreases the accuracy of the integrated navigation solution. Developing a method for tire slip compensation that does not rely on IMU measurements has the potential to overcome these issues and yield better performance of the odometry module in particular as well as the integrated algorithm in general.

Because the concept of a-posteriori variance factors is unable to deliver realistic error estimates, other techniques have to be considered to achieve the level of integrity required by safety-critical applications. One possible approach is error overbounding. Since error distribution functions derived with overbounding techniques impede outlier detection, their application might worsen the integrated solution's accuracy. In order to avoid this, each input may be assigned two different error distribution function. On the one hand a nominal error model, e. g. one of the models derived in this thesis, which is to be used for outlier detection and computation of the integrated solution. On the other hand an overbounding error model, which is to be used for obtaining protection levels for this integrated solution.

When envisioning the application of the developed algorithm as a source of input data for a vehicle's motion controller, additional capabilities have to be included. The most important one is the ability to perform all operations in real time inside the vehicle without the need to record data and process them later. Also, the algorithm needs to be able to store the current solution in memory when the vehicle gets parked and reload it when the vehicle is started again in order to avoid the initialization process, enabling the algorithm to output a complete navigation solution right away. A possibility to reinitialize the integration filter during normal operation has to implemented, as well. This function is activated in case the estimated total state vector drifts away too far from the truth, which causes too many measurements to be rejected by outlier detection and therefore makes it impossible for the integration filter to reconverge to the correct values.

References

Adjrad, M., P. D. Groves, J. C. Quick, and C. Ellul, Performance assessment of 3D –mapping–aided GNSS part 2: Environment and mapping, *NAVIGATION (Journal of the Institute of Navigation)*, *66*(2), 363–383, 2019.

Arribas, J., A. Moragrega, C. Fernández-Prades, and P. Closas, Low-cost GNSS/INS/Odometric Sensor Fusion Platform for Ground Intelligent Transportation Systems, in *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2017*, pp. 436–455, Institute of Navigation, 2017.

Betz, J. W., *Engineering Satellite-Based Navigation and Timing: Global Navigation Satellite Systems, Signals, and Receivers*, Wiley, Hoboken, New Jersey, 2016.

Bevly, D. M., and S. Cobb, *GNSS for Vehicle Control*, Artech House, Boston, Massachusetts, 2010.

Bevly, D. M., J. Ryu, and J. C. Gerdes, Integrating INS Sensors With GPS Measurements for Continuous Estimation of Vehicle Sideslip, Roll, and Tire Cornering Stiffness, *IEEE Transactions on Intelligent Transportation Systems*, *7*(4), 483–493, 2006.

Blanch, J., T. Walter, P. Enge, J. Burns, M. Mabilleau, I. Martini, J. P. Boyero, and G. Berz, A Proposed Concept of Operations for Advanced Receiver Autonomous Integrity Monitoring, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 1084–1090, Institute of Navigation, 2018.

Boehm, J., R. Heinkelmann, and H. Schuh, Short Note: A global model of pressure and temperature for geodetic applications, *Journal of Geodesy*, *81*(10), 679–683, 2007.

Cohenour, C., and F. van Graas, GPS Orbit and Clock Error Distributions, *NAVIGATION (Journal of the Institute of Navigation)*, *58*(1), 17–28, 2011.

COST Action TU1302, *SaPPART Handbook: Assessment of positioning performance in ITS applications*, Techniques et méthodes, TMI 2, IFSTTAR, Marne-la-Vallée, 2017.

DeCleene, B., Defining Pseudorange Integrity - Overbounding, in *Proceedings of the 13th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GPS) 2000*, pp. 1916–1924, Institute of Navigation, 2000.

Deng, Z., M. Fritsche, T. Nischan, and M. Bradke, *Multi-GNSS Ultra Rapid Orbit-, Clock- & EOP-Product Series*, GFZ Data Services, `http://pmd.gfz-potsdam.de/panmetaworks/showshort.php?id=escidoc:1739888`, 2016.

Duong, V., K. Harima, S. Choy, D. Laurichesse, and C. Rizos, An optimal linear combination model to accelerate PPP convergence using multi-frequency multi-GNSS measurements, *GPS Solutions*, *23*(2), 49, 2019.

Elsheikh, M., W. Abdelfatah, A. Wahdan, and Y. Gao, Low-cost PPP/INS Integration for Continuous and Precise Vehicular Navigation, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 3169–3178, Institute of Navigation, 2018.

European GNSS Service Center, *Galileo Constellation Status*, European Global Navigation Satellite Systems Agency, `https://www.gsc-europa.eu/system-status/Constellation-Information`, 2019.

Feng, Y., and Y. Zheng, Efficient interpolations to GPS orbits for precise wide area applications, *GPS Solutions*, *9*(4), 273–282, 2005.

Fernández Hernández, I., Galileo High Accuracy Service and its importance for mobility applications, in *Proceedings of the Geodätische Woche 2018*, Gesellschaft für Geodäsie, Geoinformation und Landmanagement (DVW), 2018.

Galileo OS-SIS-ICD, *Galileo Open Service Signal-in-Space Interface Control Document: Issue 1, Revision 3*, European Global Navigation Satellite Systems Agency, 2016.

Gao, J., M. G. Petovello, and M. E. Cannon, Development of Precise GPS/INS/Wheel Speed Sensor/Yaw Rate Sensor Integrated Vehicular Positioning System, in *Proceedings of the National Technical Meeting of the Institute of Navigation (ION NTM) 2006*, pp. 780–792, Institute of Navigation, 2006.

Gross, D., W. Hauger, J. Schröder, and W. A. Wall, *Technische Mechanik 3*, 14th ed., Springer, Berlin/Heidelberg, 2019.

Groves, P. D., *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed., Artech House, Boston/London, 2013.

Groves, P. D., and M. Adjrad, Performance assessment of 3D– mapping–aided GNSS part 1: Algorithms, user equipment, and review, *NAVIGATION (Journal of the Institute of Navigation)*, *66*(2), 341–362, 2019.

Gunning, K., T. Walter, and P. Enge, Multi-GNSS Constellation Anomaly Detection and Performance Monitoring, in *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2017*, pp. 1051–1062, Institute of Navigation, 2017.

Gunning, K., J. Blanch, T. Walter, L. de Groot, and L. Norman, Design and Evaluation of Integrity Algorithms for PPP in Kinematic Applications, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 1910–1939, Institute of Navigation, 2018.

Hahn, J. H., and E. D. Powers, Implementation of the GPS to Galileo time offset (GGTO), in *Proceedings of the 37th Annual Precise Time and Time Interval Systems and Applications Meeting (PTTI) 2005*, pp. 33–37, IEEE/Institute of Navigation, 2005.

Harrer, M., and P. Pfeffer (Eds.), *Steering Handbook*, Springer, Cham/Heidelberg/London, 2017.

Hewitson, S., and J. Wang, Extended Receiver Autonomous Integrity Monitoring (eRAIM) for GNSS/INS Integration, *Journal of Surveying Engineering*, *136*(1), 13–22, 2010.

Hopfield, H. S., Tropospheric Refraction Effects on Satellite Range Measurements, *APL Technical Digest*, *11*(4), 11–19, 1972.

Hsu, L.-T., S.-S. Jan, P. D. Groves, and N. Kubo, Multipath mitigation and NLOS detection using vector tracking in urban environments, *GPS Solutions*, *19*(2), 249–262, 2015.

Humphreys, T. E., M. Murrian, and L. Narula, Low-cost precise vehicular positioning in urban environments, in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (IEEE/ION PLANS) 2018*, pp. 456–471, IEEE/Institute of Navigation, 2018.

IGS, *RINEX: The Receiver Independent Exchange Format: Version 3.04*, International GNSS Service (IGS), RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104), 2018.

IS-GPS-200, *Interface Specification IS-GPS-200: NAVSTAR GPS Space Segment/Navigation User Segment Interfaces: Revision J*, Global Positioning Systems Directorate, 2018.

Joerger, M., and B. Pervan, Exploiting Satellite Motion in ARAIM: Measurement Error Model Refinement Using Experimental Data, in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2016*, pp. 1696–1712, Institute of Navigation, 2016.

Kaplan, E. D., and C. J. Hegarty (Eds.), *Understanding GPS/GNSS: Principles and Applications*, 3rd ed., Artech House, Boston/London, 2017.

Katriniok, A., *Optimal Vehicle Dynamics Control and State Estimation for a Low-Cost GNSS-based Collision Avoidance System: Dissertation (RWTH Aachen, 2013)*, *Fortschrittberichte VDI : Reihe 8, Mess-, Steuerungs- und Regelungstechnik*, vol. 1230, VDI-Verlag, Düsseldorf, 2014.

Kazmierski, K., K. Sośnica, and T. Hadas, Quality assessment of multi-GNSS orbits and clocks for real-time precise point positioning, *GPS Solutions*, *22*(1), 11, 2018.

Khanafseh, S., B. Kujur, M. Joerger, T. Walter, S. Pullen, J. Blanch, K. Doherty, L. Norman, L. de Groot, and B. Pervan, GNSS Multipath Error Modeling for Automotive Applications, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 1573–1589, Institute of Navigation, 2018.

Kouba, J., *A Guide to Using International GNSS Service (IGS) Products*, International GNSS Service (IGS), `http://acc.igs.org/UsingIGSProductsVer21.pdf`, 2009.

Laurichesse, D., and A. Blot, Fast PPP Convergence Using Multi-constellation and Triple-frequency Ambiguity Resolution, in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2016*, pp. 2082–2088, Institute of Navigation, 2016.

Li, T., M. G. Petovello, G. Lachapelle, and C. Basnayake, Ultra-tightly Coupled GPS/Vehicle Sensor Integration for Land Vehicle Navigation, *NAVIGATION (Journal of the Institute of Navigation)*, *57*(4), 263–274, 2010.

Li, X., and H.-J. Götze, Ellipsoid, geoid, gravity, geodesy, and geophysics, *GEOPHYSICS*, *66*(6), 1660–1668, 2001.

Liu, S., F. Sun, L. Zhang, W. Li, and X. Zhu, Tight integration of ambiguity-fixed PPP and INS: Model description and initial results, *GPS Solutions*, *20*(1), 39–49, 2016.

Lunze, J., *Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen*, 11th ed., Springer, Berlin/Heidelberg, 2016.

Medina, D., K. Gibson, R. Ziebold, and P. Closas, Determination of Pseudorange Error Models and Multipath Characterization under Signal-Degraded Scenarios, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 3446–3456, Institute of Navigation, 2018.

Mongrédien, C., A. Parkins, C. Hide, M. Ström, and D. Ammann, Multi-Band Multi-GNSS RTK for Mass-Market Applications, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 580–595, Institute of Navigation, 2018.

Montenbruck, O., and A. Hauschild, Code Biases in Multi-GNSS Point Positioning, in *Proceedings of the International Technical Meeting of The Institute of Navigation (ION ITM) 2013*, pp. 616–628, Institute of Navigation, 2013.

Montenbruck, O., and R. Langley, *CNAV Test Data Set*, National Aeronautics and Space Administration, `ftp://ftp.cddis.eosdis.nasa.gov/gnss/data/campaign/cnav/2013/06/aaaReadme.txt`, 2013.

Montenbruck, O., A. Hauschild, and P. Steigenberger, Differential Code Bias Estimation using Multi-GNSS Observations and Global Ionosphere Maps, *NAVIGATION (Journal of the Institute of Navigation)*, *61*(3), 191–201, 2014.

Montenbruck, O., P. Steigenberger, and A. Hauschild, Broadcast versus precise ephemerides: A multi-GNSS perspective, *GPS Solutions*, *19*(2), 321–333, 2015.

Montenbruck, O., P. Steigenberger, L. Prange, Z. Deng, Q. Zhao, F. Perosanz, I. Romero, C. Noll, A. Stürze, G. Weber, R. Schmid, K. MacLeod, and S. Schaer, The Multi-GNSS Experiment (MGEX) of the International GNSS Service (IGS) – Achievements, prospects and challenges, *Advances in Space Research*, *59*(7), 1671–1697, 2017.

Moritz, H., Geodetic reference system 1980, *Bulletin Géodésique*, *54*(3), 395–405, 1980.

Navigation Center, *GPS Constellation Status*, United States Coast Guard, `https://www.navcen.uscg.gov/?Do=constellationStatus`, 2019.

Niemeier, W., *Ausgleichungsrechnung: Statistische Auswertemethoden*, 2nd ed., de Gruyter, Berlin, 2008.

Odijk, D., and P. J. G. Teunissen, Characterization of between-receiver GPS-Galileo inter-system biases and their effect on mixed ambiguity resolution, *GPS Solutions*, *17*(4), 521–533, 2013.

Olynik, M., M. G. Petovello, M. E. Cannon, and G. Lachapelle, Temporal impact of selected GPS errors on point positioning, *GPS Solutions*, *6*(1-2), 47–57, 2002.

Pacejka, H. B., *Tyre and vehicle dynamics*, 2nd ed., Butterworth-Heinemann, Oxford, 2006.

Paziewski, J., and P. Wielgosz, Accounting for Galileo–GPS inter-system biases in precise satellite positioning, *Journal of Geodesy*, *89*(1), 81–93, 2015.

Perea, S., M. Meurer, I. Martini, M. Rippl, M. Joerger, and B. Pervan, Nominal Range Error Analysis to Support ARAIM, in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2016*, pp. 1726–1735, Institute of Navigation, 2016.

Peyret, F., Standardization of performances of GNSS-based positioning terminals for ITS applications at CEN/CENELEC/TC5, in *Proceedings of the Intelligent Transportation Systems World Congress 2013*, ITS AsiaPacific, 2013.

Pischinger, S., and U. Seiffert (Eds.), *Vieweg-Handbuch Kraftfahrzeugtechnik*, 8th ed., Springer Vieweg, Wiesbaden, 2016.

Prieto-Cerdeira, R., J. Samson, and R. Orús Pérez, Generative Statistical Multipath Error Model for Permanent Stations, in *Proceedings of the 7th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC) 2014*, IEEE, 2014.

Pullen, S., Quantifying the performance of navigation systems and standards for assisted-GNSS, *InsideGNSS*, *3*(6), 20–23, 2008.

Pullen, S., J. Kilfeather, J. Goddard, T. Nowitzky, B. Shah, W. Doong, A. Welton, D. Kagan, and K. Greer, Enhanced Navigation, Robustness, and Safety Assurance for Autonomous Vehicles as Part of the Globalstar Connected Car Program, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 1538–1565, Institute of Navigation, 2018.

Rajamani, R., *Vehicle Dynamics and Control*, Springer, New York, New York, 2006.

Rippl, M., S. Perea, S. Schlüter, and C. Günther, Global Integrity Combining ARAIM and SBAS Method and Operational Benefit, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 2598–2608, Institute of Navigation, 2018.

Salmon, D. C., and D. M. Bevly, An exploration of low-cost sensor and vehicle model Solutions for ground vehicle navigation, in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (IEEE/ION PLANS) 2014*, pp. 462–471, IEEE/Institute of Navigation, 2014.

Salos, D., C. Macabiau, A. Martineau, B. Bonhoure, and D. Kubrak, Nominal GNSS pseudorange measurement model for vehicular urban applications, in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (IEEE/ION PLANS) 2010*, pp. 806–815, IEEE/Institute of Navigation, 2010.

Schramm, D., M. Hiller, and R. Bardini, *Modellbildung und Simulation der Dynamik von Kraftfahrzeugen*, 2nd ed., Springer, Berlin, 2013.

Steigenberger, P., U. Hugentobler, S. Loyer, F. Perosanz, L. Prange, R. Dach, M. Uhlemann, G. Gendt, and O. Montenbruck, Galileo orbit and clock quality of the IGS Multi-GNSS Experiment, *Advances in Space Research*, *55*(1), 269–281, 2015.

Steinhardt, N., *Eine Architektur zur Schätzung kinematischer Fahrzeuggrößen mit integrierter Qualitätsbewertung durch Sensordatenfusion: Dissertation (TU Darmstadt, 2014)*, *Fortschrittberichte VDI : Reihe 12, Verkehrstechnik, Fahrzeugtechnik*, vol. 781, VDI-Verlag, Düsseldorf, 2014.

Strode, P. R. R., and P. D. Groves, GNSS multipath detection using three-frequency signal-to-noise measurements, *GPS Solutions*, *20*(3), 399–412, 2016.

Teunissen, P. J. G., and O. Montenbruck (Eds.), *Springer Handbook of Global Navigation Satellite Systems*, Springer, Cham, 2017.

Tijero, E. D., E. C. Pons, J. C. Calle, L. M. Fernández, P. F. N. Madrid, C. M. Varo, and M. A. Senz, Advanced GNSS Algorithms for Safe Autonomous Vehicles, in *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2017*, pp. 655–664, Institute of Navigation, 2017.

Tijero, E. D., L. M. Fernández, J. I. H. Zarzosa, J. Garca, J. Ibaez-Guzmn, E. Stawiarski, P. Xu, G. Avellone, F. Pisoni, E. Falletti, and M. Ortiz, High Accuracy Positioning Engine with an Integrity Layer for Safety Autonomous Vehicles, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 1566–1572, Institute of Navigation, 2018.

Uhlemann, M., Z. Deng, and M. Fritsche, Multi-GNSS Orbit and Clock Products of the GFZ Analysis Centre, in *Proceedings of the Geodätische Woche 2014*, Gesellschaft für Geodäsie, Geoinformation und Landmanagement (DVW), 2014.

Warnant, R., L. van de Vyvere, and Q. Warnant, Positioning with Single and Dual Frequency Smartphones Running Android 7 or Later, in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+) 2018*, pp. 284–303, Institute of Navigation, 2018.

Waypoint, *Inertial Explorer Manual*, NovAtel Inc., 2016.

Wendel, J., *Integrierte Navigationssysteme: Sensordatenfusion, GPS und inertiale Navigation*, Oldenbourg, München, 2007.

Working Group C - ARAIM Technical Subgroup, *Milestone 3 Report: Final Version*, EU-U.S. Cooperation on Satellite Navigation, 2016.

Xu, G., *GPS: Theory, algorithms and applications*, 2nd ed., Springer, Berlin, 2007.

Zumberge, J. F., and G. Gendt, The demise of selective availability and implications for the international GPS service, *Physics and Chemistry of the Earth, Part A: Solid Earth and Geodesy*, *26*(6-8), 637–644, 2001.

Own Publications

Reuper, B., M. Becker, and S. Leinen, Performance Assessment of Automotive Localization Algorithms via Metrics on a Modular Level, in *Proceedings of the International Symposium on Certification of GNSS Systems & Services (CERGAL) 2017*, Deutsche Gesellschaft für Ortung und Navigation (DGON), 2017.

Reuper, B., M. Becker, and S. Leinen, Performance Assessment of Automotive Localization Algorithms via Metrics on a Modular Level, *European Journal of Navigation*, *15*(2), 17–23, 2017.

Reuper, B., M. Becker, and S. Leinen, Performance Evaluation of a Tightly Coupled GNSS/IMU Integration Algorithm with Multi-Constellation/Multi-Frequency GNSS, in *Proceedings of the European Navigation Conference (ENC) 2018*, pp. 128–136, European Group of Institutes of Navigation (EUGIN), 2018.

Reuper, B., M. Becker, and S. Leinen, Benefits of Multi-Constellation/Multi-Frequency GNSS in a Tightly Coupled GNSS/IMU/Odometry Integration Algorithm, *Sensors*, *18*(9), 3052, 2018.

Appendix A

Test Scenario Details

This first appendix chapter provides detailed information about the test scenarios from Chapter 7. Its first section presents maps of the driven routes, its second section assesses the GNSS reception conditions in each scenario with the help of several different quality indicators.

A.1 Maps of the Routes

A map of each route is depicted in this section to illustrate the selected test scenarios. Figures A.1-A.4 give an overview of the four routes, while Figure A.5 shows a magnified section of Figure A.4 to detail the route through the city center of Frankfurt.

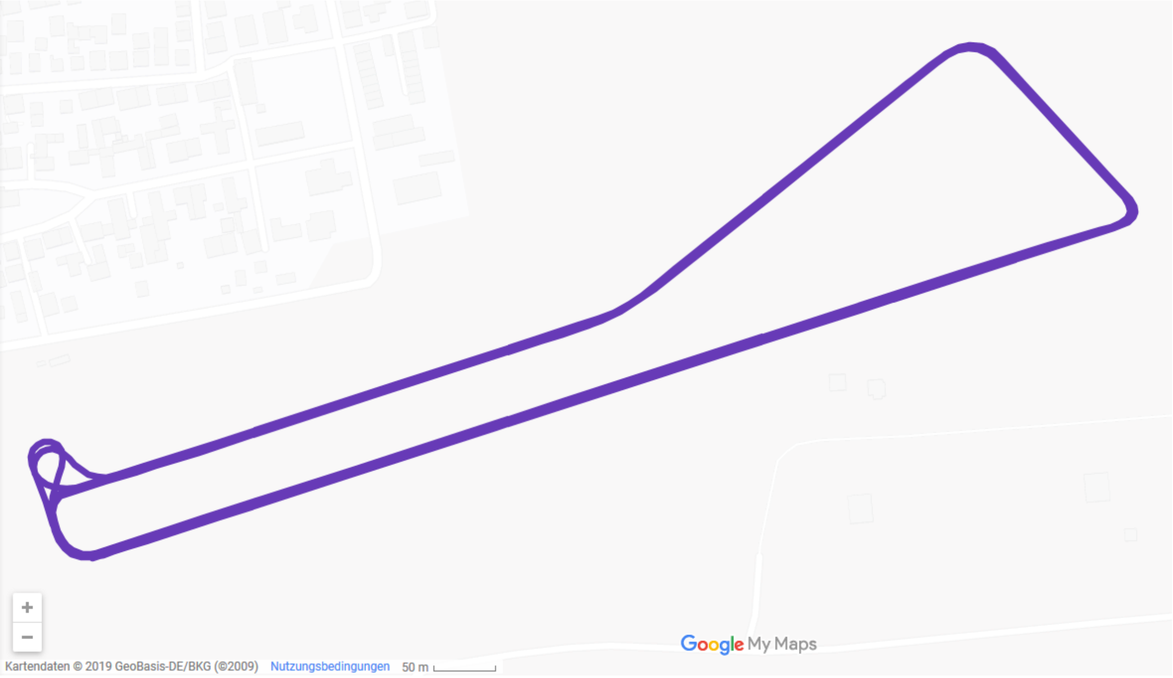

Figure A.1: Test scenario 1: Griesheim airfield (generated with Google Maps).

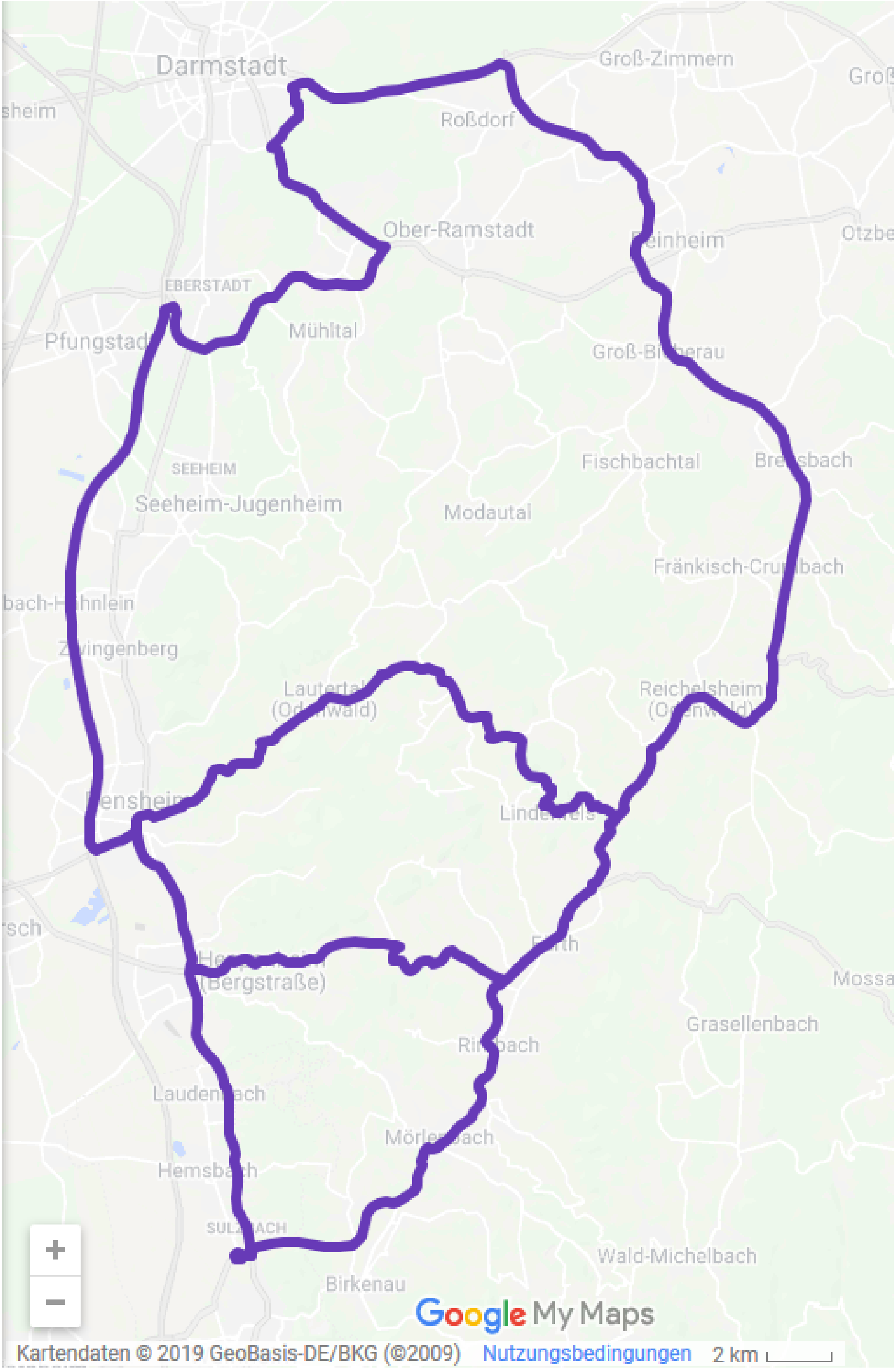

Figure A.2: Test scenario 2: Odenwald (generated with Google Maps).

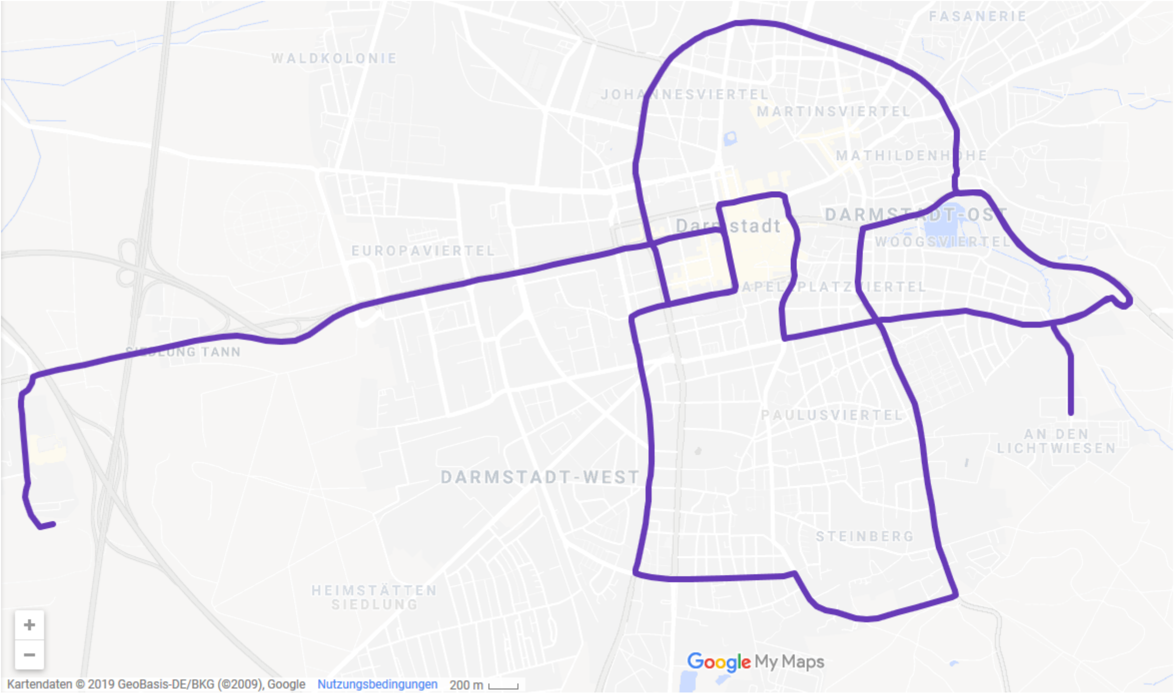

Figure A.3: Test scenario 3: Darmstadt (generated with Google Maps).

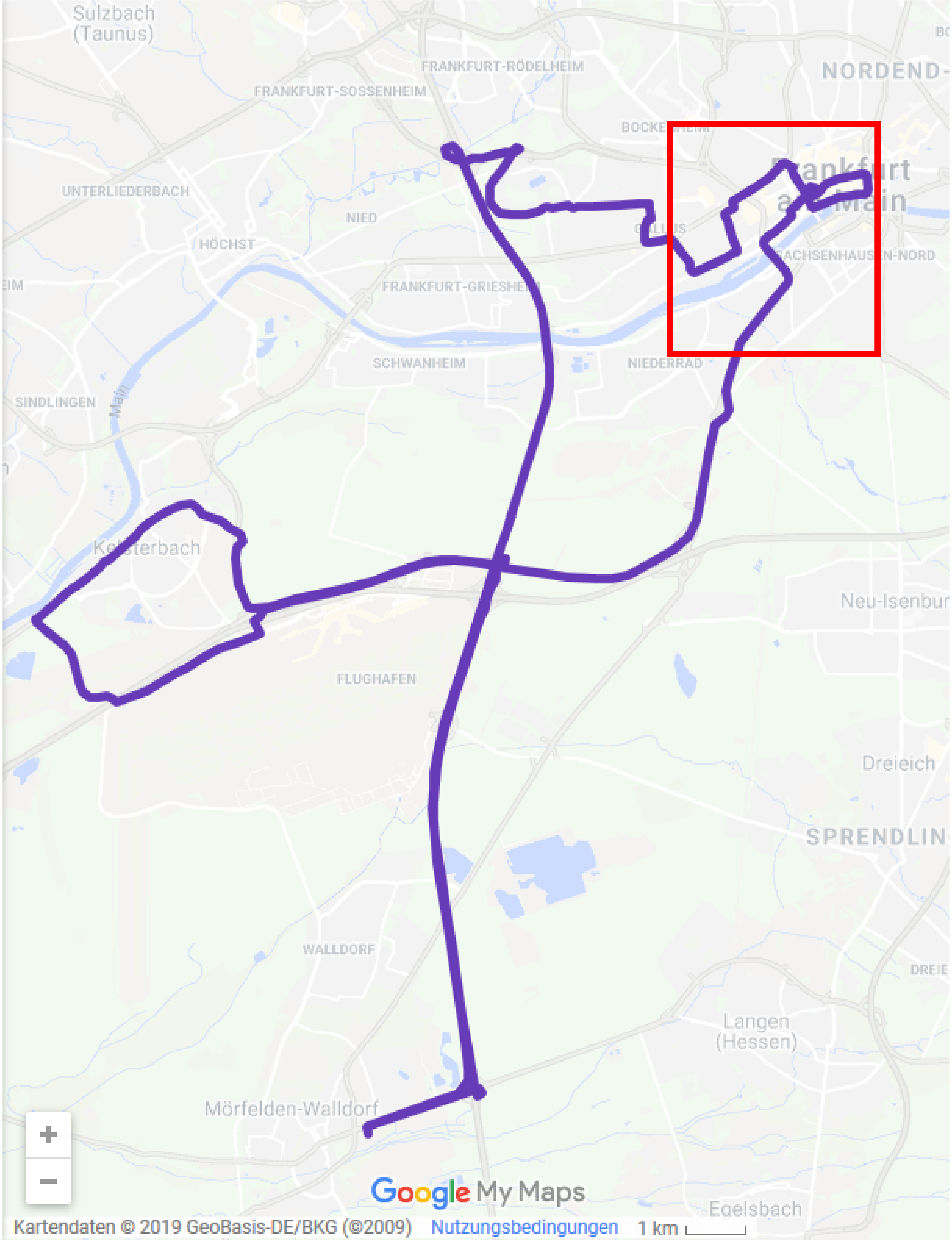

Figure A.4: Test scenario 4: Frankfurt (generated with Google Maps). The red rectangle indicates the section that is shown in detail in Figure A.5.

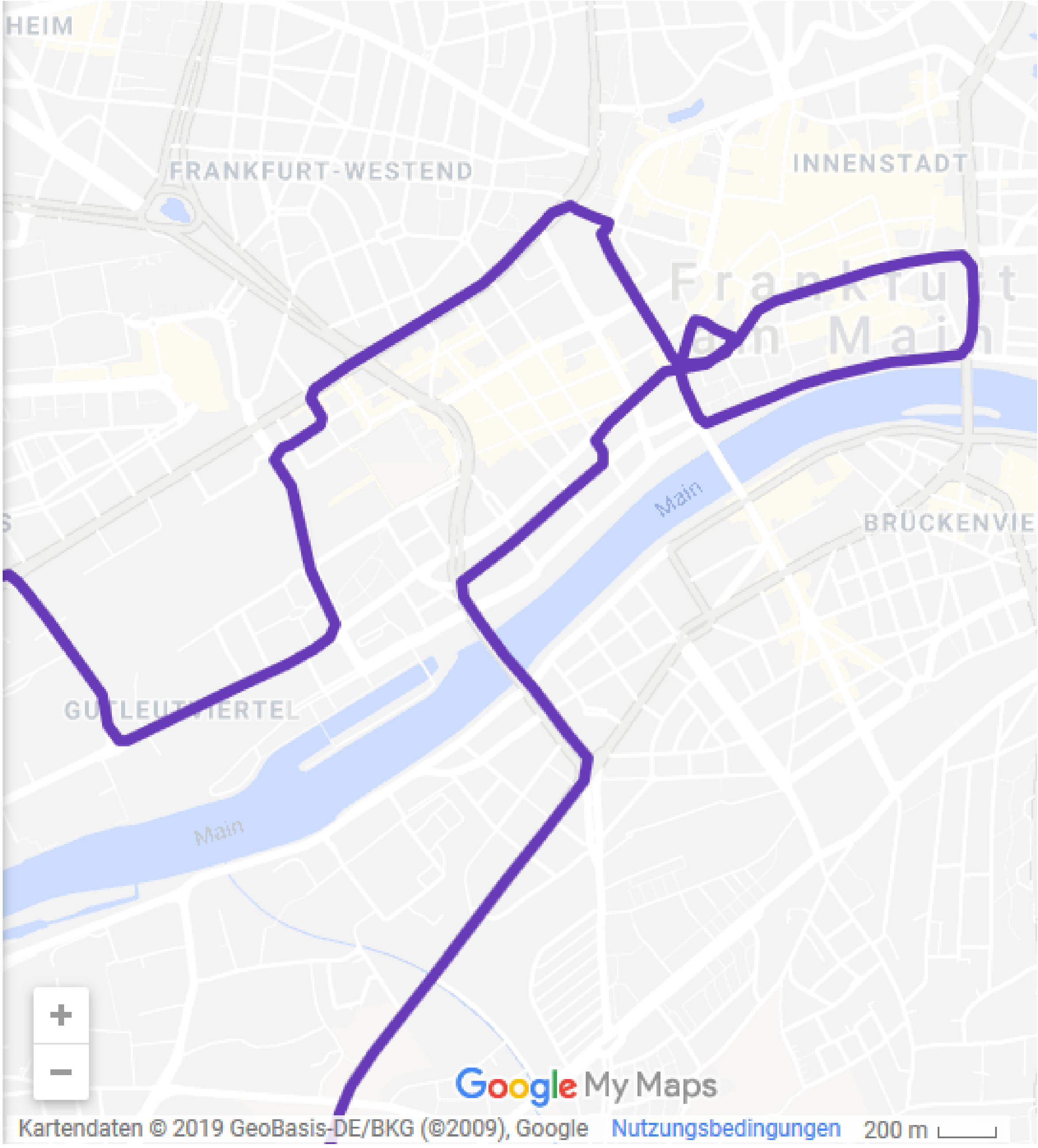

Figure A.5: Test scenario 4: Frankfurt, detailed view of city center, magnified section of Figure A.4 (generated with Google Maps).

A.2 GNSS Reception Conditions

This section assesses GNSS reception conditions in each test scenario via several quality indicators. PDOP and number of available satellites are depicted in Section A.2.1. Sections A.2.2 and A.2.3 examine code multipath and range rate errors that occur during the test scenarios. Among all GNSS error terms, these two are influenced the most by varying GNSS reception conditions. The total pseudorange error is investigated in Section A.2.4 by evaluating the accuracy of single-epoch GNSS position solutions. Finally, Section A.2.5 assesses the reference solution quality via quantities computed by Novatel's Waypoint - Inertial Explorer, the software used to obtain the reference solution. Reference solution quality correlates with GNSS reception conditions because the reference solution is based on RTK positioning. Therefore, its accuracy is predominantly influenced by carrier phase measurement quality.

A.2.1 PDOP and Number of Available Satellites

Figures A.6-A.10 present PDOP and number of available satellites for all runs of each scenario in the multi-frequency/multi-constellation configuration. Satellites are considered available if the C/N_0 in the L1 band is at least 30 dB-Hz ($C/N_{O,G1C}$ for GPS, $C/N_{O,E1X}$ for Galileo).

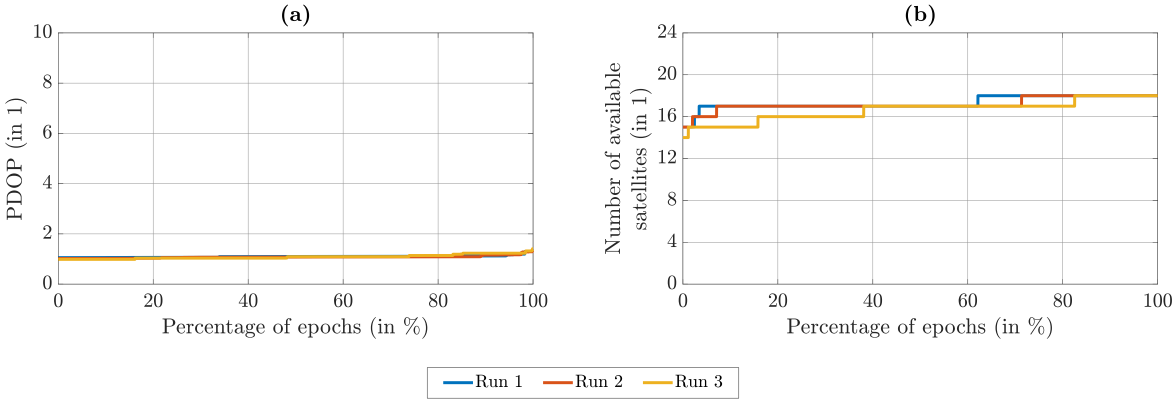

Figure A.6: PDOP and number of available satellites for Griesheim airfield data sets, moderate driving style. **(a)** Percentage of GNSS measurement epochs in which PDOP is at least as good as indicated. **(b)** Percentage of GNSS measurement epochs in which no more than the indicated number of satellites is available.

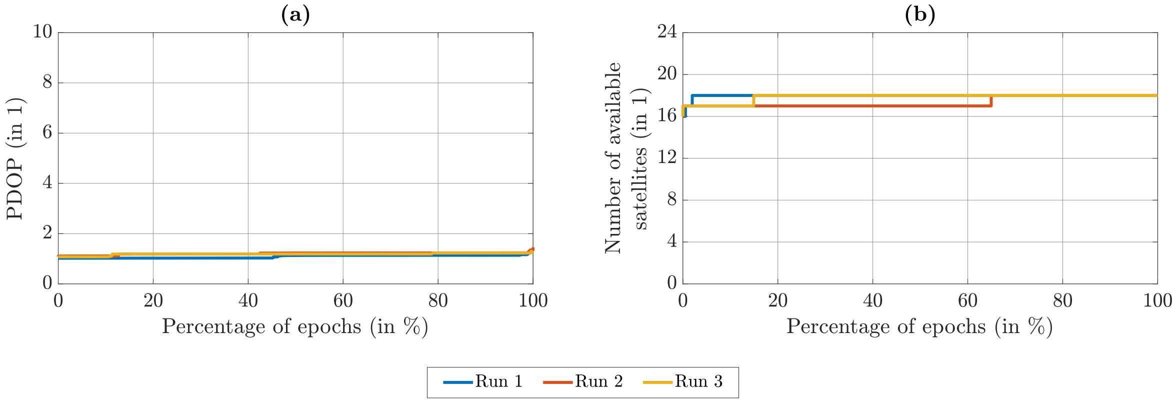

Figure A.7: PDOP and number of available satellites for Griesheim airfield data sets, dynamic driving style. **(a)** Percentage of GNSS measurement epochs in which PDOP is at least as good as indicated. **(b)** Percentage of GNSS measurement epochs in which no more than the indicated number of satellites is available.

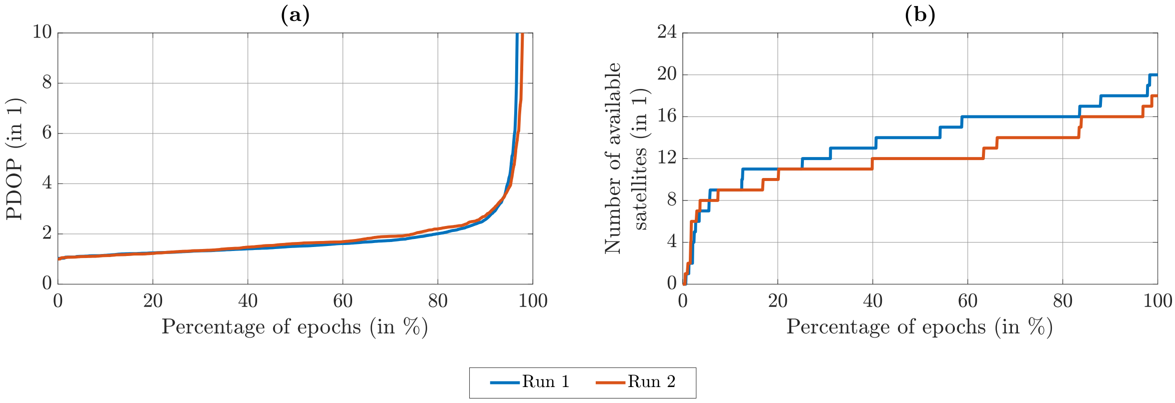

Figure A.8: PDOP and number of available satellites for Odenwald data sets. **(a)** Percentage of GNSS measurement epochs in which PDOP is at least as good as indicated. **(b)** Percentage of GNSS measurement epochs in which no more than the indicated number of satellites is available.

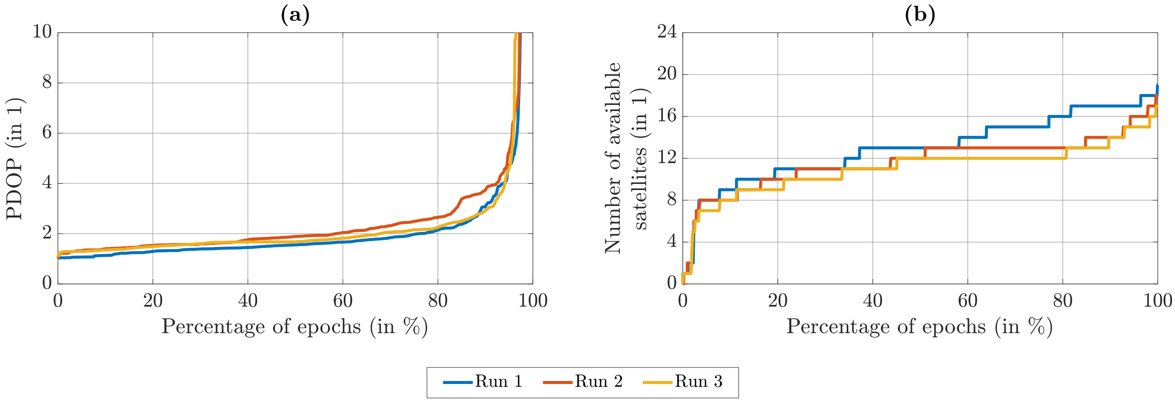

Figure A.9: PDOP and number of available satellites for Darmstadt data sets. **(a)** Percentage of GNSS measurement epochs in which PDOP is at least as good as indicated. **(b)** Percentage of GNSS measurement epochs in which no more than the indicated number of satellites is available.

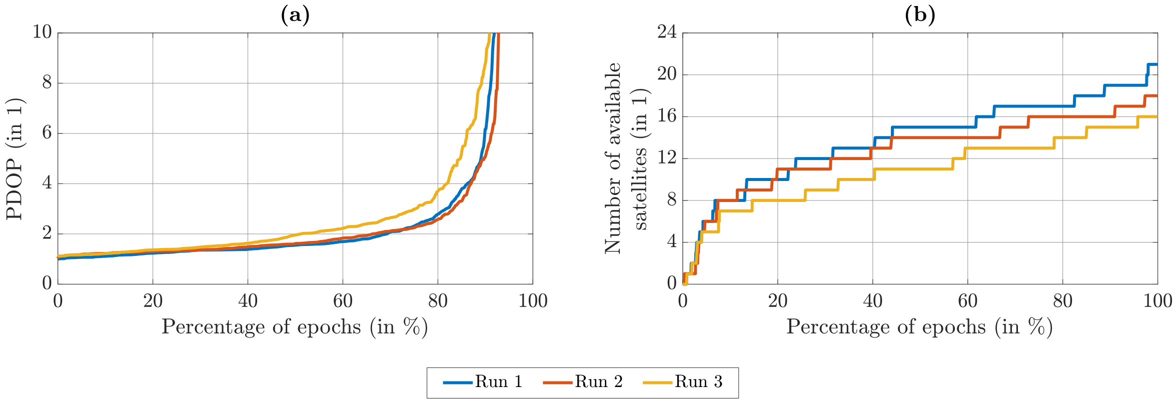

Figure A.10: PDOP and number of available satellites for Frankfurt data sets. **(a)** Percentage of GNSS measurement epochs in which PDOP is at least as good as indicated. **(b)** Percentage of GNSS measurement epochs in which no more than the indicated number of satellites is available.

A.2.2 Code Multipath Error

This section examines the code multipath errors that occur during the various test scenarios with the help of the multipath combination ρ_{MC} (3.32). ρ_{MC} mainly comprises the summed up effects of code multipath, NLOS reception errors and code tracking noise. The results for all test scenarios are presented in Figures A.11-A.16 and Tables A.1-A.6. Each figure depicts ρ_{MC} on one specific signal for all runs of each scenario. The same holds for the tables.

The data are shown in the figures in form of the *folded cumulative distribution function* (FCDF), a depiction that is commonly used for the assessment of GNSS errors (e. g. in *Cohenour and van Graas* [2011], *Perea et al.* [2016], *Joerger and Pervan* [2016]). It is especially useful if the main interest lies in the large absolute errors, which occur rather seldom. The FCDF is obtained by folding the upper half of the CDF down to the interval $[0, 0.5)$:

$$\text{FCDF} = \begin{cases} \text{CDF} & \text{for CDF} \leq 0.5 \\ 1 - \text{CDF} & \text{for CDF} > 0.5. \end{cases} \tag{A.1}$$

In order to magnify the visualization of the large absolute errors, the vertical axis of the FCDF plots is displayed logarithmically.

The results demonstrate that the goal of covering diverse GNSS reception conditions was achieved. The scenarios at Griesheim airfield exhibit the smallest multipath errors for all signals, while the Frankfurt scenario exhibits the largest multipath/NLOS errors. The Odenwald and Darmstadt scenario lie in the middle for all signals. When comparing the different signals with each other, the findings from Section 3.6.4 are confirmed: G5X, E5X and E7X exhibit the lowest multipath errors due to their high code chipping rate. G1C achieves intermediate performance, while G2X and E1X suffer from the largest multipath errors. E1X is especially vulnerable to very large errors. While an error of more than 100 m is only experienced by one of the other signals in one of the runs (G2X in the 2nd Darmstadt run, see Table A.2), E1X shows such large errors in all eight runs that did not take place at Griesheim airfield. The maximum error of more than 900 m occurs in the 1st Frankfurt run (see Table A.4). While such large errors are usually detected and eliminated via outlier detection, this reduces the number of available pseudorange observations, leading to degraded signal geometry and decreased position accuracy.

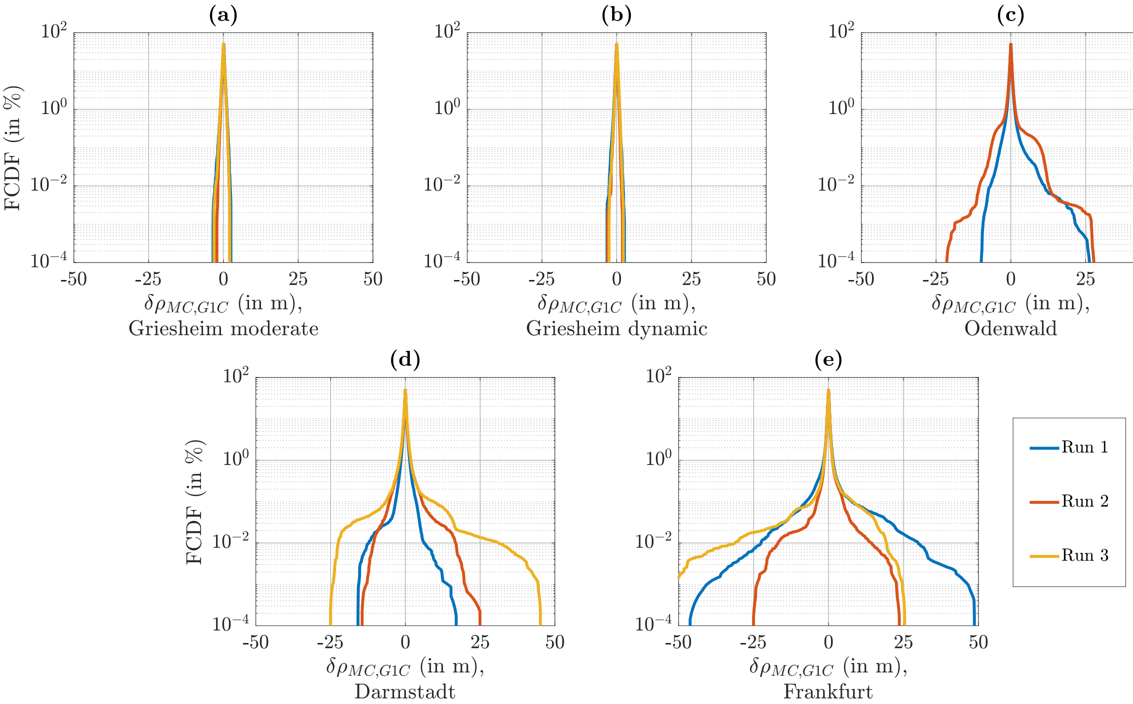

Figure A.11: FCDF of multipath combination ρ_{MC} on GPS L1 for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.1: RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on GPS L1 for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of ρ_{MC} (in m)**	**99 % Quantile of $\lvert\rho_{MC}\rvert$ (in m)**	**Max. Value of $\lvert\rho_{MC}\rvert$ (in m)**
Griesheim airfield, moderate driving style	0.36/0.36/0.36	1.32/1.16/1.24	3.46/2.18/3.12
Griesheim airfield, dynamic driving style	0.36/0.36/0.36	1.14/0.99/1.13	3.25/3.06/2.58
Odenwald	0.53/0.77	1.74/2.29	26.4/27.8
Darmstadt	0.54/0.71/0.98	1.88/2.61/2.70	17.0/25.0/45.2
Frankfurt	0.96/0.61/0.90	2.34/1.80/1.97	48.7/25.0/53.4

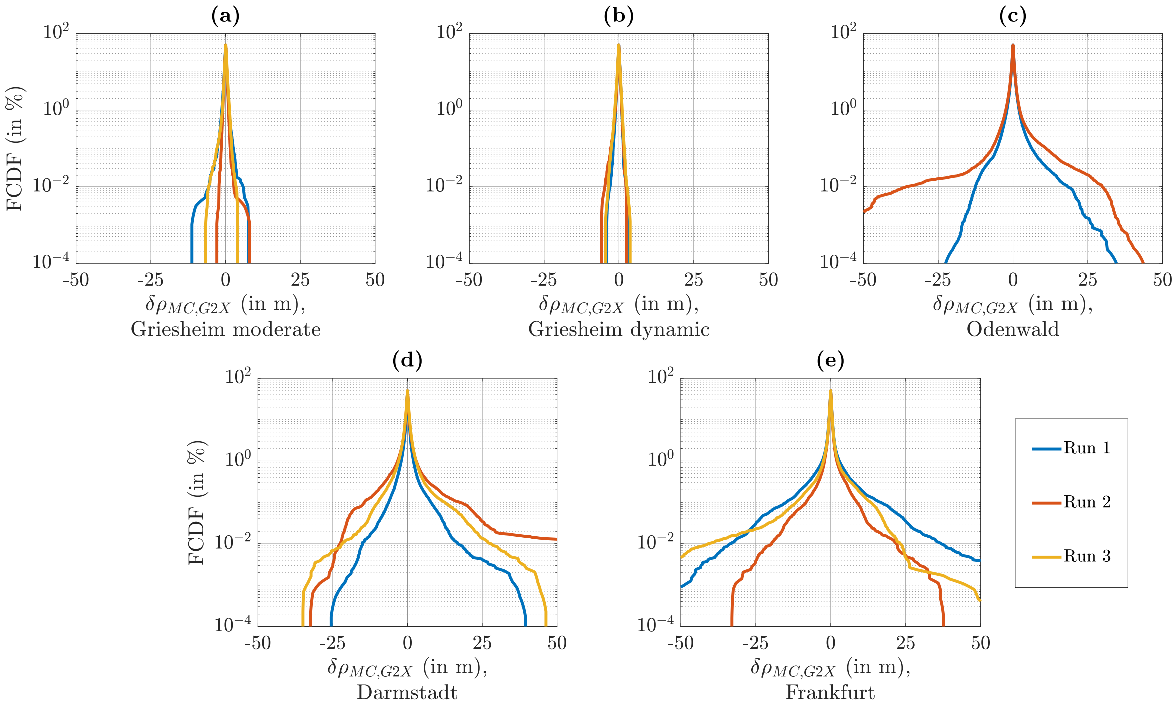

Figure A.12: FCDF of multipath combination ρ_{MC} on GPS L2C for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.2: RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on GPS L2C for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of ρ_{MC} (in m)**	**99 % Quantile of $\lvert\rho_{MC}\rvert$ (in m)**	**Max. Value of $\lvert\rho_{MC}\rvert$ (in m)**
Griesheim airfield, moderate driving style	0.40/0.40/0.40	1.56/1.11/1.51	11.3/8.07/6.67
Griesheim airfield, dynamic driving style	0.40/0.40/0.40	1.21/1.36/1.52	4.06/5.83/4.46
Odenwald	0.79/1.15	2.82/3.44	35.1/59.9
Darmstadt	0.83/1.71/1.23	2.98/5.15/4.27	39.4/114/46.3
Frankfurt	1.57/0.95/1.30	5.24/3.22/4.03	91.2/37.7/66.0

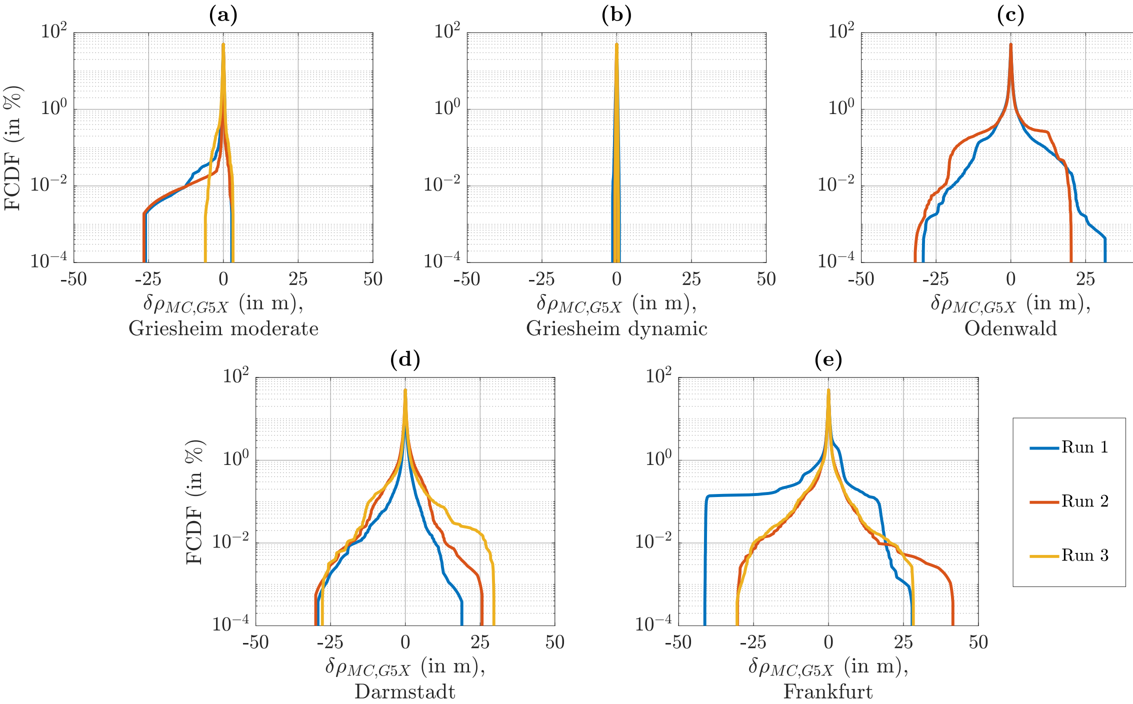

Figure A.13: FCDF of multipath combination ρ_{MC} on GPS L5 for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.3: RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on GPS L5 for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of ρ_{MC} (in m)**	**99 % Quantile of $\lvert\rho_{MC}\rvert$ (in m)**	**Max. Value of $\lvert\rho_{MC}\rvert$ (in m)**
Griesheim airfield, moderate driving style	0.21/0.21/0.21	0.77/0.49/0.95	26.0/26.6/6.02
Griesheim airfield, dynamic driving style	0.21/0.21/0.21	0.55/0.38/0.33	1.37/0.68/0.58
Odenwald	1.00/1.30	3.63/3.93	31.5/32.0
Darmstadt	0.57/0.93/1.00	1.90/4.04/3.18	29.2/30.0/29.6
Frankfurt	1.99/0.87/0.89	5.32/2.93/2.96	41.3/41.5/30.5

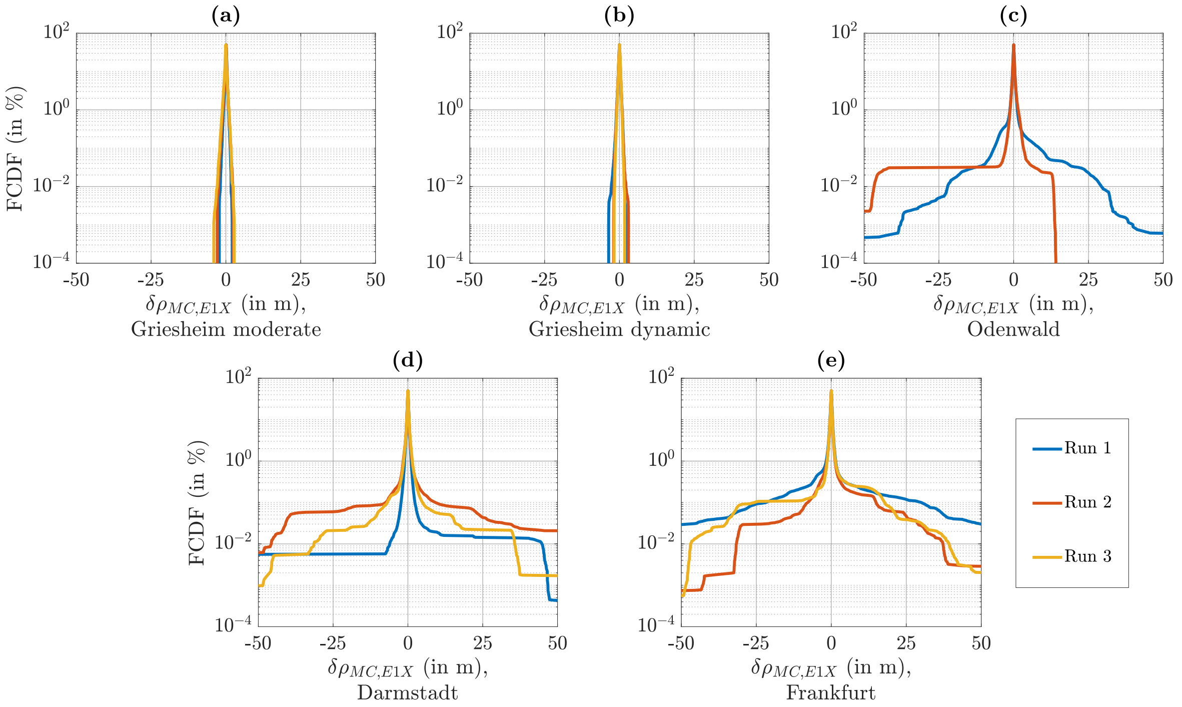

Figure A.14: FCDF of multipath combination ρ_{MC} on Galileo E1 for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.4: RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on Galileo E1 for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of ρ_{MC} (in m)**	**99 % Quantile of $\lvert\rho_{MC}\rvert$ (in m)**	**Max. Value of $\lvert\rho_{MC}\rvert$ (in m)**
Griesheim airfield, moderate driving style	0.33/0.33/0.33	1.01/1.24/1.37	2.22/2.88/4.05
Griesheim airfield, dynamic driving style	0.33/0.33/0.33	1.11/1.06/0.97	3.60/3.01/1.78
Odenwald	0.98/1.17	1.82/1.83	143/158
Darmstadt	1.09/2.53/1.15	1.37/2.08/2.00	148/165/141
Frankfurt	3.31/1.42/1.68	2.96/2.17/2.05	924/149/142

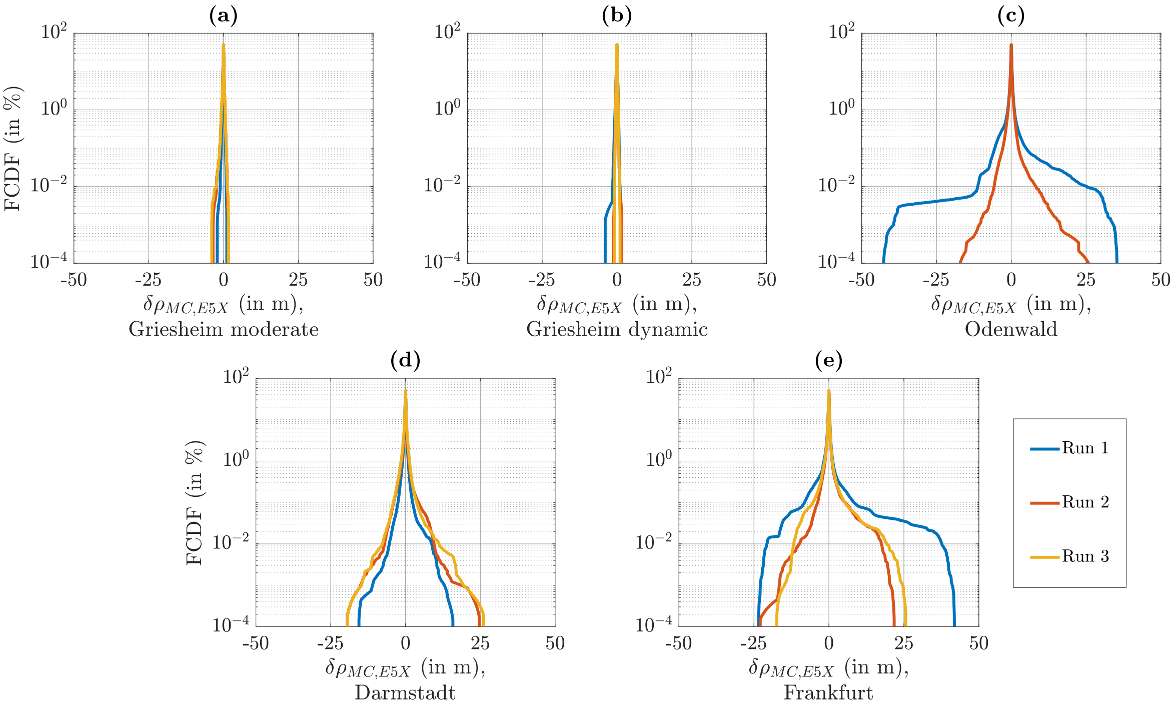

Figure A.15: FCDF of multipath combination ρ_{MC} on Galileo E5a for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.5: RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on Galileo E5a for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of ρ_{MC} (in m)**	**99 % Quantile of $\lvert\rho_{MC}\rvert$ (in m)**	**Max. Value of $\lvert\rho_{MC}\rvert$ (in m)**
Griesheim airfield, moderate driving style	0.18/0.18/0.18	0.52/0.72/0.74	2.11/3.49/3.91
Griesheim airfield, dynamic driving style	0.18/0.18/0.18	0.67/0.53/0.50	3.95/1.57/1.02
Odenwald	0.70/0.38	1.73/1.38	42.8/26.8
Darmstadt	0.36/0.48/0.52	1.29/1.74/1.97	15.8/24.7/26.1
Frankfurt	1.02/0.56/0.63	2.38/1.77/2.03	41.9/22.9/25.6

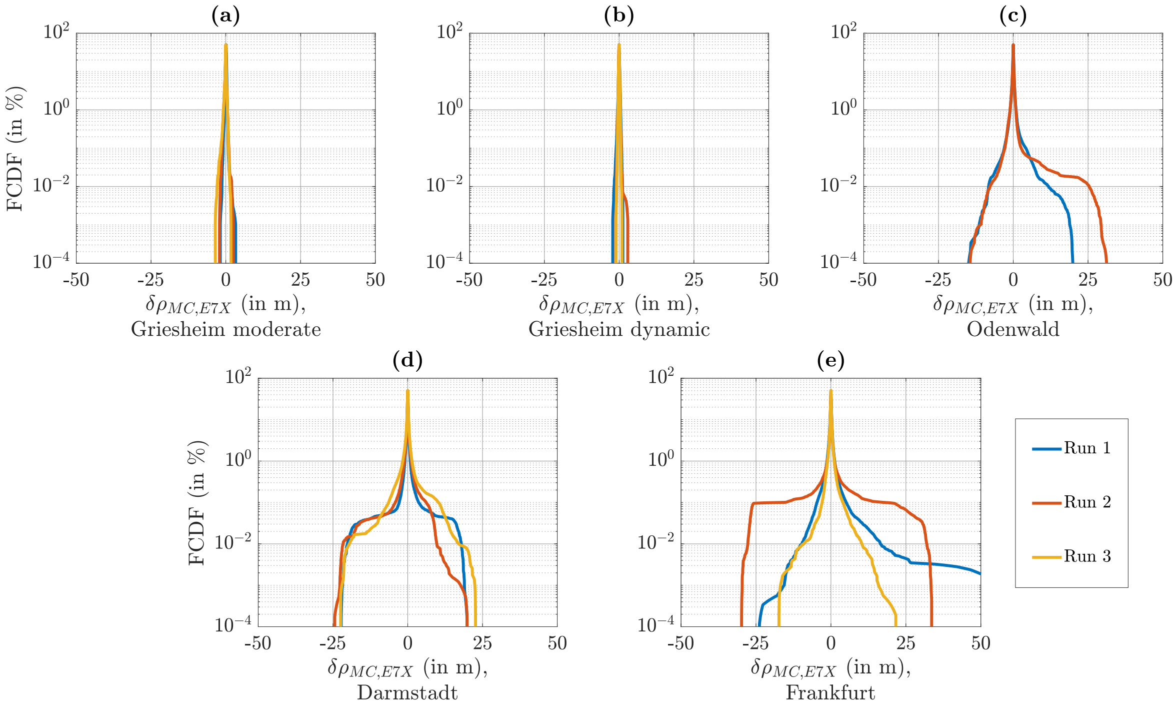

Figure A.16: FCDF of multipath combination ρ_{MC} on Galileo E5b for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.6: RMS, 99 % Quantile and Maximum of Multipath Combination ρ_{MC} on Galileo E5b for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of ρ_{MC} (in m)**	**99 % Quantile of $\lvert\rho_{MC}\rvert$ (in m)**	**Max. Value of $\lvert\rho_{MC}\rvert$ (in m)**
Griesheim airfield, moderate driving style	0.18/0.18/0.18	0.61/0.83/0.83	3.29/2.50/3.51
Griesheim airfield, dynamic driving style	0.18/0.18/0.18	0.63/0.49/0.47	2.14/2.88/1.10
Odenwald	0.43/0.53	1.39/1.36	20.0/31.3
Darmstadt	0.63/0.61/0.78	1.29/1.72/2.44	22.3/24.4/22.7
Frankfurt	0.63/1.34/0.44	1.93/2.15/1.58	56.3/33.6/21.6

A.2.3 Range Rate Error

This section examines the range rate errors that occur during the various test scenarios. The results for all runs of each test scenario are presented in Figure A.17 and Table A.7. No distinction between GPS and Galileo is made because the results are very similar for both constellations. This is in analogy to the range rate measurement model parametrized in Section 3.6.6. The data shown in Figure A.17 are depicted in form of their FCDF to illustrate the frequency with which large absolute errors occur in the different test scenarios. Just as for the multipath errors in Section A.2.2, the scenarios at Griesheim airfield feature the smallest errors, while the Frankfurt scenario exhibits the largest ones and the Odenwald scenario as well as the Darmstadt scenario lie in the middle.

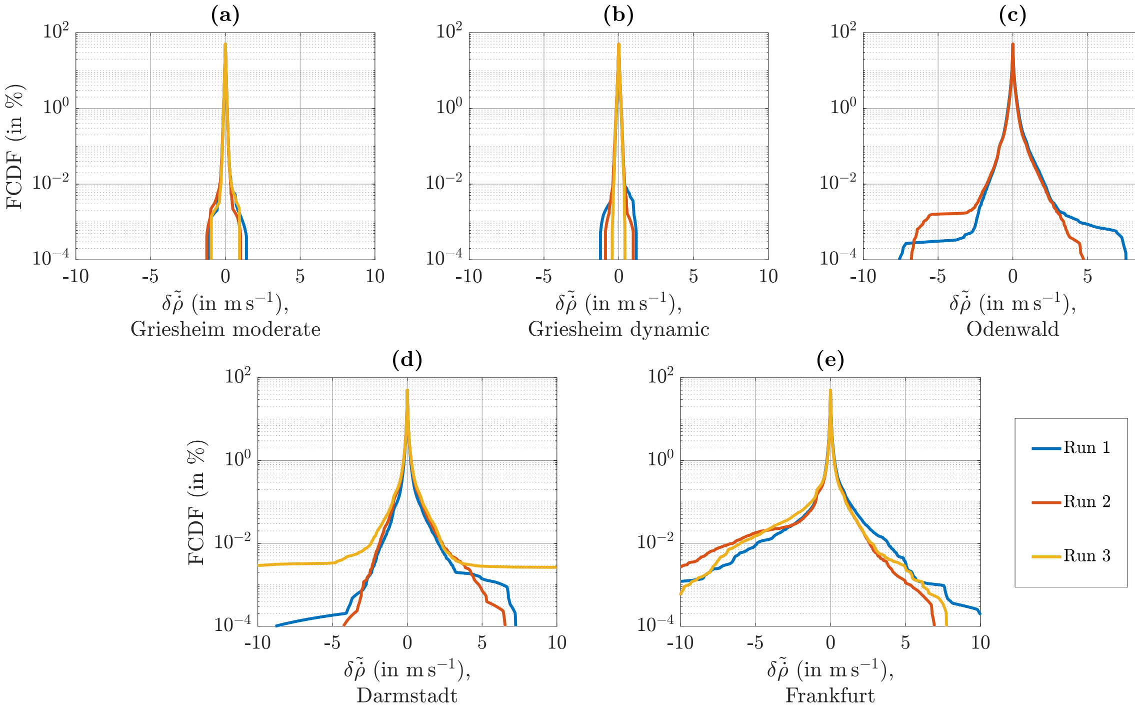

Figure A.17: FCDF of range rate error $\delta\tilde{\dot{\rho}}$ for all test scenarios. **(a)** Griesheim airfield data sets, moderate driving style. **(b)** Griesheim airfield data sets, dynamic driving style. **(c)** Odenwald data sets. **(d)** Darmstadt data sets. **(e)** Frankfurt data sets.

Table A.7: RMS, 99 % Quantile and Maximum of Range Rate Error $\delta\tilde{\dot{\rho}}$ for all Test Scenarios. Each Cell Contains the Values for all Runs of a Given Scenario

Test Scenario	**RMS of $\delta\tilde{\dot{\rho}}$ (in m s^{-1})**	**99 % Quantile of $\lvert\delta\tilde{\dot{\rho}}\rvert$ (in m s^{-1})**	**Max. Value of $\lvert\delta\tilde{\dot{\rho}}\rvert$ (in m s^{-1})**
Griesheim airfield, moderate driving style	0.04/0.04/0.04	0.14/0.15/0.14	1.41/1.24/0.95
Griesheim airfield, dynamic driving style	0.05/0.05/0.05	0.19/0.20/0.19	1.22/0.97/0.44
Odenwald	0.12/0.11	0.45/0.42	7.82/6.81
Darmstadt	0.09/0.10/0.18	0.31/0.36/0.39	10.3/6.56/23.7
Frankfurt	0.15/0.16/0.15	0.39/0.36/0.36	19.4/20.8/16.3

A.2.4 Single-Epoch Position Solution Accuracy

In this section, the position error between single-epoch GNSS position solution and a reference solution is evaluated. The reference solution is obtained in the same way as in Chapter 7, but transferred to the GNSS antenna position instead of the position of the MEMS IMU. Single-epoch GNSS position solutions are usually only needed for initialization of the navigation solution and not computed anymore after initialization is complete. For the figures in this section, the localization algorithm tries to compute a single-epoch position solution from multi-frequency/multi-constellation pseudoranges at each GNSS measurement epoch. A simple outlier detection method is applied, which is able to identify and eliminate a pseudorange outlier if its corresponding residual is significantly larger than the Euclidean norm of the residuals corresponding to the remaining pseudoranges.

The horizontal and vertical position error CDFs of this single-epoch solution for all runs of a specific test scenario are depicted in Figures A.18-A.23. Epochs in which no solution is computed because there are not enough available pseudorange observations or the positioning algorithm does not converge are considered to have an infinite error. Consequently, some error quantiles are missing in several plots. The plots from the scenarios at Griesheim airfield in Figures A.18 and A.19 have the same scale to facilitate an easy comparison. This also applies to the plots from the Odenwald, Darmstadt and Frankfurt scenarios in Figures A.20-A.22. However, the scale changes from the Griesheim plots to the other three, because the error magnitudes are too dissimilar to allow for the usage of the same scale. Because the position error reaches values of up to several hundred meters in the Frankfurt runs due to poor signal geometry, large multipath errors and NLOS reception, the results from the Frankfurt scenario are presented in Figure A.23 once again, but with a logarithmic scale.

Since the single-epoch solution experiences such large errors in the Frankfurt test scenario, the horizontal projection of the position solution from a portion of each of its three runs is depicted in Figures A.24-A.26. This portion contains the stretch through Frankfurt city center, where the worst GNSS reception conditions and therefore the largest errors are encountered. This area is also shown in the detailed map in Figure A.5 and contains deep urban canyons as well as two tunnels. In each of the figures, the single-epoch solution and the integrated solution are shown together in the same plot. Since the horizontal position error of the integrated solution stays below 8 m during all runs, it cannot be distinguished from the reference solution in these figures. Therefore, the reference solution is omitted. The horizontal errors of the single-epoch solution on the other hand are clearly visible and exceed values of 1600 m, 300 m and 1500 m during the depicted time interval for the three runs, respectively. Because pseudoranges from multiple satellites are simultaneously affected by large multipath or NLOS errors, these errors cannot be identified when computing the single-epoch solution, yielding a position solution that deviates from the true position by several hundred meters. The integrated solution remains largely uncontaminated from these large pseudorange errors, because they are identified by the outlier detection techniques described in Section 6.4 and subsequently eliminated.

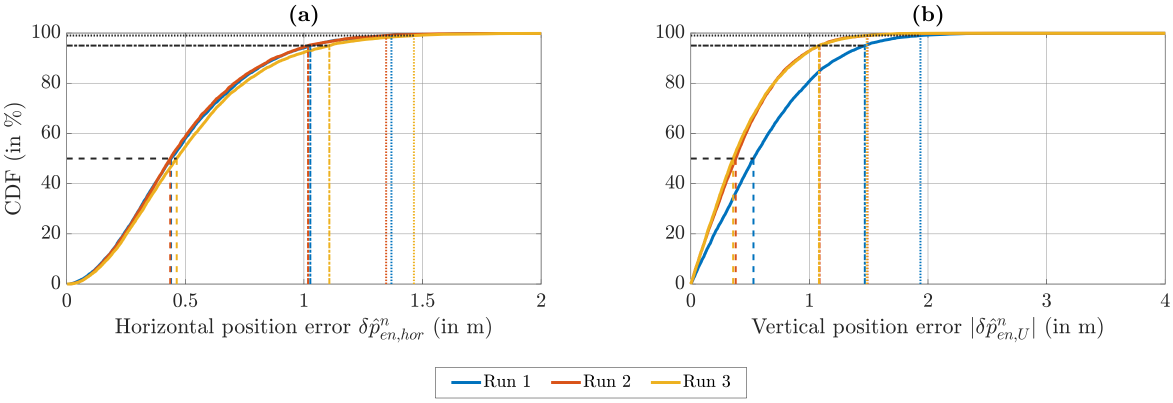

Figure A.18: Position error CDF for Griesheim airfield data sets from single-epoch solution, moderate driving style. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

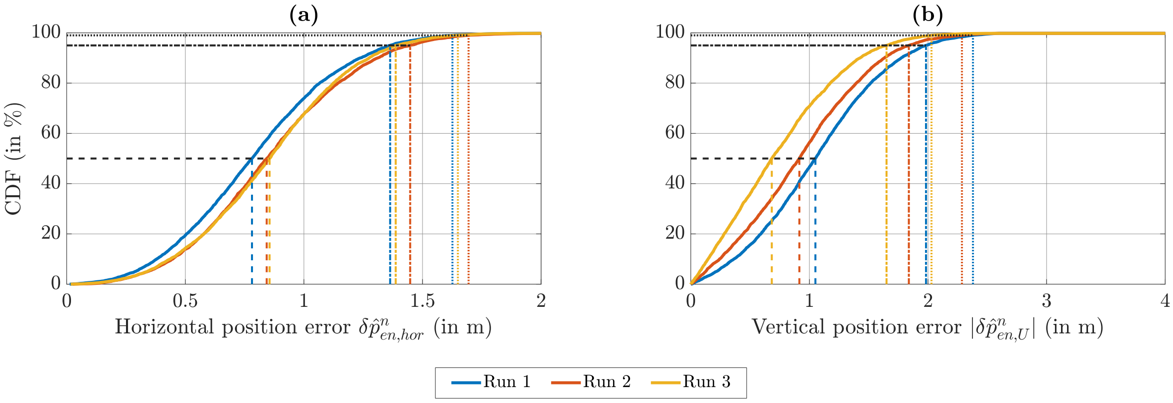

Figure A.19: Position error CDF for Griesheim airfield data sets from single-epoch solution, dynamic driving style. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

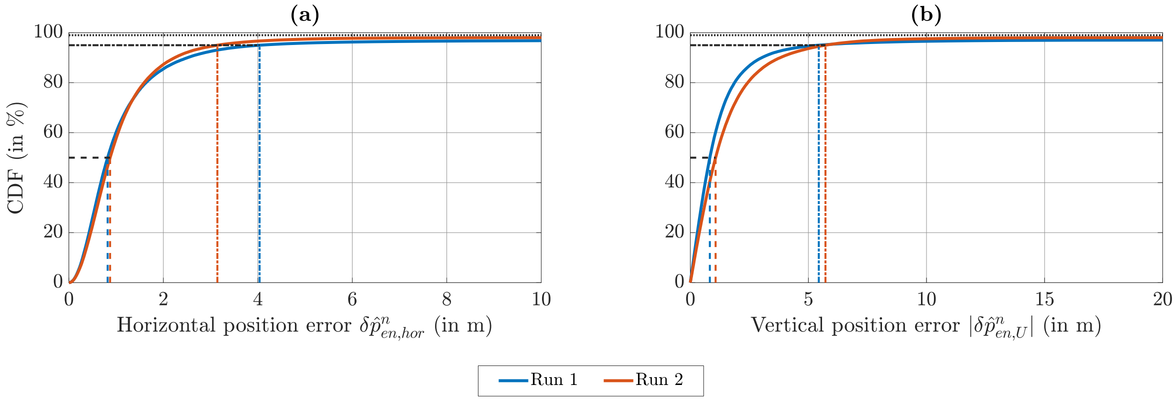

Figure A.20: Position error CDF for Odenwald data sets from single-epoch solution. 50 % and 95 % error quantiles are indicated, 99 % error quantiles do not apply since no position solution can be obtained in more than 1 % of epochs. **(a)** Horizontal error. **(b)** Vertical error.

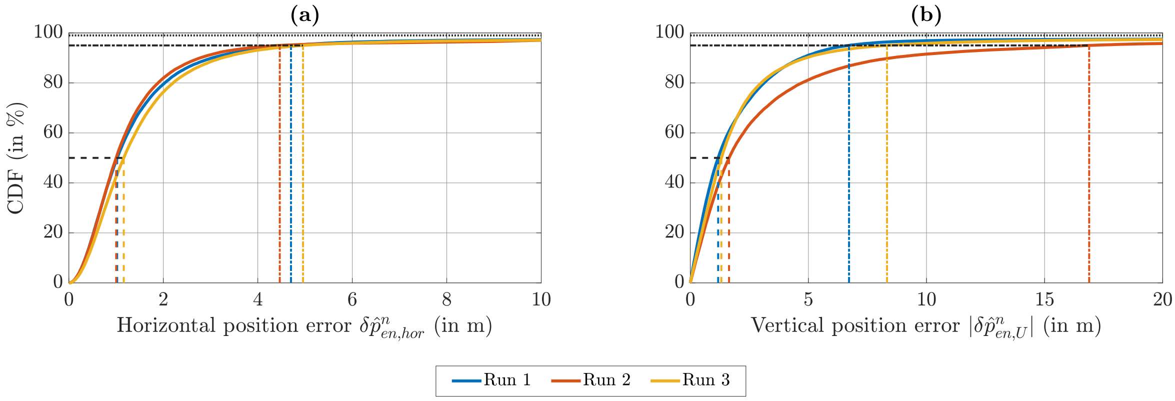

Figure A.21: Position error CDF for Darmstadt data sets from single-epoch solution. 50 % and 95 % error quantiles are indicated, 99 % error quantiles do not apply since no position solution can be obtained in more than 1 % of epochs. **(a)** Horizontal error. **(b)** Vertical error.

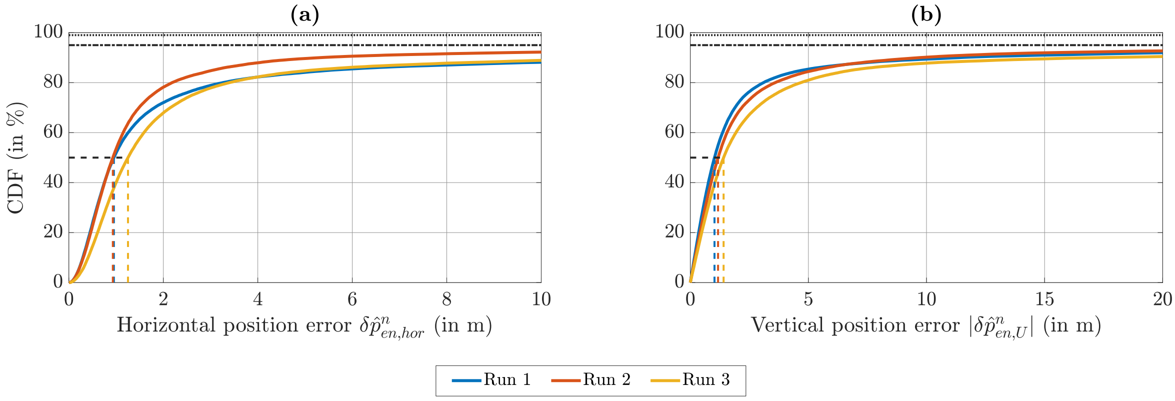

Figure A.22: Position error CDF for Frankfurt data sets from single-epoch solution. 50 % error quantiles are indicated. 95 % error quantiles for runs 1 and 2 are off the scale, but visible in Figure A.23. 95 % error quantiles for run 3 and 99 % error quantiles for all runs do not apply since no position solution can be obtained in more than 5 % of epochs for run 3 and more than 1 % of epochs for the other runs, respectively. **(a)** Horizontal error. **(b)** Vertical error.

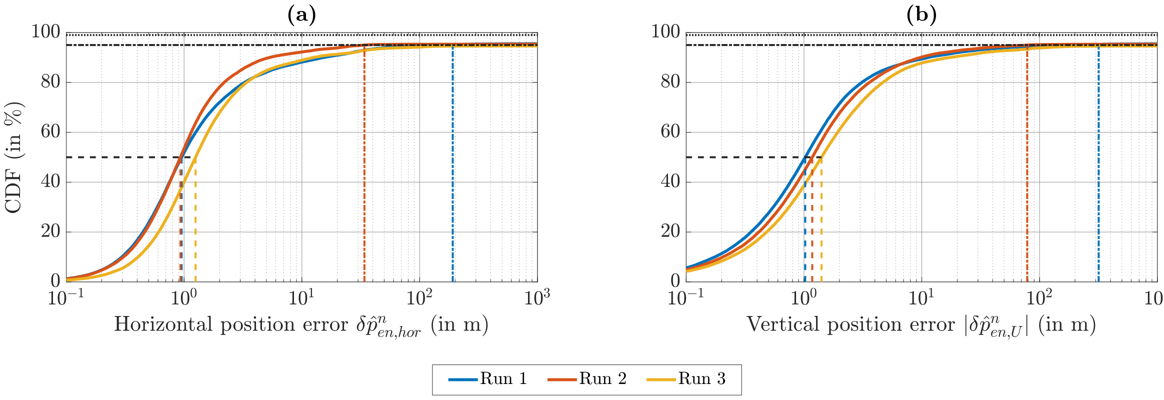

Figure A.23: Position error CDF for Frankfurt data sets from single-epoch solution, logarithmic scale. 50 % and 95 % error quantiles are indicated. 95 % error quantiles for run 3 and 99 % error quantiles for all runs do not apply since no position solution can be obtained in more than 5 % of epochs for run 3 and more than 1 % of epochs for the other runs, respectively. **(a)** Horizontal error. **(b)** Vertical error.

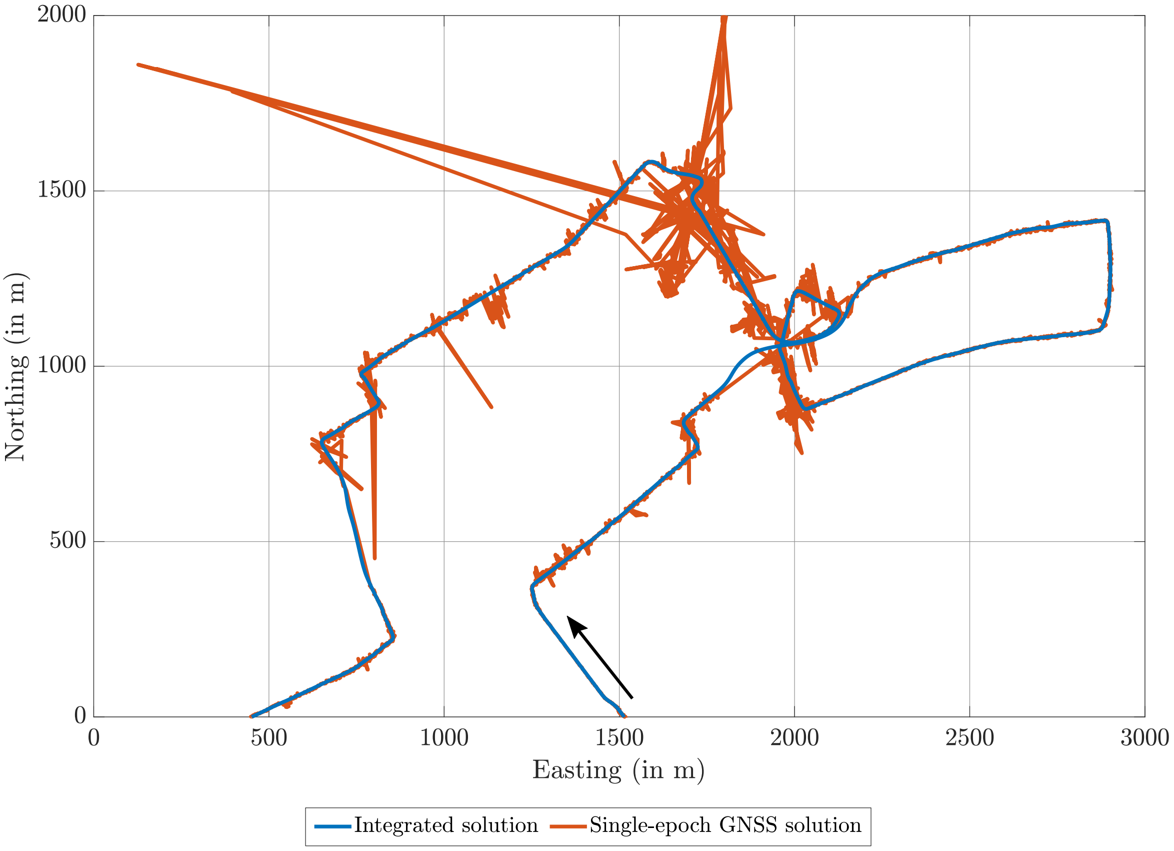

Figure A.24: Horizontal projection of position solution inside Frankfurt city center, run 1. Driving direction is indicated. The single-epoch GNSS solution exhibits errors of more than 1600 m horizontally and 1900 m vertically. On this scale, the reference solution is indistinguishable from the integrated solution and hence omitted.

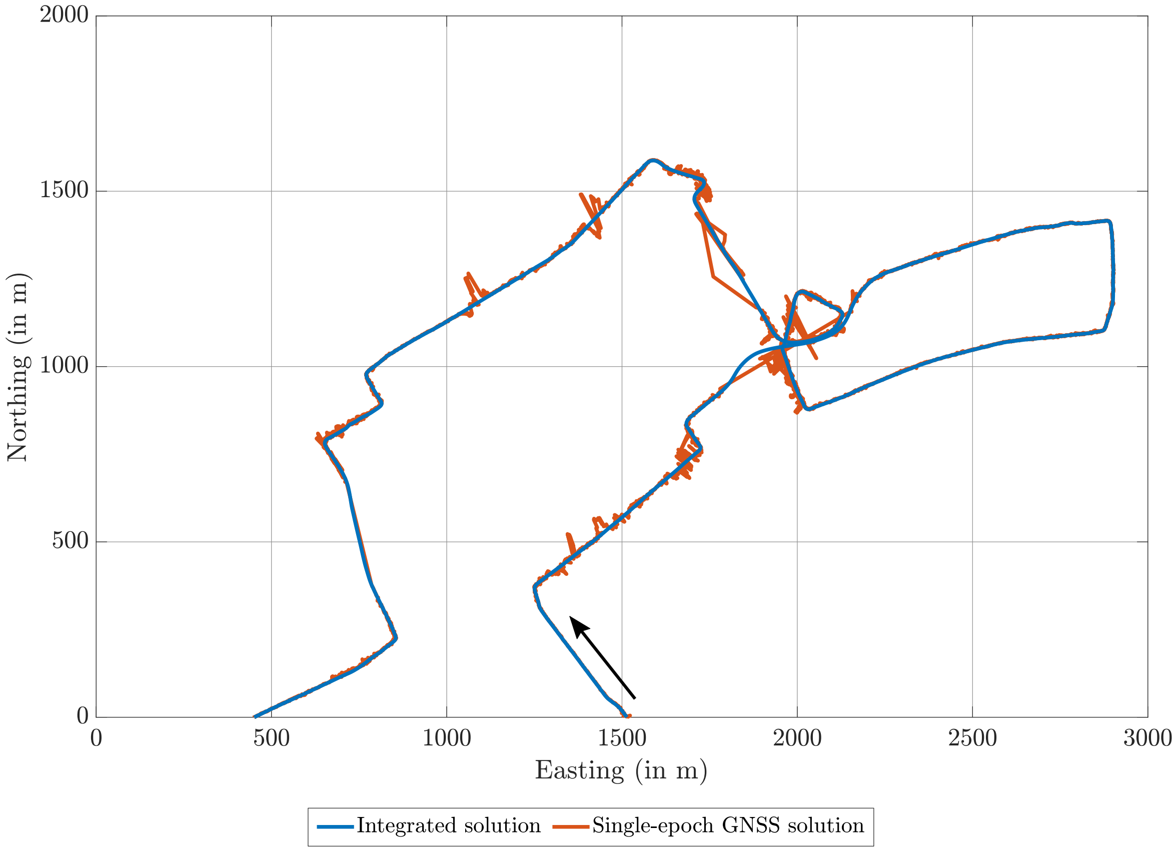

Figure A.25: Horizontal projection of position solution inside Frankfurt city center, run 2. Driving direction is indicated. The single-epoch GNSS solution exhibits errors of more than 300 m horizontally and 1500 m vertically. On this scale, the reference solution is indistinguishable from the integrated solution and hence omitted.

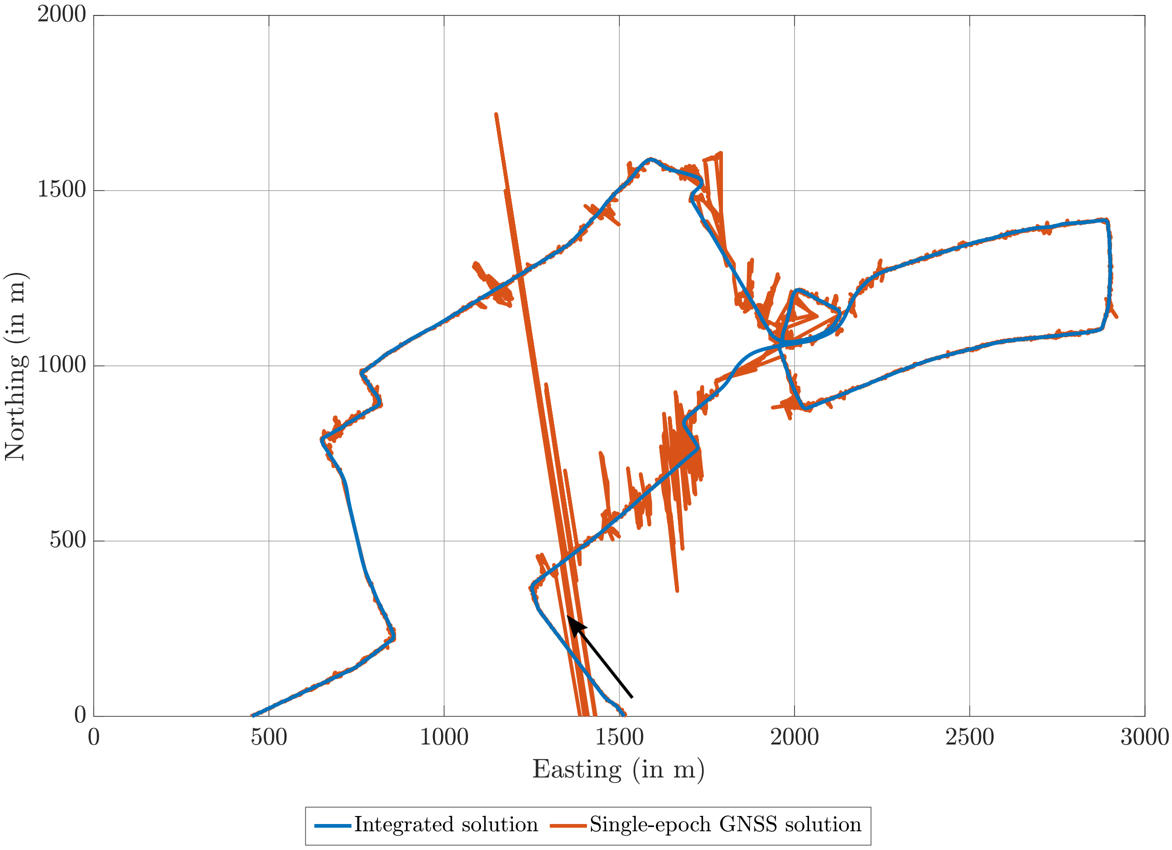

Figure A.26: Horizontal projection of position solution inside Frankfurt city center, run 3. Driving direction is indicated. The single-epoch GNSS solution exhibits errors of more than 1500 m horizontally and 1700 m vertically. On this scale, the reference solution is indistinguishable from the integrated solution and hence omitted.

A.2.5 Reference Solution Quality

This section examines the GNSS reception conditions during the various test scenarios by assessing the quality of the reference solution. Because the reference solution's accuracy is based on carrier phase positioning, its quality is mainly an indicator for carrier phase measurement quality. In contrast, the accuracy of the localization algorithm developed in this thesis is based on pseudorange positioning. Code tracking is typically much more robust than carrier tracking [*Groves*, 2013, p. 373], so carrier phase measurement quality can only serve as a coarse metric for pseudorange measurement quality.

The reference solution is obtained by processing data from a navigation grade ring laser gyroscope IMU (iMAR iNAV-RQH-1003) and a geodetic GNSS receiver mounted inside the car together with GNSS observations from a base station in Novatel's Waypoint - Inertial Explorer software. The data are processed both forward and backward in time before both solutions are combined and smoothed to form the most accurate solution possible. In addition to the navigation solution, Inertial Explorer also outputs several quantities that assess the solution's quality. One of these quantities is the estimated position standard deviation, which is included in the figures in Section 7.2.2. Another one is the so-called quality factor, a metric whose numerical values are the natural numbers in the interval $[1, 6]$, with lower values indicating higher solution quality. All possible quality factors, their description and the associated 3-D accuracy of the reference solution are listed in Table A.8. The values for 3-D accuracy appear to be quite pessimistic. Based on the RMS of the total pseudorange error according to Section 3.6.5, the algorithm developed in this thesis has an expected accuracy in the order of 1 m under good GNSS reception conditions. Consequently, only epochs with a reference quality factor of 1 or 2 (max. estimated 3-D error 0.4 m) warrant proper accuracy assessment. However, because both the algorithm under test as well as the reference solution rely on GNSS observations, the accuracy of the algorithm under test is degraded whenever the reference solution's quality is suboptimal, as well. Due to the higher capability of the reference equipment (ring laser gyroscope IMU vs. MEMS IMU, RTK positioning vs. pseudorange positioning, combining forward and backward solution vs. forward filtering only), the solution of the algorithm under test drifts away from the truth faster than the reference solution. Therefore, the reference is still able to assess the accuracy of the algorithm under test properly, even if the former does not reach a quality factor of 1 or 2. This theory is backed up by the CDF of the reference solution's estimated standard deviation, which is included in all figures concerning the position error CDF of the algorithm under test in Sections 7.2.2 and B.4.1.

Table A.8: Quality Factor Description and Associated 3-D Accuracy for Novatel's Waypoint - Inertial Explorer Software [*Waypoint*, 2016]

Inertial Explorer Quality Factor	**Description**	**3-D Accuracy of Ref. Solution (in m)**
1	Fixed integer	0.00 – 0.15
2	Converged float or noisy fixed integer	0.05 – 0.40
3	Converging float	0.20 – 1.00
4	Converging float	0.50 – 2.00
5	DGNSS	1.00 – 5.00
6	DGNSS	2.00 – 10.00

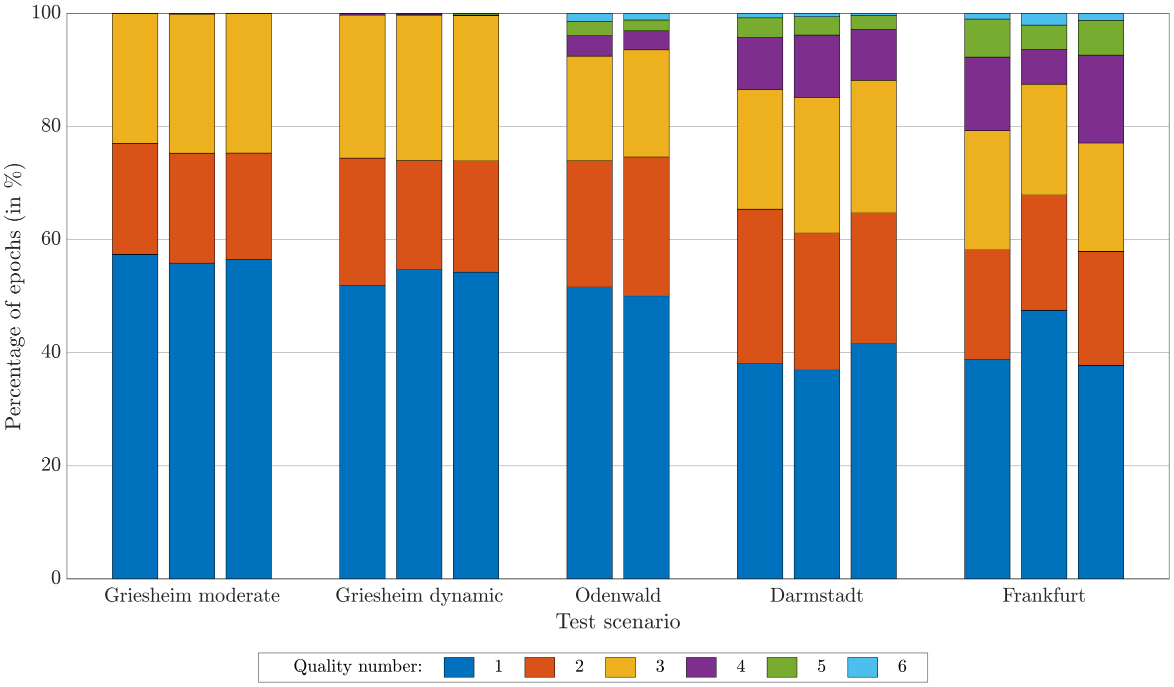

Figure A.27: Reference solution quality factor distribution for all test scenario runs. Each bar represents one run of the respective scenario. The quality factor description is given in Table A.8.

The quality factor distribution of the reference solution for all test scenarios is displayed in Figure A.27. The scenarios are ordered from left to right in decreasing reference solution quality. As expected, the solution quality decreases with more difficult GNSS reception conditions. Even under unimpaired sky view, a PDOP of less than 1.7 and at least twelve available satellites at all times, Inertial Explorer is unable to obtain a solution with quality factor 1 or 2 in more than 20 % of epochs for the test scenarios at Griesheim airfield. (In contrast to the algorithm under test, the reference solution uses GPS and GLONASS observations with an elevation mask of 10°, so the values for PDOP and number of available satellites differ from the ones for the algorithm under test in Section A.2.1.) Because the higher vehicle dynamics make carrier tracking more difficult, the reference solution quality is worse in the runs with dynamic driving style than it is in the runs with moderate driving style. While less than 1 % of epochs exhibit a quality factor of 4 or worse at Griesheim airfield, this value rises to 7.0 %, 13.4 % and 18.7 % on average for the Odenwald, Darmstadt and Frankfurt scenarios, respectively.

Appendix B

Additional Results

This appendix chapter presents results that are not included in the main part. In most cases, these results are very similar to the ones that are described in detail in the main part and are therefore redundant. The chapter contains four sections, which deal with DCBs, GNSS measurement noise, odometry measurement noise and the evaluation of the test scenarios from Chapter 7, respectively.

B.1 Differential Code Biases

This section presents the receiver-specific DCBs for all three collected data sets, each spanning 24 h. Since there is no noticeable variation over time except noise, only one of the plots appears in the main part (Section 3.4). It is repeated here for easier comparison of the days with each other. Figures B.1-B.3 display the receiver-specific DCBs computed with satellite-specific DCBs from DLR, while Figures B.4-B.6 display the receiver-specific DCBs computed with satellite-specific DCBs from the navigation data messages.

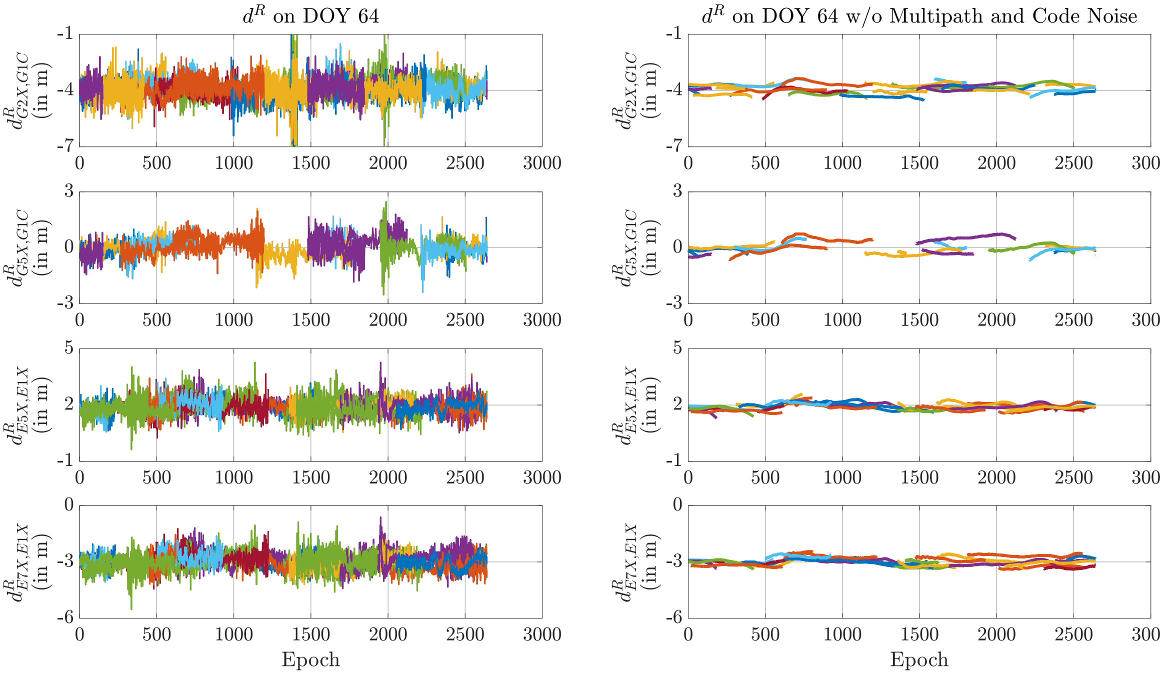

Figure B.1: Receiver-specific DCBs on DOY 64, computed with satellite-specific DCBs from DLR. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites (Repetition of Figure 3.1).

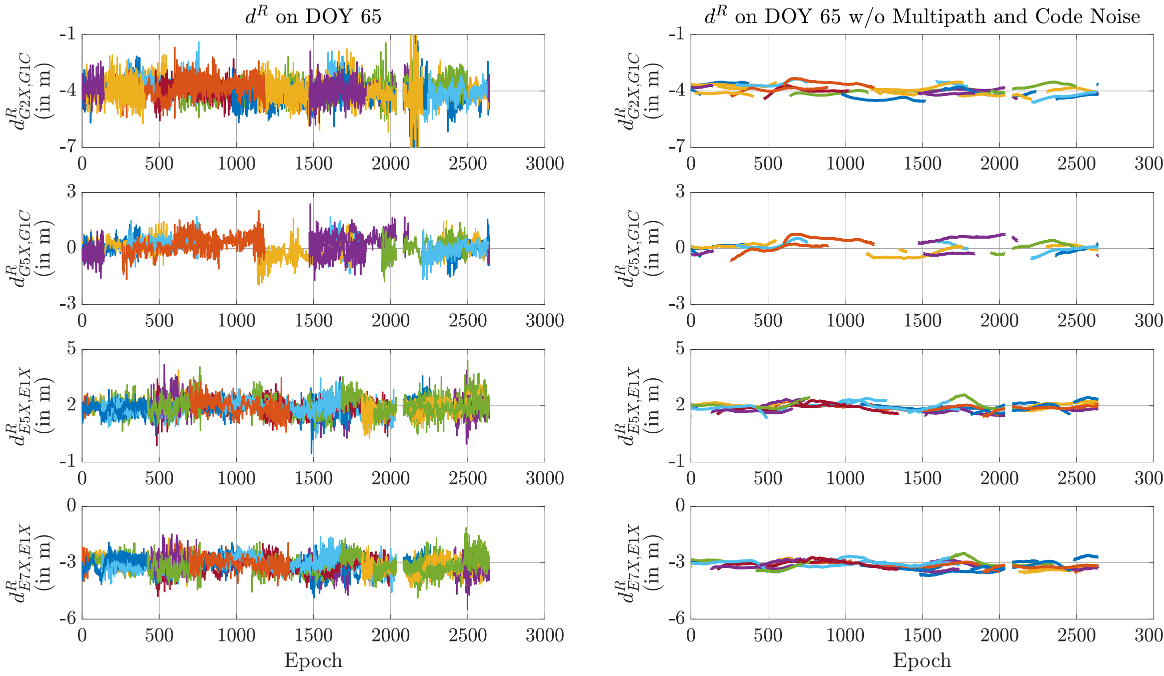

Figure B.2: Receiver-specific DCBs on DOY 65, computed with satellite-specific DCBs from DLR. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites.

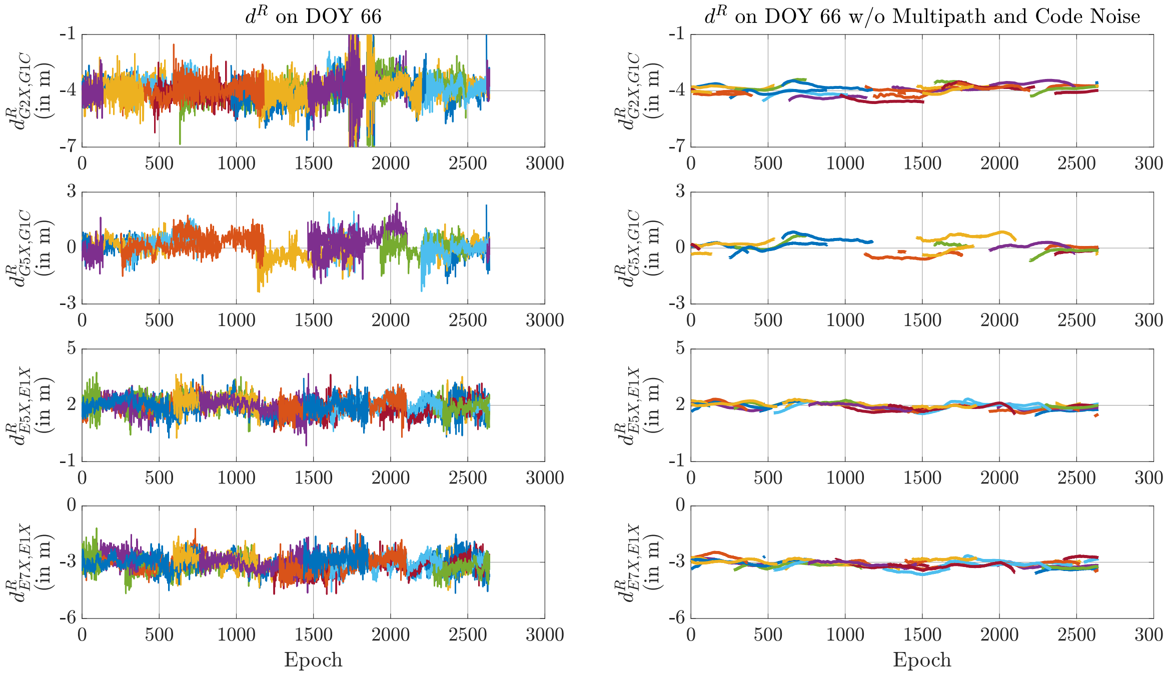

Figure B.3: Receiver-specific DCBs on DOY 66, computed with satellite-specific DCBs from DLR. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites.

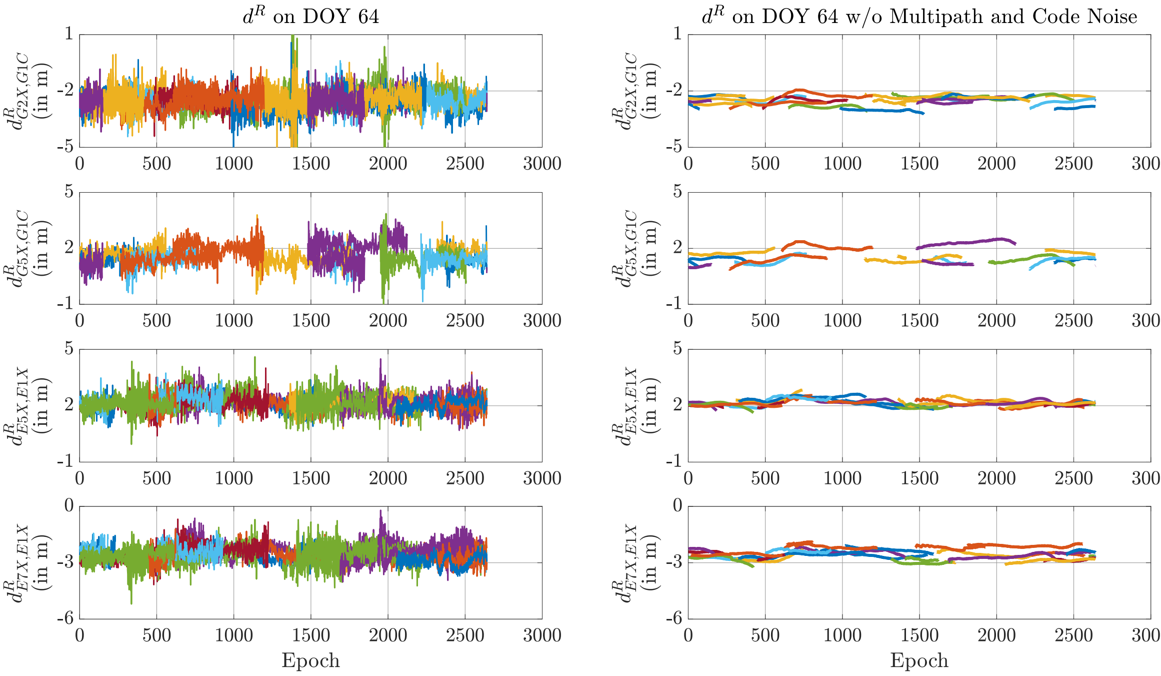

Figure B.4: Receiver-specific DCBs on DOY 64, computed with satellite-specific DCBs from the navigation data messages. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites.

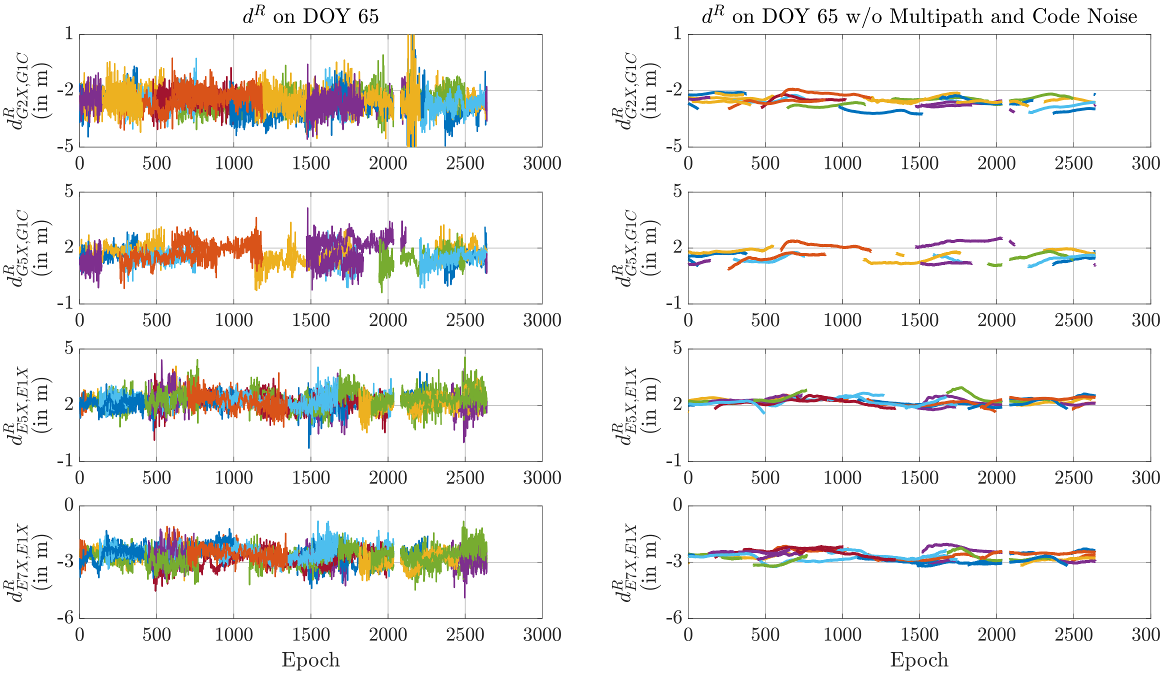

Figure B.5: Receiver-specific DCBs on DOY 65, computed with satellite-specific DCBs from the navigation data messages. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites.

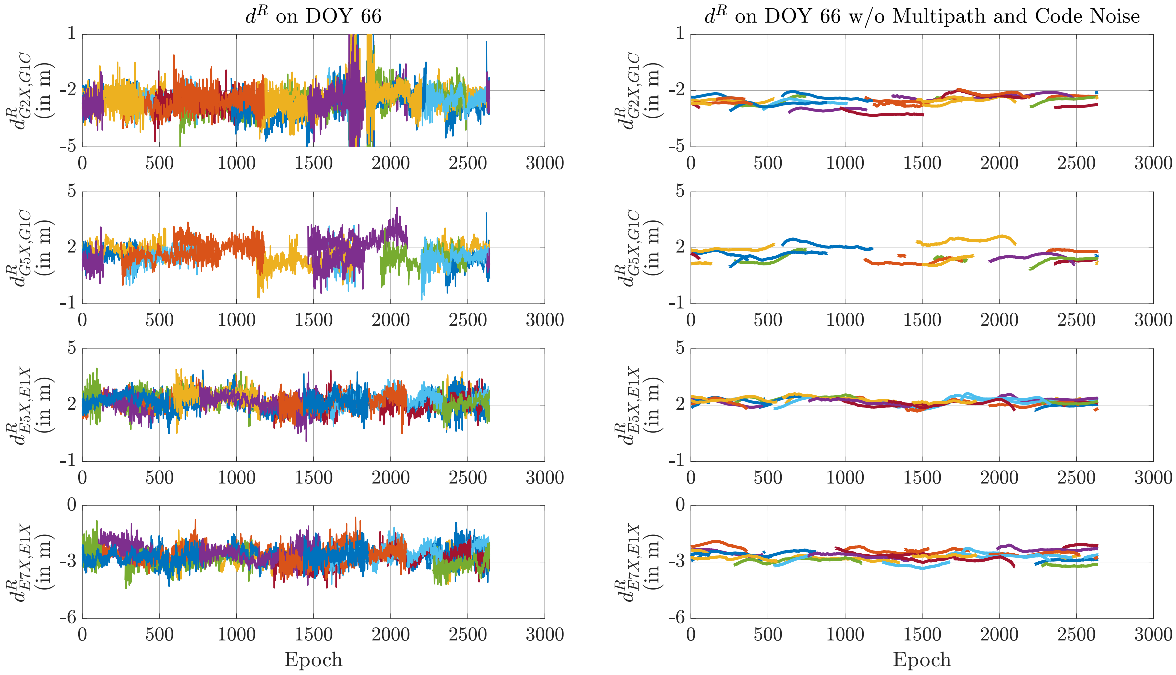

Figure B.6: Receiver-specific DCBs on DOY 66, computed with satellite-specific DCBs from the navigation data messages. The plots on the left hand side contain multipath and code noise, while these effects have been removed on the right hand side via the multipath combination. Different colors indicate different satellites.

B.2 GNSS Measurement Noise

This section contains additional figures and tables concerning the GNSS measurement noise that are not included in the corresponding Section 3.6. The results relevant to this section which are included in Section 3.6 are repeated here to facilitate comparison. Figure B.7 displays the difference between ionospheric delay obtained from IONEX and the ionospheric delay estimated from multi-frequency pseudoranges. Only one combination ($\rho_{E1X/E5X}$) is depicted in Figure 3.3 because the plots are very similar to each other. Figure B.9 shows code multipath and code tracking noise in static and kinematic data for Galileo to complement Figure 3.5/B.8, which presents the same data for GPS. Figures B.10-B.17 display the total pseudorange error: Figures B.10 and B.11 for GPS with precise orbits and clocks, Figures 3.8/B.12 and 3.9/B.13 for Galileo with precise orbits and clocks, Figures B.14 and B.15 for GPS with navigation message data as well as Figures B.16 and B.17 for Galileo with navigation message data. The corresponding numerical values are displayed in Table 3.8/B.1 for precise orbit and clock data as well as in Table B.2 for navigation message data.

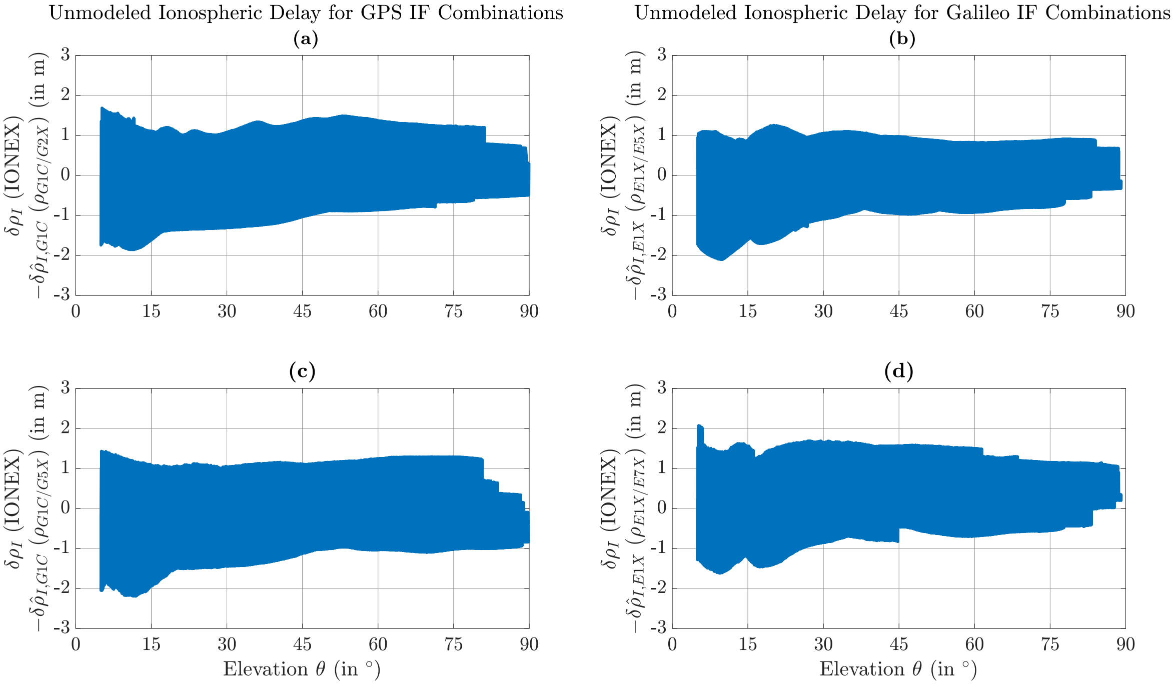

Figure B.7: Difference between reference value from IONEX and estimated value for ionospheric delay in the L1 band. **(a)** Estimated value obtained from multi-frequency pseudoranges on G1C and G2X. **(b)** Estimated value obtained from multi-frequency pseudoranges on E1X and E5X. **(c)** Estimated value obtained from multi-frequency pseudoranges on G1C and G5X. **(d)** Estimated value obtained from multi-frequency pseudoranges on E1X and E7X.

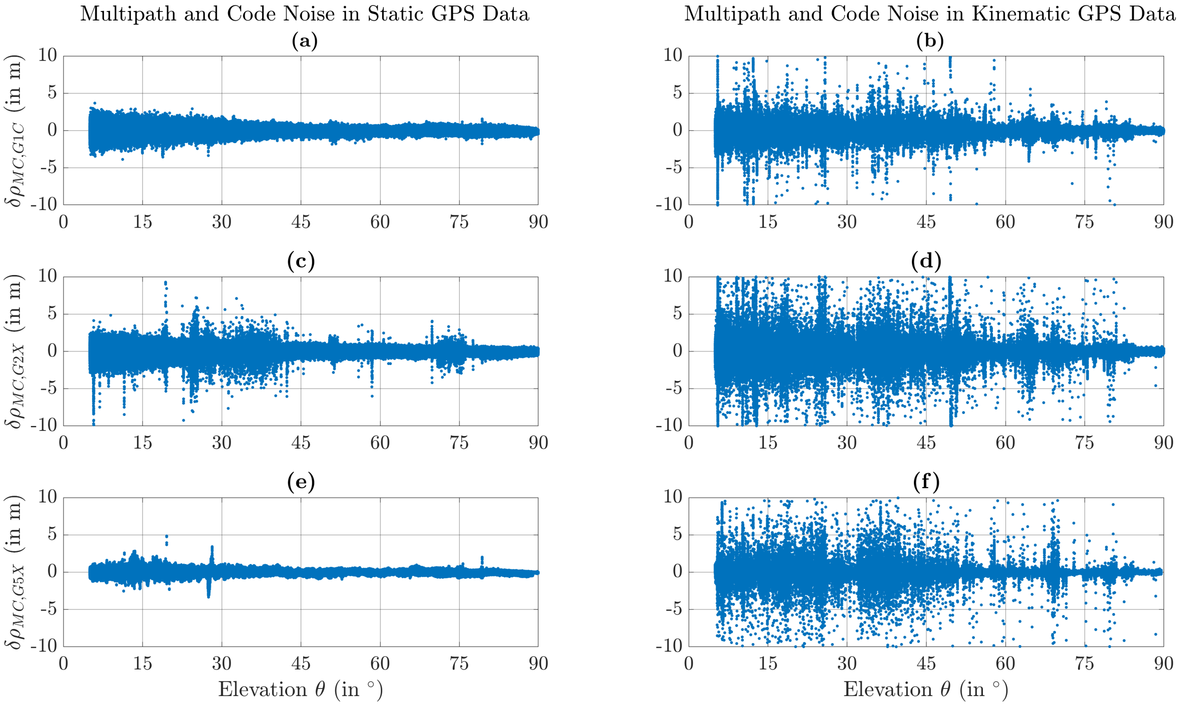

Figure B.8: Code multipath and code tracking noise for GPS, estimated via the multipath combination. **(a)**,**(c)**,**(e)** Static data for G1C, G2X and G5X. **(b)**,**(d)**,**(f)** Kinematic data for G1C, G2X and G5X (Repetition of Figure 3.5).

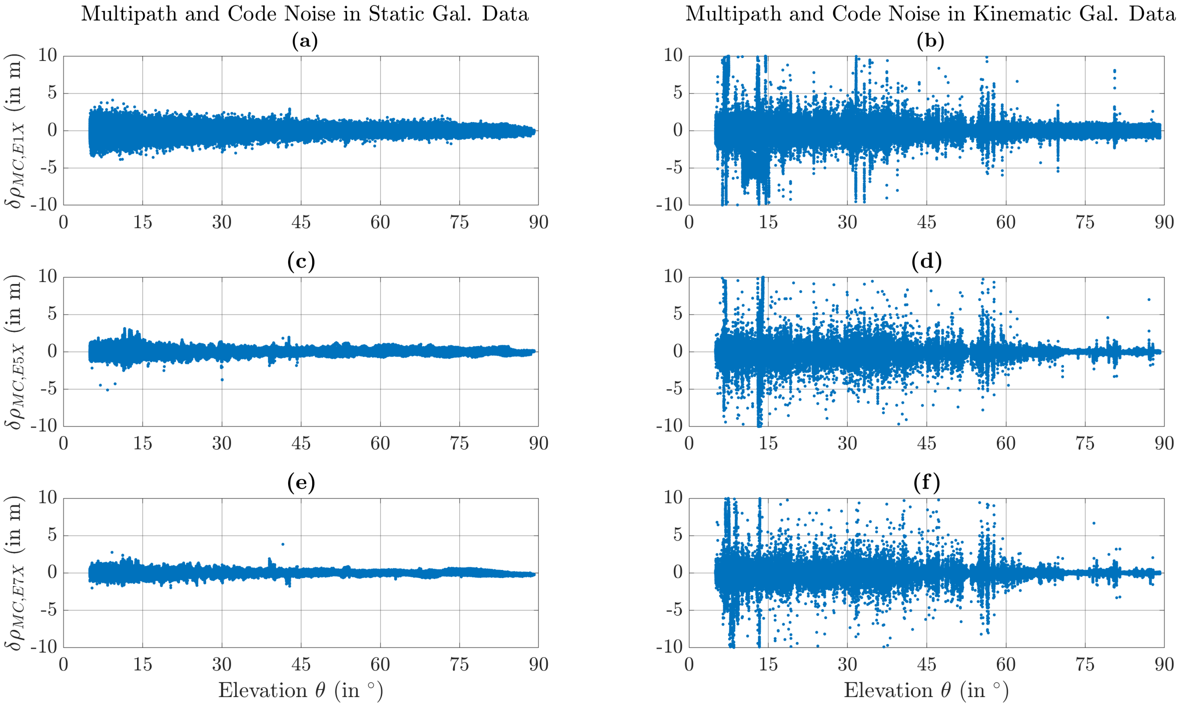

Figure B.9: Code multipath and code tracking noise for Galileo, estimated via the multipath combination. **(a)**,**(c)**,**(e)** Static data for E1X, E5X and E7X. **(b)**,**(d)**,**(f)** Kinematic data for E1X, E5X and E7X.

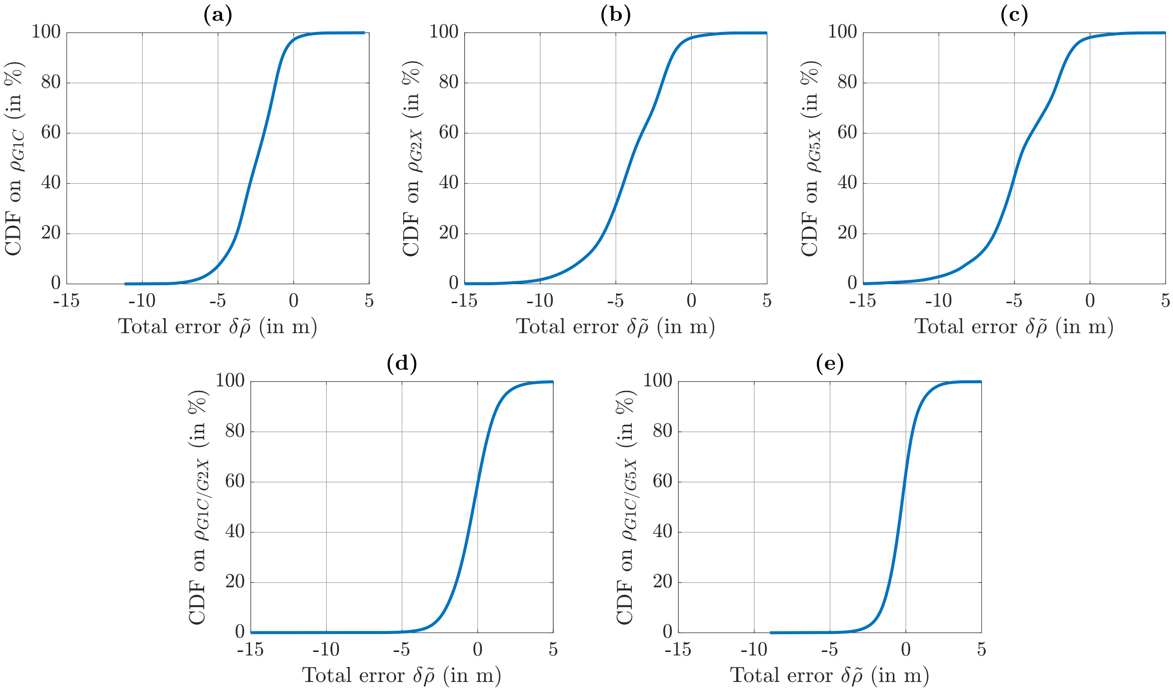

Figure B.10: CDF of total pseudorange error for GPS with predicted half of ultra-rapid precise satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on G1C, G2X and G5X. **(d)**,**(e)** IF combinations on G1C/G2X and G1C/G5X.

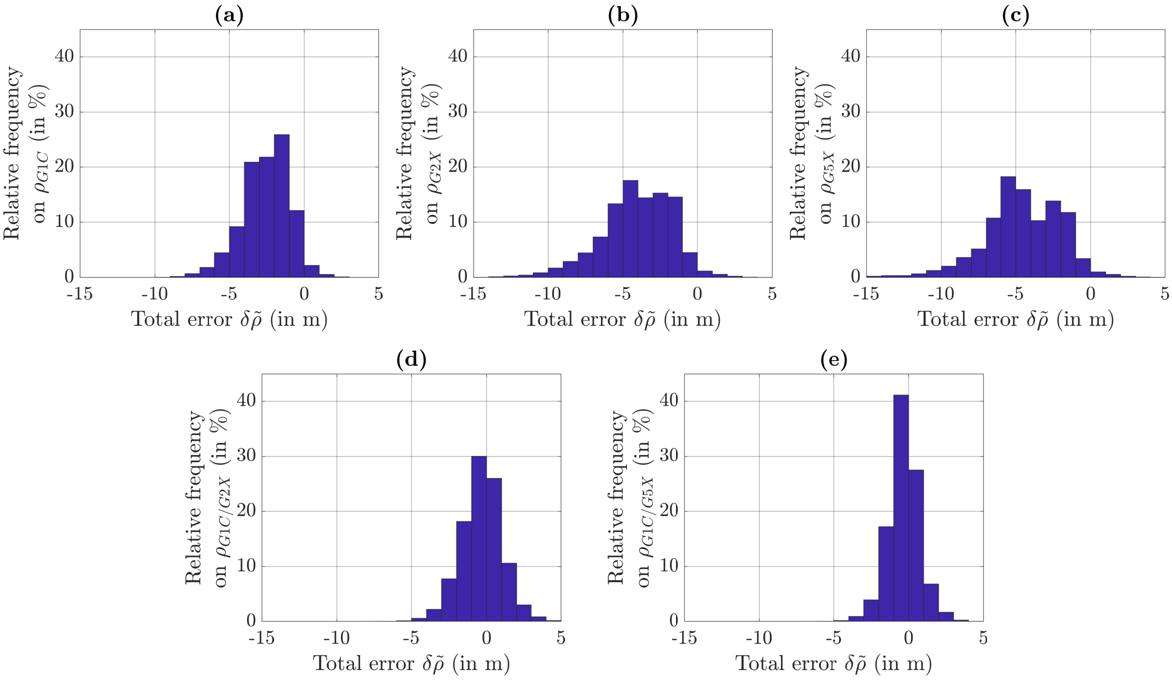

Figure B.11: Relative frequency of total pseudorange error for GPS with predicted half of ultra-rapid precise satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on G1C, G2X and G5X. **(d)**,**(e)** IF combinations on G1C/G2X and G1C/G5X.

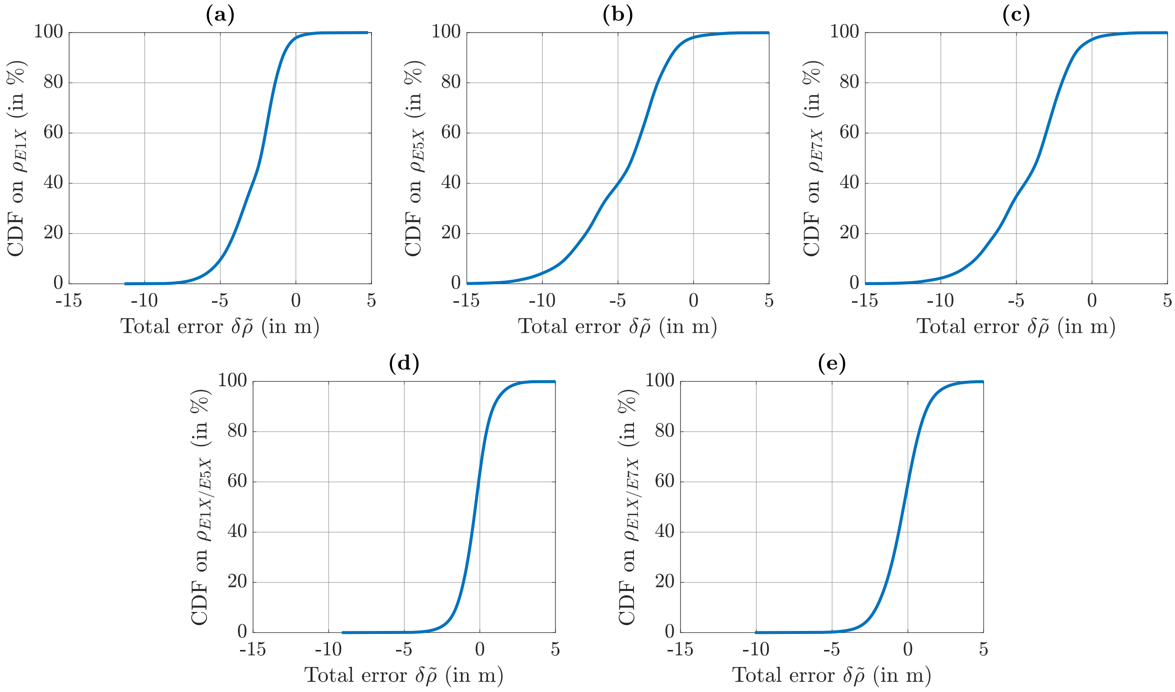

Figure B.12: CDF of total pseudorange error for Galileo with predicted half of ultra-rapid precise satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on E1X, E5X and E7X. **(d)**,**(e)** IF combinations on E1X/E5X and E1X/E7X (Repetition of Figure 3.8).

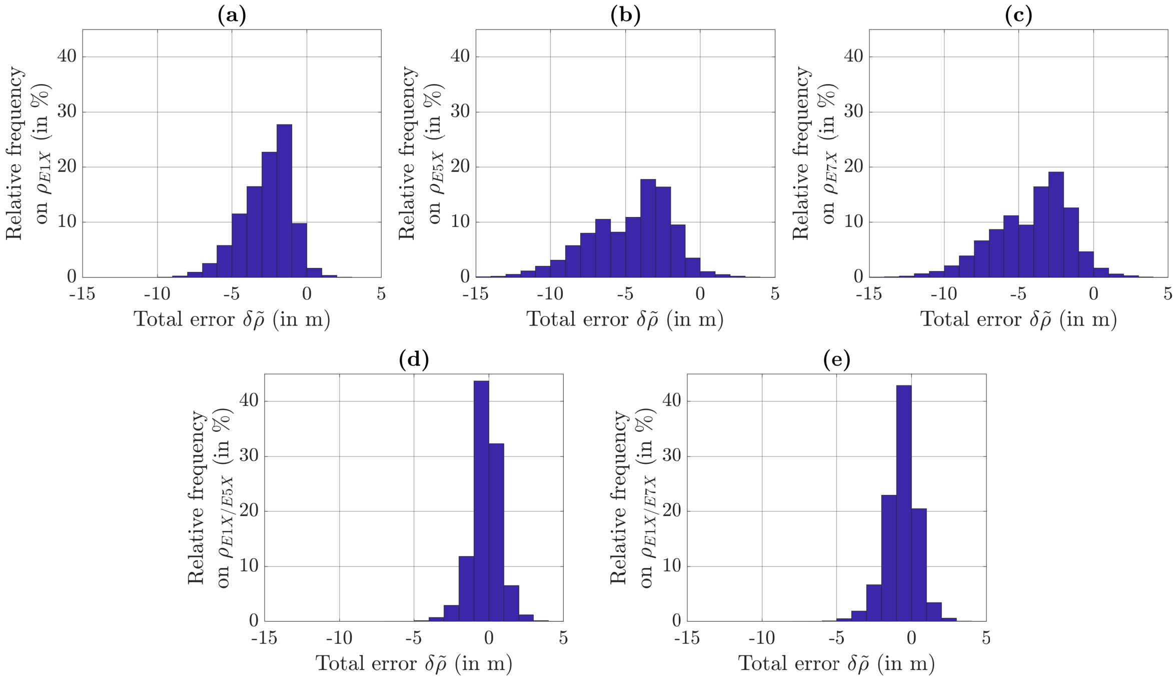

Figure B.13: Relative frequency of total pseudorange error for Galileo with predicted half of ultra-rapid precise satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on E1X, E5X and E7X. **(d)**,**(e)** IF combinations on E1X/E5X and E1X/E7X (Repetition of Figure 3.9).

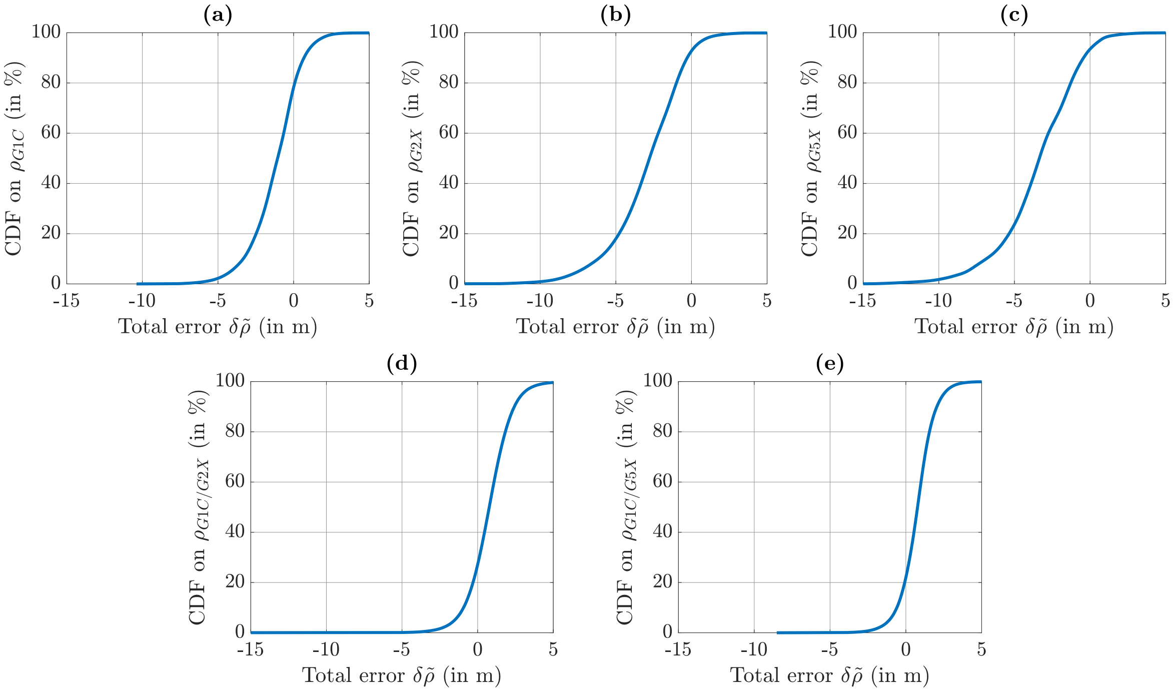

Figure B.14: CDF of total pseudorange error for GPS with LNAV satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on G1C, G2X and G5X. **(d)**,**(e)** IF combinations on G1C/G2X and G1C/G5X.

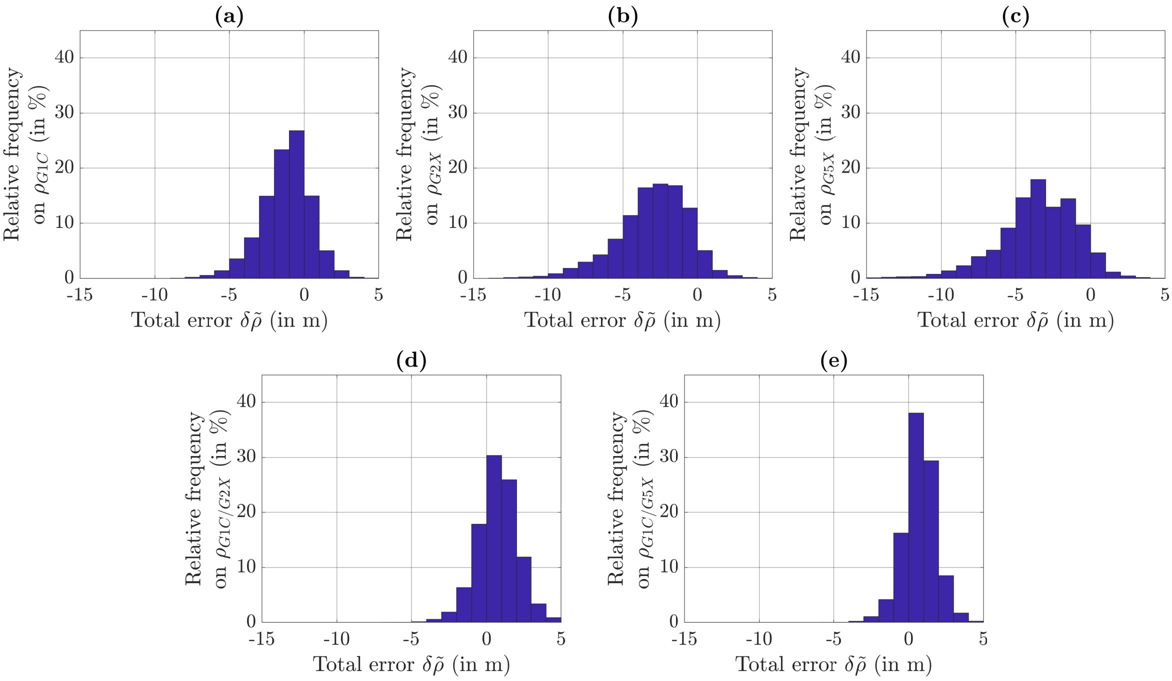

Figure B.15: Relative frequency of total pseudorange error for GPS with LNAV satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on G1C, G2X and G5X. **(d)**,**(e)** IF combinations on G1C/G2X and G1C/G5X.

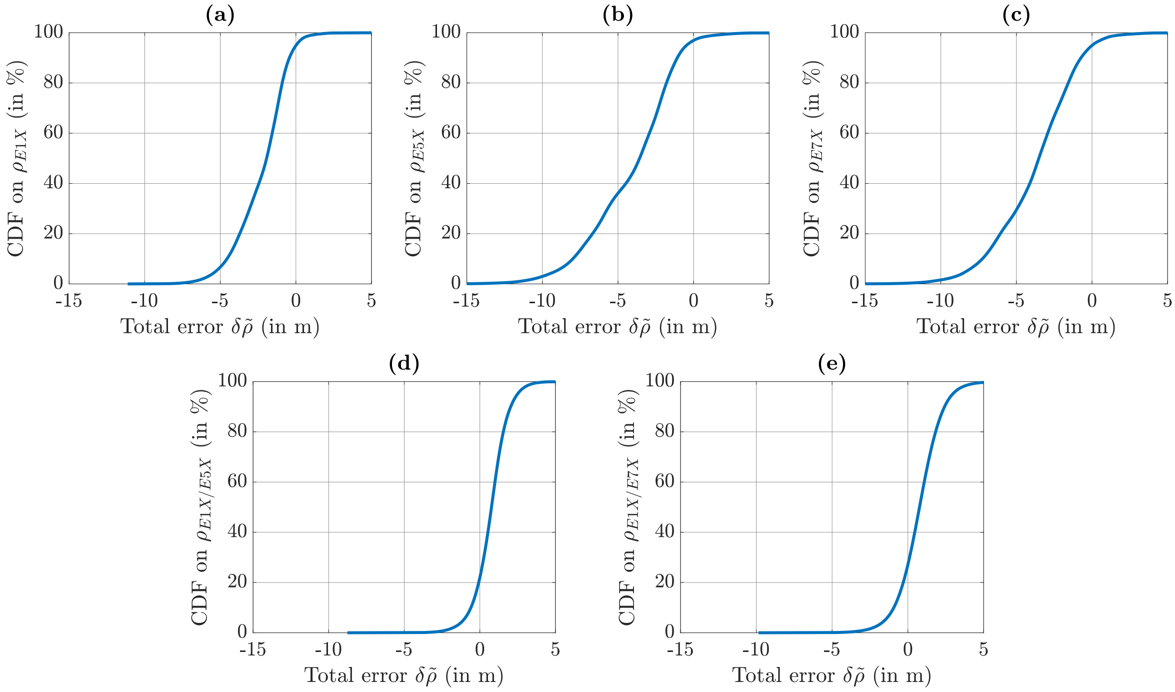

Figure B.16: CDF of total pseudorange error for Galileo with F/NAV satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on E1X, E5X and E7X. **(d)**,**(e)** IF combinations on E1X/E5X and E1X/E7X.

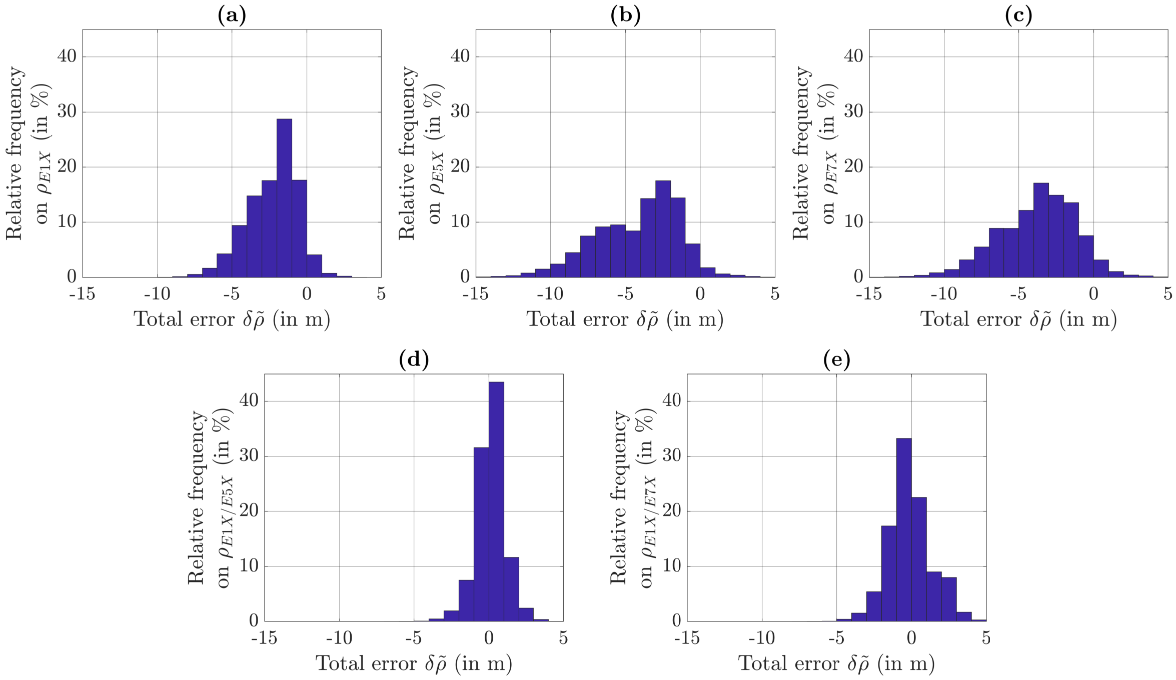

Figure B.17: Relative frequency of total pseudorange error for Galileo with F/NAV satellite orbits and clocks, corrected for all applicable DCBs. Only static data to exclude large multipath errors. **(a)**,**(b)**,**(c)** Single-frequency observations on E1X, E5X and E7X. **(d)**,**(e)** IF combinations on E1X/E5X and E1X/E7X.

Table B.1: Total Pseudorange Error With Predicted Half of Ultra-Rapid Precise Satellite Orbits and Clocks, Corrected for all Applicable DCBs. Only Static Data to Exclude Large Multipath Errors (Repetition of Table 3.8)

Signal	Mean (in m)	Standard Deviation (in m)	RMS (in m)
G1C	−2.55	1.58	3.00
G2X	−4.04	2.38	4.69
G5X	−4.53	2.56	5.20
G1C/G2X	−0.33	1.41	1.44
G1C/G5X	−0.31	1.06	1.11
E1X	−2.70	1.64	3.16
E5X	−4.13	2.60	4.88
E7X	−4.67	2.79	5.44
E1X/E5X	−0.66	1.08	1.26
E1X/E7X	−0.66	1.08	1.26

Table B.2: Total Pseudorange Error With Broadcast Satellite Orbits and Clocks, Corrected for all Applicable DCBs. Only Static Data to Exclude Large Multipath Errors

Signal	Mean (in m)	Standard Deviation (in m)	RMS (in m)
G1C	−1.21	1.65	2.04
G2X	−3.00	2.40	3.84
G5X	−3.49	2.60	4.35
G1C/G2X	0.75	1.40	1.59
G1C/G5X	0.74	1.10	1.33
E1X	−2.27	1.67	2.82
E5X	−3.78	2.61	4.59
E7X	−4.16	2.82	5.03
E1X/E5X	−0.12	1.46	1.46
E1X/E7X	−0.12	1.46	1.46

B.3 Odometry Measurement Noise

This sections contains additional figures regarding the odometry measurement noise that are not included in the corresponding Section 4.3. The results relevant to this section which are included in Section 4.3 are repeated here to facilitate comparison. Figure B.19 shows the side slip angle estimation accuracy of the linear tire model when μ_y is obtained from gyroscope measurements to complement Figure 4.7/B.18, which presents the same data for the case that μ_y is obtained from accelerometer measurements. Figure B.21 displays the longitudinal velocity error for the right wheels. It is very similar to Figure 4.10/B.20, which depicts the same for the left wheels, because the estimated longitudinal slip is identical for both wheels of each axle.

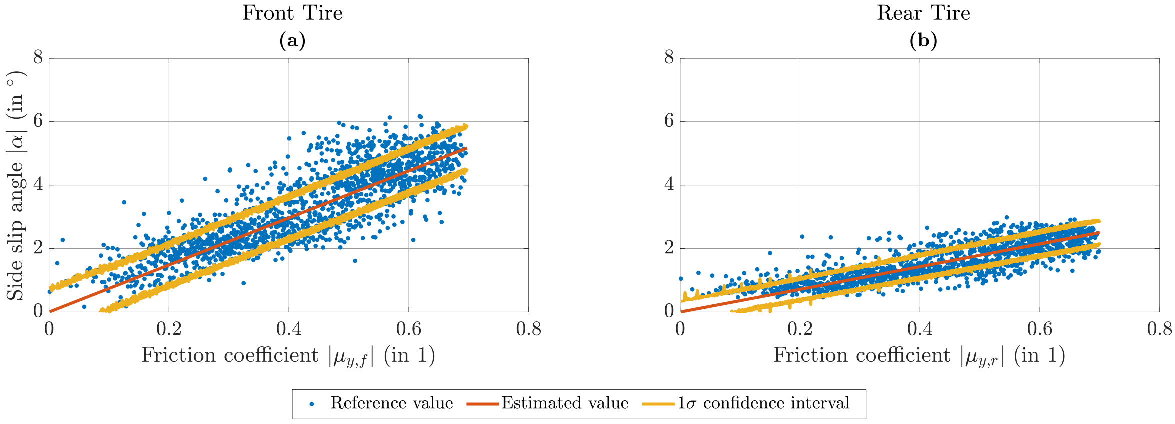

Figure B.18: Side slip angle estimation accuracy with linear tire model, based on accelerometer measurements. **(a)** Front tire. **(b)** Rear tire (Repetition of Figure 4.7).

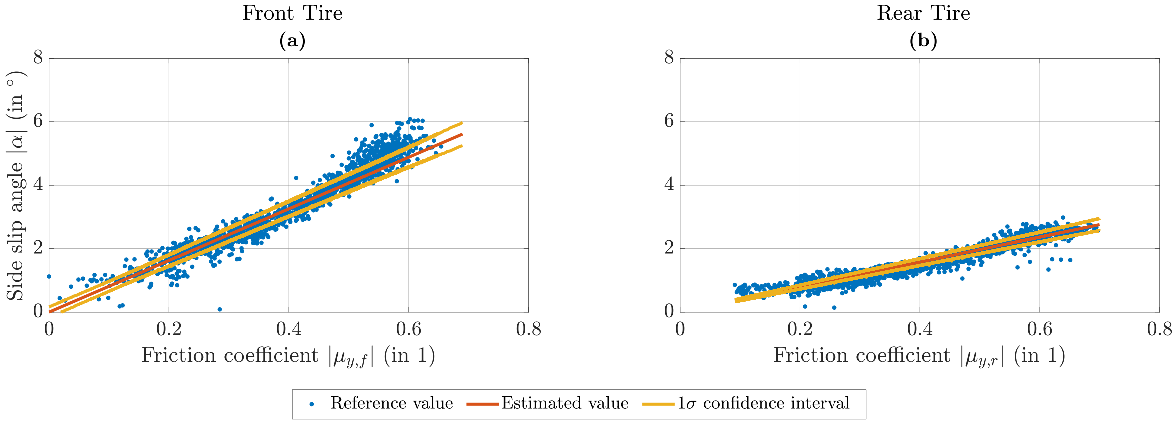

Figure B.19: Side slip angle estimation accuracy with linear tire model, based on gyroscope measurements. **(a)** Front tire. **(b)** Rear tire.

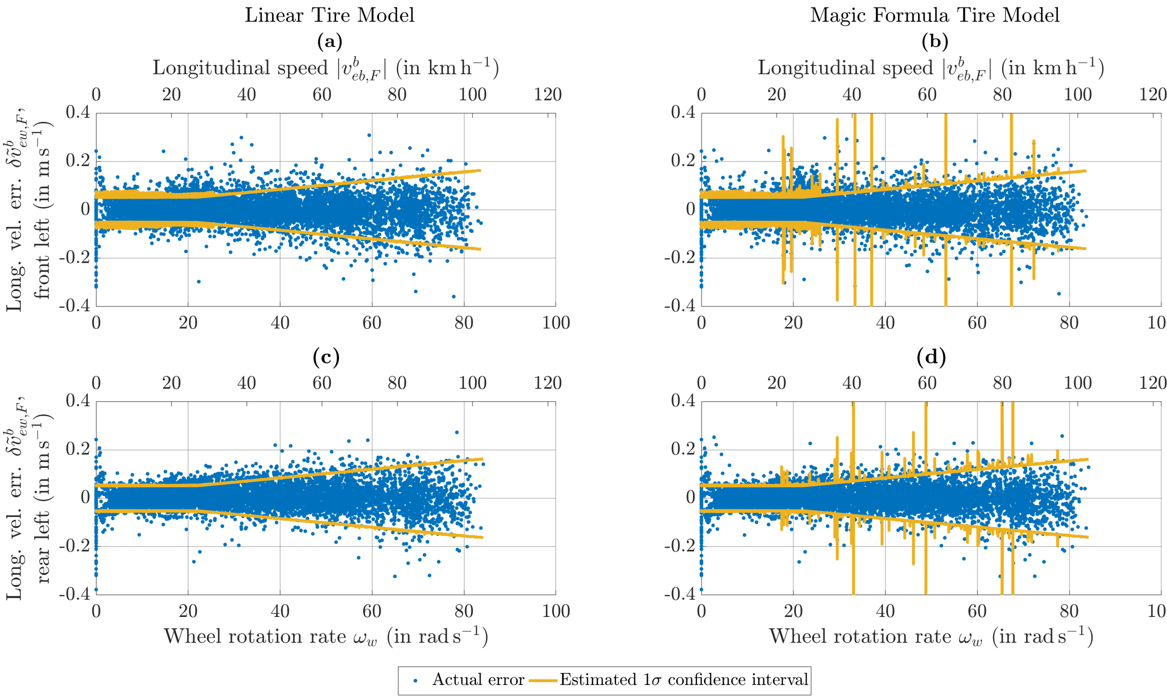

Figure B.20: Longitudinal odometry velocity error, front and rear left wheel. **(a)**,**(c)** Linear tire model. The additional horizontal axis at the top is based on the arithmetic mean of the four dynamic tire radii r_d and is included for visualization purposes only. **(b)**,**(d)** Magic Formula tire model (Repetition of Figure 4.10).

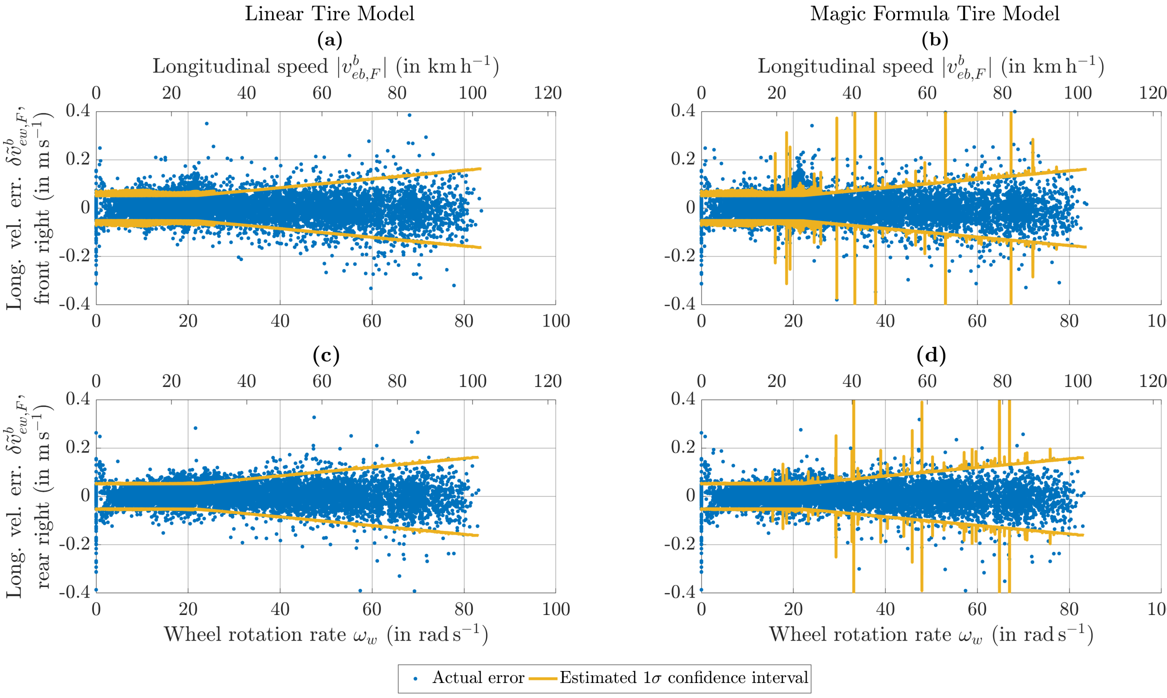

Figure B.21: Longitudinal odometry velocity error, front and rear right wheel. **(a)**,**(c)** Linear tire model. The additional horizontal axis at the top is based on the arithmetic mean of the four dynamic tire radii r_d and is included for visualization purposes only. **(b)**,**(d)** Magic Formula tire model.

B.4 Test Scenario Results

This section presents additional results from the evaluation of the test scenarios described in Chapter 7. Only one run of each scenario is featured in that chapter, while this appendix section contains all runs. The results are mostly listed without further description. However, values that deviate considerably from the ones in Chapter 7 are set in bold face and explained briefly.

B.4.1 Position Accuracy

All metrics regarding position accuracy are included in this section. Figures B.22-B.35 depict the position error CDFs for all runs of each scenario. Two different sets of scales are employed for the horizontal axes: The first set is featured in Figures B.22-B.27, which depict the results obtained at Griesheim airfield. The second set is featured in Figures B.28-B.35, which depict the results obtained in the Odenwald, in Darmstadt and in Frankfurt. This scale adjustment is performed because the error's magnitude varies due to the dissimilar GNSS reception conditions at Griesheim airfield on the one hand and during the remaining test scenarios on the other hand. However, the application of the same scales for the horizontal axes for each run of a specific scenario still allows for an easy assessment of the position error's repeatability. In the second part of the section, the numerical values for the evaluated position accuracy metrics are listed in Tables B.3-B.6. The tables compare the accuracy achieved by the multi-frequency/multi-constellation algorithm with the accuracy achieved by the single-frequency GPS L1 C/A algorithm, just as it is done in Section 7.2.2. Each table features the results from all runs of a specific test scenario.

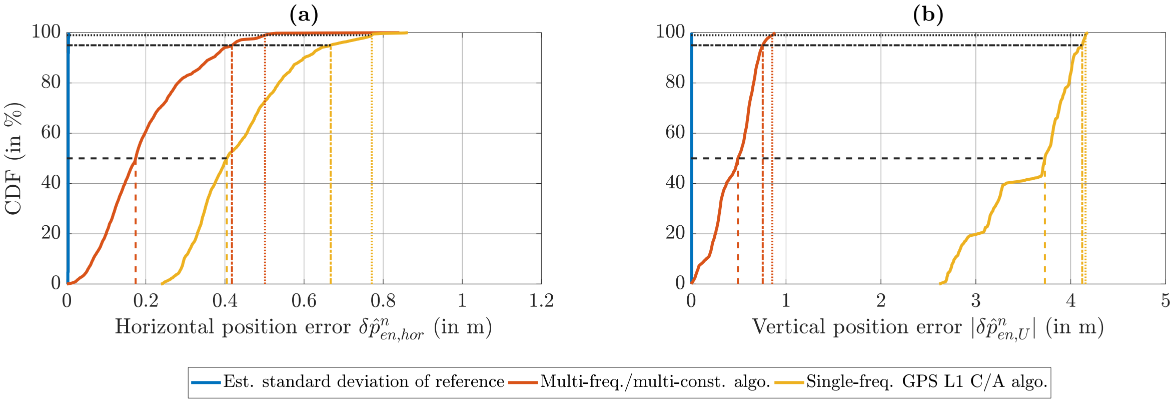

Figure B.22: Position error CDF for Griesheim airfield data set, moderate driving style, run 1. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

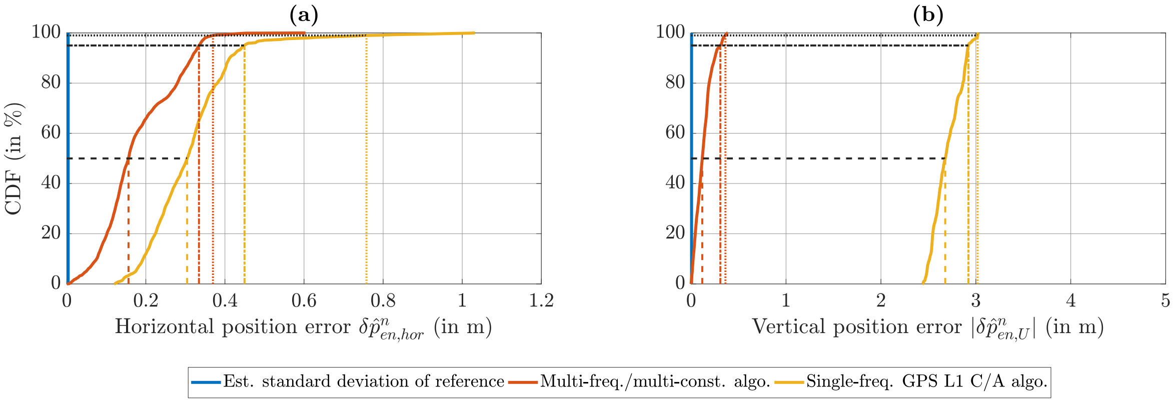

Figure B.23: Position error CDF for Griesheim airfield data set, moderate driving style, run 2. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error (Repetition of Figure 7.1).

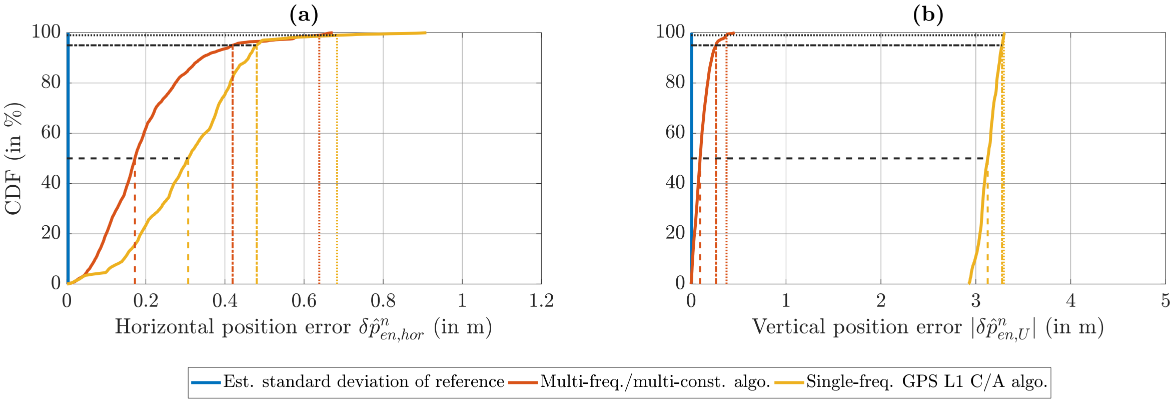

Figure B.24: Position error CDF for Griesheim airfield data set, moderate driving style, run 3. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

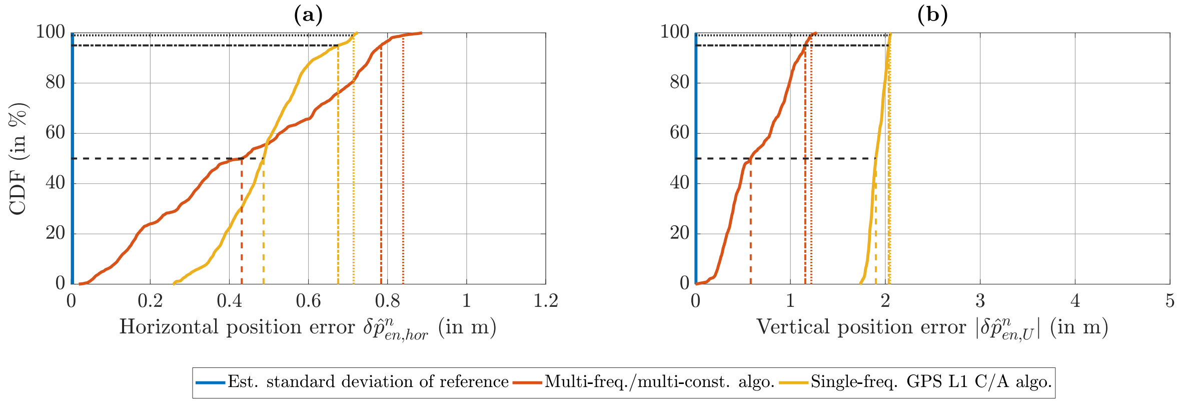

Figure B.25: Position error CDF for Griesheim airfield data set, dynamic driving style, run 1. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

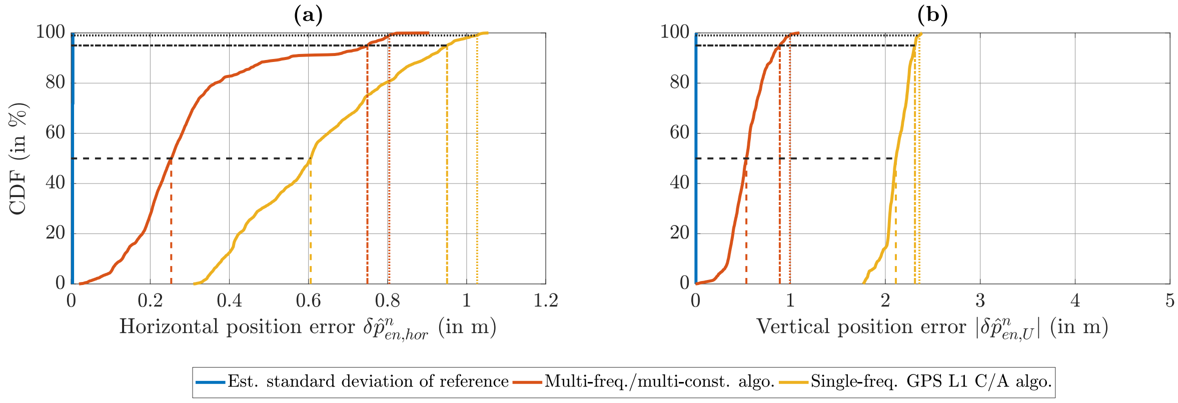

Figure B.26: Position error CDF for Griesheim airfield data set, dynamic driving style, run 2. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

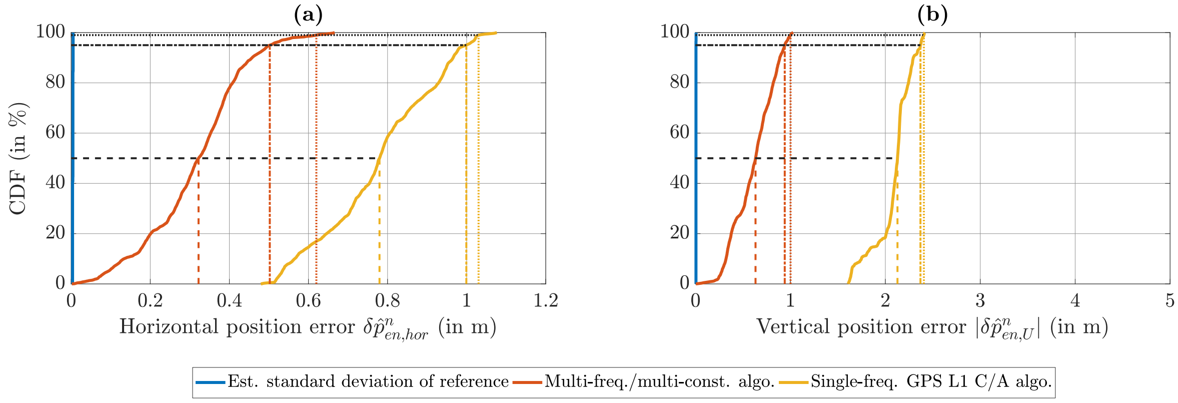

Figure B.27: Position error CDF for Griesheim airfield data set, dynamic driving style, run 3. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error (Repetition of Figure 7.2).

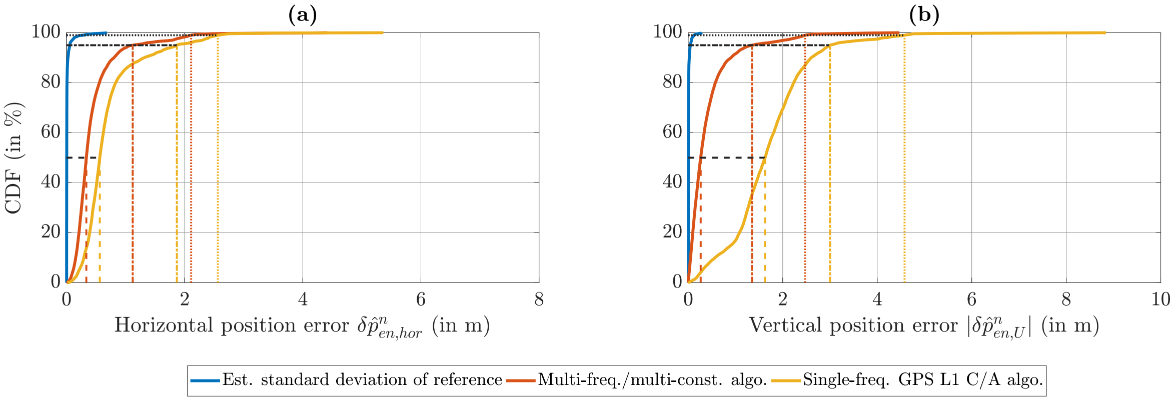

Figure B.28: Position error CDF for Odenwald data set, run 1. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error (Repetition of Figure 7.3).

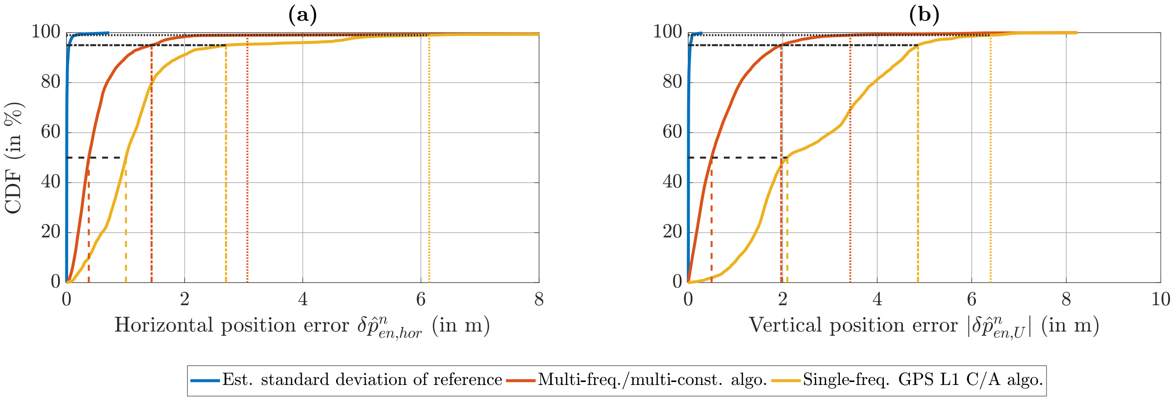

Figure B.29: Position error CDF for Odenwald data set, run 2. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

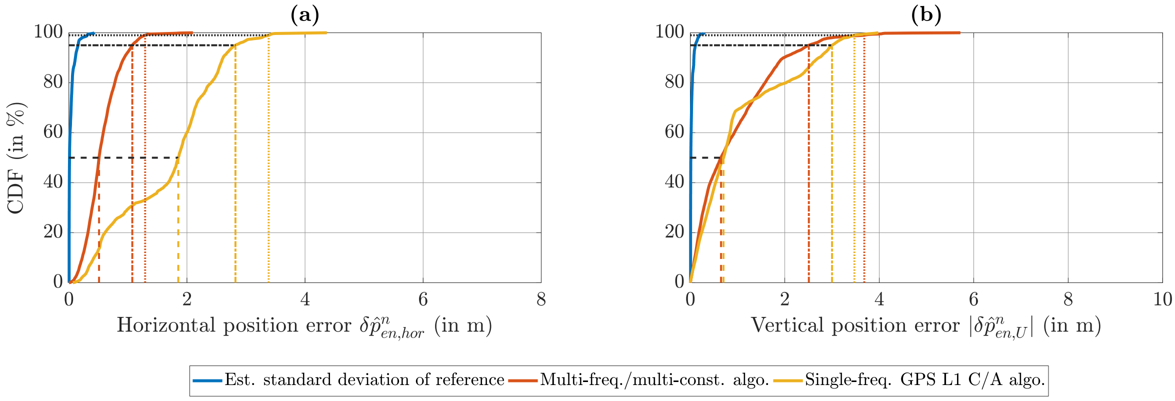

Figure B.30: Position error CDF for Darmstadt data set, run 1. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

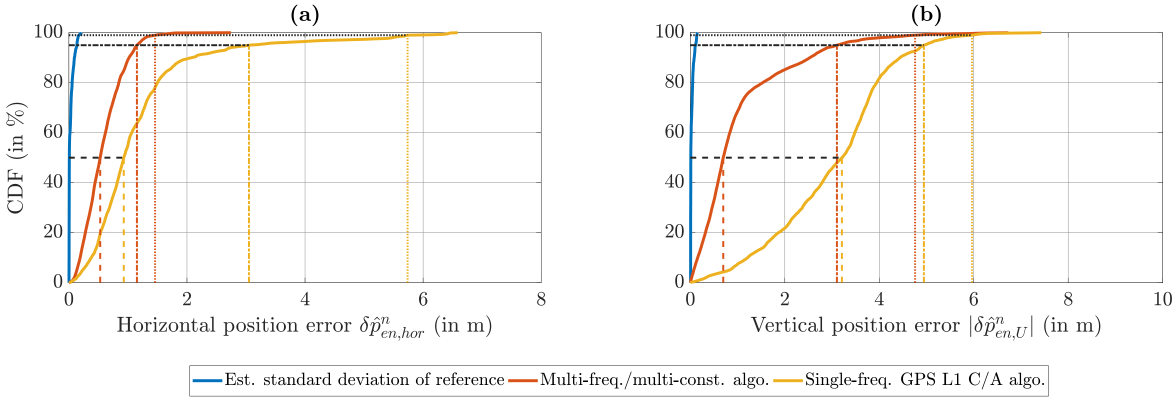

Figure B.31: Position error CDF for Darmstadt data set, run 2. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error (Repetition of Figure 7.4).

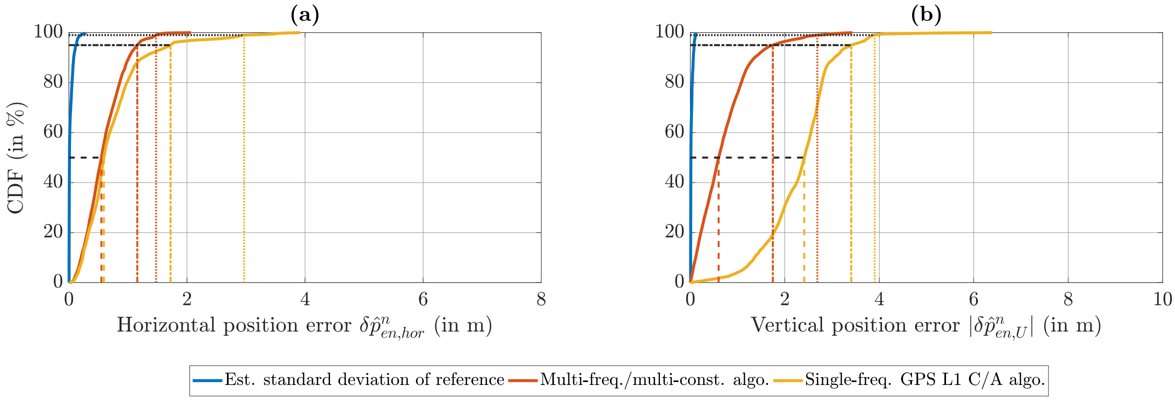

Figure B.32: Position error CDF for Darmstadt data set, run 3. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

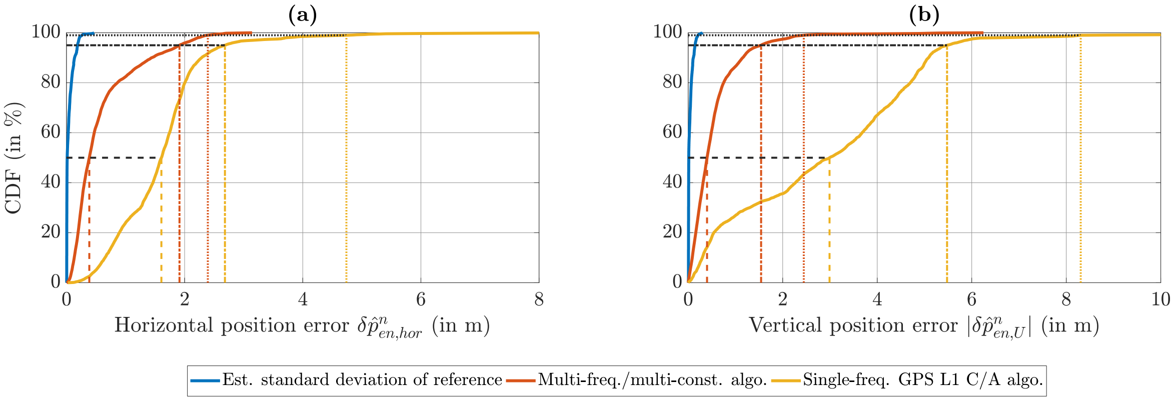

Figure B.33: Position error CDF for Frankfurt data set, run 1. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error (Repetition of Figure 7.5).

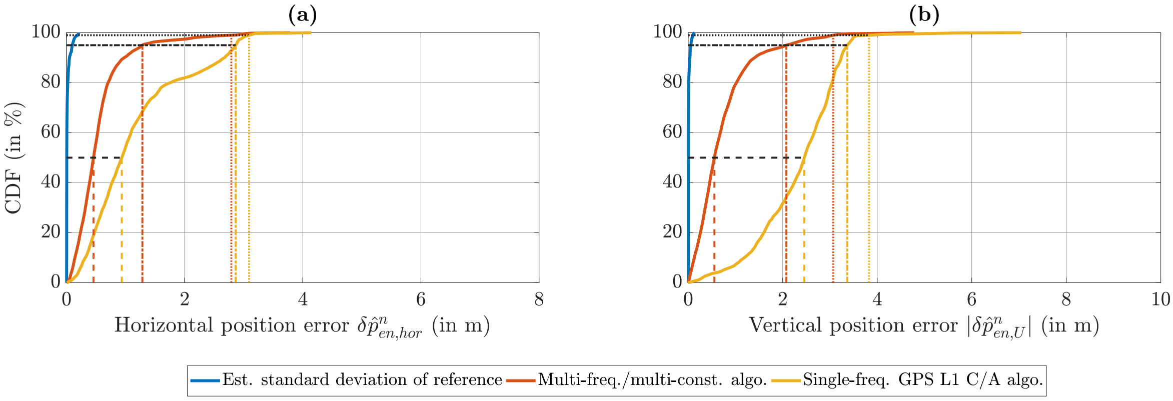

Figure B.34: Position error CDF for Frankfurt data set, run 2. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

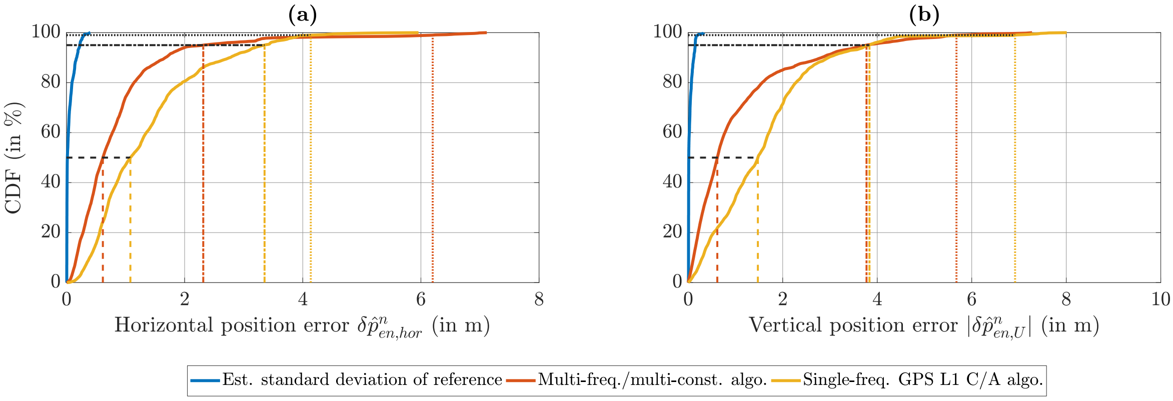

Figure B.35: Position error CDF for Frankfurt data set, run 3. 50 %, 95 % and 99 % error quantiles are indicated. **(a)** Horizontal error. **(b)** Vertical error.

Table B.3: Position Accuracy Metrics for Griesheim Airfield Data Sets. Metrics for Each Scenario: Position Error RMS (1st Row), 95 % Quantile of Horizontal and Vertical Position Error (2nd Row), 95 % Quantile of Estimated Reference Standard Deviation in Horizontal and Vertical Direction (3rd Row)

Metric	**Unit**	**Multi-Frequency/ Multi-Constellation**	**Single-Frequency GPS L1 C/A only**
Moderate driving style, run 1			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.18; 0.13; 0.50]$	$[0.35; 0.29; 3.56]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	0.42/0.75	0.67/4.12
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Moderate driving style, run 2 (repetition from Table 7.1)			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.11; 0.17; 0.15]$	$[0.16; 0.29; 2.70]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	0.33/0.31	0.45/2.92
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Moderate driving style, run 3			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.13; 0.19; 0.14]$	$[0.23; 0.24; 3.13]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	0.42/0.26	0.48/3.28
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Dynamic driving style, run 1			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.41; 0.28; 0.72]$	$[0.41; 0.28; 1.91]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	0.78/1.16	0.68/2.04
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Dynamic driving style, run 2			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.21; 0.27; 0.58]$	$[0.60; 0.24; 2.12]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	0.75/0.89	0.95/2.31
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	
Dynamic driving style, run 3 (repetition from Table 7.1)			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.17; 0.29; 0.65]$	$[0.66; 0.44; 2.10]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	0.50/0.94	1.00/2.37
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	$< 0.01/0.01$	

Table B.4: Position Accuracy Metrics for Odenwald Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Metrics for Each Scenario: Position Error RMS (1st Row), 95 % Quantile of Horizontal and Vertical Position Error (2nd Row), 95 % Quantile of Estimated Reference Standard Deviation in Horizontal and Vertical Direction (3rd Row)

Metric	Unit	Multi-Frequency/ Multi-Constellation	Single-Frequency GPS L1 C/A only
Run 1 (repetition from Table 7.1)			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.34; 0.49; 0.65]$	$[0.39; 0.77; 1.89]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%}\big/\lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.12/1.35	1.87/3.00
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.06/0.05	
Run 2			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.55; \mathbf{1.98}; 1.02]$	$[0.89; \mathbf{2.19}; 2.96]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%}\big/\lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.44/1.97	2.70/4.86
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%}\big/\lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.05/0.04	

In comparison with the first run, the position error RMS in northern direction is much larger for the second Odenwald data set in Table B.4. This applies to both the multi-frequency/multi-constellation configuration (0.49 m vs. 1.98 m) as well as the single-frequency/single-constellation configuration (0.77 m vs. 2.19 m). This increase stems from the algorithm's behavior inside a long tunnel (Saukopftunnel, 2.7 km long). As depicted in Figure 7.7, the tunnel runs in east-west direction. Consequently, a lateral position drift causes the error in northern direction to grow. While the maximum error in northern direction stays below 5 m for both configurations in the first run, it reaches up to 40 m in the second run, thereby influencing the overall RMS heavily. The large lateral position error stems from an incorrectly estimated gyroscope bias along the z^b-axis, which leads to a yaw angle error that grows over time. In the first run, the yaw error's absolute value inside the tunnel does not exceed 0.3° for both configurations. In the second run, it grows to 1.0° for the single-frequency configuration and even 1.8° for the multi-frequency/multi-constellation configuration. As soon as the vehicle leaves the tunnel, both the yaw error as well as the position error get corrected within a few measurement updates.

Table B.5: Position Accuracy Metrics for Darmstadt Data Sets. Metrics for Each Scenario: Position Error RMS (1^{st} Row), 95 % Quantile of Horizontal and Vertical Position Error (2^{nd} Row), 95 % Quantile of Estimated Reference Standard Deviation in Horizontal and Vertical Direction (3^{rd} Row)

Metric	**Unit**	**Multi-Frequency/ Multi-Constellation**	**Single-Frequency GPS L1 C/A only**
Run 1			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.44; 0.44; 1.22]$	$[0.29; 1.83; 1.41]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%}\Big/\lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.07/2.51	2.82/3.00
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%}\Big/\lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.16/0.11	
Run 2 (repetition from Table 7.1)			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.40; 0.53; 1.43]$	$[1.11; 1.12; 3.25]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%}\Big/\lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.15/3.10	3.05/4.95
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%}\Big/\lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.14/0.11	
Run 3			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.50; 0.45; 0.90]$	$[0.38; 0.82; 2.42]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%}\Big/\lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.16/1.75	1.72/3.41
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%}\Big/\lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.12/0.08	

Table B.6: Position Accuracy Metrics for Frankfurt Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Metrics for Each Scenario: Position Error RMS (1$^{\text{st}}$ Row), 95 % Quantile of Horizontal and Vertical Position Error (2$^{\text{nd}}$ Row), 95 % Quantile of Estimated Reference Standard Deviation in Horizontal and Vertical Direction (3$^{\text{rd}}$ Row)

Metric	**Unit**	**Multi-Frequency/ Multi-Constellation**	**Single-Frequency GPS L1 C/A only**
Run 1 (repetition from Table 7.1)			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.39; 0.71; 0.77]$	$[1.12; 1.38; 3.53]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.91/1.54	2.68/5.48
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.19/0.15	
Run 2			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.43; 0.57; 0.95]$	$[0.50; 1.33; 2.47]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	1.29/2.08	2.86/3.37
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.10/0.06	
Run 3			
RMS of $\delta\hat{\mathbf{p}}^n_{en}$	m	$[0.51; \mathbf{1.18}; \mathbf{1.62}]$	$[0.65; 1.49; 1.98]$
$\lvert\delta\hat{p}^n_{en,hor}\rvert_{95\,\%} \big/ \lvert\delta\hat{p}^n_{en,U}\rvert_{95\,\%}$	m	2.32/**3.77**	3.35/3.84
$\lvert\hat{\sigma}_{\breve{p}^n_{en,hor}}\rvert_{95\,\%} \big/ \lvert\hat{\sigma}_{\breve{p}^n_{en,U}}\rvert_{95\,\%}$	m	0.23/0.15	

In comparison with the other runs, the position error RMS in northern and vertical direction as well as the vertical error's 95 % quantile are significantly larger for the third Frankfurt data set in Table B.6. This only applies to the multi-frequency/multi-constellation configuration, while the single-frequency/single-constellation configuration achieves better vertical accuracy in the third run than it does in the first run, which is featured in the main part. The increased vertical error of the multi-frequency/multi-constellation configuration probably stems from one or more pseudorange measurements which are contaminated by multipath errors too small to be rejected via outlier detection before they corrupt the position solution. Both the error in northern direction as well as the one in vertical direction exceed 3 m for a time span of approx. 4 min during a stretch of 500 m inside one of the deepest urban canyons, with maximum errors of 7 m for both components. As soon as the vehicle leaves the canyon, the error drops to values below 2 m in all components.

B.4.2 Velocity Accuracy

All metrics regarding velocity accuracy are included in this section. Each of the Tables B.7-B.10 features the results from all runs of a specific test scenario. Just as in Section 7.2.3, four different odometry configurations are listed alongside each other: Magic Formula tire model with non-zero correlation in the odometry measurement noise covariance matrix $\mathbf{R}_o$, linear tire model with non-zero correlation in $\mathbf{R}_o$, linear tire model with uncorrelated measurement noise in $\mathbf{R}_o$ and odometry update turned off completely.

Table B.7: Velocity Accuracy Metrics for Griesheim Airfield Data Sets. Metrics for Each Scenario: Velocity Error RMS (1st Row), 95 % Quantile of Horizontal Velocity Error (2nd Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Moderate driving style, run 1					
RMS of $\delta\hat{\mathbf{v}}^b_{eb}$	$\mathrm{cm\,s^{-1}}$	$[1.4; 1.7; 4.0]$	$[1.4; 1.9; 3.9]$	$[1.6; 3.0; 3.6]$	$[1.4; 35.9; 3.5]$
$\lvert\delta\hat{v}^b_{eb,hor}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.0	4.5	5.9	91.0
Moderate driving style, run 2 (repetition from Table 7.2)					
RMS of $\delta\hat{\mathbf{v}}^b_{eb}$	$\mathrm{cm\,s^{-1}}$	$[1.4; 2.0; 3.5]$	$[1.5; 2.2; 3.5]$	$[1.7; 3.2; 3.4]$	$[1.2; 34.3; 3.8]$
$\lvert\delta\hat{v}^b_{eb,hor}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.5	5.0	6.5	59.3
Moderate driving style, run 3					
RMS of $\delta\hat{\mathbf{v}}^b_{eb}$	$\mathrm{cm\,s^{-1}}$	$[1.5; 2.2; 4.2]$	$[1.5; 2.5; 4.2]$	$[1.8; 3.7; 4.2]$	$[1.1; 31.2; 4.4]$
$\lvert\delta\hat{v}^b_{eb,hor}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.9	5.5	7.3	44.6
Dynamic driving style, run 1					
RMS of $\delta\hat{\mathbf{v}}^b_{eb}$	$\mathrm{cm\,s^{-1}}$	$[2.7; 3.2; 6.0]$	$[2.7; 3.7; 6.0]$	$[2.9; 5.6; 6.7]$	$[1.3; 34.8; 5.4]$
$\lvert\delta\hat{v}^b_{eb,hor}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	7.9	8.9	11.3	52.7
Dynamic driving style, run 2					
RMS of $\delta\hat{\mathbf{v}}^b_{eb}$	$\mathrm{cm\,s^{-1}}$	$[2.8; 3.4; 6.5]$	$[2.9; 3.9; 6.7]$	$[3.2; 7.0; 7.6]$	$[1.3; 34.3; 5.5]$
$\lvert\delta\hat{v}^b_{eb,hor}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	8.3	9.2	13.7	50.8
Dynamic driving style, run 3 (repetition from Table 7.2)					
RMS of $\delta\hat{\mathbf{v}}^b_{eb}$	$\mathrm{cm\,s^{-1}}$	$[2.8; 3.8; 6.0]$	$[2.9; 4.2; 6.3]$	$[3.3; 8.0; 7.5]$	$[1.4; 35.7; 5.3]$
$\lvert\delta\hat{v}^b_{eb,hor}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	8.5	9.1	14.9	52.1

Table B.8: Velocity Accuracy Metrics for Odenwald Data Sets. Metrics for Each Scenario: Velocity Error RMS (1^{st} Row), 95 % Quantile of Horizontal Velocity Error (2^{nd} Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1 (repetition from Table 7.2)					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.7; 1.7; 3.9]	[1.7; 2.0; 3.9]	[1.9; 2.5; 3.7]	[34.2; 45.5; 4.8]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.3	4.8	6.0	53.8
Run 2					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.4; 1.6; 3.9]	[1.4; 1.8; 3.7]	[1.5; 2.9; 3.6]	[21.1; 24.8; 4.1]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.1	4.6	6.5	44.3

Table B.9: Velocity Accuracy Metrics for Darmstadt Data Sets. Metrics for Each Scenario: Velocity Error RMS (1^{st} Row), 95 % Quantile of Horizontal Velocity Error (2^{nd} Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.5; 1.5; 2.6]	[1.5; 1.6; 2.5]	[1.4; 2.0; 2.7]	[4.4; 21.8; 2.9]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.3	4.5	4.7	30.6
Run 2 (repetition from Table 7.2)					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.8; 1.8; 3.0]	[1.8; 2.0; 3.0]	[1.7; 2.1; 3.2]	[6.9; 21.9; 3.6]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	5.1	5.4	5.6	47.2
Run 3					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.8; 2.0; 2.6]	[1.8; 2.2; 2.5]	[2.1; 1.9; 2.5]	[3.3; 22.2; 3.4]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	5.5	5.8	6.0	48.4

Table B.10: Velocity Accuracy Metrics for Frankfurt Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Metrics for Each Scenario: Velocity Error RMS (1st Row), 95 % Quantile of Horizontal Velocity Error (2nd Row)

Metric	Unit	Magic Formula Tire Model	Linear Tire Model	Uncorrelated Meas. Noise	Odometry Turned off
Run 1 (repetition from Table 7.2)					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.8; 1.9; 3.2]	[1.8; 2.1; 3.2]	[1.7; 2.8; 3.1]	[3.2; 31.6; 4.4]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	4.9	5.2	6.3	66.3
Run 2					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.9; **12.6**; 3.2]	[1.9; **12.5**; 3.2]	[2.0; **14.7**; 3.1]	[4.0; 24.2; 3.7]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	**25.8**	**25.5**	**29.7**	54.9
Run 3					
RMS of $\delta\hat{\mathbf{v}}_{eb}^{b}$	$\mathrm{cm\,s^{-1}}$	[1.9; 2.7; 3.4]	[1.9; 2.9; 3.3]	[2.1; 2.6; 3.2]	[2.5; 35.2; 4.5]
$\lvert\delta\hat{v}_{eb,hor}^{b}\rvert_{95\,\%}$	$\mathrm{cm\,s^{-1}}$	6.3	6.9	6.7	62.6

Lateral velocity error RMS and the horizontal error's 95 % quantile are much larger in the second Frankfurt data set than they are in the other runs in Table B.10 for all configurations with enabled odometry update. This stems from problems with the reference solution. Because the ring laser gyroscope IMU used to obtain the reference solution was not fastened properly inside the car, it changed its orientation w. r. t. the vehicle slightly throughout the run, causing a yaw error varying over time in between −0.5° and +0.5°. Because this misalignment is not constant over time, it cannot be estimated and subsequently corrected in post-processing. The incorrect yaw angle generates an incorrect lateral reference velocity, yielding the large errors for this run in Table B.10. The same phenomenon is visible in the yaw angle accuracy metrics in Table B.14. The configuration with deactivated odometry update does not experience unusually large errors, because the general level of the lateral velocity error is higher than the effect caused by the incorrect reference solution.

B.4.3 Yaw Angle Accuracy

All metrics regarding yaw angle accuracy are included in this section. Each of the Tables B.11-B.14 features the results from all runs of a specific test scenario. Just as in Section 7.2.3, four different odometry configurations are listed alongside each other: Magic Formula tire model with non-zero correlation in the odometry measurement noise covariance matrix $\mathbf{R}_o$, linear tire model with non-zero correlation in $\mathbf{R}_o$, linear tire model with uncorrelated measurement noise in $\mathbf{R}_o$ and odometry update turned off completely.

Table B.11: Yaw Angle Accuracy Metrics for Griesheim Airfield Data Sets. Metrics for Each Scenario: Yaw Angle Error RMS (1st Row), 95 % Quantile of Yaw Angle Error (2nd Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Moderate driving style, run 1					
RMS of $\delta\hat{\psi}_{nb}$	°	0.09	0.10	0.11	1.28
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.19	0.21	0.21	2.60
Moderate driving style, run 2 (repetition from Table 7.3)					
RMS of $\delta\hat{\psi}_{nb}$	°	0.07	0.09	0.10	1.14
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.17	0.20	0.22	1.66
Moderate driving style, run 3					
RMS of $\delta\hat{\psi}_{nb}$	°	0.10	0.11	0.12	1.05
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.21	0.23	0.25	1.57
Dynamic driving style, run 1					
RMS of $\delta\hat{\psi}_{nb}$	°	0.11	0.13	0.14	1.00
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.25	0.28	0.28	1.23
Dynamic driving style, run 2					
RMS of $\delta\hat{\psi}_{nb}$	°	0.14	0.15	0.17	0.97
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.23	0.27	0.33	1.20
Dynamic driving style, run 3 (repetition from Table 7.3)					
RMS of $\delta\hat{\psi}_{nb}$	°	0.14	0.14	0.19	0.98
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.27	0.33	0.36	1.16

Table B.12: Yaw Angle Accuracy Metrics for Odenwald Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Metrics for Each Scenario: Yaw Angle Error RMS (1st Row), 95 % Quantile of Yaw Angle Error (2nd Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1 (repetition from Table 7.3)					
RMS of $\delta\hat{\psi}_{nb}$	°	0.09	0.10	0.10	0.87
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.19	0.21	0.21	1.18
Run 2					
RMS of $\delta\hat{\psi}_{nb}$	°	**0.51**	**0.51**	**0.83**	0.87
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	**0.41**	**0.42**	**0.47**	1.19

Yaw angle error RMS and its 95 % quantile are much larger in the second Odenwald data set than they are in the first run in Table B.12 for all configurations with enabled odometry update. The RMS is greater than the 95 % quantile for all three odometry configurations, indicating error distributions with very thick tails. These tails stem from two situations occurring in the second run. The first one is a long tunnel (Saukopftunnel, 2.7 km long), during which the yaw angle error grows to 1.8°, 1.8° and 0.9° for the odometry configurations with Magic Formula tire model, linear tire model and uncorrelated measurement noise, respectively. This tunnel is already mentioned in Section B.4.1, because the yaw error growth inside it is responsible for the comparatively large position error in northern direction for the second Odenwald data set in Table B.4. The second situation is a standstill of 12 min due to a closed road. Since the yaw angle is unobservable when the vehicle is not moving, its error grows over time and reaches values of 4.2°, 4.2° and 2.7° for the three respective odometry configurations. As soon as the vehicle starts moving again, the error drops to its usual level of approx. 0.2°. The phenomenon of yaw error growth during standstill may be countered by applying *zero angular rate updates* (ZARUs) whenever standstill is detected [*Groves*, 2013, Section 15.3.3]. However, this problem is only encountered once in all test scenario runs, because it only occurs when the vehicle is stationary for a long time.

Table B.13: Yaw Angle Accuracy Metrics for Darmstadt Data Sets. Metrics for Each Scenario: Yaw Angle Error RMS (1^{st} Row), 95 % Quantile of Yaw Angle Error (2^{nd} Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1					
RMS of $\delta\hat{\psi}_{nb}$	°	0.18	0.19	0.19	0.85
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.41	0.42	0.43	1.15
Run 2 (repetition from Table 7.3)					
RMS of $\delta\hat{\psi}_{nb}$	°	0.22	0.23	0.22	0.99
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.47	0.48	0.48	1.36
Run 3					
RMS of $\delta\hat{\psi}_{nb}$	°	0.25	0.25	0.23	1.21
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.48	0.49	0.47	1.73

Table B.14: Yaw Angle Accuracy Metrics for Frankfurt Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Metrics for Each Scenario: Yaw Angle Error RMS (1^{st} Row), 95 % Quantile of Yaw Angle Error (2^{nd} Row)

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1 (repetition from Table 7.3)					
RMS of $\delta\hat{\psi}_{nb}$	°	0.19	0.20	0.20	1.23
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.40	0.41	0.40	1.74
Run 2					
RMS of $\delta\hat{\psi}_{nb}$	°	**0.85**	**0.85**	**0.92**	0.85
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	**0.89**	**0.90**	**1.03**	1.31
Run 3					
RMS of $\delta\hat{\psi}_{nb}$	°	0.21	0.21	0.21	1.23
$\lvert\delta\hat{\psi}_{nb}\rvert_{95\,\%}$	°	0.43	0.43	0.46	1.56

Yaw angle error RMS and its 95 % quantile are much larger in the second Frankfurt data set than they are in the other runs in Table B.14 for all configurations with enabled odometry update. As detailed in Section B.4.2 for the lateral velocity error, this stems from problems with the reference solution due to the fact that the ring laser gyroscope IMU used to obtain the reference solution was not fastened properly inside the car, causing it to change its orientation w. r. t. the vehicle slightly throughout the run.

B.4.4 Dynamic Tire Radius Accuracy

All metrics regarding dynamic tire radius accuracy are included in this section. Figures B.36-B.42 depict the transient phase of $\hat{\mathbf{r}}_d$ after initialization for selected runs of each scenario. The run chosen for evaluation in Sections 7.2 and 7.3 is always shown. The other runs are only included if their plots exhibit any unusual behavior that does not occur in the chosen runs, in which case this unusual behavior is mentioned in the figure's caption. The horizontal axes have the same scale in all figures to facilitate an easy comparison.

In the second part of the section, the numerical values for the evaluated dynamic tire radius accuracy metrics are listed in Tables B.15-B.18. Just as in Section 7.2.3, four different odometry configurations are listed alongside each other: Magic Formula tire model with non-zero correlation in the odometry measurement noise covariance matrix $\mathbf{R}_o$, linear tire model with non-zero correlation in $\mathbf{R}_o$, linear tire model with uncorrelated measurement noise in $\mathbf{R}_o$ and odometry update turned off completely. Each table features the results from all runs of a specific test scenario. Values deviating considerably from the ones listed in Chapter 7 are highlighted in the tables. As stated in Section 7.2.3, the reason for these large variations requires further investigation. Consequently, no explanation is given here.

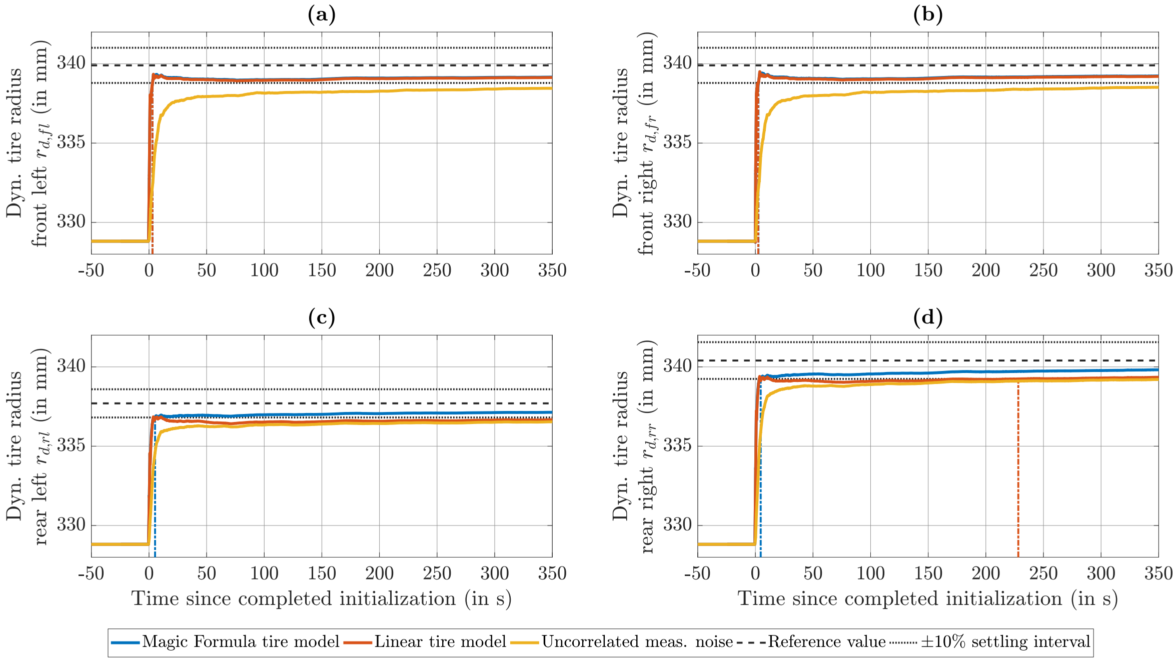

Figure B.36: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Griesheim airfield data set, moderate driving style, run 1. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. Configuration with uncorrelated measurement noise does not reach settling interval at any wheel, configuration with correlated measurement noise and linear tire model does not reach settling interval at rear left wheel. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

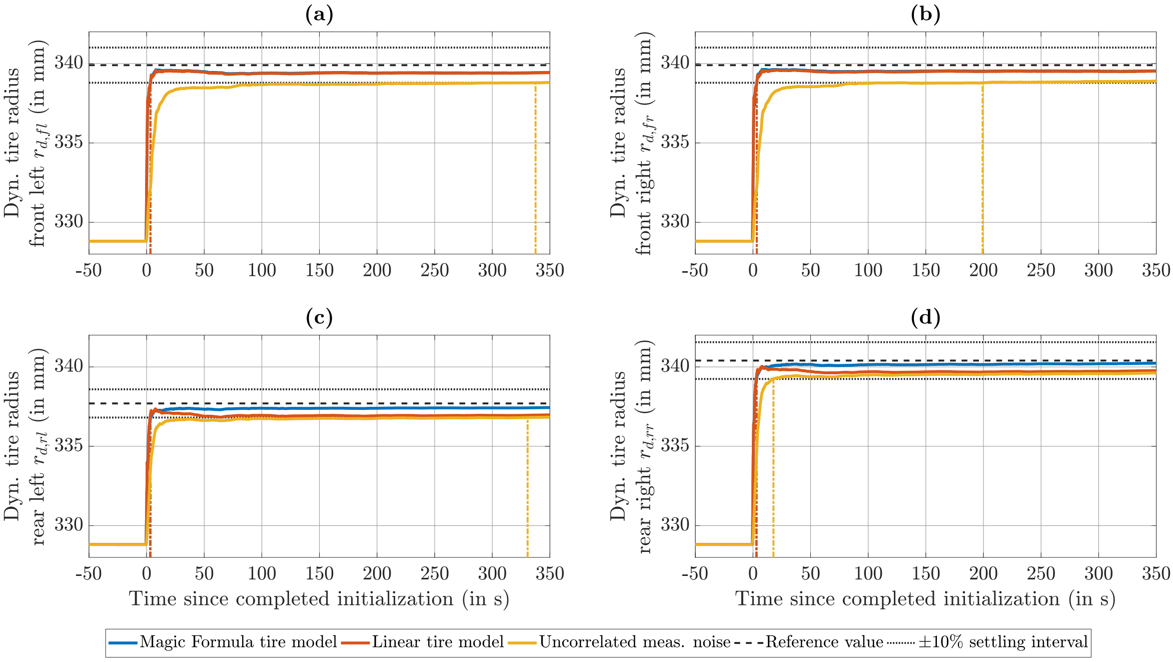

Figure B.37: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Griesheim airfield data set, moderate driving style, run 2. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. The plots from run 3 are almost identical to these ones and hence omitted. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

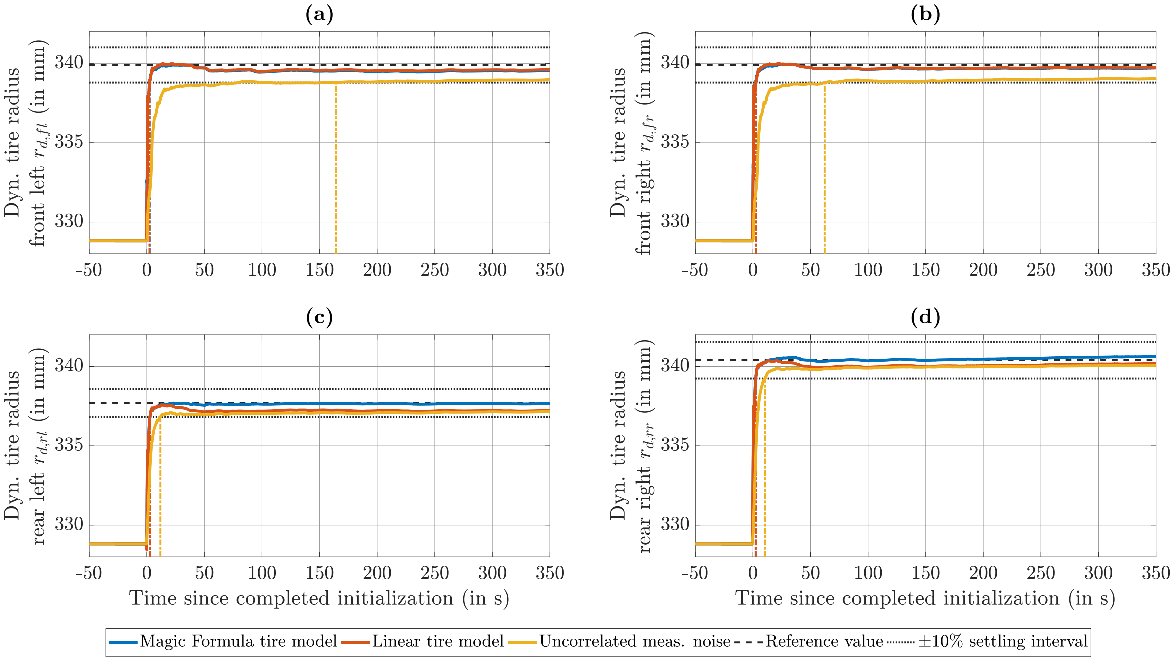

Figure B.38: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Griesheim airfield data set, dynamic driving style, run 3. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. The plots from the other runs are almost identical to these ones and hence omitted. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

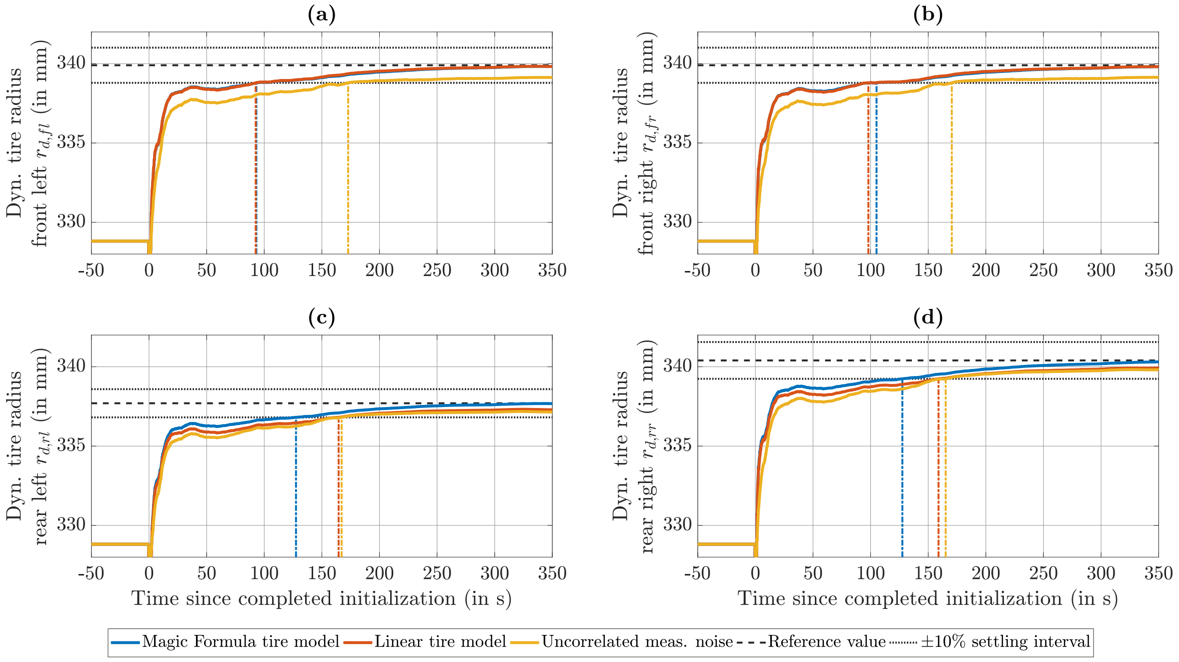

Figure B.39: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Odenwald data set, run 1. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. The plots from run 2 are almost identical to these ones and hence omitted. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

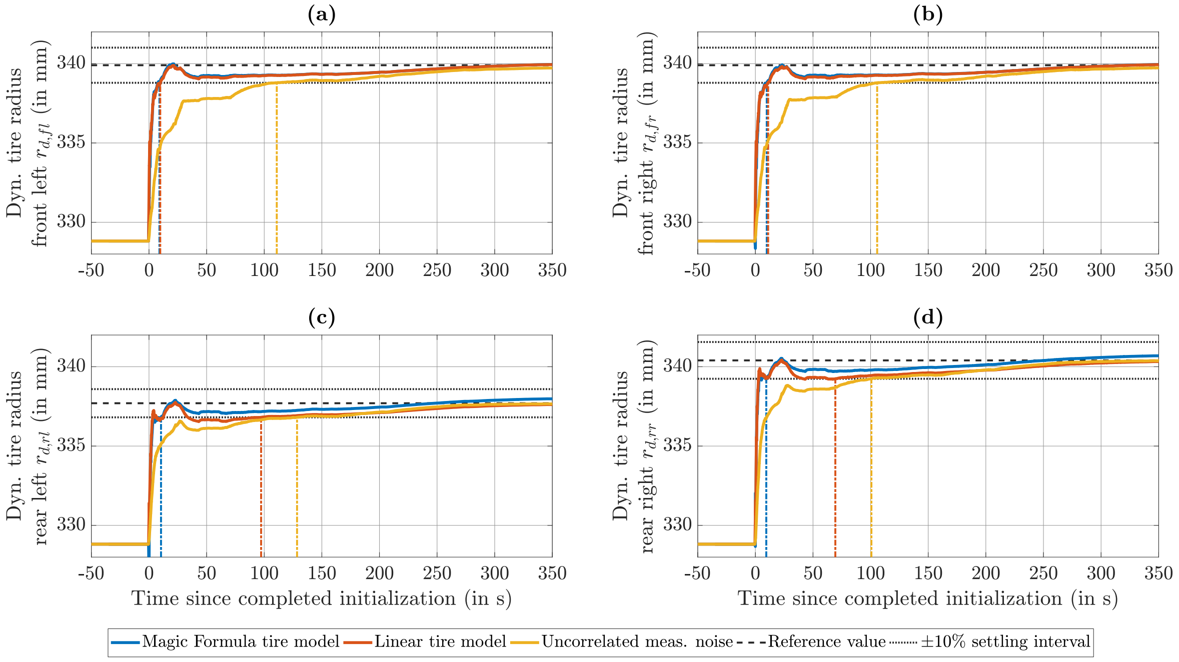

Figure B.40: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Darmstadt data set, run 2. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. The plots from the other runs are almost identical to these ones and hence omitted. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

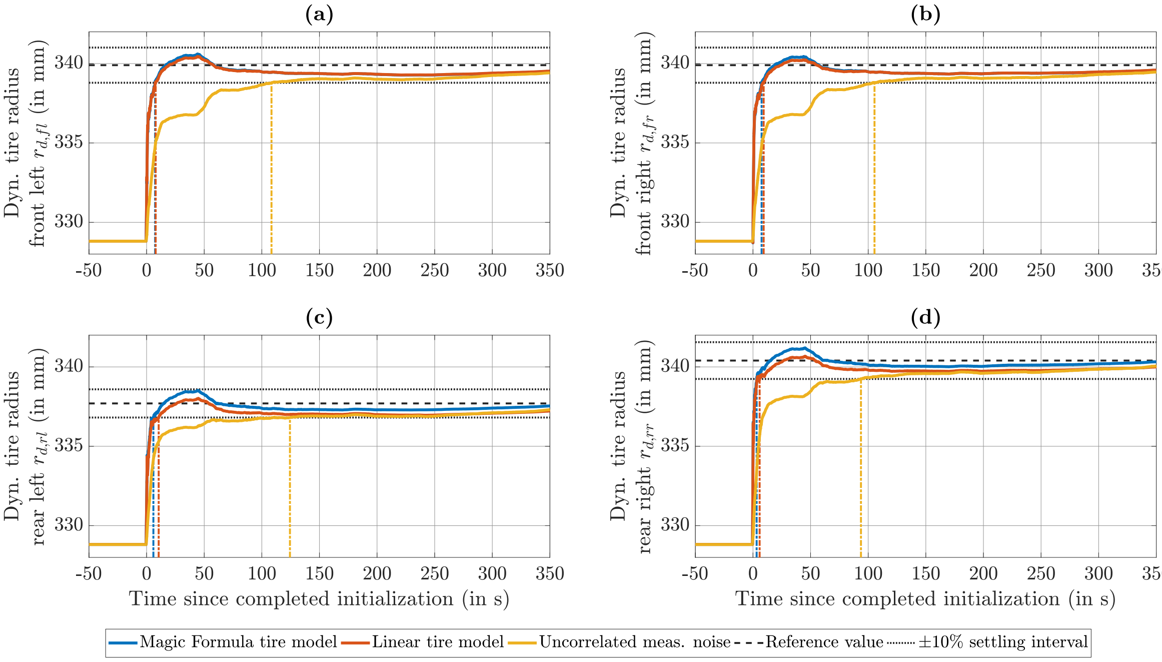

Figure B.41: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Frankfurt data set, run 1. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. The plots from run 3 are almost identical to these ones and hence omitted. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel (Repetition of Figure 7.6).

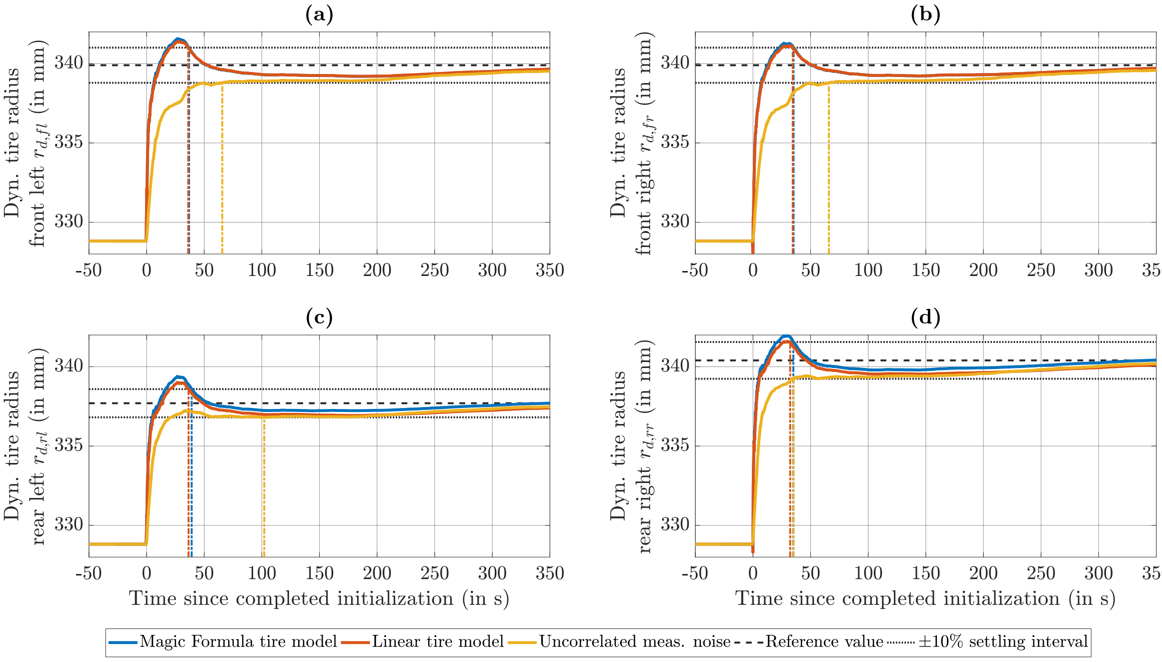

Figure B.42: Transient phase of $\hat{\mathbf{r}}_d$ after initialization for Frankfurt data set, run 2. $T_{s,10\,\%}$ is indicated. Curves for Magic Formula tire model hidden behind curves for linear tire model in **(a)**,**(b)**. This is the only run of all test scenarios in which the overshoot exceeds the ±10 % settling interval. **(a)** Front left wheel. **(b)** Front right wheel. **(c)** Rear left wheel. **(d)** Rear right wheel.

Table B.15: Dynamic Tire Radius Accuracy Metrics for Griesheim Airfield Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Only Metric is 10 % Settling Time

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Moderate driving style, run 1					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	5.3	**N/A**	**N/A**	N/A
Moderate driving style, run 2 (repetition from Table 7.4)					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	3.4	3.3	338	N/A
Moderate driving style, run 3					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	3.2	3.2	317	N/A
Dynamic driving style, run 1					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	3.2	2.9	274	N/A
Dynamic driving style, run 2					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	2.9	2.8	167	N/A
Dynamic driving style, run 3 (repetition from Table 7.4)					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	2.7	2.7	164	N/A

In both odometry configurations utilizing the linear tire model, the ±10 % settling interval is not reached in the first run with moderate driving style at Griesheim airfield in Table B.15. The configuration with uncorrelated measurement noise does not reach the settling interval at any wheel, the configuration with correlated measurement noise does not reach the settling interval at the rear left wheel.

Table B.16: Dynamic Tire Radius Accuracy Metrics for Odenwald Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Only Metric is 10 % Settling Time

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1 (repetition from Table 7.4)					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	128	164	173	N/A
Run 2					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	**21**	111	172	N/A

Table B.17: Dynamic Tire Radius Accuracy Metrics for Darmstadt Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Only Metric is 10 % Settling Time

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	2.6	**2.5**	37	N/A
Run 2 (repetition from Table 7.4)					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	10	95	128	N/A
Run 3					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	**66**	164	165	N/A

Table B.18: Dynamic Tire Radius Accuracy Metrics for Frankfurt Data Sets. Only Metric is 10 % Settling Time

Metric	**Unit**	**Magic Formula Tire Model**	**Linear Tire Model**	**Uncorrelated Meas. Noise**	**Odometry Turned off**
Run 1 (repetition from Table 7.4)					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	7.5	10	124	N/A
Run 2					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	39	37	102	N/A
Run 3					
$T_{s,10\,\%}$ of $\delta\hat{\mathbf{r}}_d$	s	30	30	201	N/A

B.4.5 Position Integrity

All metrics regarding position integrity are included in this section. Each of the Tables B.19-B.22 features the results from all runs of a specific test scenario. The tables compare the integrity achieved by the multi-frequency/multi-constellation algorithm including the a-posteriori variance factor with the integrity achieved by this algorithm without the a-posteriori variance factor, just as it is done in Section 7.3.

Table B.19: Position Integrity Metrics for Griesheim Airfield Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Metrics for Each Scenario: Percentage of Epochs for Which Horizontal Error (1st Row) and Vertical Error (2nd Row) are Inside Their Respective 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Moderate driving style, run 1			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	19.7	15.4
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	**10.0**	**9.2**
Moderate driving style, run 2 (repetition from Table 7.5)			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	18.4	14.4
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	63.7	56.8
Moderate driving style, run 3			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	16.1	12.6
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	72.1	67.4
Dynamic driving style, run 1			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	12.6	4.5
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	7.0	1.6
Dynamic driving style, run 2			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	9.6	2.9
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	6.1	1.1
Dynamic driving style, run 3 (repetition from Table 7.5)			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	10.6	3.2
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	6.8	0.8

In comparison with the other runs, the algorithm's performance in terms of vertical position integrity is much poorer for the first Griesheim airfield data set with moderate driving style in Table B.19. This applies to both configurations, with and without a-posteriori variance factor. The reason for this behavior is unclear and requires further investigation.

Table B.20: Position Integrity Metrics for Odenwald Data Sets. Metrics for Each Scenario: Percentage of Epochs for Which Horizontal Error (1^{st} Row) and Vertical Error (2^{nd} Row) are Inside Their Respective 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1 (repetition from Table 7.5)			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	8.4	6.5
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	44.8	40.3
Run 2			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	7.9	6.1
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	27.0	23.9

Table B.21: Position Integrity Metrics for Darmstadt Data Sets. Metrics for Each Scenario: Percentage of Epochs for Which Horizontal Error (1^{st} Row) and Vertical Error (2^{nd} Row) are Inside Their Respective 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	7.2	5.3
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	25.5	20.1
Run 2 (repetition from Table 7.5)			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	6.5	3.4
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	27.3	19.3
Run 3			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	7.5	3.5
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	30.3	22.7

Table B.22: Position Integrity Metrics for Frankfurt Data Sets. Metrics for Each Scenario: Percentage of Epochs for Which Horizontal Error (1^{st} Row) and Vertical Error (2^{nd} Row) are Inside Their Respective 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1 (repetition from Table 7.5)			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	11.8	8.2
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	35.0	28.2
Run 2			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	8.6	5.8
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	23.2	17.7
Run 3			
$p\left(\varepsilon_{\delta\hat{p}^n_{en,hor}} \leq \xi_{\delta\hat{p}^n_{en,hor},95\,\%}\right)$	%	8.0	5.9
$p\left(\varepsilon_{\delta\hat{p}^n_{en,U}} \leq \xi_{\delta\hat{p}^n_{en,U},95\,\%}\right)$	%	25.7	21.5

B.4.6 Velocity Integrity

All metrics regarding velocity integrity are included in this section. Each of the Tables B.23-B.26 features the results from all runs of a specific test scenario. The tables compare the integrity achieved by the multi-frequency/multi-constellation algorithm including the a-posteriori variance factor with the integrity achieved by this algorithm without the a-posteriori variance factor, just as it is done in Section 7.3.

Table B.23: Velocity Integrity Metrics for Griesheim Airfield Data Sets. Only Metric is Percentage of Epochs for Which the Horizontal Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Moderate driving style, run 1			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	54.0	48.6
Moderate driving style, run 2 (repetition from Table 7.6)			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	51.8	46.8
Moderate driving style, run 3			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	47.3	41.2
Dynamic driving style, run 1			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	24.2	19.2
Dynamic driving style, run 2			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	21.7	15.3
Dynamic driving style, run 3 (repetition from Table 7.6)			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	20.1	15.6

Table B.24: Velocity Integrity Metrics for Odenwald Data Sets. Only Metric is Percentage of Epochs for Which the Horizontal Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1 (repetition from Table 7.6)			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	75.0	53.7
Run 2			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	84.7	62.4

Table B.25: Velocity Integrity Metrics for Darmstadt Data Sets. Only Metric is Percentage of Epochs for Which the Horizontal Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	86.7	64.7
Run 2 (repetition from Table 7.6)			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	80.4	52.3
Run 3			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	80.1	53.2

Table B.26: Velocity Integrity Metrics for Frankfurt Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Only Metric is Percentage of Epochs for Which the Horizontal Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1 (repetition from Table 7.6)			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	71.0	46.6
Run 2			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	**30.1**	**15.6**
Run 3			
$p\left(\varepsilon_{\delta\hat{v}^b_{eb,hor}} \leq \xi_{\delta\hat{v}^b_{eb,hor},95\,\%}\right)$	%	69.4	38.7

In comparison with the other runs, the algorithm's performance in terms of velocity integrity is much poorer for the second Frankfurt data set in Table B.26. This applies to both configurations, with and without a-posteriori variance factor. As detailed in Section B.4.2 for lateral velocity accuracy, this stems from problems with the reference solution due to the fact that the ring laser gyroscope IMU used to obtain the reference solution was not fastened properly inside the car, causing it to change its orientation w. r. t. the vehicle slightly throughout the run. The localization algorithm cannot account for this increased lateral velocity error, since it stems from an incorrect reference solution and not from errors in the integration filter's input data.

B.4.7 Yaw Angle Integrity

All metrics regarding yaw angle integrity are included in this section. Each of the Tables B.27-B.30 features the results from all runs of a specific test scenario. The tables compare the integrity achieved by the multi-frequency/multi-constellation algorithm including the a-posteriori variance factor with the integrity achieved by this algorithm without the a-posteriori variance factor, just as it is done in Section 7.3.

Table B.27: Yaw Angle Integrity Metrics for Griesheim Airfield Data Sets. Only Metric is Percentage of Epochs for Which the Yaw Error is Inside its 95 % Confidence Interval

Metric	Unit	With A-Posteriori Variance Factor	Without A-Posteriori Variance Factor
Moderate driving style, run 1			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	74.2	73.0
Moderate driving style, run 2 (repetition from Table 7.7)			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	75.5	75.1
Moderate driving style, run 3			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	66.9	65.9
Dynamic driving style, run 1			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	58.5	52.5
Dynamic driving style, run 2			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	55.5	47.6
Dynamic driving style, run 3 (repetition from Table 7.7)			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	54.4	44.6

Table B.28: Yaw Angle Integrity Metrics for Odenwald Data Sets. Only Metric is Percentage of Epochs for Which the Yaw Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1 (repetition from Table 7.7)			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	76.3	73.8
Run 2			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	80.5	78.6

Table B.29: Yaw Angle Integrity Metrics for Darmstadt Data Sets. Only Metric is Percentage of Epochs for Which the Yaw Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	77.0	71.5
Run 2 (repetition from Table 7.7)			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	67.2	59.1
Run 3			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	63.0	54.9

Table B.30: Yaw Angle Integrity Metrics for Frankfurt Data Sets. Values That Deviate Considerably From the Ones in Chapter 7 in **Bold**. Only Metric is Percentage of Epochs for Which the Yaw Error is Inside its 95 % Confidence Interval

Metric	**Unit**	**With A-Posteriori Variance Factor**	**Without A-Posteriori Variance Factor**
Run 1 (repetition from Table 7.7)			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	71.8	66.8
Run 2			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	**11.4**	**8.1**
Run 3			
$p\left(\varepsilon_{\delta\hat{\psi}_{nb}} \leq \xi_{\delta\hat{\psi}_{nb},95\,\%}\right)$	%	52.4	46.0

In comparison with the other runs, the algorithm's performance in terms of yaw angle integrity is much poorer for the second Frankfurt data set in Table B.30. This applies to both configurations, with and without a-posteriori variance factor. As detailed in Section B.4.3 for yaw angle accuracy, this stems from problems with the reference solution due to the fact that the ring laser gyroscope IMU used to obtain the reference solution was not fastened properly inside the car, causing it to change its orientation w. r. t. the vehicle slightly throughout the run. The localization algorithm cannot account for this increased yaw angle error, since it stems from an incorrect reference solution and not from errors in the integration filter's input data.

Appendix C

Additional Equations

This appendix chapter presents equations that are not included in the main part. It contains two sections, which deal with the integration filter's system model and the Kalman filter residual covariance matrix, respectively.

C.1 System Model

As described in Section 6.2, the integrations filter's state vector $\mathbf{x}$ consists of 22 states. While the time derivatives of the last seven states (three GNSS clock errors and four dynamic tire radii) are given explicitly in that section, the time derivatives for the first 15 states are not. The reason for this is that these time derivatives are taken from *Groves* [2013, Section 14.2.4] without significant changes. Small differences exist:

- While *Groves* [2013] formulates the equations in the north-east-down version of the n-frame, the east-north-up version is employed in this thesis.
- *Groves* [2013] resolves the position error state in the e-frame with the units radian, radian and meter. In contrast, it is resolved in the n-frame with the unit meter for all three components in this thesis.
- *Groves* [2013] employs a different gravity model, so the partial derivative of $\dot{\mathbf{v}}^e_{en}$ w. r. t. the position is slightly different.
- Due to a different treatment of the IMU biases, all partial derivatives w. r. t. either $\mathbf{b}_a$ or $\mathbf{b}_\omega$ receive the opposite sign.

The resulting system model is given in this section. There is no connection to the other seven states, so all remaining off-diagonal entries of the continuous-time system matrix $\mathbf{N}$ are zero.

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\Psi_{nb}\\ \mathbf{v}^n_{en}\\ \mathbf{p}^n_{en}\\ \mathbf{b}_\omega\\ \mathbf{b}_a\end{pmatrix} = \begin{pmatrix}\mathbf{N}_{11} & \mathbf{N}_{12} & \mathbf{N}_{13} & -\mathbf{C}^n_b & \mathbf{0}\\ \mathbf{N}_{21} & \mathbf{N}_{22} & \mathbf{N}_{23} & \mathbf{0} & -\mathbf{C}^n_b\\ \mathbf{0} & \mathbf{I} & \mathbf{N}_{33} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{pmatrix}\begin{pmatrix}\Psi_{nb}\\ \mathbf{v}^n_{en}\\ \mathbf{p}^n_{en}\\ \mathbf{b}_\omega\\ \mathbf{b}_a\end{pmatrix} \tag{C.1}$$

$$\mathbf{N}_{11} = -\left[\left(\boldsymbol{\omega}_{ie}^{n} + \boldsymbol{\omega}_{en}^{n}\right)\times\right] \tag{C.2}$$

$$\mathbf{N}_{12} = \begin{pmatrix} 0 & \frac{1}{R_N+h_e} & 0 \\ -\frac{1}{R_E+h_e} & 0 & 0 \\ -\frac{\tan\varphi_e}{R_E+h_e} & 0 & 0 \end{pmatrix} \tag{C.3}$$

$$\mathbf{N}_{13} = \begin{pmatrix} 0 & 0 & -\frac{v_{en,N}^{n}}{(R_N+h_e)^2} \\ 0 & \frac{\omega_{ie}\sin\varphi_e}{R_N+h_e} & \frac{v_{en,E}^{n}}{(R_E+h_e)^2} \\ 0 & -\frac{\omega_{ie}\cos\varphi_e}{R_N+h_e} - \frac{v_{en,E}^{n}}{(R_E+h_e)(R_N+h_e)\cos^2\varphi_e} & \frac{v_{en,E}^{n}\tan\varphi_e}{(R_E+h_e)^2} \end{pmatrix} \tag{C.4}$$

$$\mathbf{N}_{21} = -\left[\mathbf{C}_{b}^{n}\mathbf{f}_{ib}^{b}\times\right] \tag{C.5}$$

$$\mathbf{N}_{22} = -\left[\mathbf{v}_{en}^{n}\times\right]\mathbf{N}_{12} - \left[\left(2\boldsymbol{\omega}_{ie}^{n} + \boldsymbol{\omega}_{en}^{n}\right)\times\right] \tag{C.6}$$

$$\mathbf{N}_{23} = \mathbf{N}_{23,A} + \mathbf{N}_{23,B} \tag{C.7}$$

$$\mathbf{N}_{33} = \begin{pmatrix} 0 & \frac{v_{en,E}^{n}\tan\varphi_e}{R_N+h_e} & -\frac{v_{en,E}^{n}}{R_E+h_e} \\ 0 & 0 & -\frac{v_{en,N}^{n}}{R_N+h_e} \\ 0 & 0 & 0 \end{pmatrix} \tag{C.8}$$

$$\mathbf{N}_{23,A} = \left[\mathbf{v}_{en}^{n}\times\right] \begin{pmatrix} 0 & 0 & \frac{v_{en,N}^{n}}{(R_N+h_e)^2} \\ 0 & -\frac{2\omega_{ie}\sin\varphi_e}{R_N+h_e} & -\frac{v_{en,E}^{n}}{(R_E+h_e)^2} \\ 0 & \frac{2\omega_{ie}\cos\varphi_e}{R_N+h_e} + \frac{v_{en,E}^{n}}{(R_E+h_e)(R_N+h_e)\cos^2\varphi_e} & -\frac{v_{en,E}^{n}\tan\varphi_e}{(R_E+h_e)^2} \end{pmatrix} \tag{C.9}$$

$$\mathbf{N}_{23,B} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -\frac{1}{R_N+h_e}\frac{\partial g_{ib,U}^{n}}{\partial\varphi_e} & -\frac{\partial g_{ib,U}^{n}}{\partial h_e} \end{pmatrix} \tag{C.10}$$

$$\begin{aligned}\frac{\partial g_{ib,U}^{n}}{\partial\varphi_e} &= g_e\cos\varphi_e\left[2c_{\varphi2}\sin\varphi_e + 4c_{\varphi4}\sin^3\varphi_e + 6c_{\varphi6}\sin^5\varphi_e + 8c_{\varphi8}\sin^7\varphi_e\right] \\ &\quad + 2c_{\varphi h}h_e\sin\varphi_e\cos\varphi_e \end{aligned} \tag{C.11}$$

$$\frac{\partial g_{ib,U}^{n}}{\partial h_e} = -\left[c_{h1} - c_{\varphi h}\sin^2(\varphi_e)\right] + 2c_{h2}h_e \tag{C.12}$$

C.2 Kalman Filter Residual Covariance

To derive the Kalman filter's residual covariance matrix $\boldsymbol{\Sigma}_{\delta\mathbf{z}^{+}}$, the filter's measurement update is expressed in terms of a Gauss-Markov model. All equations describing the general Gauss-Markov model are explained in *Niemeier* [2008, Chapter 4]. The general Gauss-Markov model and its formulation with Kalman filter terms are:

$$\mathbf{l} \quad = \quad \mathbf{A}\,\mathbf{d}\hat{\mathbf{x}} \quad - \quad \boldsymbol{\upsilon} \tag{C.13}$$

$$\begin{pmatrix} \hat{\mathbf{x}}^{-} \\ \tilde{\mathbf{z}} \end{pmatrix} = \begin{pmatrix} \mathbf{I} \\ \mathbf{H} \end{pmatrix} \hat{\mathbf{x}}^{+} - \begin{pmatrix} \hat{\mathbf{x}}^{+} - \hat{\mathbf{x}}^{-} \\ \delta\mathbf{z}^{+} \end{pmatrix}. \tag{C.14}$$

The vector on the left hand side is the observation vector of the Gauss-Markov model, $\mathbf{A}$ is its design matrix, $\mathbf{d}\hat{\mathbf{x}}$ is the vector of unknown parameters and $\boldsymbol{\upsilon}$ is the improvement vector. Improvements are the negative of residuals, so their covariance matrices are identical. The observation covariance matrix $\mathbf{\Sigma_l}$, expressed in Kalman filter terms, is:

$$\mathbf{\Sigma_l} = \begin{pmatrix} \mathbf{P}^- & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix}. \tag{C.15}$$

For the general Gauss-Markov model, the improvement's covariance matrix $\mathbf{\Sigma}_\upsilon$ is:

$$\mathbf{\Sigma}_\upsilon = \mathbf{\Sigma_l} - \mathbf{A}\mathbf{\Sigma}_{\mathbf{d\hat{x}}}\mathbf{A}^\mathsf{T}. \tag{C.16}$$

In Kalman filter terms, the parameter covariance matrix $\mathbf{\Sigma}_{\mathbf{d\hat{x}}}$ is the a-posteriori covariance matrix $\mathbf{P}^+$. Translating (C.16) to Kalman filter terms yields:

$$\begin{aligned} \mathbf{\Sigma}_\upsilon &= \begin{pmatrix} \mathbf{P}^- & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} - \begin{pmatrix} \mathbf{I} \\ \mathbf{H} \end{pmatrix} \mathbf{P}^+ \begin{pmatrix} \mathbf{I} & \mathbf{H}^\mathsf{T} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{P}^- & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} - \begin{pmatrix} \mathbf{P}^+ & \mathbf{P}^+\mathbf{H}^\mathsf{T} \\ \mathbf{H}\mathbf{P}^+ & \mathbf{H}\mathbf{P}^+\mathbf{H}^\mathsf{T} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{P}^- - \mathbf{P}^+ & -\mathbf{P}^+\mathbf{H}^\mathsf{T} \\ -\mathbf{H}\mathbf{P}^+ & \mathbf{R} - \mathbf{H}\mathbf{P}^+\mathbf{H}^\mathsf{T} \end{pmatrix}. \end{aligned} \tag{C.17}$$

From (C.14), the term in the bottom right corner is the covariance matrix of the Kalman filter measurement residual:

$$\mathbf{\Sigma}_{\delta\mathbf{z}^+} = \mathbf{R} - \mathbf{H}\mathbf{P}^+\mathbf{H}^\mathsf{T}. \tag{C.18}$$