Low-Cost Failure-Tolerant Hybrid Navigation Designs for Future Space Transportation Systems

Vom Fachbereich Mathematik/Informatik
der
UNIVERSITÄT BREMEN

zur Erlangung des Grades
Doktor-Ingenieur
genehmigte

Dissertation
von
M. Sc. Guilherme Fragoso Trigo
geboren in
Setúbal, Portugal

Gutachter: Prof. Dr. rer. nat. Christof Büskens
Fachbereich Mathematik/Informatik,
Universität Bremen

Dr.-Ing. Stephan Theil
Institut für Raumfahrtsysteme,
Deutsches Zentrum für Luft- und Raumfahrt

Tag der mündlichen Prüfung: 15. Dezember 2020

This dissertation is typeset in LaTeX. Most diagrams and plots in it are generated using `tikz` and `pgfplots`. The bibliography is typeset using `biblatex`. The template used is inspired by the `classicthesis` of André Miede. Text and math are typeset with `txfonts`.

Low-Cost Failure-Tolerant Hybrid Navigation Designs for Future Space Transportation Systems

Abstract

Global Navigation Satellite System (GNSS) technology is steadily making its way into space transportation systems. However, the most widely used launcher navigation systems to date, including those on board of European launch vehicles, are inertial-only. Despite reliable and robust, the dead-reckoning nature of these systems causes unbounded navigation solution drift, raising important mission constraints and limitations. Due to its bounded error characteristics, GNSS has long been used as remedy for such inertial drift. Combined, the two systems compensate many of each others' shortcomings.

This thesis presents the research and development of a robust and fault-tolerant GNSS/inertial hybrid navigation system for launchers based on lower grade sensory.

A comprehensive survey of architectural elements of the GNSS/inertial fusion under the scope of the envisaged application is first carried out, providing the background and justification for the proposed system concept. The design process is broken down into three stages: baseline system concept, robust order reduction, and Fault Detection, Isolation and Recovery (FDIR) scheme design.

A tightly-coupled, modular, closed-loop architecture featuring an error-state Kalman filter updated by GNSS pseudorange and time-differenced carrier phase measurements forms the baseline design. The filter runs in parallel with a strapdown inertial propagator, estimating a wide set of inertial sensor perturbations, including many associated with low-grade units, which is fed back to the propagation algorithm. GNSS receiver clock and atmospheric offsets are also mitigated within the filter. Given the large state size, filter order is then reduced through Consider state Kalman filtering. Rederiving this filtering framework, a novel perspective is drawn whereby the standard- and consider-state routines are entirely separated, enabling higher implementation flexibility. Correlation between process and measurement noise is also accounted for. Different inertial sensor filter models (high, medium, and low grade) are reduced based on extensive parametric observability and error impact analyzes using Parametric Cramér-Rao Bounds along representative launch vehicle trajectories. GNSS filter state set is also reduced. Despite using only around half of the computational load of the original full-order estimator, the resulting lower-order filter configurations maintain the robustness of the baseline design against inertial and GNSS sensor errors. Going a step further and targeting common failure modes of hybrid navigation, the developed system is suited with an FDIR module. The proposed novel scheme uses both filter innovation and GNSS measurement-set statistics to detect and isolate GNSS failures as well as inertial/strapdown faults. Threshold selection and concept validation is done through stochastic analysis of the statistical test variables under real GNSS receiver data. Required computation load is shown to be small relative to that of the filter update step.

In a stride to raise the readiness of the designed system the navigation algorithm is implemented as C/C++ software, based on DLR's Hybrid Navigation System (HNS) heritage, flight-proven code. In this process new navigation software libraries are created and several original ones updated. Despite all the additional performance, robustness, fault-tolerance enhancing features with respect to the heritage software, computational load analysis shows only a moderate burden increase, which does not considerably strain the real-time load margins of the latter.

At each design stage, performance is evaluated through model-in-the-loop Monte Carlo and/or hardware-in-the-loop testing using real GNSS data from a representative receiver stimulated with trajectories of both DLR SHEFEX-2 sounding rocket and ESA Vega launcher. The C/C++ navigation software is tested in a software- and hardware-in-the-loop set-up. The proposed system is compared to several other configurations. Among these, comparison to a representative launcher (high grade) inertial-only system suggests the possibility of a one-grade reduction of the inertial sensor with the conceived hybrid design, significantly lowering system cost.

Zusammenfassung

Die Globale Navigationssatellitensystemtechnik (GNSS) dringt immer mehr in die Raumtransportsysteme ein. Die derzeit am weitesten verwendeten Trägerraketen-Navigationssysteme, einschließlich derjenigen an Bord europäischer Trägerraketen, sind jedoch inertiale Navigationssysteme (INS). Trotz Zuverlässigkeit und Robustheit führt das Funktionsprinzip dieser Systeme dazu, dass ihre Lösung driftet und wichtige Missionsbegrenzungen verursacht. Aufgrund seines beschränkten Fehlers wird das GNSS seit langem als Mittel gegen Inertialdrift eingesetzt. In Kombination kompensieren die beiden Systeme viele Mängel des jeweils anderen.

Diese Dissertation stellt die Forschung und Entwicklung eines robusten und fehlertoleranten GNSS/inertialen Hybrid-Navigationssystems für Trägerraketen mit niedrigeren als hochgradigen Sensoren vor.

Zunächst wird eine umfassende Untersuchung der architektonischen Elemente der GNSS/inertialen Hybridnavigation im Rahmen der vorgesehenen Anwendung durchgeführt. Dies liefert den Hintergrund und die Begründung für das in dieses Arbeit vorgeschlagene Systemkonzept. Der Designprozess ist in drei Phasen unterteilt: das Konzept des Basissystems, die Filterordnungsreduktion bei konstanter Robustheit, und das Design eines Fault Detection, Isolation and Recovery (FDIR) Schemas.

Eine tightly-coupled, modulare, closed-loop Architektur mit einem error-state Kalman Filter, der mit GNSS Pseudoentfernungs- und zeitdifferenzierten Trägerphasenmessungen aktualisiert wird, bildet das Grundkonzept. Der Filter läuft parallel zu einem inertialen Propagator und schätzt eine Vielzahl von Inertialsensorparametern, darunter viele, die vorrangig für inertialen Sensoren gerringer Qualität relevant sind, die an die Propagierung rückgekoppelt werden. Die Messfehler des GNSS-Empfängeruhr und atmosphärische Störungen werden ebenfalls innerhalb des Filters gemildert. In Anbetracht des umfangs des Zustandsvektor wird die Filterordnung dann durch die Consider-Kalman-Filterung reduziert. Dieses Filter-Framework wird abgeleitet und in einer neuartigen modularen Perspektive präsentiert, in der Standard- und Consider-Zustandsroutinen vollständig getrennt sind. Dies bietet zusätzliche Flexibilität bei der Implementierung. Der Korrelation zwischen Prozess- und Messrauschen wird ebenfalls berücksichtigt. Filtermodelle für hoch-, mittel- und niedriggradig Inertialsensoren werden basierend auf umfangreichen parametrischen Beobachtungs- und Impact-Analysen unter Verwendung von parametrische Cramér-Rao-Ungleichungen entlang repräsentativer Trägerraketentrajektorien reduziert. Die daraus resultierenden Filter niedrigerer Ordnung sind in der Lage, die Robustheit gegenüber GNSS- und Inertial-Sensorstörungen aufrechtzuerhalten, obwohl sie nur ca. die Hälfte der Rechnen-Load des Vollordnungsfilters benötigen. In einem weiteren Schritt wird ein FDIR-Modul entwickelt, das auf häufige Fehler der GNSS/inertialen Hybridnavigation abzielt. Das vorgeschlagene neue Schema analysiert Innovationen im Filter sowie die Statistik der GNSS-Messwerte., um GNSS- und Inertial/Propagations Fehler zu detektieren und zu isolieren. Die Schwellenwertauswahl und Konzeptvalidierung erfolgt durch stochastische Analyse der statistischen Testvariablen unter realen GNSS-Empfängerdaten. Die erforderliche Rechenlast ist im Vergleich zum Filteraktualisierungsschritt gering.

Um den Reifegrad des Systems zu erhöhen, ist der Navigationsalgorithmus als C/C++-Software implementiert die auf dem flugerprobten Hybrid-Navigation-System-Code (HNS) des DLR basiert. Neue Programmbibliotheken der Navigationssoftware werden geschrieben und einige Originale werden aktualisiert. Trotz aller zusätzlichen Robustheit, Fehlerdetektion und Leistungsmerkmale gegenüber der Originalsoftware ist die Rechenlast nur moderat erhöht und damit werden die Echtzeit Kriterion des Systems weiterhin eingehalten.

In jeder Designphase wird die Systemleistung durch Model-in-the-Loop Monte Carlo-Tests und/oder Hardware-in-the-Loop-Tests mit realen GNSS-Daten von einer repräsentativen Empfängereinheit bewertet. Die C/C++-Navigationssoftware wird in einem Software- und Hardware-in-the-Loop-Setup getestet. Dabei werden reale Trajektorien der DLR SHEFEX-2 Höhenforschungsrakete und der ESA Vega Trägerrakete verwendet. Das vorgeschlagene System wird mit mehreren anderen Konfigurationen verglichen. Unter diesen deutet der Vergleich mit dem repräsentativen (High Grade) Inertialsystem einer Trägerrakete auf die Möglichkeit einer Gradreduktion des Inertialsensors mit dem entwickelten Hybridsystem hin. Dies kann die Systemkosten erheblich senken.

Aos meus pais, Helena e Luís.
Aos meus avós, Júlia e Jorge, Madalena e António.
À Alice e à pequena Elisa.

Contents

Abbreviations

AFSS Autonomous Flight Safety System.

ALTS Autonomous Localization and Telemetry Subsystem (Vega LV subsystem).

AR Auto-Regressive (stochastic process).

AVUM Attitude Vernier Upper Module (Vega LV upper stage).

C/N$_0$ Carrier-to-Noise-density ratio (of GNSS signals).

COTS Commercial Off-The-Shelf.

CRV Crew Return Vehicle (NASA-developed spacecraft).

DCM Direction Cosine Matrix (attitude representation).

DIO Digital Input/Output.

DLR German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt).

DoF Degree of Freedom.

DOP Dilution of Precision (GNSS metric).

ECEF Earth-Centered Earth-Fixed (reference frame).

EGSE Electrical Ground Support Equipment.

EKF Extended Kalman Filter.

ELV European Launch Vehicle, Spa.

ENU East-North-Up (reference frame).

ER Earth Rate (static filter update).

ESA European Space Agency.

FDI Fault Detection and Isolation.

FDIR Fault Detection, Isolation and Recovery.

FLOP Floating-Point Operations.

FOG Fiber-Optic Gyroscope.

FS Full Scale (same as Full Range).

GDOP Geometric Dilution of Precision (GNSS metric, RSS of DOPs along all Cartesian directions and time).

GEO Geostationary Orbit.

GLONASS Global Navigation Satellite System (Globalnaya Navigatsionnaya Sputnikovaya Sistema).

GM Gauss-Markov (stochastic process).

GNC Guidance, Navigation and Control.

GNSS Global Navigation Satellite System.

GPS Global Positioning System.

GPS MT GPS Metric Tracking system.

GTO Geostationary Transfer Orbit.

HiNAV HybrId NAVigation system (Airbus-DS-developed system).

HNS Hybrid Navigation System (DLR-developed system).

HR High Rate.

IIP Immediate Impact Point.

IMU Inertial Measurement Unit.

INS Inertial Navigation System.

KF Kalman Filter.

LEO Low Earth Orbit.

LEOP Launch Early Operation Phase.

LR Low Rate.

LS Least Squares.

LV Launch Vehicle.

MC Monte Carlo.

MD Minimum Detectable (measurement blunder).

MEMS Micro Electro-Mechanical Systems.

MEO Medium Earth Orbit.

MORABA Mobile Rocket Base (Mobile Raketenbasis).

NASA National Aeronautics and Space Administration.

NC Navigation Computer.

NED North-East-Down (reference frame).

NPI Network Partnership Initiative.

OCAM-G Online CAMera and GNSS experiment (ESA Ariane-V VA 219 payload).

PCRB Parametric Cramér-Rao Bound.

PDF Probability Density Function.

POS Position (GNSS measurement/solution).

PPS Pulse per Second (GNSS output).

PR Pseudorange (GNSS raw measurement).

PRN Pseudorandom Noise.

PRR Pseudorange-Rate (GNSS raw measurement).

PVT Position, Velocity, and Time (GNSS measurement/solution).

RAIM Receiver Autonomous Integrity Monitoring.

RLG Ring Laser Gyroscope.

RMS Root Sum of Squares.

RSS Root Sum of Squares.

RW Random Walk (stochastic process).

SBAS Satellite-based Augmentation System.

SHEFEX-2 SHarp Edge Flight EXperiment 2 (DLR sounding rocket mission).

SIL software-in-the-loop (testing set-up).

SSO Sun-Synchronous Orbit.

STR Star Tracker.

SV Satellite Vehicle (of GNSS space segment).

TDCP Time-Differenced Carrier Phase (GNSS pseudo-measurement).

TECU Total Electron Content Unit.

TMTC Telemetry and Telecommand.

UKF Unscented Kalman Filter.

UNB3 University of New Brunswick Tropospheric delay model variant 3.

VEL Velocity (GNSS measurement/solution).

VTEC Vertical Total Electron Content.

WGS-84 World Geodetic System 1984.

ZV Zero Velocity (static filter update).

Notation and Symbols

Math notation

Vector quantities, columns unless otherwise stated, are denoted by lower case bold letters (Roman or Greek), e.g., $\mathbf{v}$. Matrix quantities are denoted by upper case bold letters (Roman or Greek), e.g., $\mathbf{M}$. Functions are denoted by italic letters (Roman or Greek), with upper/lower case and regular/bold attributes according to the their codomain dimensions, e.g., scalar $f(\cdot) : \mathbb{R}^n \longrightarrow \mathbb{R}$, vector $\boldsymbol{f}(\cdot) : \mathbb{R}^n \longrightarrow \mathbb{R}^m$, $m > 1$, and matrix $\boldsymbol{F}(\cdot) : \mathbb{R}^n \longrightarrow \mathbb{R}^{l\times m}$, $l, m > 1$. Attitude quaternions are denoted as $\mathbf{q}_A^B$, representing a rotation from frame A to B, and with scalar part in the fourth position; the quaternion product convention used is $\mathbf{q}_A^C = \mathbf{q}_A^B \mathbf{q}_B^C$ (denoted $\odot$ in [105]). Direction Cosine Matrices (DCMs), also referred to here as rotation matrices, are denoted $\mathbf{C}_A^B$ and represent a rotation from A to frame B, being related to the equivalent quaternion as $\mathbf{C}_A^B = \mathbf{C}(\mathbf{q}_A^B)$; when multiplied on the left-hand side of a vector yield $\mathbf{v}^B = \mathbf{C}_A^B \mathbf{v}^A$.

Symbols

$\mathbf{0}$	=	column zero vector (of appropriate size)
$\mathbf{1}$	=	column vector of ones (of appropriate size)
$\mathbf{a}$	=	acceleration vector, m/s^2
$\mathbf{B}$	=	cross-covariance (matrix) between process and measurement noise
$\mathbf{b}$	=	bias (vector)
$\mathbf{c}$	=	consider parameter vector
$\mathbf{e}$	=	unit vector or total error (vector)
$E\{\bullet\}$	=	expected value
$\mathbf{g}$	=	gravity acceleration vector, m/s^2
$\mathbf{G}$	=	LS gain matrix
$\boldsymbol{h}$	=	measurement vector function
$\mathbf{H}$	=	measurement Jacobian matrix
$\mathbf{I}$	=	identity matrix (of appropriate size)
$\mathbf{l}$	=	lever arm, m
$\mathbf{L}$	=	Cholesky lower triangular matrix square root
$\mathbf{m}$	=	G-sensitive bias (vector)
$\mathbf{m}_i$	=	unit vector in unit direction i
$\mathbf{O}$	=	zero matrix (of appropriate size)
$\mathbf{P}$	=	covariance matrix
$\mathbf{Q}$	=	process noise covariance matrix
$\mathbf{R}$	=	measurement noise covariance matrix

$\mathbf{r}$	=	position vector, m
$\mathbf{s}$	=	scale-factor (vector)
$\mathrm{t}_{\mathrm{u},i}$	=	normalized component i of vector $\mathbf{u}$,
$\mathrm{T}_{\mathrm{u}}^{2}$	=	normalized norm squared of vector $\mathbf{u}$,
$\mathbf{v}$	=	velocity vector, m/s
$\mathbf{w}$	=	process noise vector
$\mathbf{x}$	=	state vector
$\mathbf{y}$	=	measurement vector
$\mathbf{z}$	=	innovation vector
Δ	=	increment or discrete difference
Δt	=	sample time, s
∂	=	partial derivative
δ	=	error or deviation
$\boldsymbol{\delta\theta}$	=	attitude error, rad
$\boldsymbol{\epsilon}$	=	measurement noise vector correlated to process noise
$\boldsymbol{\eta}$	=	non-orthogonality (vector), rad
$\boldsymbol{\vartheta}$	=	misalignment (vector), rad
$\boldsymbol{\nu}$	=	measurement noise vector
ρ	=	range or pseudorange, m
σ	=	standard deviation
τ	=	delay or time offset, s
$\mathbf{\Phi}$	=	state vector transition matrix
$\boldsymbol{\phi}$	=	state vector transition function
φ	=	carrier phase, cycles
$\mathbf{\Omega}$	=	skew-symmetric (or cross-product) matrix of angular rate vector, rad/s
$\boldsymbol{\omega}$	=	angular rate vector, rad/s
$[\bullet\times]$	=	skew-symmetric (or cross-product) matrix
$\hat{\bullet}$	=	estimated value
$\tilde{\bullet}$	=	measured value
$\dot{\bullet}$	=	time derivative

Subscripts

a	=	accelerometer
ant	=	GNSS receiver antenna
gnss	=	GNSS
g	=	gyroscope or gravity
I	=	ionosphere
imu	=	IMU
i	=	receiver channel index
j	=	high-rate time index
k	=	low-rate time index
kin	=	kinematics
r	=	GNSS receiver
s	=	GNSS satellite
sf	=	specific-force
T	=	troposphere

Superscripts

B	=	body-fixed reference frame
E	=	ECEF reference frame
I	=	inertial reference frame
$\top$	=	transposed
$-$	=	*a priori* estimate
$+$	=	*a posteriori* estimate

1
Introduction

This chapter presents the background for the work conducted, as well as the research objective, institutional framework and main contributions.

1.1 Background

Navigation is the process by which a system (or an individual for that matter) acquires information on how it is positioned, oriented and moving relative to its environment [27]. In a space system this means finding position, attitude and velocity with respect to the Earth, another celestial body or inertial space itself.

1.1.1 Launcher navigation and localization

Navigation information is fundamental to a launch mission. The on-board real-time navigation solution is mainly used for two tasks: navigation and localization [121, 17]. The former refers to the vehicle state estimation information (e.g., position, velocity and attitude) provided to the Guidance and Control routines, which autonomously steer the rocket. Localization, also known as safeguard, is the monitoring by the ground segment of the vehicle's state with respect the nominal trajectory for detection of any safety hazard. Given its safety critical nature, vehicle state information is typically gathered from different independent sources. For instance, at the Guiana Space Center in Kourou a set of ground-based (C-band) radar tracking antennas complement the on-board measurements sent to ground [121, 17]. This information is used on ground to estimate an Immediate Impact Point (IIP) and an uncertainty ellipse around it [25]. The typical size of this ellipse determines the imposed ground exclusion areas.

1.1.2 The inertial way

Traditional on-board launch vehicle (LV) navigation systems, used in most vehicles, are inertial-only: in an Inertial Navigation System (INS), measurements of on-board accelerometers and gyroscopes, most often strapped-down (i.e., not gimballed), are integrated at high-rate, propagating an on-ground initial navigation state. The output solution is available for navigation and safeguard. The Ariane-V, for instance, uses two identical, independent INSs running in parallel: a *primary*, fulfilling the navigation function, and a *secondary*, used for redundancy (replacing the primary in case of fault) and for safeguard [25]. The main advantages of this type of inertial-only system include:

- Long heritage: inertial navigation systems for aircraft have been under development since the 1940s, having been widely used in aeronautical, space and military applications since then [35].

- Self-containedness: the measurement principles of inertial sensors do not rely on external signals, making them robust against environmental disturbances [176].

- High-rate: most modern inertial platforms provide high-rate (hundreds to thousands of Hertz) raw/integrated inertial solutions, easing the navigation task in fulfilling most guidance and control and safeguard update rate requirements [176].

On the other hand, the dead-reckoning nature of inertial navigation together with inevitable measurement errors yield continuous solution accuracy degradation, in what is known as *inertial drift* [176]. This imposes a set of constraints on a launch mission:

- Initial condition knowledge: The accuracy of initial position and attitude of the system highly influences inertial drift [176]. This poses strict requirements on the alignment and positioning of the vehicle within the launch facility.

- High grade of inertial sensors: Mission feasibility is only achieved (in most cases) with the inertial drift levels of highly accurate sensors which are generally very costly, bulky and heavy [29, 25]. In fact, high (navigation) grade units may cost 5 to 10 times as much as medium (tactical) grade ones [68].

- Trajectory maintenance and payload injection accuracy: During propelled phases navigation error leaks to the actual vehicle state as guidance and control track the reference trajectory through the on-board solution; this makes navigation one of the main drivers of trajectory tracking accuracy, causing as much as 90% of the final payload injection error [45]. Typical delivery dispersions are such that it is common that orbital correction manoeuvres need to be performed by the injected Spacecraft. These often have high Delta-V costs, potentially reducing mission life-time by months, years even, with considerable loss of commercial/scientific profit. Alternatively, fuel margins need to be taken into account from spacecraft development phase, sacrificing valuable payload mass. In either case, several days may be required for an orbital correction planing and execution, potentially increasing Launch Early Operation Phase (LEOP) duration and cost.

- Mission duration and profile: Inertial drift strongly limits mission duration and phase profile [17, 121]. Long and complex multi-phase missions often result in less accurate payload delivery or, to avoid this, need to include lengthy ground-based orbital determination phases. Return and landing phases of reusable vehicle missions are simply not feasible with inertial-only navigation [146, 147].

- Trajectory optimality: Nominal trajectories of early ascent phases lie within visibility regions of ground tracking radars and include large margins to account for on-board navigation drift, often sacrificing fuel optimality.

- Ground exclusion area: Due to inertial drift, use of inertial navigation solutions for state monitoring and safeguard (e.g., as in Ariane-V) as support to ground radar (which has considerably limited accuracy, especially in velocity) generally results in a large growing state uncertainty ellipse around the IIP [25] forcing the definition (pre-flight) of large ground exclusion areas.

1.1.3 The GNSS/inertial solution

GNSSs (Global Navigation Satellite Systems) have long been combined with inertial navigation in land-based and aeronautical applications [69]. In a hybrid GNSS/inertial navigation set-up, the strengths of both sensor types ease each others' flaws:

- low-rate, error-bounded GNSS data correct the inertial propagation; while

- high-rate inertial measurements smoothly bridge GNSS outputs and provide solution continuity during satellite signal outages.

While vehicle position and velocity states can be derived directly from the GNSS receiver measurements, thus directly bounding inertial drift on these states, vehicle attitude can be indirectly resolved through the sensing of specific-force on both inertial and GNSS measurements. As a result, fused GNSS/inertial solutions have better attitude accuracy than inertial-only ones, especially during accelerated motion (e.g., engine burns, re-entry). An important implication of this is that attitude estimation is best at the end of a thrusting phase, which is when the highest navigation accuracy is required (in order to minimize trajectory divergence of the coasting that follows) [102]. Note also that attitude performance (i.e., gyroscope grade) is generally the limiting factor (and thus cost driver) of launcher Inertial Measurement Unit (IMU) selection [121].

Albeit an attractive solution, GNSS technology has several important vulnerabilities, especially in a launch environment: As a non-self-contained system, signal disturbances and disruptions can occur (e.g., jamming, spoofing, tropospheric and ionospheric effects) [91]. Additionally, the receiver tracking loops are not immune to the high-dynamics, vibration and shocks of launch [25]. Synergy of inertial and GNSS systems can mitigate some of these effects, promoting information redundancy and ensuring continuity. Furthermore, because in hybrid systems the inertial sensors can be continuously calibrated in flight by GNSS data, requirements on their performance may, in some instances, be relaxed (as, e.g., in [187]), representing an opportunity for significant system cost reduction by potentially allowing inertial sensor downgrading.

On-line merging of GNSS and inertial measurements can be accomplished in many ways, offering a wide universe of design possibilities. Fundamental architectural features such as coupling depth [92], loop *openness* [69], and GNSS measurement set choice provide high-level design flexibility, while lower level modeling and filter algorithm options enable system customization to the challenging scenario dynamics and errors of (potentially lower-grade) sensors. Different options of robustness-added designs (e.g., [153, 203]) and fault detection and handling schemes (e.g., [30, 130]) allow yet further design flexibility beyond unit redundancy in the fulfillment of reliability and performance requirements.

Despite the multiple existing GNSS-based LV navigation system concepts, research is still lacking on the benefits/drawbacks of many design options within this application.

1.1.4 Examples of GNSS in LV navigation

Several launch navigation/tracking systems have been developed based on or including GNSS [most often GPS (Global Positioning System)] measurements.

One of the very first systems, the Space Integrated GPS/INS, or SIGI, is a military-grade Honeywell H764G Embedded GPS/INS modified under NASA (National Aeronautics and Space Administration) contract that has been extensively tested on-board of space platforms. These included seven Space Shuttle missions [201], X-37 and X-38 flights [154, 85], and the International Space Station [60, 61]. This system was conceived to play the role of primary navigation system of the Crew Return Vehicle (CRV) and is envisaged to figure as part of the Guidance, Navigation and Control (GNC) system of the Orion space vehicle [126]. Also in the United States, GPS Metric Tracking (GPS MT) systems, a class of on-board GPS-based tracking and telemetry systems, have been targeted as replacement technology for the C-band radar tracking, as part of the initiative for decomissioning of these facilities [119, 66]. Under this initiative, United Launch Aliance has developed a GPS MT localization system with L1 and L2 bands for its launchers using Commercial Off-The-Shelf (COTS) components [33]. This system is routinely flown on Atlas V and Delta IV launchers as main tracking means [8, 40]. Another GPS-based safety tracking system for launchers, the Autonomous Flight Safety System (AFSS), was developed by NASA Goddard Space Flight Center and Wallops Flight Facility [53]. This COTS-based system follows a flexible vehicle, mission, and range-based approach, being configured and tuned prior to each flight according to the scenario. Since development Phase III of the project, it allows the inclusion of an IMU coupled with the GPS receiver. Other GPS MT systems for tracking of small launch vehicles and sounding rockets are currently under development [156, 200].

In Russia, a hybrid GNSS/inertial system meant for primary navigation and using both GPS and GLONASS has successfully operated on-board of the Fregat upper stage launched by both Soyuz-2 and Zenit-SM vehicles. After several flights, the attained orbital injection accuracy improvement with respect to purely inertial navigation was up to one order of magnitude in Medium Earth Orbit (MEO) delivery missions and two in Geostationary Transfer Orbit (GTO) and Geostationary Orbit (GEO) missions [45].

More recently, the Chinese launcher Long March 7 (CZ-7) has been suited with a loosely coupled GNSS/inertial system fulfilling the role of primary navigation [166].

European experience with GNSS receivers on-board of launch vehicles is still gaining momentum. As part of the OCAM-G (Online CAMera and GNSS experiment) payload, a set of three commercial GNSS receivers flew on-board of the Ariane-V launcher [73]. This initiative was purely experimental and realized as a cooperation between European Space Agency (ESA) and European national space agencies and industry partners. Concerning the use of GNSS in primary navigation, and adding to the feasibility studies of GNSS-aided inertial navigation on future Ariane-V models [138, 23, 121, 17], a study funded by ESA was carried out by Airbus DS and culminated in the development of HiNAV (HybrId NAVigation system) [133, 29]. This project aimed at creating a prototype coupled GNSS/inertial navigation system for European launch and re-entry vehicles and revealed promising results. The use of Galileo signals by future hybrid navigation for Ariane-V and Vega has also been investigated [54].

The SHEFEX-2 (SHarp Edge Flight EXperiment 2) sounding rocket mission [193] (Figure 1.1) included a hybrid GPS/INS system developed by German Aerospace Center (DLR, Deutsches Zentrum für Luft- und Raumfahrt), the Hybrid Navigation System (HNS) [174, 160]. This system, first flown in this mission, was designed to work as primary navigation block and employs a COTS IMU (iMAR iIMU-FCAI-MDS) [84], and DLR-developed GPS receiver [104] and Star Tracker (STR) [142]. The receiver unit, a Phoenix-HD, follows the DLR Orion GPS receiver heritage [115, 103] and has been flight-tested a number of times on satellite platforms [114], sounding rockets [104], and on Ariane-V as part of the aforementioned OCAM-G experiment [73].

Vega (Figure 1.1) is a small European launcher conceived by European Launch Vehicle, Spa (ELV) and ESA and operated by Arianespace from Guiana Space Center, Kourou [191]. Due to its cost and payload segment target, it flies a non-redundant high-grade inertial platform (identical to that used in Ariane-V) [25]. As a result, localization of Vega can only be performed through ground radars at Kourou. However, for the northward direction – Vega flies predominantly to sun-synchronous and polar orbits, i.e., heading North – only one of the three radar stations is available, causing a safeguard redundancy issue. To provide redundancy to the limited radar tracking, a COTS GPS receiver is currently flown as part of the on-board Autonomous Localization and Telemetry Subsystem (ALTS) as tracking means [187]. It has also been used for post-flight processing and trajectory reconstruction of flights VV02–VV11. This on-board GNSS receiver is planned to be coupled with a commercial IMU in the future [56]. Furthermore, a GNSS/inertial hybrid navigation system is under development for the future evolutions of the launcher. This is aimed at improving the navigation accuracy while increasing the mission duration and while reducing the equipment cost.

1.2 Research framework and objective

The work presented in this document results from a joint research initiative between the Guidance, Navigation and Control Systems section, ESTEC, ESA, the Guidance, Navigation and Control department, Institute of Space Systems, DLR, and the University of Bremen, under the Network Partnership Initiative (NPI) program contract 4000111837/14/NL/MH.

The goal of this research is to investigate, design and implement a robust, fault-tolerant hybrid navigation system based on lower grade, lower cost sensors (with respect to the trend in the field) that is able to deliver sufficient and consistent navigation performance for the launcher application.

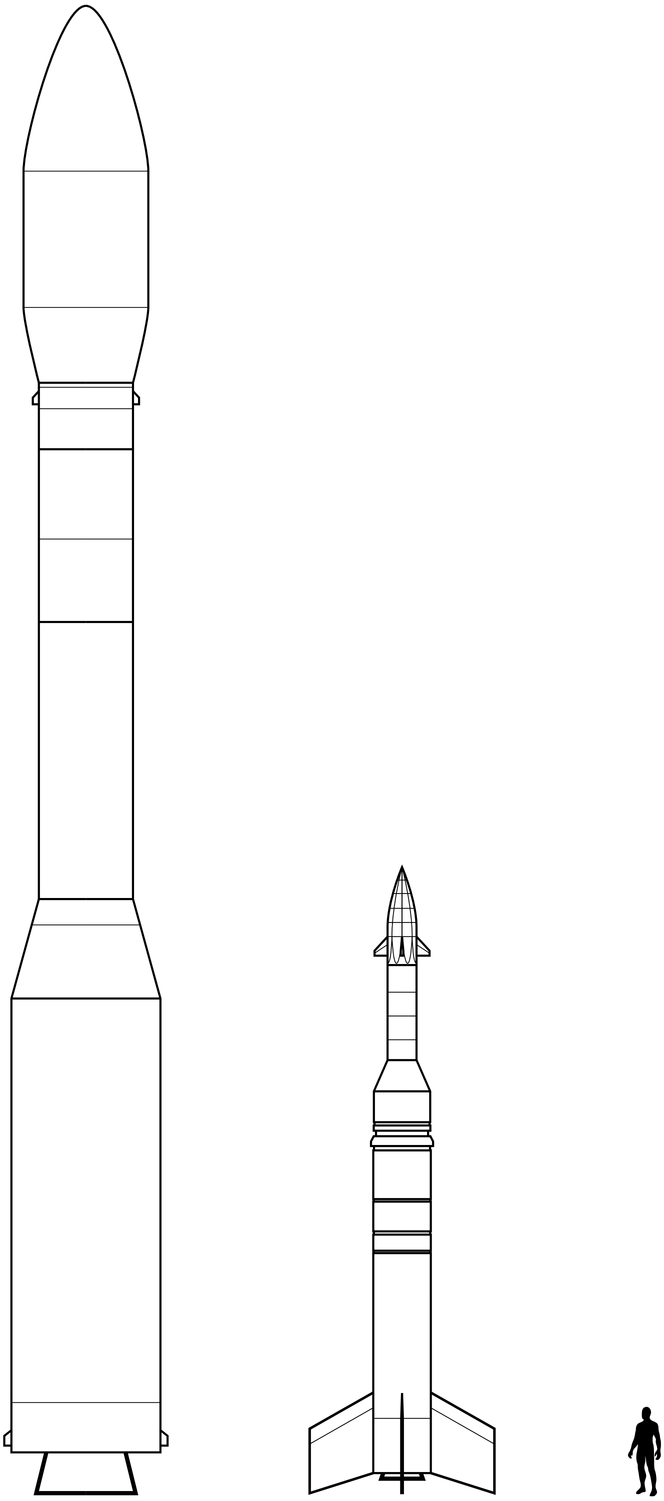


Figure 1.1: Outlines of Vega (left) and SHEFEX-2 (center) vehicles (roughly to scale)

1.3 Contributions

The research and development described in this thesis have produced several contributions to the field of space launcher navigation. The following points summarize the most prominent:

- Extensive review and analysis of architectural elements and design options of hybrid GNSS/inertial navigation systems in the light of the intended application, i.e. real-time on-board LV navigation.

- Design and real-time implementation of a robust low-order hybrid GNSS/inertial navigation system based on lower grade inertial sensory (relative to the commonly used units) and commercial grade GNSS receiver. This contribution can be broken down into several novel developments: Application of time-differenced carrier phases (TDCPs) measurements (previously used in other scenarios) to the LV navigation problem as a velocity-carrying observable for the tightly-coupled filter design; Identification of Tropospheric delay as major error source in GNSS low-noise velocity information (e.g., in TDCP) during early launch, and development of an efficient, robust method for its mitigation within the estimation algorithm; Derivation and implementation of a modular version of the error-state Consider Kalman filter, separating standard-state and consider-state routines, and accounting for correlation between measurement and process noise; Extensive analysis of observability and system impact of a wide set of typical medium/low-grade inertial measurement sensor perturbations using Parametric Cramér-Rao Bounds (PCRBs) and under representative LV trajectories; And establishment of a procedure for the structured design and robust order reduction of hybrid navigation estimators based on the Consider Kalman filter framework through the aforementioned parametric observability and impact analyzes.

- Design, analysis, and real-time implementation of a new Fault Detection, Isolation and Recovery (FDIR) scheme tailored to GNSS/inertial hybrid navigation that is able to detect and isolate different GNSS and INS faults, and is able to maintain/recover estimator coherence in their event.

- Development of a general n-axes IMU model framework for estimation that distills a redundant set of single-axis measurements into a 3-dimensional Cartesian space, minimizing estimator states while maintaining analytically sound stochastic description of the overall inertial sensor errors.

Several of the above-mentioned contributions have, as of this writing, been published in conference proceedings [181, 183], journal articles [182, 184] and a book chapter [180]. In addition, some of the materials and models developed in this thesis have directly contributed to further published work [178, 179, 141, 149].

1.4 Thesis outline

This thesis report starts, in Chapter 2, with an extensive review and analysis of GNSS/inertial hybrid navigation system architectures and design options relevant to the space transportation application. Based on the results of this review, a baseline hybrid navigation system concept is designed and tested in Chapter 3. Chapter 4 then describes the robust order reduction of the design of the previous chapter using extensive parametric impact and observability analysis. An FDIR strategy and algorithm for the conceived system architecture are created in Chapter 5, and the entire algorithm is written in real-time capable software in Chapter 6. Finally, in Chapter 7, conclusions are drawn and future development opportunities are outlined.

In addition, Appendix A describes the SHEFEX-2 mission, trajectory and carried navigation system. It also analyzes its in-flight navigation performance, and the GNSS data collected. Appendix B briefly describes the Vega launcher and the mission trajectories used in this study. The IMU simulation models used throughout this work are described in Appendix C along with the derivation of a general redundant inertial sensor model framework. Appendix D details further GNSS measurement update models (unused in the main navigation system design proposed), and Appendix E provides additional simulation result plots of the tests performed in Chapter 4.

2

Architectural Elements of Hybrid Navigation

Having in mind the launcher application target, a wide set of possible architectures and design options of hybrid navigation unfold. This chapter discusses some of the most prominent. The content herein has in large part been published in [181, 182].

2.1 Breakdown of design options

Figure 2.1 summarizes the options covered in the coming sections. The design process, which starts out by gathering vehicle and trajectory envelope and specifications, as well as system requirements, is then broken down into four main branches:

(a) Integration architecture;

(b) Inertial elements;

(c) GNSS signals and models; and

(d) Filter core algorithm.

Hardware selection, also included in the diagram, can, depending on the design case, be an iterative *variable*, adjusted or consolidated with every design iteration.

Note that these four arms are not necessarily independent: a possible option under one might lock or constrain the choices of another. An example is the selection of GNSS measurement set which, as shall be seen, defines the architecture coupling depth.

Note also that the elements under each arm are mere examples of nested design decisions and/or candidate solutions. Most of these are discussed in this chapter's text. The list displayed is thus by no means exhaustive.

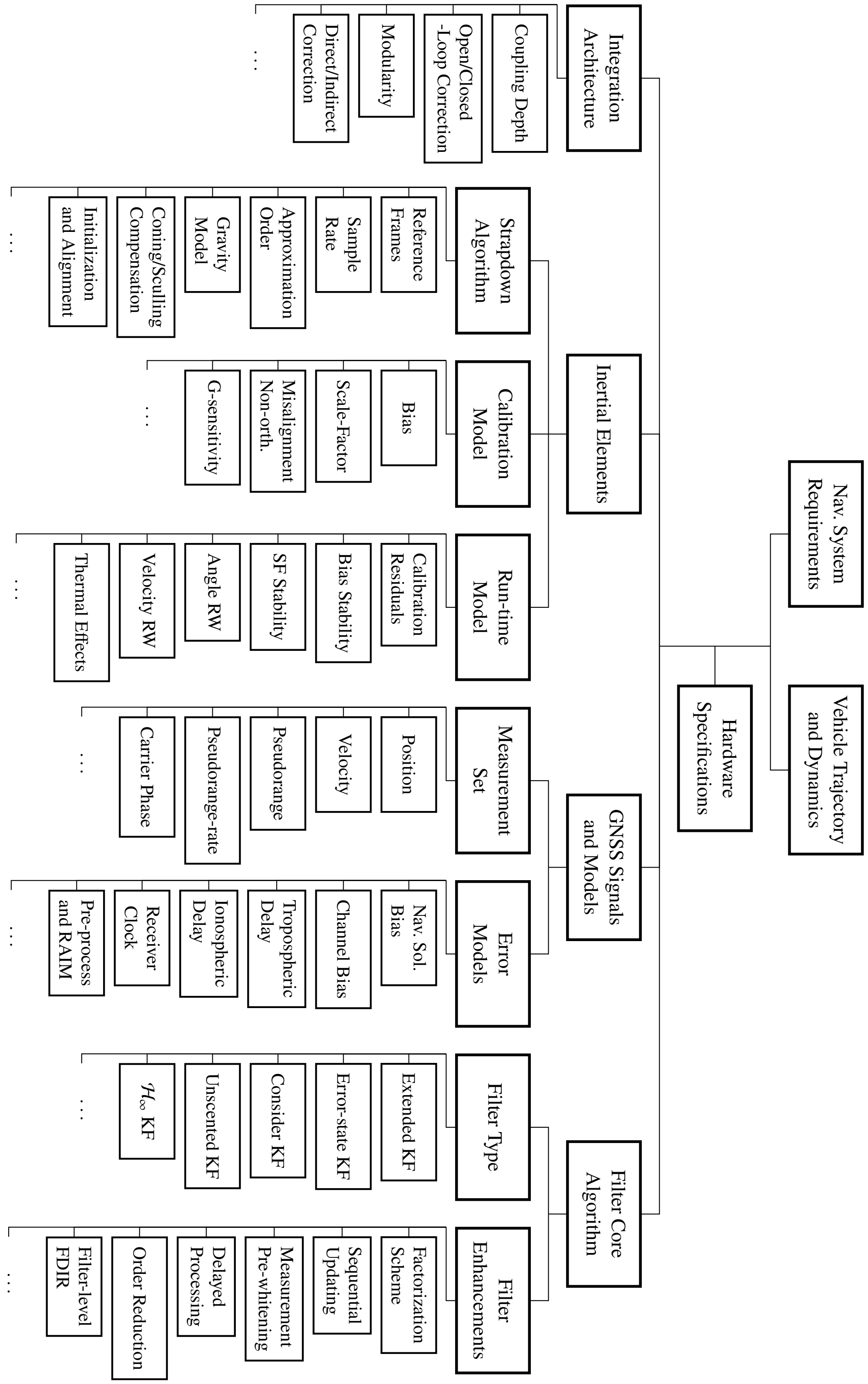


Figure 2.1: Breakdown of possible system design options in hybrid navigation

2.2 Possible requirements and design challenges

The possible requirements for a hybrid navigation system applied to space transportation can form a vast list. These may depend not only on the vehicle carrying it but also on whether the system is to act as primary navigation, as safeguard localization and tracking, or as both (serving as safeguard and switching to main navigation in case of failure of the primary system). However, as a relatively new technology in the space transportation field, regulation and requirement documentation does not abound; nevertheless, the following form a natural starting point:

- A GNSS-based on-board safeguard tracking system should have similar or better accuracy than the currently used means of vehicle tracking (e.g., radar stations);
- A GNSS-aided primary navigation system should achieve a worst case performance equivalent to that of the purely-inertial navigation systems now in use.

A further crucial requirement is output continuity; this can be expressed as:

- A GNSS-based or aided tracking or navigation system should provide a continuous stream of solutions under all predetermined and/or relevant phases of flight.

In the design process of an inertially-aided GNSS receiver for launcher tracking, Braun et al. [25] work out a set of performance requirements departing from: the accuracy of the currently used C-band tracking radars at Guiana Space Center, Kourou; and from the United States range safety standard general performance requirements for GNSS receivers used as on-board tracking devices [59]. The authors arrive at a figure for the maximum tolerable instantaneous velocity error norm (~ 8.7 m/s).

A more extensive set of design and functional requirements for a GNSS-based navigation system for safeguard and primary navigation on Ariane-V is given in [17]:

- Architecture: e.g., functional separation between GNSS receiver, inertial measurement unit, and navigation fusion computer by, for instance, forgoing dynamic state aiding to the receiver (as to maintain a fail-operational set-up);
- Output accuracy: e.g., 25 m in total position and 0.25 m/s in total velocity, both 1σ; (in terms of attitude and as an example, SHEFEX-2 had an accuracy requirement of 0.17 deg at atmospheric reentry initiation point [173];)
- Robustness: e.g., against GNSS signal interference;
- Failure tolerance: maximum failure rate of 10^{-4} per hour during ascent and Low Earth Orbit (LEO) phases and required GNSS failure detection and isolation.

These requirements represent a great challenge to the design of a hybrid navigation system. Further design challenges arise from both vehicle trajectory and dynamics, and the atmospheric environment the vehicle flies through. These may include:

- Vibration and shocks during engine burns and separation events that may not only perturb inertial measurements but can also disrupt the tracking loops of the GNSS receiver causing signal loss and measurement drop-out;

- Signal loss caused by ionospheric scintillations;
- Changing tropospheric delays during lower ascent that can disturb the receiver velocity measurement;
- Low attitude observability, particularly about roll/thrust-axis and especially in spinning vehicles, which may set a strong constraint gyroscope class.

2.3 Hybrid navigation basic architecture

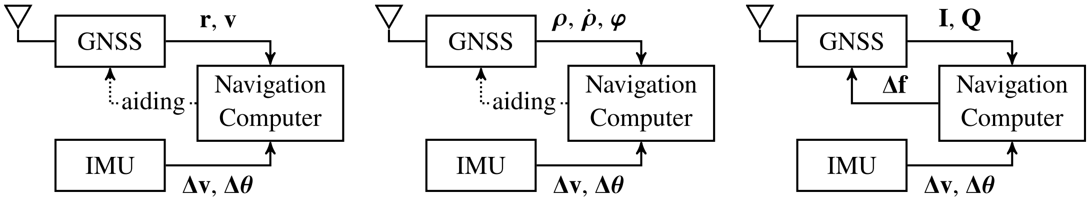
Combination of inertial and GNSS measurements not new; in fact, it has been researched since the early days of GPS [92]. Since then, various architectures have been proposed and implemented in different applications with varying requirements. This section discusses some fundamental architectural options the light of space transportation.

2.3.1 Coupling depth

The most basic architectural feature of a hybrid navigation system is the depth of coupling between inertial and GNSS sensors. Figure 2.2 depicts three levels: loose, tight and ultra-tight.

The simplest of the presented set-ups is the *loosely* coupled system (Figure 2.2a). This uses the GNSS receiver navigation solution to correct the inertial propagation through a fusion algorithm (e.g., a Kalman filter) [65]. The *tightly* coupled hybridization (Figure 2.2b) uses directly the raw GNSS measurements (e.g., Pseudorange, Pseudorange-rate, Carrier Phase, etc.), avoiding the solution step within the receiver [92]. In these two configurations the fused estimates can be provided as aiding to the GNSS receiver to expedite satellite (re)acquisition. Finally, a closer coupling, usually called *ultra-tight* or *deep*, can be achieved by using the corrected inertial estimates to drive the receiver code and carrier tracking loops (Figure 2.2c). In this set-up the navigation computer receives the accumulated correlator outputs from the receiver, fusing them directly with inertial measurements [69].

In general, the tighter the coupling the more complex the system becomes, but also the better the performance and, especially, robustness it delivers [69]. For instance, using a GNSS solution (instead of raw measurements) prevents the navigation from drawing information from sets of fewer than four tracked satellites [128]. This is limiting, for example, if an outage occurs and signals are gradually reacquired, or if the outage is partial and fewer than four signals are still tracked, as it is likely to happen during high dynamic phases of launch. A less obvious disadvantage of a loosely coupled


GNSS
$\mathbf{r}, \mathbf{v}$
aiding
Navigation Computer
IMU
$\Delta\mathbf{v}, \Delta\boldsymbol{\theta}$
GNSS
$\rho, \dot{\rho}, \varphi$
aiding
Navigation Computer
IMU
$\Delta\mathbf{v}, \Delta\boldsymbol{\theta}$
GNSS
I, Q
$\Delta\mathbf{f}$
Navigation Computer
IMU
$\Delta\mathbf{v}, \Delta\boldsymbol{\theta}$

(a) Loose (b) Tight (c) Ultra-tight

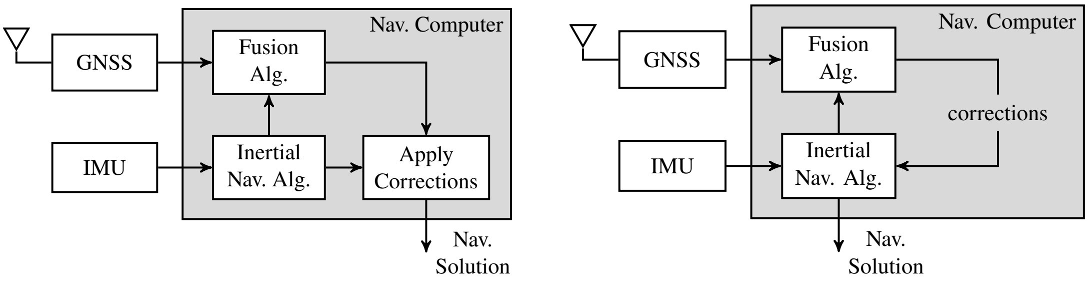
Figure 2.2: GNSS/inertial coupling architectures

scheme is the cascade of navigation filters; by having a first filter stage at the receiver (for GNSS navigation solution) the input to the navigation computer has noise components which are heavily time-correlated with variable profile. This may cause severe misspelling, potentially degrading performance and robustness, and, in the extreme case, lead to instability [92]. Receiver requirements for loose and tight integrations do not dramatically differ, as off-the-shelf receivers usually output raw measurements together with their navigation solution. The ultra-tight hybridization, having a close link between receiver tracking loops and inertial propagation and correction, can achieve the very best results. It can withstand considerably lower C/N$_0$ conditions and operate under higher dynamics. However, its design requires either extensive access to the internal functioning of the receiver or parallel development of both receiver and hybrid system sides. The design and development complexity of such a system is thus far greater than the lower coupled options. Moreover, in this set-up a fail-operational architecture may not be achievable given the inter-dependence of GNSS tracking loops and inertial data. The two less coupled configurations offer, more or less readily, the possibility to isolate GNSS and Inertial platform functions, being thus preferable for the application at hand. For the reasons just stated, the ultra-tight architecture will not be pursued further in the study herein.

As an additional note on coupling, it is worth mentioning that an *uncoupled* architecture is also possible. An example is the system described by Belin et al. [17] for the tracking of Ariane-V, which parallelly runs an inertial platform and a GNSS receiver (with no connection between them). The solution of both devices is sent to ground for safeguard and state monitoring.

2.3.2 Open- *vs* closed-loop

Another architectural option lies in the open-/closed-loop nature of the estimated navigation corrections [69]. Figure 2.3 shows the difference between open- and closed-loop hybridizations. In the open-loop configuration (Figure 2.3a) the fusion algorithm estimates corrections to be applied to the inertial propagator output (position, velocity and attitude). These corrections may grow indefinitely as the inertial solution drifts. This architecture has been used in [146] for the design of a GNSS/inertial system for a reusable launch vehicle experiment. A different approach has the inertial propagation regularly reset using the fused estimates in a closed-loop set-up (Figure 2.3b). This makes sure that the filter remains close to the origin, reducing linearization errors and



(a) Open-loop

(b) Closed-loop

Figure 2.3: Open- and closed-loop configurations of a GNSS/Inertial system

numerical issues due to unbounded state growth. Moreover, it allows for the inertial sensor online calibration, which offers higher robustness and greatly improves performance during outage. Both of these features are fundamental for the studied application. Indeed, this type of architecture has been used in multiple launcher hybrid navigation system concepts (e.g., [161, 187, 25]). Note that, for the fulfillment of a fail-operational requirement (mentioned in Section 2.2), a second uncorrected inertial propagation may be carried out parallelly (within the inertial unit or in a separate computer).

2.3.3 Modularity and direct/indirect filtering

In a closed-loop configuration the degree of modularity between the propagation step of the filter and the strapdown algorithm is another design option. If the two are independently defined, with the filter estimating error quantities, the set-up is called an *indirect* filtering scheme (modular integration). Instead, if the filter estimates total kinematics quantities and the inertial propagation is done as part of the filter's state propagation, the set-up is known as *direct* filtering (non-modular). Although architecturally distinct the performances of these two set-ups can be made virtually equivalent [69]. Potential differences in behaviour may however arise from the way the corrections are done [198]. in general, the direct filtering scheme is more computationally intensive [198] and offers less design flexibility. Furthermore, as Steffes [159] shows, the indirect filtering architecture provides a simple yet powerful way to deal with measurement latency in the real-time implementation. This is a crucial feature as the outputs (both raw and processed) of most GNSS receivers carry considerable delays (e.g., [157]).

2.4 Inertial navigation elements

Given the highly non-linear and uncertain launch vehicle dynamics, forces and torques are generally not modeled within the navigation system using stability derivatives and actuator commands as commonly done in aircraft systems. Inertial navigation is used instead. This is the process by which measurements given out by accelerometers and gyroscopes are used to track the vehicle's position and attitude with respect to a known starting condition [27]. This work focuses on strapped-down platforms, as stable (gimballed) ones have become far less common in the field of launcher navigation. The absence of gimbals or frames makes strapdown mechanically simpler, smaller and lighter; however, it requires considerably more complex algorithms [176]. This section briefly discusses this integration process, typical figures of different sensor quality classes and ways to account for their errors. The achievable hybrid navigation performance for different sensor grades is estimated by covariance analysis.

2.4.1 Reference frames

In this document three different reference frames are used:

ECEF (Earth-Centered Earth-Fixed) WGS-84 (World Geodetic System 1984) [42], here denoted E, is one of the most used reference frames in navigation on and around the Earth. It is approximately defined as having its origin at the Earth's

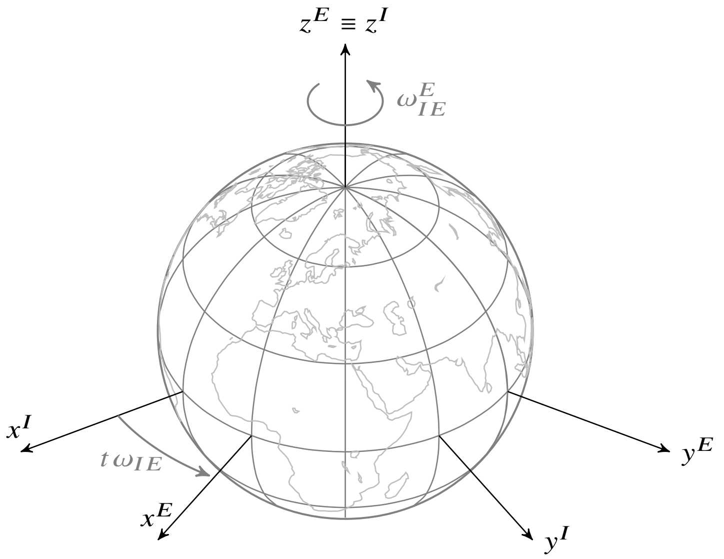


Figure 2.4: ECEF and Inertial axes

center of mass and its axes fixed with respect to Earth. It's x-axis points along the surface point of null latitude and longitude, it's z-axis extends through the geodetic North along the planet's spin axis, and it's y-axis completes the orthogonal right-handed coordinate system. Figure 2.4 shows the ECEF frame, where ω^E_{IE} is the angular velocity vector of ECEF with respect to inertial space, written in ECEF.

Inertial reference frame, I, is used to support the definition and handling of absolute physical quantities (e.g., inertial sensor measurements). being here aligned with the ECEF frame at $t = 0$ s, i.e., when the system is turned-on. The relation between such Inertial frame and the instantaneous ECEF frame is given simply by the rotation of the latter in the elapsed time since the definition of the former. Figure 2.4 shows the Inertial axes depicted with respect to ECEF.

Body reference frame, B, is a vehicle-fixed frame here defined as being centered in the IMU with its axes aligned with the vehicle: z along the vehicle longitudinal direction, pointing in the opposite direction with respect to the rocket tip; the remaining axes form an orthogonal system. Figure 2.5 depicts this frame.

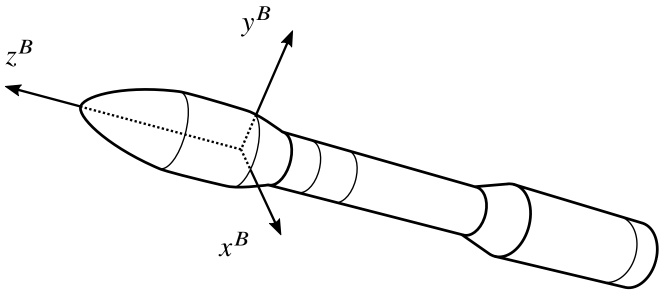


Figure 2.5: Vehicle body axes

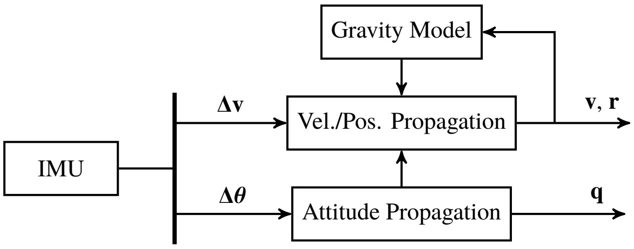


Figure 2.6: Strapdown inertial propagation process

2.4.2 Strapdown integration

Figure 2.6 presents a diagram of the propagation of inertial measurements used to generate a navigation solution. This process can be done either by the on-board (avionics) computer or by the IMU itself, which in such case it is referred to as INS. Absolute angular velocities and specific-force (non-gravitational) accelerations ($\boldsymbol{\omega}_{IB}^{B}$, $\mathbf{a}_{\mathrm{sf}}^{B}$) or, more commonly, angular and velocity increments ($\Delta\boldsymbol{\theta}_{IB}^{B}$, $\Delta\mathbf{v}_{\mathrm{sf}}^{B}$) are measured by the IMU in a body-fixed frame (here B) and, generally, output at a constant rate. Kinematics propagation from t_j to t_{j+1} can be written in inertial frame I as [51, 27]

$$\mathbf{C}_{B_{j+1}}^{I} = \mathbf{C}_{B_j}^{I} \exp\left(\int_{t_j}^{t_{j+1}} \left[\boldsymbol{\omega}_{IB}^{B}\times\right] \mathrm{d}t\right) = \mathbf{C}_{B_j}^{I} \Delta\mathbf{C}_{B_{j+1}}^{B_j} \tag{2.1a}$$

$$\mathbf{v}_{j+1}^{I} = \mathbf{v}_{j}^{I} + \int_{t_j}^{t_{j+1}} \dot{\mathbf{v}}^{I}\, \mathrm{d}t = \mathbf{v}_{j}^{I} + \Delta\mathbf{v}_{\mathrm{sf},j+1}^{I} + \Delta\mathbf{v}_{\mathrm{g},j+1}^{I} \tag{2.1b}$$

$$\mathbf{r}_{j+1}^{I} = \mathbf{r}_{j}^{I} + \int_{t_j}^{t_{j+1}} \mathbf{v}^{I}\, \mathrm{d}t = \mathbf{r}_{j}^{I} + \Delta\mathbf{r}_{j+1}^{I}\ , \tag{2.1c}$$

where $\Delta\mathbf{v}_{\mathrm{g},j+1}^{I}$ is the gravitational correction and $\Delta\mathbf{C}_{B_{j+1}}^{B_j}$, $\Delta\mathbf{v}_{\mathrm{sf},j+1}^{I}$, and $\Delta\mathbf{r}_{j}^{I}$ are the motion integrals computed from the inertial increments measured. The skew-symmetric matrix operator $[\bullet\times]$ can be defined with arbitrary vectors $\mathbf{u}$ and $\mathbf{w}$ as

$$[\mathbf{u}\times]\,\mathbf{w} = \mathbf{u}\times\mathbf{w}\,. \tag{2.2}$$

Note that, as shown in Figure 2.6, the propagated attitude and its increments are used also in linear motion integration, making it a crucial element in the strapdown method.

Reference frame considerations

While navigation mechanisation is simplest in an Inertial reference frame (as in (2.1)), it may be advantageous to propagate the strapdown solution in a different frame.

GNSS outputs are generally expressed in ECEF (see Section 2.4.1), thus it is common for GNSS/inertial applications to integrate the kinematics solution with respect to this frame. The rotating nature of ECEF introduces a Coriolis term and frame rotation transformations which slightly complicate the mechanisation (see, e.g., [176, 69, 89, 35]). However, GNSS measurement update models in the fusion algorithm are simpler.

Although far more used in aeronautical applications, Local Geocentric or Geodesic reference frames (also referred to as *Local-level frames* or *n-frames*), such as the North-East-Down (NED) frame or the East-North-Up (ENU) frame, can also be used (see definitions in, e.g., [27]). In this case, the mechanisation equations complicate further as an additional frame rotation motion needs to be considered, accounting for the change in the local surface tangent caused by the vehicle translation (see, e.g., [68, 89, 176]).

Finally, in on-board tracking systems intended to operate only during the first few minutes of flight it may make sense to use a launch-pad/radar-station fixed frame (referred to in [27] as *tangent frame*) for the mechanisation. Tracked states could then be azimuth, slant-range (or downrange), and elevation (or altitude). Such configuration may be especially suited for the localization function of sounding-rocket-like applications that cover very short downrange distances.

Integration rate and dynamic effects

Numerous algorithms have been derived to solve the motion integrals in (2.1) [109, 83, 144, 145]. The two greatest concerns behind the development and selection of such methods are computational efficiency, and robustness against dynamics effects. While the former has an obvious origin: the on-board computing power is a limited resource; the latter is considerably more complex to grasp and tackle.

The rotation integral in (2.1a) can be exactly computed from non-infinitesimal angular increments if the angular velocity direction is constant within the integration period. In an unconstrained vehicle, however, this is generally not the case. The simplest approximation then is to assume that the sampling period (between t_j and t_{j+1}) is small enough so that the changes in rate direction are negligible, i.e.,

$$\begin{aligned} \int_{t_j}^{t_{j+1}} \boldsymbol{\omega}^B \, \mathrm{d}t &\approx \boldsymbol{\Delta\theta}_{j+1}^{B_{j+1}} \\ \int_{t_j}^{t_{j+1}} \dot{\mathbf{v}}^I \, \mathrm{d}t &\approx \mathbf{C}_{B_{j+1}}^I \boldsymbol{\Delta}\mathbf{v}_{j+1}^{B_{j+1}} + \boldsymbol{\Delta}\mathbf{v}_{\mathrm{g},j+1}^I \,, \end{aligned} \tag{2.3}$$

where $\boldsymbol{\Delta\theta}_{j+1}^{B_{j+1}}$ and $\boldsymbol{\Delta}\mathbf{v}_{j+1}^{B_{j+1}}$ are the gyroscope and accelerometer incremental measurements for such interval. The attitude, velocity and position propagation equations (2.1a)-(2.1c) can then be solved. This corresponds to a first order, full-rate propagation. Higher order solutions of the integrals in (2.3) are possible [109, 183].

Decreasing the computational cost of this process is possible by widening the integration interval of (2.3), simply summing up inertial increments in batches as

$$\begin{aligned} \int_{t_j}^{t_{j+N}} \boldsymbol{\omega}^B \, \mathrm{d}t &\approx \sum_{k=1}^{N} \boldsymbol{\Delta\theta}_{j+k}^{B_{j+k}} \\ \int_{t_j}^{t_{j+N}} \dot{\mathbf{v}}^I \, \mathrm{d}t &\approx \mathbf{C}_{B_{j+N}}^I \sum_{k=1}^{N} \boldsymbol{\Delta}\mathbf{v}_{j+k}^{B_{j+k}} + \boldsymbol{\Delta}\mathbf{v}_{\mathrm{g},j+N}^I \,, \end{aligned} \tag{2.4}$$

and using these to solve (2.1a)-(2.1c) at a lower rate with respect to the IMU output rate. This downsampling, however, can lead to severe inaccuracies as the validity of the “constant angular rate direction” assumption weakens. A classical example of an especially malign motion in such case is *coning*, which occurs when the angular velocity vector is itself rotating (describing a cone) [55, 176]. A similarly malign motion, known as *sculling*, affects velocity and position integration [176]. Coning motion is rather common in spinning rockets. Moreover, high-frequency vibrations, as those experienced in launch, can also cause similar effects. While in a hybrid navigation scheme sculling is well mitigated by the direct measurement of position and velocity by the GNSS receiver, coning poses a higher threat as attitude is not directly observed. The simplest way to prevent such effects is to perform the integration at the highest possible rate, i.e., at the output rate of the IMU. More efficient methods perform the integration at a lower rate, but apply better (and more complex) approximations than the simple summation in (2.4). These are commonly known as coning- and sculling-compensation algorithms, and can be high-rate iterative algorithms, as in [144, 145], allowing any rate-reduction ratio, or predefined laws optimized for a given number of subsamples [83]. Naturally, the inertial sensor output rate must be such that (at least) at full-rate integration the dynamic effects are tolerable. Often, these compensations are done internally within the inertial sensor [19]. Interestingly, an equivalency between coning and sculling has been drawn which allows compensation algorithms derived for one of such motion types to be converted to tackle the other one [139].

Figure 2.7 compares the strapdown propagation error (with respect to the full-rate integration at 400 Hz) for different levels of integration rate reduction with and without dynamical compensation. The compensation methods used for the 2, 3, 4 and 5 fold rate reductions are, respectively, Algorithms 2, 6, 8 and 10 in [83]. The trajectory followed is a simulated SHEFEX-2 trajectory [160] (see Appendix A) and the inertial increments used in the integration are error-free; i.e., the plotted errors arise solely from dynamics effects on the propagation. Rate reduction leads to a clear increase of integration error; and while compensation algorithms ease this effect, especially for position and velocity, attitude accuracy loss quickly becomes significant. Mild reduction of inertial integration rate (2-3 times) in such a highly dynamic application is possible, but should be carefully traded-off against the effects of vehicle dynamics and trajectory. It should be noted that in a hybrid navigation system this accuracy loss will have the most impact on longer dead-reckoning periods, i.e., GNSS outages.

2.4.3 Inertial sensor class

Being central to the navigation system, inertial sensors go through thorough calibration and noise characterization routines before and after assembly within the avionics platform. Nevertheless, residual inaccuracies remain and calibrated quantities drift during operation due to the internal and external factors (temperature and pressure changes, vibration, shocks, etc.). Units are commonly arranged in quality grades depending on how well such calibration remains accurate over time and over turn-on/turn-off cycles. Grades, or classes, range from Consumer- to Inertial/Strategic-grade [176, 14]. The strapdown gyroscope units used in launcher applications are generally based on optical technology: Fiber-Optic Gyroscope (FOG) and Ring Laser Gyroscope (RLG). European launchers Ariane-V and Vega, for instance, carry Navigation-grade RLG-based units [25]. In sounding rockets and smaller launchers relatively lower grade units are used. For example, the SHEFEX-2 sounding rocket employed a Tactical-grade FOG-based IMU [160]. In recent years, great effort has been put into the development and improvement of Micro Electro-Mechanical Systems (MEMS) for space applica-

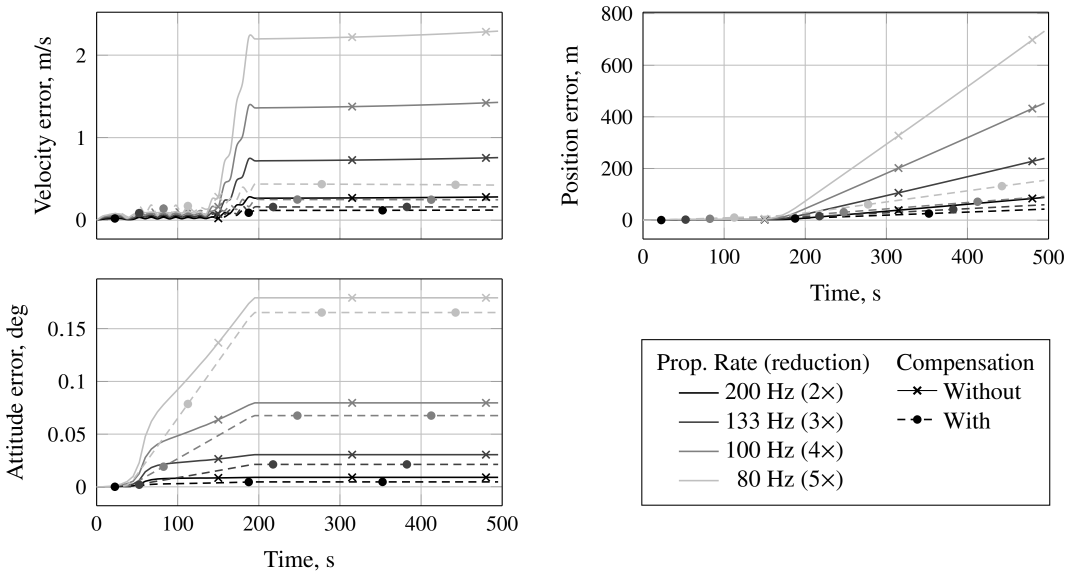


Figure 2.7: Propagation errors with respect to a full-rate integration (400 Hz), of several rate reductions (2, 3, 4 and 5 times) with and without dynamic compensation. Error-free inertial measurements of a SHEFEX-2 trajectory (lift-off at 0 s).

tions [62, 46]. This type of miniaturized, low-cost, low-power, silicon-based sensors can deliver performances that range from Consumer- to Tactical-grade, while being substantially lighter, smaller and cheaper than regular optical-based devices [93].

Table 2.1 displays exemplary specifications for a low, a medium and a high grade sensors. As in [187], these grades correspond to high-end MEMS-based sensor (low-grade), a FOG-based unit (medium-grade), and a RLG-based unit (high-grade) representative of that on-board of Vega. Note that the value for G-sensitive bias of the low-grade sensor was deemed optimistic and is here increased. A standard for the terminology of the different gyro and accelerometer error sources can be found in [79], while an overview on the inertial system output consequences of them is given in [98]. In this section only a subset of these errors is discussed.

Errors and error modeling

As illustrated by the exemplary sensor specifications on Table 2.1, the perturbations affecting lower grade units are generally stronger. Some, as bias and scale-factor, also have faster rates of change. While in a purely-inertial set-up bias and scale-factor stability is absolutely crucial in ensuring the initial (on-ground) calibration remains valid as long as possible during flight, in a hybrid navigation scheme such requirement may be relaxed given the continuous calibration provided by the GNSS. However, if lower grade sensors are to be used, additional effort must be put into the modeling of inertial measurement perturbations within the fusion algorithm, or robustness and filter coherence may be at risk.

Stochastic errors of medium and lower class sensors tend to be time-correlated, for instance due to thermal effects, and to include strong bias instability levels (flicker noise). Allan variance analysis can be used to map and model these stochastic features through

Table 2.1: Error specifications (1σ) of the IMU grades considered [187]

		Sensor Class		
Perturbation	Unit	Low	Medium	High
	Gyroscope			
Angle RW	$\deg/\sqrt{\mathrm{h}}$	0.15	0.01	0.005
Bias on/off repeatability	$\deg/\mathrm{h}$	1	0.1	0.02
Rate RW	$\deg/\mathrm{h}/\sqrt{\mathrm{h}}$	1	0.3	0.005
Bias instability	$\deg/\mathrm{h}$	5	0.01	0.001
Bias thermal residual	$\deg/\mathrm{h}$	5	0.5	0.01
Bias G-sensitivity	$\deg/\mathrm{h}/\mathrm{g}$	3	0.1	0.01
Bias vibration sensitivity	$\deg/\mathrm{h}/\mathrm{g}^2$	1	0.1	0.01
Scale-factor on/off repeatability	ppm	1000	400	30
Scale-factor drift (RW)	$\mathrm{ppm}/\sqrt{\mathrm{month}}$	1000	300	10
Scale-factor non-linearity	ppm FS	500	100	20
Scale-factor G-sensitivity	ppm/g	50	5	1
Axis misalignment	mrad	1	0.3	0.06
Axis non-orthogonality	mrad	0.5	0.15	0.03
	Accelerometer			
Velocity RW	$\mathrm{mg}/\sqrt{\mathrm{Hz}}$	10	1	0.1
Bias on/off repeatability	mg	2	0.5	0.1
Acceleration RW	$\mathrm{mg}/\sqrt{\mathrm{h}}$	1.5	0.05	0.01
Bias instability	mg	0.05	0.01	0.01
Bias thermal residual	mg	1	0.1	0.01
Bias vibration rectification	mg/g^2	1	0.15	0.05
Scale-factor on/off repeatability	ppm	400	200	50
Scale-factor drift (RW)	$\mathrm{ppm}/\sqrt{\mathrm{month}}$	300	100	50
Scale-factor non-linearity	ppm FS	100	80	50
Axis misalignment	mrad	1	0.3	0.06
Axis non-orthogonality	mrad	0.5	0.15	0.03

the fitting of Gauss-Markov (GM) processes [76]. Alternatively, auto-regressive (AR) modeling directly produces the process shaping filter coefficients [122]. Note that, the longer the operation time (i.e., the launch mission), the closer the modeling needs to fit the medium and long term features of the noise profile.

Turn-on scale-factor and scale-factor drift are considerably higher in lower grade sensors [68]. While a simple random-constant plus RW model can be used to account for the constant and time-dependent components, the operation region dependency (known as non-linearity) is much more difficult to tackle. On-ground characterization can be used to mitigate the deterministic (and static) part of the scale-factor curve, however, the residual non-linearity may still result in significant error. Farrell [51] adds 18 states to the Kalman filter accelerometer model, accounting for such non-linearity effects in this sensor up to second order (including cross axes terms), greatly increasing filter state size. In a lighter solution, Steffes [161] inflates the random walk values of the filter scale-factor model to render it robust to the non-linearity residual in a sounding rocket flight. Another approach is to perform this additional “injection” of uncertainty not continuously, but as a function of the change of operation region. In either case, these *ad-hoc* solutions should be carefully tuned according to the sensor and trajectory.

Due to higher manufacturing tolerances, the mutual alignment of sensing axes (known as non-orthogonality) and the alignment of the triad to the sensor casing also tend to carry higher uncertainties with decreasing sensor grade [93]. Consequently, in addition to pre-flight alignment, residual misalignments (which also include those of the final assembly in the launch vehicle) should be accounted for in the fusion algorithm.

Depending on the design and working principle, gyroscopes can have g-sensitive bias and scale-factor components [176]. Whereas in units based on optical technology this dependence is usually negligible, MEMS units, based on mechanical principles, are particularly sensitive [204, 196]. Given the strong accelerations of launch, this is certainly one of the most important criteria in gyroscope selection. Modeling of these errors within the fusion filter is possible and should be considered if the sensor shows such sensitiveness. A *full on* example is given in [51], where a 27-element g-sensitive parameter is added to the Kalman filter gyro model. Such a large state set may not be required as shown in [10], where a 3-element g-sensitivity set achieved the same calibration performance as a 9-element set in a MEMS inertial application.

An example of general gyro and accelerometer single-axis measurement model is

$$\Delta\tilde{\theta}_i = \left(1 + \mathrm{s}_{\mathrm{g},i}\right) {\mathbf{e}_{\mathrm{g},i}^{B}}^{\top} \left(\mathbf{I} + [\boldsymbol{\vartheta}_{\mathrm{g},i}^{B}\times]\right) \boldsymbol{\Delta\theta}_{IB}^{B} + \Delta t\, \mathrm{b}_{\mathrm{g},i} + \nu_{\mathrm{g},i} \tag{2.5}$$

$$\Delta\tilde{\mathrm{v}}_{\mathsf{sf},i} = \left(1 + \mathrm{s}_{\mathrm{a},i}\right) {\mathbf{e}_{\mathrm{a},i}^{B}}^{\top} \left(\mathbf{I} + [\boldsymbol{\vartheta}_{\mathrm{a},i}^{B}\times]\right) \mathbf{\Delta v}_{\mathsf{sf}}^{B} + \Delta t\, \mathrm{b}_{\mathrm{a},i} + \nu_{\mathrm{a},i}\ , \tag{2.6}$$

where $i \in 1, 2, \ldots, n$ is the axis index (within an n-axis IMU) and $\mathbf{e}_i^B$ its unit direction in B frame. The subscripts g and a denote *gyro* and *accelerometer*, respectively.

The error terms in (2.5)–(2.6) are:

$\mathrm{b}_{\mathrm{g}}, \mathrm{b}_{\mathrm{a}}$ are gyro and accelerometer bias terms (scalars). These may include a turn-on (initial) component and a time varying one (rate/acceleration RW); a time-correlated process may be added to model thermal effects. In the case of the gyro, this bias can also have an acceleration (or G-) sensitive component.

$\mathrm{s}_{\mathrm{g}}, \mathrm{s}_{\mathrm{g}}$ are gyro and accelerometer scale-factor terms (scalars). These may include a turn-on component and a time varying one, which is generally modeled as random walk. A G-sensitive term may in certain cases also be added. In signal simulation, a non-linearity component can also be included.

$\boldsymbol{\vartheta}_{\mathrm{g}}^{B}, \boldsymbol{\vartheta}_{\mathrm{a}}^{B}$ are gyro and accelerometer axis misalignement (3×1) vectors. These can generally be modeled as random constants.

$\nu_{\mathrm{g}}, \nu_{\mathrm{a}}$ are gyro and accelerometer noise terms (scalars). These capture angular/velocity random walk and may, if needed, be added a time-correlated component to account for flicker noise (also known as pink noise or bias instability).

A more detailed version of model (2.5)–(2.6) meant for simulation is described in Appendix C.2. This model further includes bandwidth, discretization, sensor and output saturation, and scale-factor non-linearity effects. For units with redundant axes, the separate single-axis measurements can be easily combined into a single 3-element measurement, for instance, using a least-squares method, as described in C.1.

If used for strapdown and estimation purposes, the single-axis gyro and accelerometer models (2.5)–(2.6) may be far too large. These can be reworked, as done in

Appendix C.3, to yield 3-orthogonal-axes sensor model formulations which are more conducive to such application. After this process, the final 3-dimensional model can be written [as in (C.59)–(C.60)] as

$$\Delta\tilde{\boldsymbol{\theta}}^B = \left(\mathbf{I} + \mathrm{diag}\left(\mathbf{s}_\mathrm{g}^B\right) + \left[\boldsymbol{\vartheta}_\mathrm{g}^B \times\right] + \mathbf{S}\left(\boldsymbol{\eta}_\mathrm{g}^B\right)\right) \Delta\boldsymbol{\theta}^B + \mathbf{M}_\mathrm{g}^B \Delta\mathbf{v}_\mathrm{sf}^B + \Delta t\mathbf{b}_\mathrm{g}^B + \boldsymbol{\nu}_\mathrm{g}^B \,, \tag{2.7}$$

$$\Delta\tilde{\mathbf{v}}_\mathrm{sf}^B = \left(\mathbf{I} + \mathrm{diag}\left(\mathbf{s}_\mathrm{a}^B\right) + \left[\boldsymbol{\vartheta}_\mathrm{a}^B \times\right] + \mathbf{S}\left(\boldsymbol{\eta}_\mathrm{a}^B\right)\right) \left(\Delta\mathbf{v}_\mathrm{sf}^B + \Delta\mathbf{v}_\mathrm{cent}^B + \Delta\mathbf{v}_\mathrm{euler}^B\right) + \Delta t\mathbf{b}_\mathrm{a}^B + \boldsymbol{\nu}_\mathrm{a}^B \,, \tag{2.8}$$

where $\mathbf{s}^B$, $\boldsymbol{\vartheta}^B$, $\boldsymbol{\eta}^B$, $\mathbf{b}^B$ and $\boldsymbol{\nu}^B$ are the $(3x1)$ overall scale-factor, misalignment, non-orthogonality, bias, and noise, i.e., those resulting from all single-axis units combined. Again, the subscripts g and a denote *gyro* and *accelerometer*, respectively. The matrix $\mathbf{M}_\mathrm{g}^B$ represents the gyro G-sensitivity. These overall IMU uncertainties can be easily modeled as estimator states. The Delta-V terms $\Delta\mathbf{v}_\mathrm{cent}^B$ and $\Delta\mathbf{v}_\mathrm{euler}^B$ are due to centrifugal and Euler accelerations that arise due to the lever arm of each accelerometer sensing axis with respect to the origin of B. These terms are given for each single axis in (C.23)–(C.24) and for the collapsed model in (C.61)–(C.62).

Sensor class and navigation performance

As previously mentioned, the performance of purely-inertial navigation depends greatly on the quality of the inertial sensors employed. An example is given in Figure 2.8, which shows the inertial propagation error covariances (velocity, position and attitude) for the three IMU profiles on Table 2.1 under Vega VV04 trajectory (see Appendix B). Considering the clear error divergence it becomes obvious the need for very accurate inertial sensors in such purely-inertial schemes. Figure 2.9, in turn, shows the expected error covariance levels if the same inertial information is fused with position and velocity fixes. The error covariance of the aiding position and velocity information was scheduled with altitude to coarsely emulate the effect of GNSS atmospheric errors (see Section 2.5.2). While position estimation accuracy is almost identical with all three sensor classes, depending entirely on the aiding quality, velocity and attitude show variation with the inertial sensor grade. This dependence is particularly high for attitude, as it is not directly measured; nonetheless, fusion renders it partially observable, considerably improving its estimation with respect to the pure inertial integration. As a result, the attitude covariance levels of the aided configurations in this analysis show roughly a one-grade improvement over the dead-reckoning ones: the aided medium-grade configuration approaches the level of the unaided high-grade sensor, whereas the aided low-grade achieves an attitude accuracy level similar to the unaided medium-grade sensor. This direct comparison is shown in Figure 2.10.

2.4.4 Initialization and alignment

As a dead-reckoning process, inertial navigation error greatly depends on initialization accuracy [176]. Initial position is usually measured, to acceptable accuracy, by an external sensor (e.g., GNSS receiver) and initial velocity is in most cases set to zero (stationary vehicle) [89]. Initial attitude, however, may be more difficult to obtain

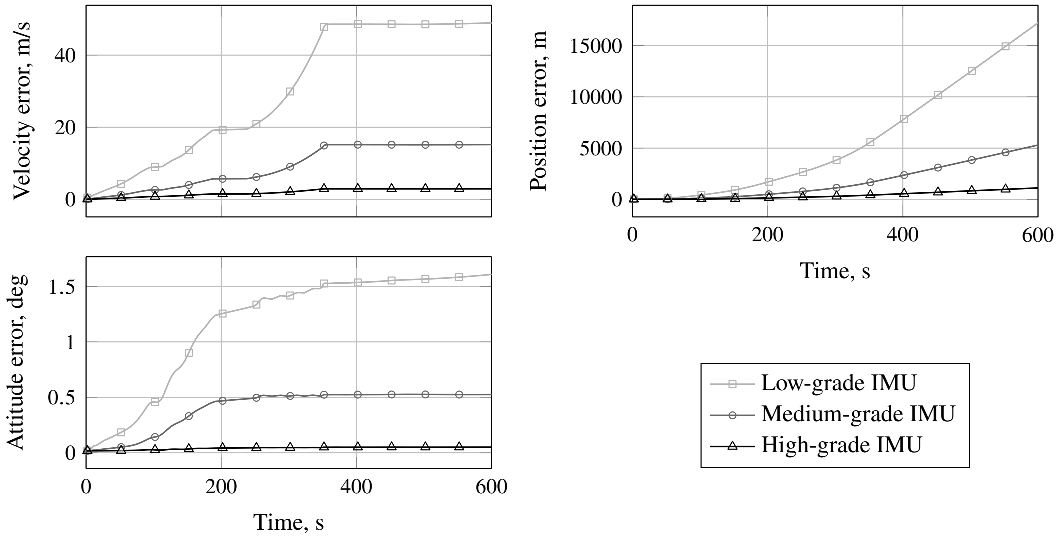


Figure 2.8: Error covariance (1σ) of the propagation of inertial measurements from the IMU profiles on Table 2.1. Vega VV04 trajectory with lift-off at 0 s.

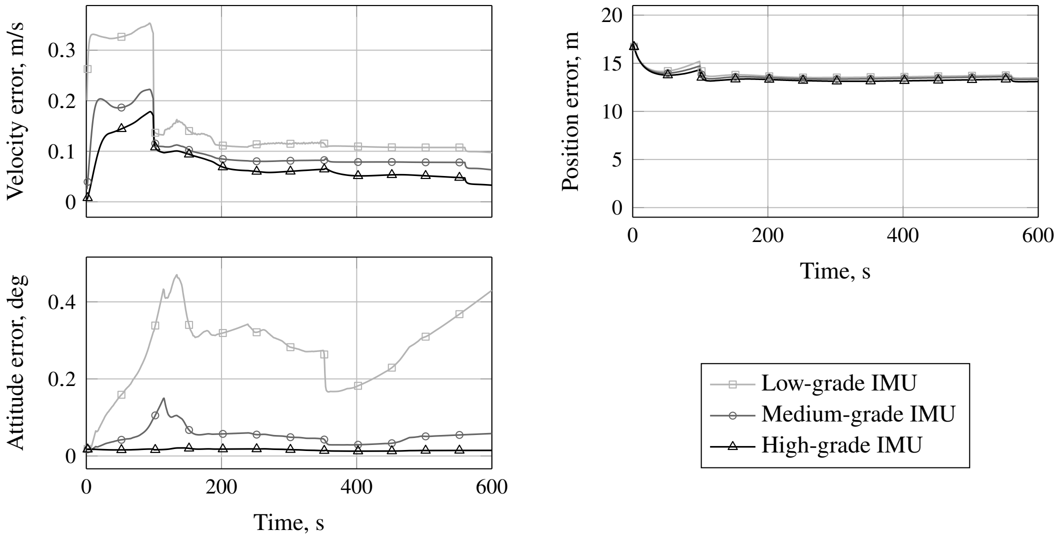


Figure 2.9: Error covariance (1σ) of the fusion of inertial data from the IMU units on Table 2.1 and Position and Velocity fixes of GNSS-like quality. Vega VV04 trajectory with lift-off at 0 s.

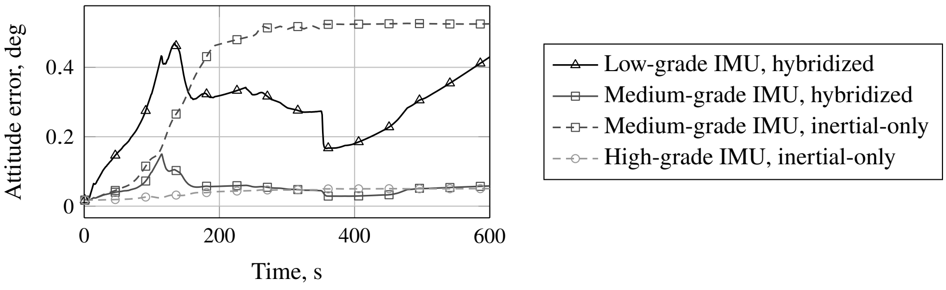


Figure 2.10: Comparison of attitude error covariance (1σ) of pure-inertial and aided solutions from Figures 2.8 and 2.9, respectively

accurately [137]. This is also true for hybrid navigation systems, in which position and velocity are provided by a GNSS unit and attitude (if single-antenna) is only indirectly observable through vehicle dynamics [69]. In fact, initial attitude error of the hybrid navigation is estimated to be the main driver of launch injection accuracy on an Ariane-V GTO launch trajectory [23].

Alignment is the process by which the initial attitude of the system is measured or estimated [176]. Stationary alignment of a strapdown inertial system can be done by:

(a) external attitude measurement/reference;

(b) coarse analytical procedure; and

(c) fine feedback procedure.

An external attitude measurement or reference can be obtained through the known orientation of a calibrated feature or landmark adjacent to or in the vicinity of the vehicle (e.g., launch pad, launch tower or launch rail). This process is akin to the *one-shot transfer alignment* described in [176] and, in general, results only in coarse alignment as many small deviations can still persist.

The coarse (stationary) analytical alignment (also known as ground alignment) is a widely used method of inertial system initialization (see, e.g., [175, 27, 35]). It uses the fact that the transformation between two frames is completely determined with two non-collinear vector observations in both frames. With known (or estimated) initial position, the local Earth rotation vector and Earth gravity vector can be obtained from models in ECEF. Additionally, in a stationary condition (with respect to Earth), the measurements of an inertial measurement unit $\boldsymbol{\omega}_{IB}^{B}$ and $\mathbf{a}_{\mathrm{sf}}^{B}$ provide the needed set of vectors in Body frame to render the system

$$\boldsymbol{\omega}_{IB}^{B} = \mathbf{C}_{E}^{B}\boldsymbol{\omega}_{IE}^{E} + \underbrace{\boldsymbol{\omega}_{EB}^{B}}_{=0} \tag{2.9a}$$

$$\mathbf{a}_{\mathrm{sf}}^{B} = -\mathbf{C}_{E}^{B}\mathbf{g}^{E} + \underbrace{\mathbf{a}_{EB}^{B}}_{=0} , \tag{2.9b}$$

determined with respect to the elements of $\mathbf{C}_{E}^{B}$, i.e., to the alignment of the vehicle (i.e., its attitude) in ECEF. (Note that the centripetal acceleration due to the Earth's rotation is included in the gravitational term.) Adding a vector perpendicular to both $\boldsymbol{\omega}_{IB}^{B}$ and $\mathbf{a}^{B}$, e.g., $\mathbf{a}^{B} \times \boldsymbol{\omega}_{IB}^{B}$, to system (2.9) and solving it for $\mathbf{C}_{E}^{B}$ gives

$$\mathbf{C}_{E}^{B} = \begin{bmatrix} \left(\boldsymbol{\omega}_{IE}^{E}\right)^{\mathsf{T}} \\ -\left(\mathbf{g}^{E}\right)^{\mathsf{T}} \\ -\left(\mathbf{g}^{E} \times \boldsymbol{\omega}_{IE}^{E}\right)^{\mathsf{T}} \end{bmatrix}^{-1} \begin{bmatrix} \left(\boldsymbol{\omega}_{IB}^{B}\right)^{\mathsf{T}} \\ \left(\mathbf{a}_{\mathrm{sf}}^{B}\right)^{\mathsf{T}} \\ \left(\mathbf{a}_{\mathrm{sf}}^{B} \times \boldsymbol{\omega}_{IB}^{B}\right)^{\mathsf{T}} \end{bmatrix} . \tag{2.10}$$

$\boldsymbol{\omega}_{IB}^{B}$ and $\mathbf{a}_{\mathrm{sf}}^{B}$ can be a set of single (simultaneous) inertial measurements or time-averages of these as suggested in [28]. Note that for collinear $\boldsymbol{\omega}_{IB}^{B}$ and $\mathbf{a}_{\mathrm{sf}}^{B}$ the matrices on the right-hand-side of (2.10) are singular. In such case, only two degrees of freedom

(DoFs) of the initial attitude can be obtained in this manner. Furthermore, as shown in [176], the accuracy of this method is maximum around the Equator (where $\boldsymbol{\omega}_{IB}^{B}$ and $\mathbf{a}_{\mathrm{sf}}^{B}$ are strictly perpendicular) degrading with increasing latitude modulus. Inertial sensor errors (e.g., bias and noise) also degrade the performance of this initialization method. Detailed error analyzes can be found in [176, 35, 137].

Finer alignment can be attained with a feedback (closed-loop) mechanism after a coarse initial condition has been established through one of the methods above. In gimballed/stabilized systems, the physical platform is adjusted until its axes are aligned with the local vertical (i.e., leveled) and with the North direction, in a procedure known as *gyrocompassing* [27]. In strapdown systems similar but purely analytical feedback loops can be used, which are often referred to as *analytic gyrocompassing* [176, 27].

A simple way to implement such analytical gyrocompassing when a navigation Kalman filter (as that in a hybrid navigation system) is already part of the system is through stationary (or static) updates [137, 51, 69]. These are in fact pseudo-measurements (i.e., not actual sensor measurements) that inform the filter of (or impose on it) the stationary condition of the physical system with respect to Earth (i.e., ECEF). In essence, this yields a constrained filter (see, e.g., [152]).

Two very common static updates are *zero velocity* and *Earth rate* [137]. These impose null ECEF velocity and null angular rate with respect to ECEF, i.e.,

$$\mathbf{y}_{\mathrm{ZV}} = \mathbf{v}^{E} = \mathbf{0} \tag{2.11}$$

$$\mathbf{y}_{\mathrm{ER}} = \mathbf{C}_{B}^{E}\boldsymbol{\Delta\theta}^{B} - \Delta t\boldsymbol{\omega}_{IE}^{E} = \mathbf{0}\ , \tag{2.12}$$

where $\boldsymbol{\Delta\theta}^{B}$ is the angular increment measurable by the gyro during a time step Δt. The subscripts ZV and ER stand for zero velocity and Earth rate, respectively. Although, the equality constraints (or pseudo-measurements) (2.12) have no noise, in practice, due to unavoidable vibrations and other residual motion, measurement noise is generally added [161, 135]. This makes (2.12) a set of soft constraints rather than perfect equality constraints, which also prevents the numerical issues associated with the latter type [152]. As with the coarse alignment method, the level of fine alignment accuracy achievable with this type of solution (i.e., analytic gyrocompassing) varies with both sensor grade and geographic location (latitude specifically) [51]. Velocity and attitude states (all axes) are rendered observable through this method. However, azimuth (angle to North) is less observable than the two other tilt angles [137]. Observable are also gyroscope and accelerometer biases (in all three directions), with down direction of the latter being observed to a lesser degree [135].

Zero velocity and Earth rate updates have been used in the alignment of the HNS on-board of SHEFEX-2 [161, 160].

2.5 GNSS signals and models

In the design of loosely and tightly coupled hybrid navigation systems the selection of the GNSS measurements to be used is fundamental. Not only does it set the coupling depth (in the choice between processed and raw measurements), but also the achievable performance and robustness, by using or not velocity-based measurements as support to the more common position-based ones. Further flexibility lies in the modeling of the selected measurements. This section discusses several GNSS outputs and possible models for the errors that affect them in the studied application. Hybrid navigation performance in a launch scenario for different measurement sets is also looked into.

2.5.1 GNSS receiver outputs

A GNSS receiver may have several outputs. The most commonly used are the position, velocity, and time (PVT) solutions, which are derived from the *raw* measurements produced by the receiver's ranging processor. These are generally code and phase (or frequency) observables from each tracked satellite.

Pseudorange and Pseudorange-rate

Pseudoranges (PRs) are generated from the time-of-flight measurements obtained by the receiver ranging processor (code tracking loop) [92]. Parallelly, pseudorange-rates (PRRs) are derived from the Doppler shifts measured by the carrier wave tracking loop [127]. These measurements can be given in terms of the real range and range-rate, ρ_i and $\dot{\rho}_i$, between receiver and satellite i at epoch k as

$$\tilde{\rho}_{i,k} = \rho_{i,k} + \rho_{\mathrm{e},i,k} + \nu_{\rho,i,k} \tag{2.13}$$

$$\tilde{\dot{\rho}}_{i,k} = \dot{\rho}_{i,k} + \dot{\rho}_{\mathrm{e},i,k} + \nu_{\dot{\rho},i,k} \ , \tag{2.14}$$

where $\rho_{\mathrm{e},i,k}$ and $\dot{\rho}_{\mathrm{e},i,k}$ are the range and range-rate errors, which may be induced, for instance, by receiver and satellite clock, atmospheric and multipath effects. $\nu_{\rho,i,k}$ and $\nu_{\dot{\rho},i,k}$ are receiver channel noises. Some of the errors affecting these measurements are discussed in Section 2.5.2.

Pseudorange is a position-based measurement, while pseudorange-rate is velocity-based. To keep the navigation system complexity low, it is not uncommon to use only Pseudoranges as update to the fusion algorithm. Indeed, the SHEFEX-2 HNS used only this GNSS observable [161]. However, in highly dynamic applications such as launchers or sounding rockets, it may be beneficial to further include a measure of velocity. Braun et al. [25] fuse both Pseudorange and Pseudorange-rate with inertial measurements in a navigation system meant for launcher localization.

The range quantity depends on vehicle and satellite antenna positions,

$$\rho_{i,k} = \left\| \mathbf{C}^{E_k}_{E(t_{\mathrm{s},i,k})} \mathbf{r}^{E}_{\mathrm{s},i}(t_{\mathrm{s},i,k}) - \mathbf{r}^{E}_{\mathrm{ant},k} \right\| \ , \tag{2.15}$$

where the position receiver antenna $\mathbf{r}^{E}_{\mathrm{ant},k}$ is taken at the reception instant $t_{\mathrm{r},i,k}$, which is equivalent to the nominal time of epoch k (i.e., $t_{\mathrm{r},i,k} = t_k$). The satellite position $\mathbf{r}^{E}_{\mathrm{s},i}$ is evaluated at time of emission $t_{\mathrm{s},i,k}$ and translated to the ECEF frame at the time of reception. The two time instants are related by

$$t_{\mathrm{s},i,k} = t_{\mathrm{r},i,k} - \Delta t_{i,k} = t_k - \frac{\rho_{i,k}}{c} \ , \tag{2.16}$$

where $\Delta t_{i,k}$ is the signal travel time and c is the speed of light in vacuum. Note that the ECEF rotation during the signal travel has to be accounted for; the satellite position at transmission epoch, expressed in ECEF at such epoch $E(t_{\mathsf{s},i,k})$, must be transformed to the ECEF coordinates as of epoch t_k.

Because $t_{\mathsf{s},i,k}$ is dependent on the range through (2.16), its estimated value is computed iteratively. The satellite position at $t_{\mathsf{s},i,k}$ is also computed iteratively within this process. Neglecting the ECEF frame rotation, i.e., assuming $E(t_{\mathsf{s},i,k}) \equiv E(t_k)$, leads to a pseudorange error up to 40 meters, whereas neglecting the signal travel time altogether, i.e., taking $t_{\mathsf{s},i,k} = t_k$, can result in a deviation of up to 300 meters [69].

The range-rate, as derived in Appendix D.2 [expression (D.20)], to given by

$$\dot{\rho}_{i,k} = \frac{\mathsf{d}\rho_{i,k}}{\mathsf{d}t} = \frac{{\mathbf{e}^E_{\rho,i,k}}^{\mathsf{T}} \left(\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})} \left(\mathbf{v}^E_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) \right) - \mathbf{v}^E_{\mathsf{ant},k} - \mathbf{\Omega}^E_{IE}\mathbf{r}^E_{\mathsf{ant},k} \right)}{1 + \frac{1}{c}\,{\mathbf{e}^E_{\rho,i,k}}^{\mathsf{T}} \mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})} \left(\mathbf{v}^E_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) \right)} \;, \tag{2.17}$$

where $\mathbf{e}^E_{\rho,i,k}$ is the unit range vector from receiver to satellite i, $\mathbf{v}^E_{\mathsf{s},i}$ is the satellite velocity, $\mathbf{\Omega}^E_{IE}$ is the skew-symmetric matrix of the Earth angular velocity vector in ECEF.

The satellite states (e.g., position and velocity) are computed using the ephemeris information broadcasted within the GNSS navigation message [124]. Examples of algorithms used for this purpose can be found in [124, 118, 69].

Integrated Carrier Phase

The integrated carrier phase, also referred to as integrated Doppler, is a quantity obtained by the receiver through the accumulation of the phase increments from the carrier tracking loop [127]. This quantity is directly proportional to the range increment (Delta) since the beginning of the integration. It is the GNSS (raw) observable with the lowest noise level [70]. However, the moment when the carrier is acquired and the loop is closed, starting the accumulation, is unknown. This gives rise to an ambiguity term in the relation between the integrated carrier phase and the current receiver-to-satellite range

$$\varphi_{i,k} = \frac{1}{\lambda}\rho_{i,k} + \mathrm{N}_i \;, \tag{2.18}$$

where λ is the wave-length of the carrier, and N_i is an unknown constant, which for null initial phase accumulation, is $\mathrm{N}_i = -\rho_{0,i}/\lambda$. Although N_i can be estimated using ambiguity resolution algorithms (usually in differencial GPS), allowing the integrated carrier measurement to be used as a range observable (see, e.g., [92]), it is commonly used in stand-alone (i.e., single-receiver) navigation to smooth noisy pseudorange measurements [127]. Another way of using this observable is as an explicit Delta-range measurement through its time-differencing; it then becomes a measure of displacement (or average range-rate) rather than of range. This time-differenced carrier phase (TDCP) does not include the influence of N_i, being given by

$$\tilde{\varphi}_{i,k} - \tilde{\varphi}_{i,k-1} \equiv \Delta\tilde{\varphi}_{i,k} = \frac{1}{\lambda}\left(\Delta\rho_{i,k} + \Delta\rho_{\mathrm{e},i,k}\right) + \Delta\nu_{\varphi,i,k} \;, \tag{2.19}$$

where the error term $\Delta\rho_{\mathrm{e},i,k}$ includes contributions from receiver and satellite clocks, atmospheric delays and multipath. $\Delta\nu_{\varphi,i,k}$ is measurement noise. Note that, although akin to pseudorange smoothing where carrier-phase differences are also used, the TDCP pseudo-measurements can be used independently of pseudorange ones.

The velocity nature of TDCPs has led to its use in highly dynamic platforms such as unmanned aerial vehicle [206] and missile [199]. Its dual-epoch origin, however, makes its use in Kalman filtering more complex than regular measurements. Several filter state configurations (or state augmentations) can be used to map the delta-range in the measurement model (2.19). These include:

1. Augmenting the filter state with a delta-range quantity $\Delta\rho_i$ for each tracked satellite vehicle (SV) accumulated since the last update epoch k and propagated as
$$\Delta\rho_{i,j+1|k} = \Delta\rho_{i,j|k} + \Delta t_j \dot{\rho}_{i,j} \ , \tag{2.20}$$
where the evaluation of $\dot{\rho}_{i,j}$ [e.g., as in (2.17)] requires the calculation of satellite position and velocity (from ephemeris) at the rate of filter propagation, which adds considerable computational cost to the already costly large state augmentation.

2. An equivalent but lighter approach, augments the filter state only with the elapsed GNSS antenna position since the last GNSS update, $\mathbf{\Delta r}^E_{\mathsf{ant},j|k}$, which is used to map the previous epoch k antenna position as
$$\Delta\rho_{i,j|k} = \rho_{i,j} - \rho_{i,k} = \rho_{i,j} - \left\| \mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})} \mathbf{r}^E_{\mathsf{s},i}(t_{\mathsf{s},i,k}) - \mathbf{r}^E_{\mathsf{ant},j} + \mathbf{\Delta r}^E_{\mathsf{ant},j|k} \right\| \ , \tag{2.21}$$
being $\mathbf{\Delta r}^E_{\mathsf{ant},j|k}$ then a simple integrator state of the vehicle antenna velocity $\mathbf{v}^E_{\mathsf{ant},j}$ which is reset after each completed measurement update.

3. An alternative way to map the previous epoch antenna state is to back-propagate the current position state [199]. In lieu of state augmentation, this approach uses an inverse transition matrix (from the current propagation time to the previous update epoch) to project the state and covariance onto the previous GNSS instant
$$\mathbf{x}_k = \mathbf{\Phi}^{-1}_{j|k}\mathbf{x}_j \ , \tag{2.22}$$
where the matrix $\mathbf{\Phi}_{j|k}$ is obtained by chain product of the state forward transition matrices $\mathbf{\Phi}_{j|j-1}$ of all propagation steps since the last update step. This process, despite forgoing state augmentation, depending on the filter state size, may require significant computational expense. Moreover, it yields a relatively involved update model with correlated system and measurement noise and mutually-correlated measurements within the set of channels.

4. The simplest way, however, is to make the delayed antenna position (at the previous GNSS epoch) a filter state [183, 206]. Because this state describes a quantity at a fixed point in time, it has no dynamics, making its propagation model very simple. This approach also yields a measurement update model of the regular form (i.e., without state-measurement correlations) and can handle sequential measurement updating, which may be used to reduce the update computational burden and compensate for the extra cost of the 3-component additional filter state.

Method 4. is used in the navigation system designed in Chapter 3.

Receiver Navigation Solution

A navigation solution is computed by the receiver using several or all of the raw measurements described thus far. Depending on the unit's software, this is either done through single-epoch methods or filtering algorithms (e.g., Kalman filter). The position (POS) and velocity (VEL) observables are usually modelled in ECEF frame as

$$\tilde{\mathbf{r}}^E_{\mathtt{ant}} = \mathbf{r}^E_{\mathtt{ant}} + \mathbf{b}^E_{\mathrm{r}} + \boldsymbol{\nu}_{\mathrm{r}} \tag{2.23}$$

$$\tilde{\mathbf{v}}^E_{\mathtt{ant}} = \mathbf{v}^E_{\mathtt{ant}} + \mathbf{b}^E_{\mathrm{v}} + \boldsymbol{\nu}_{\mathrm{v}} \ , \tag{2.24}$$

where $\mathbf{b}^E_{\mathrm{r}}$ and $\mathbf{b}^E_{\mathrm{v}}$ are biases, and $\boldsymbol{\nu}_{\mathrm{r}}$ and $\boldsymbol{\nu}_{\mathrm{v}}$ are noise terms. The subscript ant refers to the receiver antenna. As a result of the filtered nature of the receiver solution, the bias and noise in (2.23)–(2.24) may be heavily time-correlated. As this time correlation is generally unknown (and potentially variable) it is common to update the hybrid navigation algorithm with it at a lower rate than output by the receiver [69]. This may not be acceptable for highly dynamic applications such as the one at hand.

2.5.2 Errors and disturbances

The Pseudorange, Pseudorange-rate and Time-differenced Carrier Phase measurement errors in (2.13), (2.14) and (2.19) denoted $\rho_{\mathrm{e},i}$, $\dot{\rho}_{\mathrm{e},i}$ and $\Delta\rho_{\mathrm{e},i}$ may have a variety of origins. These can be expanded as [51, 170]

$$\rho_{\mathrm{e},i} = c(\tau_{\mathrm{r}} + \tau_{\mathrm{s},i} + \tau_{\mathrm{T},i} + \tau_{\mathrm{I},i} + \tau_{\mathrm{M},i}) \tag{2.25}$$

$$\dot{\rho}_{\mathrm{e},i} = c(\dot{\tau}_{\mathrm{r}} + \dot{\tau}_{\mathrm{s},i} + \dot{\tau}_{\mathrm{T},i} - \dot{\tau}_{\mathrm{I},i} + \dot{\tau}_{\mathrm{M},i}) \tag{2.26}$$

$$\Delta\rho_{\mathrm{e},i} = c(\Delta\tau_{\mathrm{r}} + \Delta\tau_{\mathrm{s},i} + \Delta\tau_{\mathrm{T},i} - \Delta\tau_{\mathrm{I},i} + \Delta\tau_{\mathrm{M},i}) \ , \tag{2.27}$$

where τ_{r} and $\dot{\tau}_{\mathrm{r}}$ are receiver clock bias and drift, respectively; τ_{s} is the satellite clock error; T and I denote Tropospheric- and Ionospheric-induced delays, respectively; and M denotes multipath effects.

Handling of receiver clock errors, and Tropospheric and Ionospheric delays is discussed in the following points. Multi-path errors, that may occur due to reflections on vehicle surfaces, can be mitigated by proper antenna placement within the vehicle. Satellite clock offsets are removed using parameters from the broadcasted navigation message, and relativistic effects (omitted in the expansion) are corrected using satellite position and velocity information [124, 69]. Any remaining residuals (e.g., due to ephemeris errors) can be accounted for in the filter as bias states.

Receiver clock errors

Receiver clock error affects all channels equally. While clock bias disturbs range measurements (pseudorange and carrier phase), clock drift affects the range-rate (Doppler) ones. These effects are commonly modeled in the filter by a second order system as

$$\begin{bmatrix} \ddot{\tau}_{\mathrm{r}} \\ \dot{\tau}_{\mathrm{r}} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\tau}_{\mathrm{r}} \\ \tau_{\mathrm{r}} \end{bmatrix} + \mathbf{w}_{\tau_{\mathrm{r}}} \ , \tag{2.28}$$

where the characteristics of the noise $\mathbf{w}_{\tau_{\mathrm{r}}} \sim N\left(\mathbf{0}, \mathbf{Q}_{\tau_{\mathrm{r}}}\right)$ are obtained, for instance, through Allan variance analysis [44].

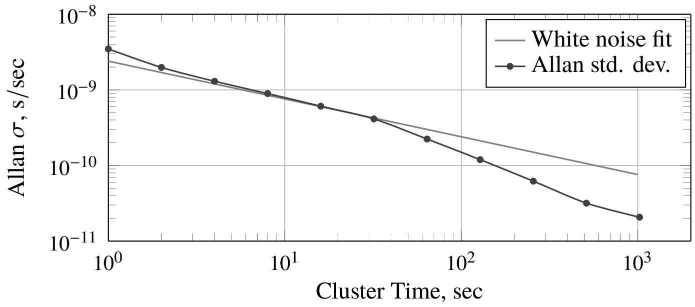


Figure 2.11: Clock frequency error Allan std. dev. for a DLR Phoenix-HD receiver

The receiver used in the present study, a DLR Phoenix-HD unit, uses the same internal clock model to correct both pseudorange and integrated carrier phase measurements. The TDCP common-mode error can be seen as an average clock frequency error over the differencing interval Δt (here 1s), i.e.,

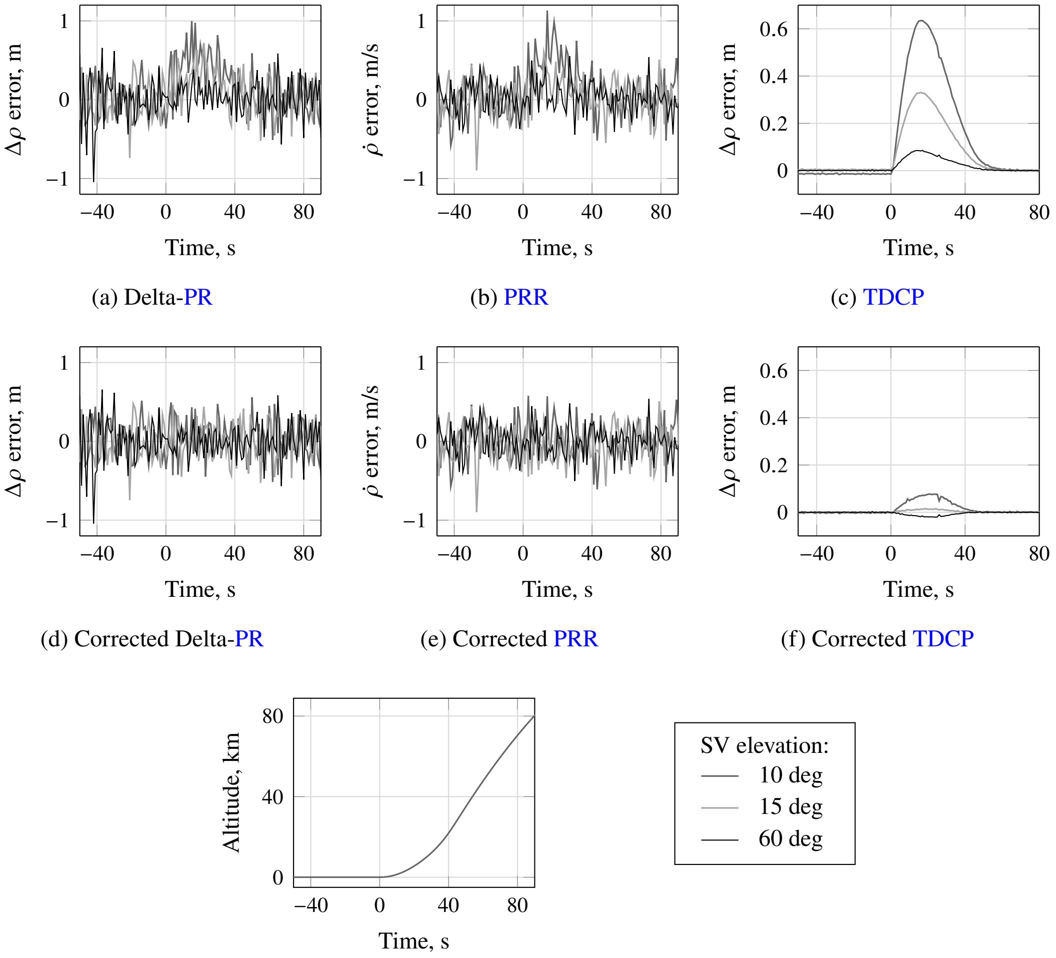
$$\Delta\tau_{\mathrm{r},k} = \tau_{\mathrm{r},k} - \tau_{\mathrm{r},k-1} \approx \Delta t \dot{\tau}_{\mathrm{r},k} \ . \tag{2.29}$$

Figure 2.11 shows the Allan standard deviation of the clock frequency error of a DLR Phoenix-HD receiver during a static test [180]. This was obtained from pseudoranges preprocessed to extract the remaining errors. It is clear that the frequency error is dominated by white noise, leading the clock phase (or bias) to approximately behave as random walk. This feature will be used in Section 3.9.2 to derive a low-order state model for the receiver clock dynamics. Receiver clock errors are the largest disturbance in TDCP observables derived from the receiver in this study. Also, the common mode error in pseudorange-rate measurements does correspond to the drift of the common mode offset in pseudoranges (and carrier phases) [112]. Careful study of receiver outputs should precede modeling as error characteristics may differ between receivers.

An alternative to the explicit modeling of clock error is the cross-differencing of raw GNSS measurements done, e.g., by the HNS in SHEFEX-2 [161]. The pseudo-measurement obtained subtracting two measurements of the same type is free of the common mode error, and thus of clock bias/drift. This method yields modest computational savings by freeing up two state slots. While performance-wise equivalent to explicit modeling, cross-differencing has a drawback: two or more channel measurements are required in a filter update, reducing robustness to partial outages. Moreover, cross-differencing couples the noise between different measurements, rendering the measurement covariance matrix non-diagonal which hinders the use of a sequential measurement updating scheme by requiring measurement pre-whitening [21].

Tropospheric errors

Refraction of GNSS signals in the Troposphere (up to 50 km altitude) introduces a delay that depends on signal travel path and atmospheric conditions [92]. For a land-based receiver this delay is fairly constant, affecting mostly position estimation. In a rocket carried receiver, the most important effect is not in position (though it is still affected) but in velocity. In fact, the ascending rocket motion through the atmosphere makes the tropospheric delay on each channel change quickly, yielding a hump-like



(a) Delta-PR

(b) PRR

(c) TDCP

(d) Corrected Delta-PR

(e) Corrected PRR

(f) Corrected TDCP

(g) Altitude

Figure 2.12: Tropospheric error in Delta-PR, PRR and TDCP (at 1 Hz) during ascent (lift-off at 0 s) before and after model-based correction

error in range-rate (or delta-range), i.e., in velocity information. While for differenced pseudorange (Delta-PR) and pseudorange-rate (PR-rate) this effect is mostly buried in measurement noise, for TDCP the considerably lower (single-channel) noise renders this effect flagrant. This is shown in Figures 2.12a to 2.12c for three different satellite elevations. The trajectory followed is that of SHEFEX-2 sounding rocket, whose altitude profile is displayed in Figure 2.12g. The signals were measured using a DLR Phoenix-HD GPS receiver stimulated by a GSS7700 SPIRENT GNSS signal emulator running a NATO STANAG troposphere model [123].

If uncorrected or unaccounted for in the filter, the effect of this changing Tropospheric delay can cause velocity estimation errors to surpass covariance bounds during the ascent. This can worsen strapdown divergence during GNSS outage, which is rather likely during this flight leg given the violent dynamic events that may then occur.

Numerous models have been derived for tropospheric delay compensation. These can be divided into two types according to their intended application: geodetic models and navigation models [37]. The former, generally based on extended analysis of atmospheric physics, are more accurate and more complex, making use of on-site atmospheric data. These include Saastamoinen and Hopfield models, and modified versions thereof [74, 70]. Navigation-oriented models, on the olther hand, are simpler and use either look-up tables of average regional/global atmospheric values or none at all (i.e., approximate these with a single expression and parameter set). An example of a navigation-oriented model is that used by the Satellite-based Augmentation System (SBAS) in GPS navigation for civil aviation, which is a modified version of the University of New Brunswick Tropospheric delay model variant 3 (UNB3) model proposed in [37]. Of the same type is also the aforementioned NATO STANAG model [123]. Examples of even simpler models, without atmospheric data tables, with exponential altitude dependence and simple elevation mapping functions are described in [51, 111, 158]. A model of this kind is used internally by the DLR Phoenix-HD receiver [117, 116]. Thorough overview and comparison of different Tropospheric delay models can be found in [111, 37, 86, 49]. Figures 2.12d to 2.12f show the errors in differenced pseudorange, pseudorange-rate and TDCP after being corrected using an airborne-navigation-oriented model, without parameter look-up, given by [158] as

$$\tau_{\mathrm{T},i} = h_{\tau_{\mathrm{T}},i}\left(\mathbf{x}\right) = \frac{1}{c} M\left(\mathrm{E}_i\right)\,\Delta\left(\mathrm{h}_{\mathsf{ant}}\right) \ , \tag{2.30a}$$

with

$$M\left(\mathrm{E}_i\right) = \frac{1.0121}{\sin\left(\mathrm{E}_i\right) + 0.0121} \tag{2.30b}$$

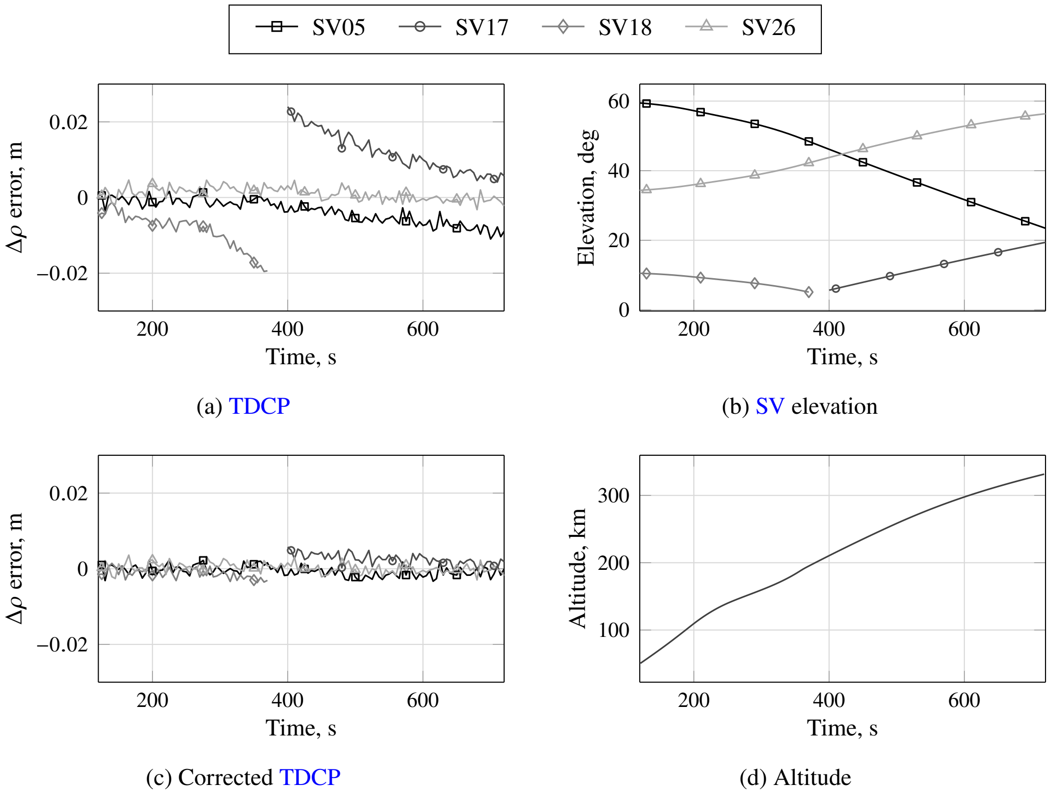
$$\Delta\left(\mathrm{h}_{\mathsf{ant}}\right) = 2.4405\ \exp\left(-0.133\times 10^{-3}\ \mathrm{h}_{\mathsf{ant}}\right) \ , \tag{2.30c}$$

where the *Zenit* delay $\Delta(\,\cdot\,)$ is a function of receiver altitude $\mathrm{h}_{\mathsf{ant}}$, and the mapping function $M(\,\cdot\,)$ depends on the satellite apparent elevation E_i. Note that for TDCP (Figure 2.12f) the residual after correction is still considerably larger than the channel noise level. An effective way of dealing with this residual is derived in Section 3.9.3.

Ionospheric errors

The Earth's Ionosphere (around 50–1000 km of altitude) is a dispersive medium that affects GNSS signal propagation. Like the Troposphere, it causes refraction, delaying the signal modulation, Pseudorandom Noise (PRN) code. However, it advances the carrier phase the same amount [51]; hence the opposite sign of Ionospheric contributions to PR and PRR or TDCP measurements in (2.25) and (2.26)–(2.27). As a dispersive medium it causes wave propagation velocity to vary with signal frequency. This allows the elimination of Ionospheric errors in dual-frequency operation (e.g., L1/L2 in GPS). In single-frequency, non-differential operation correction requires the use models.

As the Tropospheric delay, the Ionospheric effect, usually fairly constant for land-based slow-moving receivers, varies quickly on a launch trajectory. This is due to both the relatively faster changing apparent elevation of the tracked satellites, and the thinning of the effective Ionospheric layer above the receiver as the vehicle climbs. Figure 2.13a shows the error induced by the Ionospheric delay-rate on TDCP for a Vega trajectory section starting at 50 km altitude. The measurements were again collected by a DLR Phoenix-HD receiver fed by a GSS7700 SPIRENT GNSS emulator. The altitude



(a) TDCP

(b) SV elevation

(c) Corrected TDCP

(d) Altitude

Figure 2.13: Ionospheric error profile in TDCP (at 1 Hz) during ascent (lift-off at 0 s) before and after correction on a Vega VV04 trajectory

profile of the trajectory used is displayed in Figure 2.13d, while Figure 2.13b shows the apparent elevation history of the tracked satellites. The errors in PR and PRR are much lower than the noise levels in these measurements and are thus omitted. Figure 2.13a shows that the Ionospheric errors in TDCP are considerably lower than those induced by the Troposphere in the previous section. The error shows an increasing trend with lower elevation and with faster elevation rate.

As mentioned, correction can be done with a delay model. This should account for satellite elevation and changing Ionospheric vertical thickness. Montenbruck et al. [113] propose a model of the form

$$\tau_{\mathrm{I},i} = h_{\tau_{\mathrm{I}},i}(\mathbf{x}) = \frac{40.3}{cf^2} M(\mathrm{E}_{\mathrm{IP},i})\,\mathrm{VTEC}\left(\mathbf{r}_{\mathrm{IP},i}\right) \ , \tag{2.31a}$$

accounting for changing VTEC (Vertical Total Electron Content) with receiver altitude

$$\mathrm{VTEC}\left(\mathbf{r}_{\mathrm{IP},i}\right) = \frac{e - \exp\left(1 - \exp\left(-\mathrm{z}_{\mathrm{IP},i}\right)\right)}{e - \exp\left(1 - \exp\left(\mathrm{h}_0/\mathrm{H}\right)\right)}\,\mathrm{VTEC}\left(\mathbf{r}_{\mathrm{IP},i,0}\right) \ , \tag{2.31b}$$

with

$$\mathrm{z}_{\mathrm{IP}} = \frac{\mathrm{h}_{\mathrm{IP},i} - \mathrm{h}_0}{\mathrm{H}} \ , \tag{2.31c}$$

where $\mathbf{r}_{\mathrm{IP},i,0}$ is the surface (vertical) projection of the Ionospheric point $\mathbf{r}_{\mathrm{IP},i}$. This is the point, on the line-of-sight from receiver to satellite i, that lies at the altitude of 50 percentile of residual Ionosphere, i.e., the altitude at which half of the VTEC from receiver altitude to infinity is achieved. f is the L1 frequency, and h_0 and H are, respectively, the inflection point altitude and scale height of the Chapman profile. $M(\cdot)$ is the mapping function accounting for the apparent satellite elevation E_{IP}, in this case with respect to the Ionospheric point. While in [113] $M(\cdot)$ is modelled as a cosecant function, here the mapping function introduced by Lear [100] is used:

$$M(\mathrm{E}_{\mathrm{IP},i}) = \frac{2.037}{\sin\left(\mathrm{E}_{\mathrm{IP},i}\right) + \sqrt{\sin\left(\mathrm{E}_{\mathrm{IP},i}\right)^2 + 0.076}} \,. \tag{2.32}$$

VTEC $\left(\mathbf{r}_{\mathrm{IP},i,0}\right)$ can be set as a constant or computed with a regular Klobuchar model [95].

Figure 2.13c shows the residuals after correction of the TDCP measurements with the model just described, using a constant surface VTEC of 20 TECU (Total Electron Content Unit). The residual magnitude, unlike that of the tropospheric delay effect, is clearly small enough to simply be neglected.

2.5.3 GNSS measurement set and navigation performance

Figure 2.14 compares the navigation covariance performance obtained if the strapdown propagation is updated with the following sets of GNSS measurements:

- GNSS position and velocity (POS+VEL);
- Pseudorange (PR);
- Pseudorange and Pseudorange-rate (PR+PRR); and
- Pseudorange and Time-Differenced Carrier Phase (PR+TDCP).

The inertial sensor is a Medium-grade unit (Table 2.1) and the GNSS signal performance levels are those of a DLR Phoenix-HD receiver fed by a GSS7700 SPIRENT emulator. (Other receivers with different internal tuning may yield different overall performance.) Trajectory is that of Vega VV04 flight. The results assume Tropospheric and Ionospheric corrections: The raw GNSS measurement models (PR, PRR and TDCP) include covariance compensation for the correction uncertainty (see Section 2.5.2). The position and velocity measurement models assume these corrections are done by the receiver. Therefore, a (slightly) more conservative (altitude scheduled) tuning is used to account for correction residuals. The GNSS output rate is 1 Hz.

Comparing the different velocity covariance curves it is clear that the inclusion of velocity-based measurements yields improved estimation of this state. Attitude, on the other hand, is only very slightly influenced by the different GNSS information used. Among the raw measurement (tightly coupled) configurations, the one using TDCP pseudo-measurements clearly provides the most accurate velocity estimates. The PR+PRR set-up presents somewhat marginal improvements over the PR-only configuration. This is explained by the rather high noise level on the Pseudorange-rates retrieved by receiver tested, which has its internal tracking loop tuning set loose enough to minimize loss of lock under high dynamics (hence the initials HD in its name). The performance achieved by the loosely coupled set-up (POS+VEL) approaches that

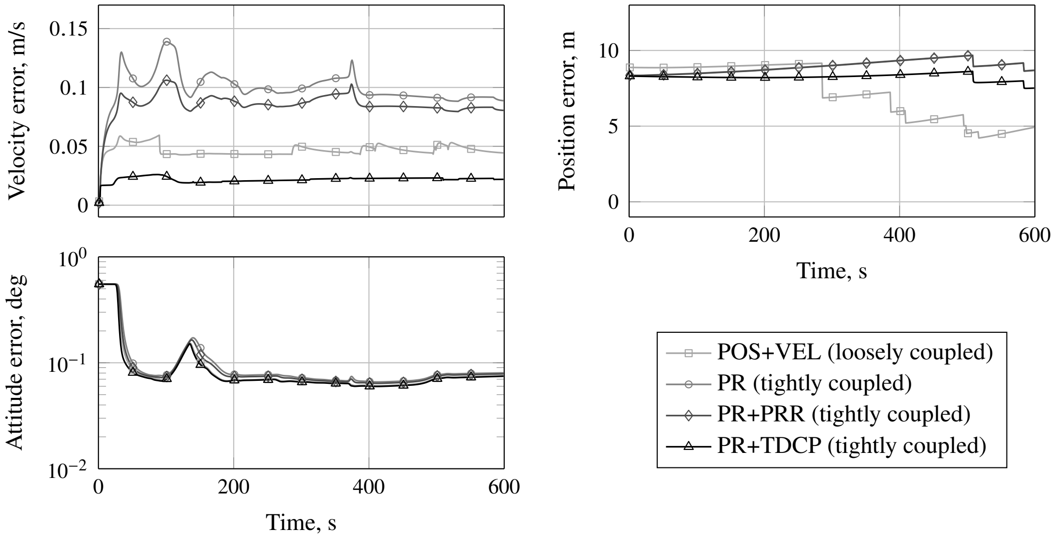


Figure 2.14: Hybrid navigation performance (1σ cov.) for different GNSS measurement sets on a Vega VV04 trajectory

of PR+TDCP because the receiver navigation solution uses Carrier Phase smoothing. The remaining difference is due to the conservative tuning of the loosely coupled filter against atmospheric correction residuals. Expected position accuracy is almost identical among tightly coupled set-ups. The POS+VEL filter shows a slightly different curve as its bias states are reset upon every change in tracked satellite set, here resulting in a lower error. Finally, the velocity estimation levels here shown are lower than those in Figure 2.9, as the tuning of the models used then was set conservatively enough (i.e., with higher uncertainty) to account for uncorrected atmospheric errors.

2.6 Filter, robustness and fault-tolerance

2.6.1 Filter algorithm considerations

The propagation and measurement models of a GNSS/inertial system may be written

$$\begin{bmatrix} \mathbf{x}_{\mathsf{kin},j+1} \\ \mathbf{x}_{\mathsf{imu},j+1} \\ \mathbf{x}_{\mathsf{gnss},j+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\phi}_{\mathsf{kin},j}\left(\mathbf{x}_{\mathsf{kin},j}, \hat{\mathbf{u}}_{j+1}\left(\tilde{\mathbf{u}}_{j+1}, \mathbf{x}_{\mathsf{imu},j}\right)\right) \\ \boldsymbol{\phi}_{\mathsf{imu},j}\left(\mathbf{x}_{\mathsf{imu},j}\right) \\ \boldsymbol{\phi}_{\mathsf{gnss},j}\left(\mathbf{x}_{\mathsf{gnss},j}\right) \end{bmatrix} + \begin{bmatrix} \mathbf{w}_{\mathsf{kin},j} \\ \mathbf{w}_{\mathsf{imu},j} \\ \mathbf{w}_{\mathsf{gnss},j} \end{bmatrix} , \qquad (2.33)$$

and

$$\mathbf{y}_{\mathsf{gnss},j} = \boldsymbol{h}_{\mathsf{gnss},j}\left(\mathbf{x}_{\mathsf{kin},j}, \mathbf{x}_{\mathsf{gnss},j}\right) + \boldsymbol{\nu}_{\mathsf{gnss},j} , \qquad (2.34)$$

where $\mathbf{x}$ is the state, $\mathbf{w}$ is the process noise and $\boldsymbol{\nu}$ is measurement noise.

The state vector is split into: kinematics states, denoted kin, comprising, e.g., velocity, position and attitude; IMU states, imu, including biases, scale-factors, misalignments, and others; and GNSS measurement model states, gnss, for instance, receiver clock bias/drift and channel/solution biases. The kinematics propagation model, $\boldsymbol{\phi}_{\mathsf{kin}}$, uses corrected inertial measurements $\hat{\mathbf{u}}$, computed from actual measurements $\tilde{\mathbf{u}}$ and IMU perturbation states $\mathbf{x}_{\mathsf{imu}}$.

A closer look at the structure of the propagation and measurement functions reveals several instances of non-linearities. The attitude propagation law (in $\boldsymbol{\phi}_{\mathsf{kin}}$) is non-linear. So is the gravity model and some of the IMU error corrections in $\hat{\mathbf{u}}$, e.g., scale-factors and misalignments (multiplicative perturbations). The measurement mappings of the raw GNSS measurements are either of range or range-rate which are also non-linear. On the other hand, position and velocity measurements (in a loosely coupled design) are given by linear models. Linear is also the propagation of both IMU and GNSS measurement states (functions $\boldsymbol{\phi}_{\mathsf{imu}}$ and $\boldsymbol{\phi}_{\mathsf{gnss}}$).

The most widely used approach for such a system is the Extended Kalman Filter (EKF). This uses the full non-linear model for state integration and innovation computation, and a linearisation of this for covariance propagation and update (Kalman) gain calculation. The non-linear nature of the system has motivated extensive experimentation with more complex non-linear filtering schemes. Wendel et al. [197] observed only marginal performance improvement of the Unscented Kalman Filter (UKF), also known as Sigma-Point Kalman filter, in an GPS/INS system with respect to the EKF. (Other authors have reported similar results [67, 131].) The non-linearities in the GNSS raw measurement models and in the kinematics state propagation of both medium and low grade sensors were found to be moderate, and thus well suited for the EKF. Significantly faster convergence by the UKF was however observed for very high errors at initialization. Given the application at hand, this may be of little importance in on-ground initialization (at the launch pad), but rather advantageous in the recovery from long GNSS outages during flight. The considerably higher computational complexity of the UKF makes it less attractive in this trade-off.

Another important point to consider in the filter algorithm design is the rate of its internal routines. Assuming a modular (indirect) filtering approach, as defined in Section 2.3.3, the fusion algorithm may perform error-state propagation, covariance propagation and measurement update at different rates [183, 159]. The GNSS measurement updates can be performed at the receiver's output rate (generally 1 Hz) to avoid loss of information. As for the filter prediction steps, as covariance propagation is comparatively more "expensive" than state propagation, this step may be performed at a lower rate than the filter error-state integration. The latter can, in turn, be performed at a (considerably) lower rate than the parallel strapdown propagation. Selection of propagation rates must account for target vehicle dynamics and trajectory.

2.6.2 Filter robustness

As previously discussed, IMU and GNSS measurements can be corrupted by numerous errors. A natural way of making the navigation robust against these is by modeling and estimating their effect within the filtering algorithm. This is not always possible nor practical. From lack of suitable/accurate models, to growth of filter size, there are several drawbacks to this approach. Robustness can still be achieved through other means. Conservative tuning methods as pseudonoise addition [108] and measurement under-weighting [205] are commonly used. The following two instances of this approach were mentioned in previous sections. The SHEFEX-2 HNS uses inflated process noise values for inertial sensor scale-factor states to compensate for the unmodeled non-linearities [160]; the HNS also heavily under-weighs pseudorange measurements at lower altitudes to render the filter robust against Tropospheric effects [161].

It may be argued that many of the robust (Kalman-like) filter implementations found in literature (e.g., fading memory filters [108, 153]) fall into this set of conservative tuning techniques. Others depart from it by requiring some level of model or uncertainty structure (e.g., $\mathcal{H}_\infty$ Kalman filtering [153]). Within this spectrum, and somewhat closer to full modeling, lie the Schmidt Kalman filtering (or *consider*-state filtering).

Introduced in [148], the Schmidt Kalman filter includes a set of parameters, the consider-state set, that, despite statistically modeled in all covariance operations, does not feature in the state vector and is therefore not estimated [203, 202, 38]. This is equivalent to a regular Kalman filter in which the gain entries of certain (considered) states are zeroed-out. In practice, computation of these components is skipped, effectively yielding a reduced-order filter. This technique has the advantage of reducing the computational cost of accounting for nuisance system parameters (e.g., unresolvable biases, colored noises); or, from a full-filter perspective, it allows state reduction without sacrificing robustness. It has the drawback of yielding degraded performance with respect to the full-order filter if the considered parameters are observable. Derivation and implementation are also more involved than in a regular Kalman filter.

2.6.3 Tolerance to faults

Integrity and reliability requirements for launch navigation systems are generally very stringent given the mission- (and even safety-) critical nature of this subsystem. Fault detection is thus a crucial element in the navigation design. As previously mentioned, in “traditional” inertial launcher navigation fault detection and contingency are usually accomplished through redundancy of inertial units (e.g., Ariane-V [25]). The inclusion of GNSS measurements adds an important source of redundant information which can be used for Fault Detection and Isolation (FDI), as recommended in [101] for GNSS-based hybrid navigation systems in future re-usable and expendable LVs.

Most GNSS receivers use Receiver Autonomous Integrity Monitoring (RAIM) routines to verify the sanity of their raw measurements. These employ self-consistency tests to detect and potentially isolate receiver channel blunders. The most common algorithms operate on single-epoch data (snapshot) and exploit the statistical properties of Least Squares (LS) solutions. One example is the LS-Residuals method in which the norm of the difference between the actual measurements and those predicted using the LS solution is compared to some sanity threshold [30]. A more heuristic and more intensive approach compares the LS solutions for all sets of $N - 1$ GNSS raw measurements (N being the number of tracked channels); if a single channel blunder is present then the solution that excludes it will be the one with the maximum separation from the remaining. This is known as Maximum Solution Separation method [30].

In hybrid GNSS/inertial systems, *a-priori* (i.e., predicted, or before measurement update) state information can be used to test the sanity of each new set of GNSS measurements. This is easily done by testing measurement innovations within the filter. Rather than a self-consistency test, this method compares the new GNSS information with the inertially propagated previous GNSS history, effectively increasing GNSS failure detection levels. Testing single innovations also allows the identification and rejection of single-channel outliers [130].

Having access to both raw and filter predicted measurements an approach mixing RAIM checks and filter innovation testing can be conceived. A combination of RAIM-like self-consistency algorithms with innovation testing could allow identification of not only single/multiple-channel outliers, but also common-mode (receiver clock) faults and inertial propagation faults. Note that both receiver clock faults and excessive inertial propagation divergence (during GNSS latency or outage) will result in generalized innovation failure (violation of some pre-defined threshold). By parallelly checking the self-consistency of the GNSS measurement set, the system is able to tell these faults apart from multiple channel failure (e.g., severe Ionospheric perturbations or multipath effects). In case of consistent measurement set, distinction between common-mode receiver fault and excessive inertial divergence can then be done through the comparison of GNSS-only position/clock solution and the corresponding filter state values. If needed, filter recovery could be done by covariance inflation of the affected states (position/velocity or clock bias/drift) followed by regular measurement update. This scheme is fully developed in Chapter 5.

3
Baseline Hybrid Navigation System Design

This chapter describes the baseline launcher navigation system architecture and algorithm design. The concept follows the findings and conclusions of the previous chapter. The core of the design presented in this chapter has, in part, been published in [183, 180]. The strapdown algorithm and modular closed-loop architecture concept have also been published in [178, 179].

3.1 Motivation and design guidelines

As stated in the introductory chapter, the goal of this work is to conceive a robust, fault-tolerant primary navigation system for launch application able to provide the necessary performance level using lower grade sensor units. Here, *necessary performance level* can be seen, as discussed in Section 2.2, as that delivered by the inertial-only systems currently employed in the envisage launch application (be it small/large launcher or sounding rocket). In terms of actual performance target, while a fixed set of required minimum figures is not set for the design process at hand, the velocity and position numbers proposed by [17] (0.25 m/s and 25 m, 1σ, respectively) and the attitude requirement of SHEFEX-2 [160] (0.17 deg, 1σ, at final coasting) are taken as guidelines. (Both sets of performance requirements are covered in Section 2.2.) This design decision is based on two points:

- Firstly, the system here proposed shall represent, as far as possible, a general design concept (and design process) of a launcher navigation system which shall be translatable to platforms other than the ones here exemplified;
- Secondly, and in part as a result of the first point, the inertial measurement unit profiles used (see Table 2.1) do not exactly correspond to any specific product but represent entire classes of sensors; changes in these specifications to those of a specific (real) unit would naturally change the achieved final performance.

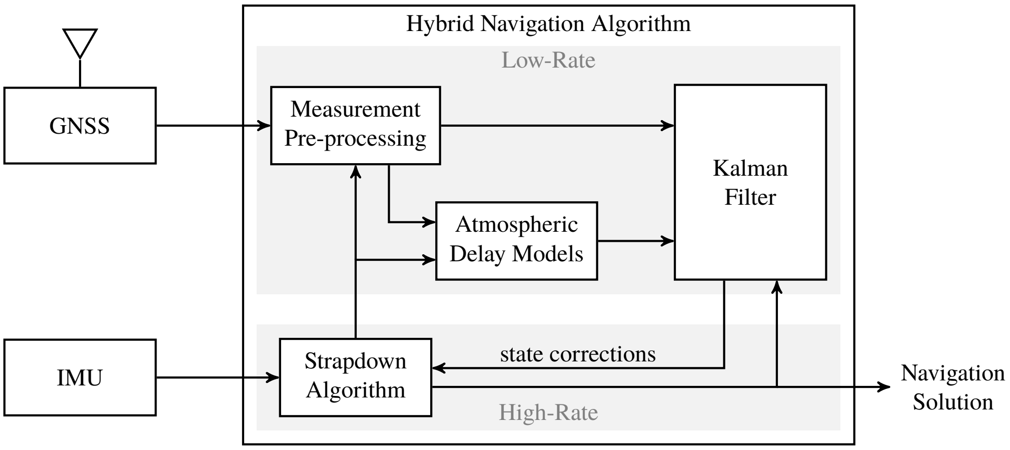


Figure 3.1: Hybrid navigation algorithm architecture

In addition to these guidelines, the system shall guarantee continuous navigation solution, even in the temporary absence of GNSS aiding. This is crucial as Guidance and Control functions depend on this information to steer the vehicle. Furthermore, it shall be robust against disturbances, including inertial sensor errors and atmospheric effects on the GNSS signals, and it shall be tolerant to faults on both sensors. Functional separation between system components shall be ensured as much as feasible.

Robustness of the design is only in part tackled in this chapter; it is further explored in Chapter 4. Fault-tolerance is covered entirely in Chapter 5.

3.2 System architecture

The proposed hybrid navigation architecture, shown in Figure 3.1, features:

- Strapdown propagation and filter algorithm modularly set up (see Section 2.3.3), running in parallel at different rates: strapdown runs at the (high) output rate of the IMU (time index j) while the filter update is done at the (low) rate of the GNSS outputs (time index k). Figure 3.2 depicts this timing sequence, with N being the number of high-rate cycles within a low-rate cycle.
- Filter estimates of kinematics and sensor parameter errors regularly fed to the strapdown to correct the propagation and calibrate subsequent inertial increments. This follows a closed-loop indirect filtering architecture described in Sections 2.3.2 and 2.3.3. The filter is then reset, remaining close to the origin and thus reducing linearization errors. This modular set-up also allows for a clear separation between strapdown kinematics states propagation and filter operation.

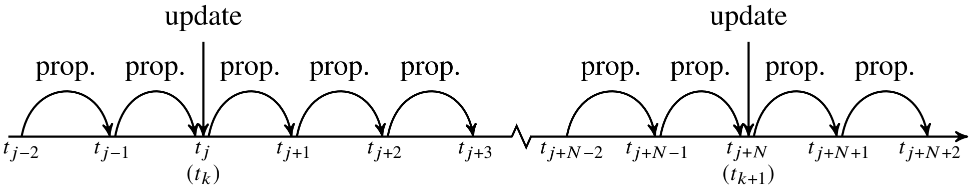


Figure 3.2: Timing sequence in terms of strapdown propagation and filter update

- Static measurement updates (see Section 2.4.4) while on the launchpad. These allow for system alignment while stabilizing/improving the estimation of kinematic and sensor states before the launch. Earth-rate and Zero-velocity pseudo-measurements are used.

- GNSS pseudorange and time-differenced carrier phase measurment updates (i.e., tightly-coupled architecture, see Section 2.3.1). Tropospheric and Ionospheric delays affecting these measurements are corrected using simple models. Uncertainty in Tropospheric correction is accounted for in the update models. Receiver clock offsets are carefully modeled for the receiver used (Phoenix-HD).

- An Extended Kalman filter to handle the moderate non-linearities present in the kinematics propagation (especially attitude) and range-based measurement models. Adequacy of this type of algorithm has been discussed in Section 2.6.1.

- Strapdown running within the navigation computer, as opposed to within the inertial platform, fulfilling the functional separation requirement, discussed in Section 2.2. (Inertial platform integration can still take place, providing an extra inertial-only solution, only without the calibration feedback.) Aiding of GNSS (re)acquisition is likewise foregone.

3.3 Strapdown inertial propagation

As described in Section 2.4.2, the strapdown algorithm integrates inertial measurements, propagating attitude, velocity and position. The IMU measures absolute (inertial) angular and velocity increments in Body frame coordinates. Integration is here done in ECEF coordinates. To minimize the effects of changing Body frame within an integration interval (see Section 2.4.2) propagation is done at the output rate of the IMU. The strapdown algorithm here developed has been further used in [178, 179].

Attitude is integrated as

$$\mathbf{q}_{B_{j+1}}^{E_{j+1}} = \left(\mathbf{q}_{B_j}^{B_{j+1}}\right)^{-1} \mathbf{q}_{B_j}^{E_j}\mathbf{q}_{E_j}^{E_{j+1}} \ , \tag{3.1}$$

with $\mathbf{q}_{B_j}^{B_{j+1}}$ computed with the third-order law [109]

$$\mathbf{q}_{B_j}^{B_{j+1}} = \begin{bmatrix} \frac{1}{2}\Delta\boldsymbol{\theta}_j^B - \frac{1}{48}\|\Delta\boldsymbol{\theta}_j^B\|^2\Delta\boldsymbol{\theta}_j^B - \frac{1}{24}\Delta\boldsymbol{\theta}_{j-1}^B \times \Delta\boldsymbol{\theta}_j^B \\ 1 - \frac{1}{8}\|\Delta\boldsymbol{\theta}_j^B\|^2 \end{bmatrix} \ . \tag{3.2}$$

Velocity and position propagation is obtained translating (2.1b)–(2.1c) to ECEF, as

$$\begin{aligned} \mathbf{v}_{j+1}^E &= -\boldsymbol{\Omega}_{IE}^E\mathbf{r}_{j+1}^E + \mathbf{C}_{E_j}^{E_{j+1}}\left(\mathbf{v}_j^E + \boldsymbol{\Omega}_{IE}^E\mathbf{r}_j^E\right) + \mathbf{C}_I^{E_{j+1}}\left(\Delta\mathbf{v}_{\mathrm{sf},j}^I + \Delta\mathbf{v}_{\mathrm{g},j}^I\right) & (3.3) \\ \mathbf{r}_{j+1}^E &= \mathbf{C}_{E_j}^{E_{j+1}}\mathbf{r}_j^E + \mathbf{C}_I^{E_{j+1}}\Delta\mathbf{r}_j^I \ , & (3.4) \end{aligned}$$

where $\boldsymbol{\Omega}_{IE}^E = [\boldsymbol{\omega}_{IE}^E\times]$, i.e., the skew-symmetric matrix of the Earth's absolute angular velocity written in ECEF coordinates. As in (2.1b), the acceleration integral is split into specific-force and gravitational Delta-V terms, $\Delta\mathbf{v}_{\mathrm{sf},j}^I$ and $\Delta\mathbf{v}_{\mathrm{g},j}^I$, respectively.

Using a trapezoidal integration on the velocity integral in (2.1c), $\mathbf{\Delta r}^I_j$ in (3.4), gives

$$\begin{aligned}
\mathbf{\Delta r}^I_j &\approx \frac{\Delta t_j}{2}\left(\mathbf{v}^I_j + \mathbf{v}^I_{j+1}\right) && (3.5)\\
&\approx \frac{\Delta t_j}{2}\left(\mathbf{C}^I_{E_j}\left(\mathbf{v}^E_j + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_j\right) + \mathbf{C}^I_{E_{j+1}}\left(\mathbf{v}^E_{j+1} + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_{j+1}\right)\right) && (3.6)\\
&\approx \frac{\Delta t_j}{2}\left(\mathbf{C}^I_{E_j}\left(\mathbf{v}^E_j + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_j\right) + \mathbf{C}^I_{E_j}\left(\mathbf{v}^E_j + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_j\right) + \mathbf{\Delta v}^I_{\mathsf{sf},j} + \mathbf{\Delta v}^I_{\mathsf{g},j}\right) && (3.7)\\
&\approx \Delta t_j \mathbf{C}^I_{E_j}\left(\mathbf{v}^E_j + \mathbf{\Omega}^E_{IE}\mathbf{r}^E_j\right) + \frac{\Delta t_j}{2}\left(\mathbf{\Delta v}^I_{\mathsf{sf},j} + \mathbf{\Delta v}^I_{\mathsf{g},j}\right) \ . && (3.8)
\end{aligned}$$

This can be replaced in (3.3) and (3.4) yielding

$$\begin{bmatrix}\mathbf{v}^E_{j+1}\\ \mathbf{r}^E_{j+1}\end{bmatrix} \approx \begin{bmatrix}\mathbf{I} - \Delta t_j \mathbf{\Omega}^E_{IE} & -\Delta t_j {\mathbf{\Omega}^E_{IE}}^2\\ \Delta t_j \mathbf{I} & \mathbf{I} + \Delta t_j \mathbf{\Omega}^E_{IE}\end{bmatrix}\begin{bmatrix}\mathbf{C}^{E_{j+1}}_{E_j}\mathbf{v}^E_j\\ \mathbf{C}^{E_{j+1}}_{E_j}\mathbf{r}^E_j\end{bmatrix} + \begin{bmatrix}\mathbf{I} - \frac{\Delta t_j}{2}\mathbf{\Omega}^E_{IE}\\ \frac{\Delta t_j}{2}\mathbf{I}\end{bmatrix}\mathbf{C}^{E_{j+1}}_I\left(\mathbf{\Delta v}^I_{\mathsf{sf},j} + \mathbf{\Delta v}^I_{\mathsf{g},j}\right) \ . \tag{3.9}$$

The specific-force Delta-V in (3.9) is assumed expressed in coordinates of Body frame at the middle of the integration interval, as

$$\mathbf{\Delta v}^I_{\mathsf{sf},j} \approx \mathbf{C}^I_{E_j}\mathbf{C}^{E_j}_{B_j}\mathbf{C}^{B_j}_{B_{j+\frac{1}{2}}}\mathbf{\Delta v}^B_j \ , \tag{3.10}$$

where $\mathbf{\Delta v}^B_j$ is the accelerometer integrated measurement from t_j to t_{j+1}. This reduces the error caused by the rotation of Body frame during accelerometer sampling. To understand how, consider the following derivation.

The IMU performs the integration

$$\Delta\mathbf{v}^B_j = \int_{t_j}^{t_{j+1}} \mathbf{a}^B \, \mathsf{d}t \ , \tag{3.11}$$

in which B is moving with respect to inertial space during the integration cycle, so $\Delta\mathbf{v}^B_j$ in (3.11) is not exactly expressed in coordinates of B_{j+1} or B_j. To have it written, for instance, in B_j the integration would instead be

$$\Delta\mathbf{v}^{B_j}_j = \int_{t_j}^{t_{j+1}} \mathbf{C}^{B_j}_B \mathbf{a}^B \, \mathsf{d}t \ . \tag{3.12}$$

Assuming that both angular velocity and specific-force acceleration are constant in Body frame during one sample period yields

$$\begin{aligned}
\Delta\mathbf{v}^{B_j}_j &\approx \int_{t_j}^{t_{j+1}} \left(\mathbf{I} - \left(t_j - t\right)\mathbf{\Omega}^B_{IB}\right)\mathbf{a}^B \, \mathsf{d}t && (3.13)\\
&\approx \int_{t_j}^{t_{j+1}} \mathbf{a}^B \, \mathsf{d}t + \int_{t_j}^{t_{j+1}} \left(t - t_j\right)\mathbf{\Omega}^B_{IB}\mathbf{a}^B \, \mathsf{d}t && (3.14)\\
&\approx \Delta\mathbf{v}^B_j + \frac{\Delta t_j}{2}\mathbf{\Omega}^B_{IB}\Delta\mathbf{v}^B_j && (3.15)\\
&\approx \mathbf{C}^{B_j}_{B_{j+\frac{1}{2}}}\Delta\mathbf{v}^B_j \ , && (3.16)
\end{aligned}$$

which is analogous to assuming that $\Delta\mathbf{v}_j^B$ is effectively expressed in coordinates of $B_{j+\frac{1}{2}}$ frame, i.e., the midpoint Body frame. This approximation is, in practice, a one-sample sculling correction scheme (cf., e.g., [145]).

The gravity-induced Delta-V in (3.9) is also evaluated at the midpoint of the interval

$$\begin{aligned}\Delta\mathbf{v}_{g,j}^I &= \int_{t_j}^{t_{j+1}} \mathbf{g}^I \,\mathrm{d}t \\ &\approx \Delta t_j \; \mathbf{C}_{E_{j+\frac{1}{2}}}^I \boldsymbol{g}\left(\mathbf{r}_{j+\frac{1}{2}}^E\right) \\ &\approx \Delta t_j \; \mathbf{C}_{E_{j+\frac{1}{2}}}^I \boldsymbol{g}\left(\mathbf{C}_{E_j}^{E_{j+\frac{1}{2}}}\left(\mathbf{r}_j^E + \frac{\Delta t_j}{2}\mathbf{v}_j^E\right)\right) . \end{aligned} \tag{3.17}$$

where $\Delta t_j = t_{j+1} - t_j$. As in [161], a spherical harmonic gravity model (EGM2008 [129]) of 9th order and degree is used.

3.4 Error-state Kalman filter

The error-state Kalman filter uses two sets of states: Whole-states, denoted $\mathbf{x}$; and Error-states, denoted $\boldsymbol{\delta}\mathbf{x}$. Error-states hold small perturbations around the (usually) larger numeric quantities held by the whole-states. While the filter propagation process acts on both sets, the update step operates directly on the error-states; it computes the filter correction as a perturbation of the whole-state. After an update cycle of a single or multiple update steps, the filter is reset, i.e., the error-state quantities are integrated into the whole-state set. The error-states are then zeroed. A mode diagram depicting this process is shown in Figure 3.3. The index superscripts − and + denote *a priori* (meaning, before correction) and *a posteriori* (meaning, after correction), respectively.

This typical error-state Kalman filter, here in the Extended form (e.g., [132, 161]), is re-derived to accommodate correlation between measurement and process noise, that, as shall be shown, arises in the modeling of clock error for the GNSS receiver used in this study. The filter algorithm is summarized by the flow diagram in Figure 3.4 and is derived as follows.

The whole-state system and measurement models are written as

$$\mathbf{x}_{k+1} = \boldsymbol{\phi}_{k+1|k}(\mathbf{x}_k) + \mathbf{w}_k \tag{3.18}$$

$$\mathbf{y}_k = \boldsymbol{h}_k(\mathbf{x}_k) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \;, \tag{3.19}$$

with $\mathbf{w}_k \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{Q}_{k+1|k}\right)$, $\boldsymbol{\epsilon}_k \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{R}_{\epsilon,k}\right)$ and $\boldsymbol{\nu}_k \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{R}_{\nu,k}\right)$. The measurement noise portion $\boldsymbol{\epsilon}$ in (3.19) is assumed correlated with the process noise $\mathbf{w}$, giving

$$E\left\{\boldsymbol{\epsilon}_k \mathbf{w}_{k-1}^\mathsf{T}\right\} = \mathbf{B}_k^\mathsf{T} \;. \tag{3.20}$$

The error-state system and measurement models are derived from (3.18)–(3.19) as

$$\begin{aligned}\boldsymbol{\delta}\mathbf{x}_{k+1} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1} \\ &= \boldsymbol{\phi}_{k+1|k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - \boldsymbol{\phi}_{k+1|k}(\hat{\mathbf{x}}_k) + \mathbf{w}_k \;, \end{aligned} \tag{3.21}$$

$$\begin{aligned}\boldsymbol{\delta}\mathbf{y}_k &= \mathbf{y}_k - \hat{\mathbf{y}}_k \\ &= \boldsymbol{h}_k(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - \boldsymbol{h}_k(\hat{\mathbf{x}}_k) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \;. \end{aligned} \tag{3.22}$$

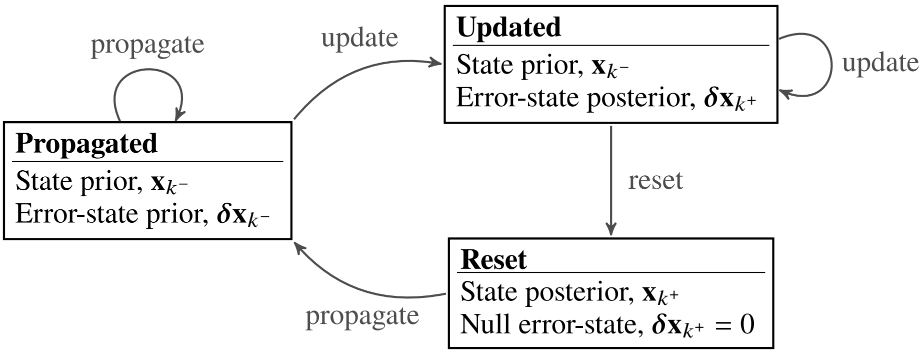

propagate
update
Updated
State prior, $\mathbf{x}_{k^-}$
Error-state posterior, $\delta\mathbf{x}_{k^+}$
update
Propagated
State prior, $\mathbf{x}_{k^-}$
Error-state prior, $\delta\mathbf{x}_{k^-}$
reset
Reset
State posterior, $\mathbf{x}_{k^+}$
Null error-state, $\delta\mathbf{x}_{k^+} = 0$
propagate

Figure 3.3: Mode diagram of the derived error-state Kalman filter

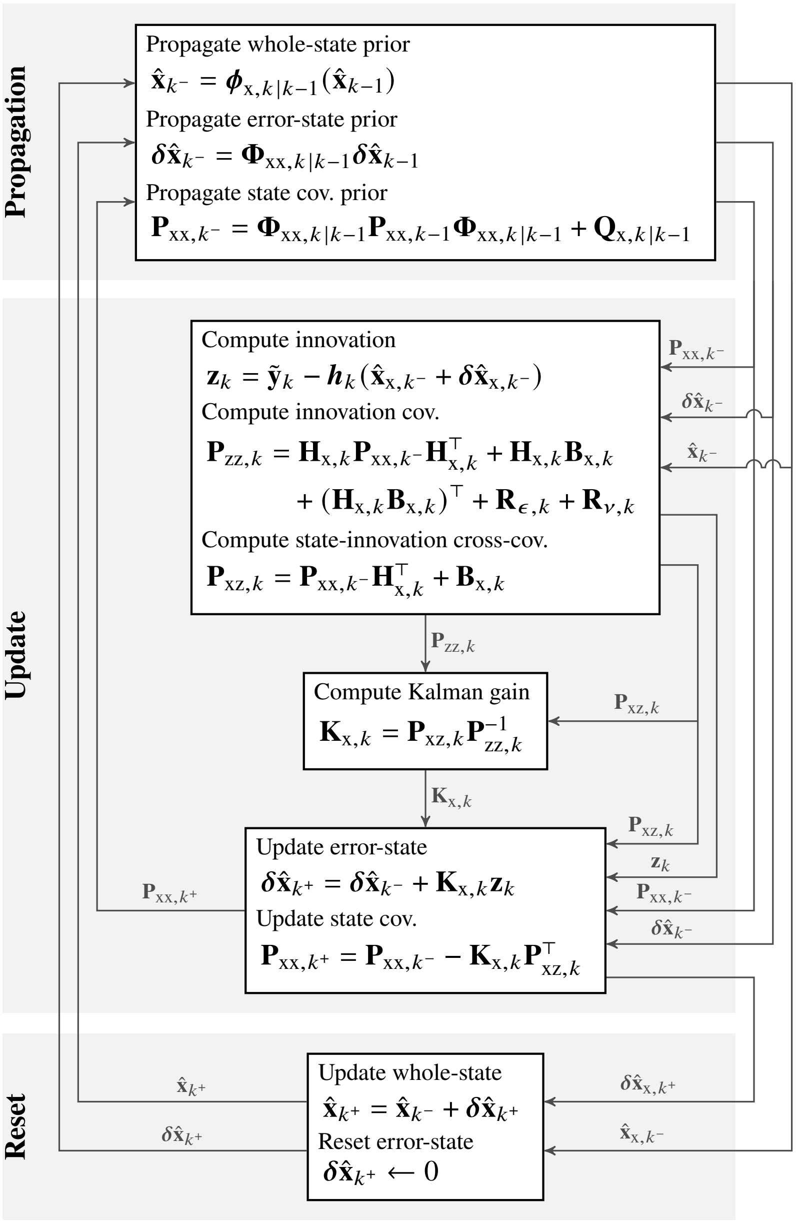

Propagation
Propagate whole-state prior
$\hat{\mathbf{x}}_{k^-} = \boldsymbol{\phi}_{\mathrm{x},k|k-1}(\hat{\mathbf{x}}_{k-1})$
Propagate error-state prior
$\delta\hat{\mathbf{x}}_{k^-} = \mathbf{\Phi}_{\mathrm{xx},k|k-1}\delta\hat{\mathbf{x}}_{k-1}$
Propagate state cov. prior
$\mathbf{P}_{\mathrm{xx},k^-} = \mathbf{\Phi}_{\mathrm{xx},k|k-1}\mathbf{P}_{\mathrm{xx},k-1}\mathbf{\Phi}_{\mathrm{xx},k|k-1} + \mathbf{Q}_{\mathrm{x},k|k-1}$
Update
Compute innovation
$\mathbf{z}_k = \tilde{\mathbf{y}}_k - \boldsymbol{h}_k(\hat{\mathbf{x}}_{\mathrm{x},k^-} + \delta\hat{\mathbf{x}}_{\mathrm{x},k^-})$
Compute innovation cov.
$\mathbf{P}_{\mathrm{zz},k} = \mathbf{H}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}_{\mathrm{x},k}^\top + \mathbf{H}_{\mathrm{x},k}\mathbf{B}_{\mathrm{x},k}$
$+ (\mathbf{H}_{\mathrm{x},k}\mathbf{B}_{\mathrm{x},k})^\top + \mathbf{R}_{\epsilon,k} + \mathbf{R}_{\nu,k}$
Compute state-innovation cross-cov.
$\mathbf{P}_{\mathrm{xz},k} = \mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}_{\mathrm{x},k}^\top + \mathbf{B}_{\mathrm{x},k}$
$\mathbf{P}_{\mathrm{xx},k^-}$
$\delta\hat{\mathbf{x}}_{k^-}$
$\hat{\mathbf{x}}_{k^-}$
$\mathbf{P}_{\mathrm{zz},k}$
Compute Kalman gain
$\mathbf{K}_{\mathrm{x},k} = \mathbf{P}_{\mathrm{xz},k}\mathbf{P}_{\mathrm{zz},k}^{-1}$
$\mathbf{P}_{\mathrm{xz},k}$
$\mathbf{K}_{\mathrm{x},k}$
$\mathbf{P}_{\mathrm{xz},k}$
$\mathbf{z}_k$
$\mathbf{P}_{\mathrm{xx},k^-}$
$\delta\hat{\mathbf{x}}_{k^-}$
$\mathbf{P}_{\mathrm{xx},k^+}$
Update error-state
$\delta\hat{\mathbf{x}}_{k^+} = \delta\hat{\mathbf{x}}_{k^-} + \mathbf{K}_{\mathrm{x},k}\mathbf{z}_k$
Update state cov.
$\mathbf{P}_{\mathrm{xx},k^+} = \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{K}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xz},k}^\top$
Reset
$\hat{\mathbf{x}}_{k^+}$
$\delta\hat{\mathbf{x}}_{k^+}$
Update whole-state
$\hat{\mathbf{x}}_{k^+} = \hat{\mathbf{x}}_{k^-} + \delta\hat{\mathbf{x}}_{k^+}$
Reset error-state
$\delta\hat{\mathbf{x}}_{k^+} \leftarrow 0$
$\delta\hat{\mathbf{x}}_{\mathrm{x},k^+}$
$\hat{\mathbf{x}}_{\mathrm{x},k^-}$

Figure 3.4: Data flow diagram of the derived error-state Kalman filter

Note that the error-state system can be linearized around the origin, yielding

$$\delta\mathbf{x}_{k+1} = \boldsymbol{\Phi}_{k+1|k}\delta\mathbf{x}_k + \mathbf{w}_k \ . \tag{3.23}$$

As described in [132], the filter estimate for $\mathbf{x}$ is actually $\hat{\mathbf{x}}+\delta\hat{\mathbf{x}}$, which has the covariance

$$\mathbf{P}_{\mathrm{xx},k} = E\left\{(\mathbf{x}_k - \hat{\mathbf{x}}_k - \delta\hat{\mathbf{x}}_k)\left(\mathbf{x}_k - \hat{\mathbf{x}}_k - \delta\hat{\mathbf{x}}_k\right)^\mathsf{T}\right\} = E\left\{(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_k)\left(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_k\right)^\mathsf{T}\right\} \ , \tag{3.24}$$

which is propagated as

$$\mathbf{P}_{\mathrm{xx},(k+1)^-} = \boldsymbol{\Phi}_{k+1|k}\mathbf{P}_{\mathrm{xx},k^+}\boldsymbol{\Phi}^\mathsf{T}_{k+1|k} + \mathbf{Q}_{k+1|k} \ . \tag{3.25}$$

The process transition matrix $\boldsymbol{\Phi}_{k+1|k}$ is the Jacobian of the vector function $\boldsymbol{\phi}_{k+1|k}(\mathbf{x}_k)$ evaluated at the *a posteriori* whole-state estimate at time step k

$$\boldsymbol{\Phi}_{k+1|k} = \left.\frac{\partial\boldsymbol{\phi}_{k+1|k}(\mathbf{x}_k)}{\partial\mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k^+}} \ . \tag{3.26}$$

The filter correction step is given by

$$\delta\hat{\mathbf{x}}_{k^+} = \delta\hat{\mathbf{x}}_{k^-} + \mathbf{K}_k\mathbf{z}_k \ , \tag{3.27}$$

having its covariance worked out as

$$\begin{aligned}\mathbf{P}_{\mathrm{xx},k^+} &= E\left\{(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_{k^+})\left(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_{k^+}\right)^\mathsf{T}\right\} \\ &= E\left\{\left(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_{k^-} - \mathbf{K}_{\mathrm{x},k}\mathbf{z}_k\right)\left(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_{k^-} - \mathbf{K}_{\mathrm{x},k}\mathbf{z}_k\right)^\mathsf{T}\right\} \\ &= \mathbf{P}_{\mathrm{xx},k^-} - E\left\{(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_{k^-})\,\mathbf{z}_k^\mathsf{T}\right\}\mathbf{K}_k^\mathsf{T} - \mathbf{K}_k E\left\{\mathbf{z}_k\left(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_{k^-}\right)^\mathsf{T}\right\} + \mathbf{K}_k E\left\{\mathbf{z}_k\mathbf{z}_k^\mathsf{T}\right\}\mathbf{K}_k^\mathsf{T} \\ &= \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{P}_{\mathrm{xz},k^-}\mathbf{K}_k^\mathsf{T} - \mathbf{K}_k\mathbf{P}^\mathsf{T}_{\mathrm{xz},k^-} + \mathbf{K}_k\mathbf{P}_{\mathrm{zz},k}\mathbf{K}_k^\mathsf{T} \ . \end{aligned} \tag{3.28}$$

The Kalman gain $\mathbf{K}$ is obtained through the minimization of the trace of (3.28) (e.g., as in [13]) as

$$\begin{aligned}\frac{\partial\,\mathsf{Tr}\left(\mathbf{P}_{\mathrm{xx},k^+}\right)}{\partial\mathbf{K}_k} &= -2\mathbf{P}_{\mathrm{xz},k^-} + 2\mathbf{K}_k\mathbf{P}_{\mathrm{zz},k} = 0 \\ \Rightarrow \quad \mathbf{K}_k &= \mathbf{P}_{\mathrm{xz},k^-}\mathbf{P}^{-1}_{\mathrm{zz},k} \ . \end{aligned} \tag{3.29}$$

The innovation $\mathbf{z}_k$ in (3.27) is

$$\mathbf{z}_k = \delta\mathbf{y}_k - \boldsymbol{h}_k(\hat{\mathbf{x}}_{k^-} + \delta\hat{\mathbf{x}}_{k^-}) + \boldsymbol{h}_k(\hat{\mathbf{x}}_{k^-}) \, , \tag{3.30}$$

with $\delta\mathbf{y}_k = \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k$, and $\tilde{\bullet}$ denoting the actual measurement. Note that combining (3.30) with (3.22) reduces the innovation model to

$$\mathbf{z}_k = \boldsymbol{h}_k(\hat{\mathbf{x}}_{k^-} + \delta\mathbf{x}_k) - \boldsymbol{h}_k(\hat{\mathbf{x}}_{k^-} + \delta\hat{\mathbf{x}}_{k^-}) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \ . \tag{3.31}$$

which, depending on the non-linearity level of $\boldsymbol{h}_k(\mathbf{x}_k)$, can be linearized to

$$\mathbf{z}_k = \mathbf{H}_k(\delta\mathbf{x}_k - \delta\hat{\mathbf{x}}_k) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \ , \tag{3.32}$$

with measurement sensitivity matrix

$$\mathbf{H}_k = \left.\frac{\partial \boldsymbol{h}_k(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k^-}} . \tag{3.33}$$

The state-to-innovation cross-covariance is

$$\mathbf{P}_{\mathrm{xz},k} = E\left\{(\boldsymbol{\delta}\mathbf{x}_k - \boldsymbol{\delta}\hat{\mathbf{x}}_k)\,\mathbf{z}_k^\top\right\} = \mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}_k^\top + \mathbf{B}_k \ , \tag{3.34}$$

and the innovation covariance $\mathbf{P}_{\mathrm{zz},k}$ is

$$\mathbf{P}_{\mathrm{zz},k} = E\left\{\mathbf{z}_k\mathbf{z}_k^\top\right\} = \mathbf{H}_k\mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}_k^\top + \mathbf{H}_k\mathbf{B}_k + \mathbf{B}_k^\top\mathbf{H}_k^\top + \mathbf{R}_{\epsilon,k} + \mathbf{R}_{\nu,k} \,. \tag{3.35}$$

The measurement model Jacobian is as in (3.33) and the covariance correlation matrix between measurement and process noise is as in (3.20).

Finally, the covariance update law (3.28) can be simplified using (3.29) as

$$\begin{aligned}\mathbf{P}_{\mathrm{xx},k^+} &= \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{P}_{\mathrm{xz},k^-}\mathbf{K}_k^\top - \mathbf{K}_k\mathbf{P}_{\mathrm{xz},k^-}^\top + \mathbf{K}_k\mathbf{P}_{\mathrm{zz},k}\mathbf{K}_k^\top \\ &= \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{P}_{\mathrm{xz},k^-}\mathbf{P}_{\mathrm{zz},k}^{-1}\mathbf{P}_{\mathrm{xz},k^-}^\top \\ &= \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{K}_k\mathbf{P}_{\mathrm{zz},k}\mathbf{K}_k^\top \,. \end{aligned} \tag{3.36}$$

After the correction step (3.27), the whole-state is updated with the estimated error-state, being the latter reset

$$\begin{aligned}\hat{\mathbf{x}}_{k^+} &= \hat{\mathbf{x}}_{k^-} + \boldsymbol{\delta}\hat{\mathbf{x}}_{k^+} \\ \boldsymbol{\delta}\hat{\mathbf{x}}_{k^+} &\leftarrow \mathbf{0} \,. \end{aligned} \tag{3.37}$$

3.5 State vector

As in (2.33), the state vector $\mathbf{x}_k$ is split into kinematics states $\mathbf{x}_{\mathsf{kin},k}$, inertial sensor states $\mathbf{x}_{\mathsf{imu},k}$, and GNSS measurement states $\mathbf{x}_{\mathsf{gnss},k}$ as

$$\mathbf{x}_k = \begin{bmatrix}\mathbf{x}_{\mathsf{kin},k} \\ \mathbf{x}_{\mathsf{imu},k} \\ \mathbf{x}_{\mathsf{gnss},k}\end{bmatrix} . \tag{3.38}$$

The kinematics state set $\mathbf{x}_{\mathsf{kin},k}$ is

$$\mathbf{x}_{\mathsf{kin},k} = \begin{bmatrix}\mathbf{v}_k^E \\ \mathbf{r}_k^E \\ \mathbf{q}_{B_k}^{E_k}\end{bmatrix} . \tag{3.39}$$

In general, the error-states $\boldsymbol{\delta}\mathbf{x}$ are additive disturbances of the whole-states $\mathbf{x}$ [as in (3.21)], being added (or fed back) to the latter after each filter update and then zeroed [as in (3.37)]. Attitude is an exception. Similar to the usual multiplicative quaternion error filter designs (e.g., [39]), attitude error-state is here defined as 3-element angle,

$$\begin{bmatrix}\frac{1}{2}\boldsymbol{\delta\theta}_k^B \\ 1\end{bmatrix} \approx \mathbf{q}_{B_k}^{E_k}\left(\hat{\mathbf{q}}_{B_k}^{E_k}\right)^{-1} , \tag{3.40}$$

being fed back and reset as

$$\left\{\hat{\mathbf{q}}^{E_k}_{B_k}\right\}^+ = \begin{bmatrix} \frac{1}{2}\left\{\delta\boldsymbol{\theta}^B_k\right\}^+ \\ 1 \end{bmatrix} \left\{\hat{\mathbf{q}}^{E_k}_{B_k}\right\}^- , \tag{3.41}$$

$$\left\{\delta\boldsymbol{\theta}^B_k\right\}^+ \leftarrow \mathbf{0} . \tag{3.42}$$

The state sets $\mathbf{x}_{\mathsf{imu},k}$ and $\mathbf{x}_{\mathsf{gnss},k}$ are defined in the points to follow.

3.6 Inertial sensor calibration model

The assumed IMU unit has a triad of single-axis gyros and another of accelerometers, both oriented along the elemental directions of B frame, as shown in Figure 3.5.

This set-up yields a special, simpler case of the general n-axes model for estimation derived in Appendix C.3 (and therein discussed).

The inertial sensor calibration model, used to map the contribution of $\mathbf{x}_{\mathsf{imu}}$ to the propagation of $\mathbf{x}_{\mathsf{kin}}$, is obtained by approximate inverse of the measurement corruption model (C.59)–(C.60), giving:

$$\begin{aligned}\Delta\hat{\boldsymbol{\theta}}^B_j \approx & \left(\mathbf{I} - \mathsf{diag}\left(\hat{\mathbf{s}}^B_{\mathsf{g},j}\right) - \left[\hat{\boldsymbol{\vartheta}}^B_{\mathsf{g},j}\times\right] - \boldsymbol{S}\left(\hat{\boldsymbol{\eta}}^B_{\mathsf{g},j}\right)\right)\left(\Delta\tilde{\boldsymbol{\theta}}^B_j - \Delta t_j \hat{\mathbf{b}}^B_{\mathsf{g,on},j} - \Delta t_j \hat{\mathbf{b}}^B_{\mathsf{g,thrm},j}\right. \\ & \left. - \mathsf{diag}\left(\hat{\mathbf{m}}^B_{\mathsf{g},j}\right)\Delta\tilde{\mathbf{v}}^B_j\right)\end{aligned} \tag{3.43}$$

$$\Delta\hat{\mathbf{v}}^B_j \approx \left(\mathbf{I} - \mathsf{diag}\left(\hat{\mathbf{s}}^B_{\mathsf{a},j}\right) - \left[\hat{\boldsymbol{\vartheta}}^B_{\mathsf{a},j}\times\right] - \boldsymbol{S}\left(\hat{\boldsymbol{\eta}}^B_{\mathsf{a},j}\right)\right)\left(\Delta\tilde{\mathbf{v}}^B_j - \Delta t_j \hat{\mathbf{b}}^B_{\mathsf{a,on},j} - \Delta t_j \hat{\mathbf{b}}^B_{\mathsf{a,thrm},j}\right) , \tag{3.44}$$

where $\Delta\hat{\boldsymbol{\theta}}^B_j$ and $\Delta\hat{\mathbf{v}}^B_j$ are calibrated measurements, and $\hat{\bullet}$ and $\tilde{\bullet}$ denote estimated and measured value, respectively. The calibration parameters are all 3-component quantities expressed in Body axes and form the IMU filter state portion

$$\mathbf{x}_{\mathsf{imu},j} = \begin{bmatrix} \mathbf{b}^B_{\mathsf{g,on},j} \\ \mathbf{b}^B_{\mathsf{g,thrm},j} \\ \mathbf{s}^B_{\mathsf{g},j} \\ \boldsymbol{\vartheta}^B_{\mathsf{g},j} \\ \boldsymbol{\eta}^B_{\mathsf{g},j} \\ \mathbf{m}^B_{\mathsf{g},j} \\ \mathbf{b}^B_{\mathsf{a,on},j} \\ \mathbf{b}^B_{\mathsf{a,thrm},j} \\ \mathbf{s}^B_{\mathsf{a},j} \\ \boldsymbol{\vartheta}^B_{\mathsf{a},j} \\ \boldsymbol{\eta}^B_{\mathsf{a},j} \end{bmatrix} . \tag{3.45}$$

The symbols in (3.43)–(3.44) have the same meaning as in (C.59)–(C.60) (and in Table C.1), with a small difference: gyro and accelerometer biases, $\mathbf{b}^B_{\mathsf{g}}$ and $\mathbf{b}^B_{\mathsf{a}}$, respec-

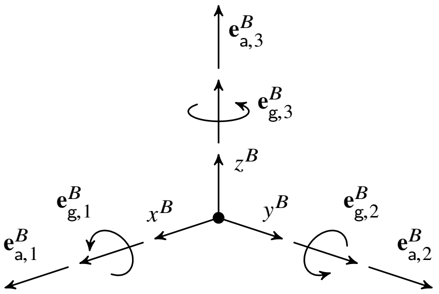

$\mathbf{e}^B_{\mathrm{a},3}$
$\mathbf{e}^B_{\mathrm{g},3}$
z^B
$\mathbf{e}^B_{\mathrm{g},1}$
x^B
y^B
$\mathbf{e}^B_{\mathrm{g},2}$
$\mathbf{e}^B_{\mathrm{a},1}$
$\mathbf{e}^B_{\mathrm{a},2}$

Figure 3.5: IMU axes orientation

tively, are split into repeatability components, $\mathbf{b}^B_{\mathrm{g,on}}$ and $\mathbf{b}^B_{\mathrm{a,on}}$, and thermal induced components, $\mathbf{b}^B_{\mathrm{g,thrm}}$ and $\mathbf{b}^B_{\mathrm{a,thrm}}$.

Table 3.1 describes the process models of each filter state in (3.45). The bias quantities denoted on described above capture the initial on/off (repeatability) component and any stochastic non-thermally induced drift (i.e., that observed in an isothermic Allan variance analysis). The thermal bias states, denoted thrm, are modeled as 1st-order GM processes [see (C.15)] with a small initial uncertainty and a time constant of 10 minutes, to simulate the fast changing thermal conditions of rocket flight. Vandersteen et al. [187] observed significant propagation dispersions resulting from thermal residuals in similar sensor profiles (see Table 2.1). Similarly high impact was also observed from the G-sensitive gyro bias $\mathbf{m}^B_{\mathrm{g},j}$ of the low-grade gyro. Conversely, G-dependencies in the gyro scale-factor yielded negligible errors in all grades, and are thus excluded from the current filter design.

Table 3.1: IMU calibration state process implementation

State	Symbol	Size	Process
Gyro bias repeatability	$\mathbf{b}^B_{\mathrm{g,on}}$	3	random constant (bias repeatability) plus random walk (bias drift)
Gyro thermal bias	$\mathbf{b}^B_{\mathrm{g,thrm}}$	3	1st-order GM
Gyro scale-factor	$\mathbf{s}^B_{\mathrm{g}}$	3	random constant (scale-factor repeatability) plus random walk (scale-factor drift)
Gyro misalignment	$\boldsymbol{\vartheta}^B_{\mathrm{g}}$	3	random constant
Gyro non-orthogonality	$\boldsymbol{\eta}^B_{\mathrm{g}}$	3	random constant
Gyro G-sensitive bias	$\mathbf{m}^B_{\mathrm{g}}$	3	random constant
Accel. bias repeatability	$\mathbf{b}^B_{\mathrm{a,on}}$	3	random constant (bias repeatability) plus random walk (bias drift)
Accel. thermal bias	$\mathbf{b}^B_{\mathrm{a,thrm}}$	3	1st-order GM
Accel. scale-factor	$\mathbf{s}^B_{\mathrm{a}}$	3	random constant (scale-factor repeatability) plus random walk (scale-factor drift)
Accel. misalignment	$\boldsymbol{\vartheta}^B_{\mathrm{a}}$	3	random constant
Accel. non-orthogonality	$\boldsymbol{\eta}^B_{\mathrm{a}}$	3	random constant

3.7 Filter propagation model

The kinematics whole-states are propagated at high-rate (index j) by the strapdown algorithm described in Section 3.3. Linearizing the error-state system (3.21) as

$$\begin{aligned}\boldsymbol{\delta}\mathbf{x}_{j+1} &= \boldsymbol{\phi}_{j+1|j}(\hat{\mathbf{x}}_j + \boldsymbol{\delta}\mathbf{x}_j) - \boldsymbol{\phi}_{j+1|j}(\hat{\mathbf{x}}_j) + \mathbf{w}_j && (3.46)\\ &\approx \boldsymbol{\Phi}_{j+1|j}\boldsymbol{\delta}\mathbf{x}_j + \mathbf{w}_j \ , && (3.47)\end{aligned}$$

and assuming that the accumulated system noise over one low-rate step (index k) is lower than the total state uncertainty, the low-rate transition and process noise covariance matrices can be approximated as

$$\boldsymbol{\Phi}_{k+1|k} \approx \prod_{j=0}^{N-1} \boldsymbol{\Phi}_{j+1|j} \tag{3.48}$$

$$\mathbf{Q}_{k+1|k} \approx \sum_{j=0}^{N-1} \mathbf{Q}_{j+1|j} \ , \tag{3.49}$$

where $t_k = t_{j=0}$ and $t_{k+1} = t_{j=N}$.

Following the system structure in (2.33), which yielded the state partitioning described in Section 3.5, the state transition matrix $\boldsymbol{\Phi}_{j+1|j}$ can be broken into

$$\boldsymbol{\Phi}_{j+1|j} = \begin{bmatrix} \boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin}} & \boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{kin}} & \mathbf{O} \\ \mathbf{O} & \boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{imu}} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \boldsymbol{\Phi}^{\mathsf{gnss}}_{\mathsf{gnss}} \end{bmatrix}_{j+1|j} \ , \tag{3.50}$$

where the kinematics rows of the transition matrix $\boldsymbol{\Phi}_{j+1|j}$ can be obtained through the derivation of the linearized kinematics error-state system (3.47). The sequential product (3.48) can be split into submatrix computations, given in a recursive form as

$$\begin{aligned}\boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin},j+1|0} &= \boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin},j+1|j}\boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin},j|0} && (3.51)\\ \boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{kin},j+1|0} &= \boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin},j+1|j}\boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{kin},j|0} + \boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{kin},j+1|j}\boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{imu},j|0} && (3.52)\\ \boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{imu},j+1|0} &= \boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{imu},j+1|j}\boldsymbol{\Phi}^{\mathsf{imu}}_{\mathsf{imu},j|0} && (3.53)\\ \boldsymbol{\Phi}^{\mathsf{gnss}}_{\mathsf{gnss},j+1|0} &= \boldsymbol{\Phi}^{\mathsf{gnss}}_{\mathsf{gnss},j+1|j}\boldsymbol{\Phi}^{\mathsf{gnss}}_{\mathsf{gnss},j|0} \ . && (3.54)\end{aligned}$$

As described in Section 3.5, the kinematics error-states include velocity $\boldsymbol{\delta}\mathbf{v}^E$, position $\boldsymbol{\delta}\mathbf{r}^E$ and attitude $\boldsymbol{\delta\theta}^B$. The attitude error propagation law can be derived by differentiating the attitude error Direction Cosine Matrix (DCM), making use of the definition of $\boldsymbol{\delta\theta}^B$ from (3.40), as

$$\begin{aligned}\frac{\mathsf{d}}{\mathsf{d}t}\mathbf{C}(\boldsymbol{\delta\theta}^B_j) &= \frac{\mathsf{d}}{\mathsf{d}t}\left(\hat{\mathbf{C}}^{B_j}_{E_j}\mathbf{C}^{E_j}_{B_j}\right) = \frac{\mathsf{d}}{\mathsf{d}t}\left(\hat{\mathbf{C}}^{B_j}_{I}\hat{\mathbf{C}}^{I}_{E_j}\mathbf{C}^{E_j}_{I}\mathbf{C}^{I}_{B_j}\right) = \frac{\mathsf{d}}{\mathsf{d}t}\left(\hat{\mathbf{C}}^{B_j}_{I}\mathbf{C}^{I}_{B_j}\right) && (3.55)\\ &= -\hat{\boldsymbol{\Omega}}^B_{IB,j}\hat{\mathbf{C}}^{B_j}_{I}\mathbf{C}^{I}_{B_j} + \hat{\mathbf{C}}^{B_j}_{I}\mathbf{C}^{I}_{B_j}\left(\hat{\boldsymbol{\Omega}}^B_{IB,j} + \boldsymbol{\delta\Omega}^B_{IB,j}\right) \\ &= \mathbf{C}(\boldsymbol{\delta\theta}^B_j)\left(\hat{\boldsymbol{\Omega}}^B_{IB,j} - \mathbf{C}(\boldsymbol{\delta\theta}^B_j)^{\mathsf{T}}\hat{\boldsymbol{\Omega}}^B_{IB,j}\mathbf{C}(\boldsymbol{\delta\theta}^B_j) + \boldsymbol{\delta\Omega}^B_{IB,j}\right) \ , && (3.56)\end{aligned}$$

where, in (3.55), $\hat{\mathbf{C}}_{E_j}^{I}\mathbf{C}_{I}^{E_j} \approx \mathbf{I}$. Using this result to propagate the original DCM gives

$$\begin{aligned}\mathbf{C}(\boldsymbol{\delta\theta}_{j+1}^{B}) &\approx \mathbf{C}(\boldsymbol{\delta\theta}_{j}^{B}) + \Delta t_j \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{C}(\boldsymbol{\delta\theta}_{j}^{B}) \\ &\approx \mathbf{C}(\boldsymbol{\delta\theta}_{j}^{B}) + \Delta t_j \mathbf{C}(\boldsymbol{\delta\theta}_{j}^{B})\hat{\boldsymbol{\Omega}}_{IB,j}^{B} - \Delta t_j \hat{\boldsymbol{\Omega}}_{IB,j}^{B}\mathbf{C}(\boldsymbol{\delta\theta}_{j}^{B}) + \Delta t_j \mathbf{C}(\boldsymbol{\delta\theta}_{j}^{B})\boldsymbol{\delta\Omega}_{IB,j}^{B} \ ,\end{aligned} \tag{3.57}$$

which, using the small angle approximation, $\mathbf{C}(\boldsymbol{\delta\theta}^{B}) \approx \mathbf{I} - [\boldsymbol{\delta\theta}^{B}\times]$, on both sides yields

$$\begin{aligned} \mathbf{I} - \left[\boldsymbol{\delta\theta}_{j+1}^{B}\times\right] &\approx \mathbf{I} - \left[\boldsymbol{\theta}_{j}^{B}\times\right] - \Delta t_j \left[\boldsymbol{\theta}_{j}^{B}\times\right]\hat{\boldsymbol{\Omega}}_{IB,j}^{B} + \Delta t_j \hat{\boldsymbol{\Omega}}_{IB,j}^{B}\left[\boldsymbol{\delta\theta}_{j}^{B}\times\right] \\ &\qquad + \Delta t_j \left(\mathbf{I} - \left[\boldsymbol{\delta\theta}_{j}^{B}\times\right]\right)\boldsymbol{\delta\Omega}_{IB,j}^{B} \\ \Rightarrow \qquad \left[\boldsymbol{\delta\theta}_{j+1}^{B}\times\right] &\approx \left[\boldsymbol{\delta\theta}_{j}^{B}\times\right] - \Delta t_j \left[\hat{\omega}_{IB,j}^{B} \times \boldsymbol{\delta\theta}_{j}^{B}\times\right] - \Delta t_j \boldsymbol{\delta\Omega}_{IB,j}^{B} \\ &\approx \left[\left(\mathbf{I} - \Delta t_j \hat{\boldsymbol{\Omega}}_{IB,j}^{B}\right)\boldsymbol{\delta\theta}_{j}^{B}\times\right] - \Delta t_j \boldsymbol{\delta\Omega}_{IB,j}^{B} \ . \end{aligned} \tag{3.58}$$

To arrive at (3.58), the triple cross-product identity, $\left[\mathbf{u}\times\mathbf{v}\times\right] = \left[\mathbf{u}\times\right]\left[\mathbf{v}\times\right] - \left[\mathbf{v}\times\right]\left[\mathbf{u}\times\right]$, was used and the second order error terms were neglected. Undoing the skew symmetric and approximating $\Delta t_j \boldsymbol{\delta\Omega}_{IB}^{B} \approx \boldsymbol{\delta\Delta\theta}^{B}$, the relation (3.58) finally gives

$$\boldsymbol{\delta\theta}_{j+1}^{B} \approx \hat{\mathbf{C}}_{B_j}^{B_{j+1}}\boldsymbol{\delta\theta}_{j}^{B} - \boldsymbol{\delta\Delta\theta}_{j}^{B} + \mathbf{w}_{\theta,j} \ . \tag{3.59}$$

where the noise term $\mathbf{w}_{\theta}$ is introduced to account for all the approximations performed. The attitude transition error term $\delta\boldsymbol{\Delta\theta}^{B}$ accounts for the errors in the estimation of the gyroscope uncertainties (those described in Section 3.6) and is defined as

$$\delta\boldsymbol{\Delta\theta}_{j}^{B} = \boldsymbol{\Delta\theta}_{j}^{B} - \boldsymbol{\Delta}\hat{\boldsymbol{\theta}}_{j}^{B} \ . \tag{3.60}$$

The translational error-states (velocity and position) transition model can be derived directly from the strapdown propagation law (3.9) as

$$\begin{aligned}\begin{bmatrix}\boldsymbol{\delta}\mathbf{v}_{j+1}^{E} \\ \boldsymbol{\delta}\mathbf{r}_{j+1}^{E}\end{bmatrix} &\approx \begin{bmatrix}\mathbf{I} - \Delta t_j \boldsymbol{\Omega}_{IE}^{E} & -\Delta t_j \boldsymbol{\Omega}_{IE}^{E\ 2} \\ \Delta t_j \mathbf{I} & \mathbf{I} + \Delta t_j \boldsymbol{\Omega}_{IE}^{E}\end{bmatrix}\begin{bmatrix}\mathbf{C}_{E_j}^{E_{j+1}}\boldsymbol{\delta}\mathbf{v}_{j}^{E} \\ \mathbf{C}_{E_j}^{E_{j+1}}\boldsymbol{\delta}\mathbf{r}_{j}^{E}\end{bmatrix} \\ &\quad + \begin{bmatrix}\mathbf{I} - \frac{\Delta t_j}{2}\boldsymbol{\Omega}_{IE}^{E} \\ \frac{\Delta t_j}{2}\mathbf{I}\end{bmatrix}\mathbf{C}_{I}^{E_{j+1}}\left(\boldsymbol{\delta\Delta}\mathbf{v}_{\mathsf{sf},j}^{I} + \boldsymbol{\delta\Delta}\mathbf{v}_{\mathsf{g},j}^{I}\right) + \begin{bmatrix}\mathbf{w}_{\mathrm{v},j} \\ \mathbf{w}_{\mathrm{r},j}\end{bmatrix} \ . \end{aligned} \tag{3.61}$$

However, the Delta-V error terms $\boldsymbol{\delta\Delta}\mathbf{v}_{\mathsf{sf}}^{I}$ and $\boldsymbol{\delta\Delta}\mathbf{v}_{\mathsf{g}}^{I}$ need to be expanded to expose further error-state contributions. The term $\boldsymbol{\delta\Delta}\mathbf{v}_{\mathsf{sf}}^{I}$ can be obtained from (3.10) as

$$\begin{aligned}\boldsymbol{\delta\Delta}\mathbf{v}_{\mathsf{sf},j}^{I} &\approx \mathbf{C}_{E_j}^{I}\mathbf{C}_{B_j}^{E_j}\mathbf{C}_{B_{j+\frac{1}{2}}}^{B_j}\boldsymbol{\Delta}\mathbf{v}_{j}^{B} - \mathbf{C}_{E_j}^{E_{j+1}}\hat{\mathbf{C}}_{B_j}^{E_j}\hat{\mathbf{C}}_{B_{j+\frac{1}{2}}}^{B_j}\boldsymbol{\Delta}\mathbf{v}_{j}^{B} \\ &\approx \mathbf{C}_{E_j}^{I}\hat{\mathbf{C}}_{B_j}^{E_j}\left(\mathbf{I} - \left[\boldsymbol{\delta\theta}_{j}^{B}\times\right]\right)\left(\hat{\mathbf{C}}_{B_{j+\frac{1}{2}}}^{B_j} + \frac{1}{2}\left[\delta\boldsymbol{\Delta\theta}_{j}^{B}\times\right]\right)\left(\boldsymbol{\Delta}\hat{\mathbf{v}}_{j}^{B} + \boldsymbol{\delta\Delta}\mathbf{v}_{j}^{B}\right) \\ &\qquad - \mathbf{C}_{E_j}^{I}\hat{\mathbf{C}}_{B_j}^{E_j}\hat{\mathbf{C}}_{B_{j+\frac{1}{2}}}^{B_j}\boldsymbol{\Delta}\hat{\mathbf{v}}_{j}^{B} \\ &\approx \mathbf{C}_{E_j}^{I}\hat{\mathbf{C}}_{B_j}^{E_j}\left(\left[\hat{\mathbf{C}}_{B_{j+\frac{1}{2}}}^{B_j}\boldsymbol{\Delta}\hat{\mathbf{v}}_{j}^{B}\times\right]\boldsymbol{\delta\theta}_{j}^{B} + \hat{\mathbf{C}}_{B_{j+\frac{1}{2}}}^{B_j}\boldsymbol{\delta\Delta}\hat{\mathbf{v}}_{j}^{B} - \frac{1}{2}\left[\boldsymbol{\Delta}\hat{\mathbf{v}}_{j}^{B}\times\right]\boldsymbol{\delta\Delta\theta}_{j}^{B}\right) \ , \end{aligned} \tag{3.62}$$

where second order error terms have been neglected.

The gravitational Delta-V error term in (3.61) can be derived from (3.17) as

$$\begin{aligned}\delta\Delta\mathbf{v}^I_{\mathrm{g},j} &\approx \Delta t_j\ \mathbf{C}^I_{E_{j+\frac{1}{2}}}\left(\mathbf{g}^E_{j+\frac{1}{2}} - \hat{\mathbf{g}}^E_{j+\frac{1}{2}}\right)\\ &\approx \Delta t_j\ \mathbf{C}^I_{E_{j+\frac{1}{2}}}\boldsymbol{G}\left(\hat{\mathbf{r}}^E_{j+\frac{1}{2}}\right)\mathbf{C}^{E_{j+\frac{1}{2}}}_{E_j}\left(\delta\mathbf{r}^E_j + \frac{\Delta t_j}{2}\delta\mathbf{v}^E_j\right) ,\end{aligned} \tag{3.63}$$

where the gravity gradient $\boldsymbol{G}(\mathbf{r})$ is given in [51] as

$$\boldsymbol{G}(\mathbf{r}) = \frac{\partial \boldsymbol{g}(\mathbf{r})}{\partial \mathbf{r}} \approx -\frac{\mu_\oplus}{\|\mathbf{r}\|^3}\left(\mathbf{I} - 3\frac{\mathbf{r}\mathbf{r}^\mathsf{T}}{\|\mathbf{r}\|^2}\right) , \tag{3.64}$$

with $\mu_\oplus$ as the Earth's gravitational parameter.

From (3.59), (3.61), (3.62) and (3.63), the kinematics transition matrix $\boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin}}$ can be easily filled in. It is structured as

$$\boldsymbol{\Phi}^{\mathsf{kin}}_{\mathsf{kin}} = \begin{bmatrix}\boldsymbol{\Phi}_{\mathrm{vv}} & \boldsymbol{\Phi}_{\mathrm{vr}} & \boldsymbol{\Phi}_{\mathrm{v}\theta}\\ \boldsymbol{\Phi}_{\mathrm{rv}} & \boldsymbol{\Phi}_{\mathrm{rr}} & \boldsymbol{\Phi}_{\mathrm{r}\theta}\\ \mathbf{O} & \mathbf{O} & \boldsymbol{\Phi}_{\theta\theta}\end{bmatrix} , \tag{3.65}$$

where

$$\boldsymbol{\Phi}_{\mathrm{vv}} = \boldsymbol{\Psi}^E_{IE}\left(2\right) + \frac{\Delta t_j^2}{2}\boldsymbol{\Psi}^E_{IE}\left(1\right)\boldsymbol{G}\left(\hat{\mathbf{r}}^E_{j+\frac{1}{2}}\right)\boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right) \tag{3.66}$$

$$\boldsymbol{\Phi}_{\mathrm{vr}} = -\Delta t_j\boldsymbol{\Omega}^E_{IE}{}^2\boldsymbol{\Psi}^E_{IE}\left(1\right) + \Delta t_j\boldsymbol{\Psi}^E_{IE}\left(1\right)\boldsymbol{G}\left(\hat{\mathbf{r}}^E_{j+\frac{1}{2}}\right)\boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right) \tag{3.67}$$

$$\boldsymbol{\Phi}_{\mathrm{v}\theta} = \boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right)\hat{\mathbf{C}}^{E_{j+1}}_{B_j}\left[\hat{\mathbf{C}}^{B_j}_{B_{j+\frac{1}{2}}}\Delta\hat{\mathbf{v}}^B_j\times\right] \tag{3.68}$$

$$\boldsymbol{\Phi}_{\mathrm{rr}} = \mathbf{I} + \frac{\Delta t_j^2}{2}\boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right)\boldsymbol{G}\left(\hat{\mathbf{r}}^E_{j+\frac{1}{2}}\right)\boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right) \tag{3.69}$$

$$\boldsymbol{\Phi}_{\mathrm{rv}} = \Delta t_j\boldsymbol{\Psi}^E_{IE}\left(1\right) + \frac{\Delta t_j^3}{4}\boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right)\boldsymbol{G}\left(\hat{\mathbf{r}}^E_{j+\frac{1}{2}}\right)\boldsymbol{\Psi}^E_{IE}\left(\frac{1}{2}\right) \tag{3.70}$$

$$\boldsymbol{\Phi}_{\mathrm{r}\theta} = \frac{\Delta t_j}{2}\hat{\mathbf{C}}^{E_{j+1}}_{B_j}\left[\hat{\mathbf{C}}^{B_j}_{B_{j+\frac{1}{2}}}\Delta\hat{\mathbf{v}}^B_j\times\right] \tag{3.71}$$

$$\boldsymbol{\Phi}_{\theta\theta} = \hat{\mathbf{C}}^{B_{j+1}}_{B_j} . \tag{3.72}$$

The gravitational gradient $\boldsymbol{G}(\mathbf{r})$ is given by (3.64). As in [140], the transition matrices between consecutive instances of the E frame (e.g., $\mathbf{C}^{E_{j+1}}_{E_j}$ and $\mathbf{C}^{E_{j+1/2}}_{E_j}$) are approximated by the small angle rotation matrix function $\boldsymbol{\Psi}^E_{IE}\left(\bullet\right)$ which is given by

$$\boldsymbol{\Psi}^E_{IE}\left(\alpha\right) = \left(\mathbf{I} - \alpha\Delta t_j\boldsymbol{\Omega}^E_{IE}\right) , \tag{3.73}$$

where $\boldsymbol{\Omega}^E_{IE}$ is the skew-symmetric matrix of the Earth's rotational velocity $\boldsymbol{\omega}^E_{IE}$. The argument α is the fraction of the elapsed sampling time Δ_j to which the transition corresponds, e.g.,

$$\mathbf{C}_{E_j}^{E_{j+\frac{1}{2}}} \approx \boldsymbol{\Psi}_{IE}^{E}\left(\frac{1}{2}\right)\,, \quad \text{or} \quad \mathbf{C}_{E_j}^{E_{j+1}} \approx \boldsymbol{\Psi}_{IE}^{E}\left(1\right)\,. \tag{3.74}$$

The matrix $\boldsymbol{\Phi}_{\mathsf{kin}}^{\mathsf{imu}}$ in (3.50) is split into

$$\boldsymbol{\Phi}_{\mathsf{kin}}^{\mathsf{imu}} = \begin{bmatrix} \boldsymbol{\Phi}_{\mathsf{kin}}^{\Delta\theta} & \boldsymbol{\Phi}_{\mathsf{kin}}^{\Delta\mathrm{v}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_{\Delta\theta}^{\mathsf{imu}} \\ \boldsymbol{\Phi}_{\Delta\mathrm{v}}^{\mathsf{imu}} \end{bmatrix}\,, \tag{3.75}$$

where the sensitivities of the kinematics states $\mathbf{x}_{\mathsf{kin}}$ to the inertial sensor inputs $(\boldsymbol{\Delta\theta}, \boldsymbol{\Delta}\mathbf{v})$ is

$$\begin{aligned} \boldsymbol{\Phi}_{\mathrm{v}\Delta\mathrm{v}} &= \boldsymbol{\Psi}_{IE}^{E}\left(\frac{1}{2}\right)\hat{\mathbf{C}}_{B_j}^{E_{j+1}} \\ \boldsymbol{\Phi}_{\mathrm{r}\Delta\mathrm{v}} &= \frac{\Delta t_j}{2}\hat{\mathbf{C}}_{B_j}^{E_{j+1}} \\ \boldsymbol{\Phi}_{\theta\Delta\theta} &= -\mathbf{I}\,, \end{aligned} \tag{3.76}$$

and the mappings from the sensor calibration states in $\mathbf{x}_{\mathsf{imu}}$ to $\boldsymbol{\Delta\theta}$ and $\boldsymbol{\Delta}\mathbf{v}$ are derived from (3.43) and (3.44)

$$\boldsymbol{\Phi}_{\Delta\theta\mathrm{b}_{\mathrm{g},\mathsf{on}}} = \boldsymbol{\Phi}_{\Delta\theta\mathrm{b}_{\mathrm{g},\mathsf{thrm}}} = -\Delta t_j\mathbf{I} \tag{3.77}$$

$$\boldsymbol{\Phi}_{\Delta\theta\mathrm{s}_{\mathrm{g}}} = -\,\mathsf{diag}(\boldsymbol{\Delta}\tilde{\boldsymbol{\theta}}_j^B) \tag{3.78}$$

$$\boldsymbol{\Phi}_{\Delta\theta\vartheta_{\mathrm{g}}} = [\boldsymbol{\Delta}\tilde{\boldsymbol{\theta}}_j^B\times] \tag{3.79}$$

$$\boldsymbol{\Phi}_{\Delta\theta\eta_{\mathrm{g}}} = -\boldsymbol{R}(\boldsymbol{\Delta}\tilde{\boldsymbol{\theta}}_j^B) \tag{3.80}$$

$$\boldsymbol{\Phi}_{\Delta\theta\mathrm{m}_{\mathrm{g}}} = -\,\mathsf{diag}(\boldsymbol{\Delta}\tilde{\mathbf{v}}_j^B) \tag{3.81}$$

$$\boldsymbol{\Phi}_{\Delta\mathrm{vb}_{\mathrm{a},\mathsf{on}}} = \boldsymbol{\Phi}_{\Delta\mathrm{vb}_{\mathrm{a},\mathsf{thrm}}} = -\Delta t_j\mathbf{I} \tag{3.82}$$

$$\boldsymbol{\Phi}_{\Delta\mathrm{vs}_{\mathrm{a}}} = -\,\mathsf{diag}(\boldsymbol{\Delta}\tilde{\mathbf{v}}_j^B) \tag{3.83}$$

$$\boldsymbol{\Phi}_{\Delta\mathrm{v}\vartheta_{\mathrm{a}}} = [\boldsymbol{\Delta}\tilde{\mathbf{v}}_j^B\times] \tag{3.84}$$

$$\boldsymbol{\Phi}_{\Delta\mathrm{v}\eta_{\mathrm{a}}} = -\boldsymbol{R}(\boldsymbol{\Delta}\tilde{\mathbf{v}}_j^B)\,. \tag{3.85}$$

Finally, the inertial sensor states transition submatrix $\boldsymbol{\Phi}_{\mathsf{imu}}^{\mathsf{imu}}$ [in (3.50)] is diagonal with elements filled according to the stochastic models described in Section 3.6, while that of the measurement state-set $\boldsymbol{\Phi}_{\mathsf{gnss}}^{\mathsf{gnss}}$ is built in the text to follow.

3.8 Static measurement update models

While on the launch pad, the vehicle is (for the most part) in a static condition. As described in Section 2.4.4, this insight can be provided to the navigation filter as a pseudo-measurement (or constraint), which not only stabilizes/improves the estimation of the kinematic states but also renders observable several IMU error states. Pre-flight fine alignment is of critical for the in-flight accuracy of the navigation [23].

3.8.1 Zero-velocity update

A static condition on the launch pad means that the vehicle has zero velocity in ECEF. This information can be conveyed as a null velocity pseudo-measurement ($\tilde{\mathbf{v}}_k^E = \mathbf{0}$). The update model can be written as

$$\begin{aligned} \mathbf{y}_{\mathrm{ZV},k} &= \boldsymbol{h}_k(\mathbf{x}_k) + \boldsymbol{\nu}_k && (3.86) \\ &= \mathbf{v}_k^E + \boldsymbol{\nu}_k \ , && (3.87) \end{aligned}$$

which yields the error-measurement model

$$\begin{aligned} \delta\mathbf{y}_{\mathrm{ZV},k} &= \boldsymbol{h}_k(\hat{\mathbf{x}}_k + \delta\mathbf{x}_k) - \boldsymbol{h}_k(\hat{\mathbf{x}}_k) + \boldsymbol{\nu}_k && (3.88) \\ &= \delta\mathbf{v}_k^E + \boldsymbol{\nu}_k \ . && (3.89) \end{aligned}$$

The covariance $\mathbf{R}_k$ of the Gaussian noise $\boldsymbol{\nu}_k$ shall be close to null, accounting only for possible vibrations that might induce *real* velocities.

The only non-null submatrix of the measurement Jacobian is with respect to velocity

$$\mathbf{H}_{\mathrm{ZV},k}^{\mathrm{v}} = \frac{\partial\delta\mathbf{y}_k}{\partial\delta\mathbf{v}_k^E} = \mathbf{I} \ . \qquad (3.90)$$

3.8.2 Earth rate update

While standing on the launch pad the vehicle describes an inertial rotation equal to that of the Earth, i.e., $\boldsymbol{\omega}_{IB}^E = \boldsymbol{\omega}_{IE}^E$. This means that in an sample-time Δt the vehicle experiences a rotation with angular amplitude $\Delta t \boldsymbol{\omega}_{IE}^E$. Depending on the gyroscope class, this angular change can be sensed and compared to the theoretical value in a pseudo-measurement to the filter. The absolute angular displacement is

$$\mathbf{y}_{\mathrm{ER},k} = \Delta t_k \boldsymbol{\omega}_{IE}^E \ , \qquad (3.91)$$

which can be modeled as a function of the filter states as

$$\begin{aligned} \mathbf{y}_{\mathrm{ER},k} &= \boldsymbol{h}_k(\mathbf{x}_k) + \boldsymbol{\nu}_k \\ &= \mathbf{C}_{B_k}^{E_k} \mathbf{C}_{B_{k-\frac{1}{2}}}^{B_k} \Delta\boldsymbol{\theta}_{k-1}^B + \boldsymbol{\nu}_k \ , && (3.92) \end{aligned}$$

where $\Delta\boldsymbol{\theta}_{k-1}^B$ corresponds to the rotation of the Body from t_{k-1} to t_k. As done before for the Body-rotation compensation of the gyroscope measurement, this Delta-angle is assumed to be expressed in the Body frame at middle point of the time-interval, i.e., $t_{k-\frac{1}{2}}$. The Gaussian noise $\boldsymbol{\nu}_k$ has covariance matrix $\mathbf{R}_k$ which shall be small but not null since, as it will be seen, the model includes several approximations.

The error-measurement model is obtained from (3.92) as

$$\begin{aligned}
\delta\mathbf{y}_{\mathrm{ER},k} &= \boldsymbol{h}_k(\mathbf{x}_k+\delta\mathbf{x}_k)-\boldsymbol{h}_k(\hat{\mathbf{x}}_k)+\boldsymbol{\nu}_k \\
&\approx \hat{\mathbf{C}}^{E_k}_{B_k}\left(\mathbf{I}-\left[\delta\boldsymbol{\theta}^B_k\times\right]\right)\left(\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}-\frac{1}{2}\left[\delta\boldsymbol{\Delta\theta}^B_k\times\right]\right)\left(\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}+\delta\boldsymbol{\Delta\theta}^B_{k-1}\right) \\
&\qquad -\hat{\mathbf{C}}^{E_k}_{B_k}\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}+\boldsymbol{\nu}_k \\
&\approx \hat{\mathbf{C}}^{E_k}_{B_k}\left[\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}\times\right]\delta\boldsymbol{\theta}^B_k+\hat{\mathbf{C}}^{E_k}_{B_k}\underbrace{\left(\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}+\frac{1}{2}\left[\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}\times\right]\right)}_{\approx\mathbf{I}}\delta\boldsymbol{\Delta\theta}^B_{k-1}+\boldsymbol{\nu}_k \\
&\approx \hat{\mathbf{C}}^{E_k}_{B_k}\left[\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}\times\right]\delta\boldsymbol{\theta}^B_k+\hat{\mathbf{C}}^{E_k}_{B_k}\delta\boldsymbol{\Delta\theta}^B_{k-1}+\boldsymbol{\nu}_k\ , \qquad (3.93)
\end{aligned}$$

where the second order error-state terms were neglected.

The attitude transition error angle $\delta\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B$ depends on the gyroscope error-states as described by the sensitivity matrix $\boldsymbol{\Phi}^{\mathsf{imu}}_{\Delta\theta}$ in (3.75), giving

$$\delta\boldsymbol{\Delta\theta}^B_{k-1}\approx\boldsymbol{\Phi}^{\mathsf{imu}}_{\Delta\theta,k-1}\delta\mathbf{x}_{\mathsf{imu},k-1}\ . \qquad (3.94)$$

Assuming the change in the IMU error-states is neglectable over one time-step, i.e., $\delta\mathbf{x}_{\mathsf{imu},k}\approx\delta\mathbf{x}_{\mathsf{imu},k-1}$, then the error-measurement model (3.93) becomes

$$\delta\mathbf{y}_{\mathrm{ER},k}\approx\hat{\mathbf{C}}^{E_k}_{B_k}\left[\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}\times\right]\delta\boldsymbol{\theta}^B_k+\hat{\mathbf{C}}^{E_k}_{B_k}\boldsymbol{\Phi}^{\mathsf{imu}}_{\Delta\theta,k}\delta\mathbf{x}_{\mathsf{imu},k}+\boldsymbol{\nu}_k\ , \qquad (3.95)$$

with non-null sensitivity sub matrices

$$\mathbf{H}^{\theta}_{\mathrm{ER},k}=\frac{\partial\delta\mathbf{y}_k}{\partial\delta\boldsymbol{\theta}^B_k}\approx\hat{\mathbf{C}}^{E_k}_{B_k}\left[\hat{\mathbf{C}}^{B_k}_{B_{k-\frac{1}{2}}}\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^B_{k-1}\times\right] \qquad (3.96)$$

$$\mathbf{H}^{\mathrm{x}_{\mathsf{imu}}}_{\mathrm{ER},k}=\frac{\partial\delta\mathbf{y}_k}{\partial\delta\mathbf{x}_{\mathsf{imu},k}}\approx\hat{\mathbf{C}}^{E_k}_{B_k}\boldsymbol{\Phi}^{\mathsf{imu}}_{\Delta\theta,k}\ . \qquad (3.97)$$

3.9 GNSS measurement update models

This section covers the GNSS measurement models used by the navigation system.

3.9.1 GNSS measurement set

As previously mentioned, a DLR Phoenix-HD GPS receiver [104] is used in this study. This COTS-based receiver is routinely flown in sounding rocket missions and, as described in Section 1.1.4, has been tested on board of Ariane-V as part of the OCAM-G initiative [25, 73]. It is capable of outputting GPS navigation solutions as well as raw measurements of pseudorange, pseudorange-rate and integrated carrier phase.

Post-flight analysis of the DLR HNS version flown in SHEFEX-2 [164] revealed somewhat slow dynamics estimation of the pseudorange-updated filter. As shown in

Section 2.5.3, GNSS velocity-based measurements improve state estimation, particularly during engine burns. This advantage is especially important since accuracy is critical during propelled flight phases both for optimal vehicle steering and for safety monitoring. Following the results displayed and discussed in Section 2.5.3, time-differenced carrier phase are selected as complement to the pseudorange measurements as filter updates, providing delta-range information. Recall that the time-differencing cancels out the phase ambiguity terms, avoiding the need for ambiguity fixing by differential operation or ambiguity resolution [64].

Receiver clock errors, discussed in Section 2.5.2, are handled through a reduced-order clock bias model which is described in Section 3.9.2. Tropospheric and ionospheric delays are corrected as shall be described in Sections 3.9.3 and 3.9.4, with tropospheric corrections being accounted for in filter design. Satellite clock error is removed with information from the navigation message, and relativistic effects are corrected using the computed satellite positions and velocities [124, 69]. The multipath error occurs in reflective conditions (e.g., on the launch pad) and is thus neglected.

3.9.2 Receiver clock error model

As shown in Section 2.5.2, the clock frequency error of the DLR Phoenix-HD receiver used in this study behaves nearly as white-noise. This suggests a random-walk description of the clock bias as

$$\tau_{\mathrm{r},k+1} = \tau_{\mathrm{r},k} + \mathrm{w}_{\tau_\mathrm{r},k} \ , \tag{3.98}$$

where the clock frequency noise is $\mathrm{w}_{\tau_\mathrm{r},k} \sim \mathcal{N}(0, \sigma^2_{\tau_\mathrm{r},k})$.

The TDCP measurement model is, on the other hand,

$$\mathbf{y}_k = \boldsymbol{h}_k(\mathbf{x}_k) + c\Delta\tau_{\mathrm{r},k}\mathbf{1} + \boldsymbol{\nu}_k \ , \tag{3.99}$$

where $\mathbf{1}$ is a column of ones, $\Delta\tau_{\mathrm{r},k} \sim \mathcal{N}(0, \sigma^2_{\Delta\tau_\mathrm{r},k})$, and $\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_\nu)$. Since, as described Section 2.5.2, the receiver used employs the same internal clock model in both pseudorange and integrated carrier phase measurements, the TDCP clock noise $\Delta\tau_{\mathrm{r},k}$ is simply the average of the clock drift. This had been expressed in (2.29) and is equivalent to writing

$$\Delta\tau_{\mathrm{r},k} = \mathrm{w}_{\tau_\mathrm{r},k-1} \ . \tag{3.100}$$

A consequence of (3.100) is the (perfect) correlation between TDCP clock noise $\Delta\tau_{\mathrm{r},k}$ and clock bias process noise $\mathrm{w}_{\tau_\mathrm{r},k-1}$ as

$$E\left\{\mathrm{w}_{\tau_\mathrm{r},k-1}\,\Delta\tau_{\mathrm{r},k}\right\} = \sigma^2_{\tau_\mathrm{r},k-1} \ . \tag{3.101}$$

This correlation is accounted for in the filter by filter (see Section 3.4) through matrix $\mathbf{B}$, which shall be specified upon measurement update model definition. Modeling clock drift as white noise forgoes one filter state, easing computational burden.

3.9.3 Tropospheric delay correction

As shown in Section 2.5.2, atmospheric effects perturb the GNSS pseudorange and TDCP measurements. Troposphere causes the most severe effects. In the present system these perturbations are mitigated through a feed-forward correction using the model (2.30). As discussed in Section 2.5.2, the leftover Tropospheric correction residuals are still relatively large and need to be accounted for in the update model.

Robustness against these residuals can be achieved by considering a fictitious scale-factor uncertainty affecting the correction model. In this way the true tropospheric delay and delta-delay in channel i is written in terms of the modeled quantities as

$$\begin{aligned} \tau_{\mathrm{T},i,k} &= (1 + s_{\mathrm{T},i})\ h_{\tau_{\mathrm{T}},i,k}(\mathbf{x}_k) && (3.102)\\ \Delta\tau_{\mathrm{T},i,k} &= (1 + s_{\mathrm{T},i})\Delta h_{\tau_{\mathrm{T}},i,k}(\mathbf{x}_k) = (1 + s_{\mathrm{T}})\left(h_{\tau_{\mathrm{T}},i,k}(\mathbf{x}_k) - h_{\tau_{\mathrm{T}},i,k-1}(\mathbf{x}_k)\right)\ , && (3.103) \end{aligned}$$

where $s_{\mathrm{T},i} \sim \mathcal{N}(0, \sigma^2_{s_{\mathrm{T}},i})$ is a constant unknown scale-factor. $h_{\tau_{\mathrm{T}},i,k}(\mathbf{x}_k)$ is the tropospheric delay correction model (2.30). As it shall be described in Section 3.9.6, a delayed GNSS position state within the filter allows for $\tau_{\mathrm{T},i,k-1}$, and thus $\Delta\tau_{\mathrm{T},i,k}$, to be given only in terms of $\mathbf{x}_k$, needing not $\mathbf{x}_{k-1}$.

The covariances of the arrays of post-correction residuals (i.e., of all channels in use) for delay and delta-delay, $\boldsymbol{\delta\tau}_{\mathrm{T},k}$ and $\boldsymbol{\delta\Delta\tau}_{\mathrm{T},k}$, are then

$$\begin{aligned} E\left\{\boldsymbol{\delta\tau}_{\mathrm{T},k}\boldsymbol{\delta\tau}^{\mathsf{T}}_{\mathrm{T},k}\right\} &= E\left\{\left(\boldsymbol{\tau}_{\mathrm{T},k} - \hat{\boldsymbol{\tau}}_{\mathrm{T},k}\right)\left(\boldsymbol{\tau}_{\mathrm{T},k} - \hat{\boldsymbol{\tau}}_{\mathrm{T},k}\right)^{\mathsf{T}}\right\}\\ &= \mathrm{diag}\left(\boldsymbol{h}_{\tau_{\mathrm{T}}}(\hat{\mathbf{x}}_k)\right)\mathbf{R}_{s_{\mathrm{T}}}\mathrm{diag}\left(\boldsymbol{h}_{\tau_{\mathrm{T}}}(\hat{\mathbf{x}}_k)\right) + \mathbf{H}_{\tau_{\mathrm{T}},k}\mathbf{P}_k\mathbf{H}^{\mathsf{T}}_{\tau_{\mathrm{T}},k}\ , \end{aligned} \qquad (3.104)$$

and

$$\begin{aligned} E\left\{\boldsymbol{\delta\Delta\tau}_{\mathrm{T},k}\boldsymbol{\delta\Delta\tau}^{\mathsf{T}}_{\mathrm{T},k}\right\} &= E\left\{\left(\boldsymbol{\Delta\tau}_{\mathrm{T},k} - \boldsymbol{\Delta}\hat{\boldsymbol{\tau}}_{\mathrm{T},k}\right)\left(\boldsymbol{\Delta\tau}_{\mathrm{T},k} - \boldsymbol{\Delta}\hat{\boldsymbol{\tau}}_{\mathrm{T},k}\right)^{\mathsf{T}}\right\}\\ &= \mathrm{diag}\left(\boldsymbol{h}_{\Delta\tau_{\mathrm{T}}}(\hat{\mathbf{x}}_k)\right)\mathbf{R}_{s_{\mathrm{T}}}\mathrm{diag}\left(\boldsymbol{h}_{\Delta\tau_{\mathrm{T}}}(\hat{\mathbf{x}}_k)\right) + \mathbf{H}_{\Delta\tau_{\mathrm{T}},k}\mathbf{P}_k\mathbf{H}^{\mathsf{T}}_{\Delta\tau_{\mathrm{T}},k}\ , \end{aligned} \qquad (3.105)$$

with Jacobians

$$\boldsymbol{\Delta h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k) = \boldsymbol{h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k) - \boldsymbol{h}_{\tau_{\mathrm{T}},k-1}(\hat{\mathbf{x}}_k)\ , \qquad \mathbf{H}_{\Delta\tau_{\mathrm{T}},k} = \mathbf{H}_{\tau_{\mathrm{T}},k} - \mathbf{H}_{\tau_{\mathrm{T}},k-1}\ , \qquad (3.106)$$

and

$$\mathbf{H}_{\tau_{\mathrm{T}},k} = \left.\frac{\partial \boldsymbol{h}_{\tau_{\mathrm{T}},k}(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_k}\ , \qquad \mathbf{H}_{\tau_{\mathrm{T}},k-1} = \left.\frac{\partial \boldsymbol{h}_{\tau_{\mathrm{T}},k-1}(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_k}\ . \qquad (3.107)$$

$\mathbf{P}_k$ is the filter state estimate covariance matrix and $\mathbf{R}_{s_{\mathrm{T}}}$ is the covariance matrix of the set of fictitious correction model scale-factor errors and has the form

$$\mathbf{R}_{s_{\mathrm{T}}} = \begin{bmatrix} \sigma^2_{s_{\mathrm{T}},1} & & \\ & \ddots & \\ & & \sigma^2_{s_{\mathrm{T}},n} \end{bmatrix}\ . \qquad (3.108)$$

The value of each standard deviation $\sigma_{s_{\mathrm{T}},i}$ can be tuned according to the performance of the correction algorithm in use to ensure filter robustness against the delay residuals. Different values might be selected for different channels, for instance, depending on the satellite elevation. Alternatively, the scale-factor error can be defined as a single scalar s_{T} affecting the Tropospheric correction of all channels, then $\mathbf{R}_{s_{\mathrm{T}}}$ reduces to

$$\mathbf{R}_{s_{\mathrm{T}}} = \sigma^2_{s_{\mathrm{T}}}\mathbf{1}\mathbf{1}^{\mathsf{T}}\ . \qquad (3.109)$$

To properly model the set of scale-factors $\mathrm{s}_{\mathrm{T},i}$ (or the single s_{T} if this is the case) as strictly constant uncertainties in the filter, it is possible to include them as filter states (as standard or *considered* ones). This will increase the computational burden of this method. A lighter alternative is to treat $\mathrm{s}_{\mathrm{T},i}$ (or s_{T}) as measurement noise. Among these alternatives, as shall be shown in Section 3.10.2, using a single scalar scale-factor treated as measurement noise yields the best performance results.

Tropospheric Delay Model Jacobian

The Jacobian vector of the tropospheric delay correction model (2.30) with respect to the error-state vector can be given for satellite i at epoch k as

$$\begin{aligned}\mathbf{h}_{\tau_{\mathrm{T}},i,k} &= \frac{\partial h_{\tau_{\mathrm{T}},i,k}(\mathbf{x}_k)}{\partial \boldsymbol{\delta}\mathbf{x}_k} \\ &= -\frac{1}{c}\, M(\mathrm{E}_{i,k})\Delta(\mathrm{h}_{\mathsf{ant},k}) \left(0.133\times 10^{-3}\frac{\partial \mathrm{h}_{\mathsf{ant},k}}{\partial \boldsymbol{\delta}\mathbf{x}_k} + M(\mathrm{E}_{i,k})\frac{\partial \sin \mathrm{E}_{i,k}}{\partial \boldsymbol{\delta}\mathbf{x}_k}\right)\ , \qquad (3.110)\end{aligned}$$

with receiver antenna position and altitude, and satellite apparent elevation given as

$$\mathbf{r}^E_{\mathsf{ant},k} = \mathbf{r}^E_k + \mathbf{C}^{E_k}_{B_k}\mathbf{l}_{\mathsf{ant}} \qquad (3.111)$$

$$\mathrm{h}_{\mathsf{ant},k} = \|\mathbf{r}^E_{\mathsf{ant},k}\| - \mathrm{R}_\oplus \qquad (3.112)$$

$$\sin\left(\mathrm{E}_{i,k}\right) = \left(\mathbf{e}^E_{\mathsf{ant},k}\right)^{\mathsf{T}} \mathbf{e}^E_{\rho,i,k}\ , \qquad (3.113)$$

where $\mathrm{R}_\oplus$ is the Earth radius, $\mathbf{r}^E_{\mathsf{ant},k}$ is the receiver antenna position, $\mathbf{e}^E_{\mathsf{ant},k}$ is the unit direction from the ECEF origin to the receiver antenna, and $\mathbf{e}^E_{\rho,i,k}$ is the unit range vector from receiver to satellite i. The non-null partial derivatives in (3.110) are

$$\frac{\partial \sin\left(\mathrm{E}_{i,k}\right)}{\partial \boldsymbol{\delta}\mathbf{r}^E_k} = \frac{\hat{\mathbf{e}}^E_{\mathsf{ant},k}}{\|\hat{\mathbf{r}}^E_{\mathsf{ant},k}\|}^{\mathsf{T}} \left[\hat{\mathbf{e}}^E_{\rho,i,k}\times\right]^2 - \frac{\hat{\mathbf{e}}^E_{\rho,i,k}}{\hat{\rho}_{i,k}}^{\mathsf{T}} \left[\hat{\mathbf{e}}^E_{\mathsf{ant},k}\times\right]^2 \qquad (3.114)$$

$$\frac{\partial \mathrm{h}_{\mathsf{ant},k}}{\partial \boldsymbol{\delta}\mathbf{r}^E_k} = \left(\hat{\mathbf{e}}^E_{\mathsf{ant},k}\right)^{\mathsf{T}} \qquad (3.115)$$

$$\frac{\partial \sin\left(\mathrm{E}_{i,k}\right)}{\partial \boldsymbol{\delta\theta}^E_k} = \frac{\partial \sin\left(\mathrm{E}_{i,k}\right)}{\partial \boldsymbol{\delta}\mathbf{r}^E_k}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^B_{\mathsf{ant}}\times\right] \qquad (3.116)$$

$$\frac{\partial \mathrm{h}_{\mathsf{ant},k}}{\partial \boldsymbol{\delta\theta}^E_k} = \frac{\partial \mathrm{h}_{\mathsf{ant},k}}{\partial \boldsymbol{\delta}\mathbf{r}^E_k}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^B_{\mathsf{ant}}\times\right]\ . \qquad (3.117)$$

3.9.4 Ionospheric delay correction

Like the tropospheric delay, the ionospheric effects are feed-forward corrected, this time using the altitude-dependent VTEC model (2.31). However, as discussed in Section 2.5.2, the residuals of this correction are small enough to be safely neglected.

3.9.5 Pseudorange update

The model for a set of pseudorange measurements has the standard form

$$\mathbf{y}_{\rho,k} \;=\; \boldsymbol{h}_{\rho,k}(\mathbf{x}_k) + \boldsymbol{\nu}_{\rho,k} \;, \tag{3.118}$$

where $\boldsymbol{\nu}_{\rho,k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_{\rho,k}\right)$ is the pseudorange channel noise, and the measurement vector function $\boldsymbol{h}_{\rho}(\mathbf{x}_k)$ is for channel i

$$h_{\rho,i,k}(\mathbf{x}_k) = \rho_{i,k}\left(\mathbf{x}_k\right) + c\left(\tau_{\mathsf{r},k} + \tau_{\mathrm{T},i,k}\left(\mathbf{x}_k\right) + \tau_{\mathrm{I},i,k}\left(\mathbf{x}_k\right)\right) + \mathrm{b}_{\rho,i,k} \;, \tag{3.119}$$

with range given by

$$\rho_{i,k}\left(\mathbf{x}_k\right) = \left\|\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\mathbf{r}^{E}_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) - \mathbf{r}^{E}_{k} - \mathbf{C}^{E_k}_{B_k}\mathbf{l}^{B}_{\mathsf{ant}}\right\| \;. \tag{3.120}$$

The satellite position $\mathbf{r}^{E}_{\mathsf{s},i}$ is evaluated at the emitting instant $t_{\mathsf{s},i,k}$ [given in (2.16)] and translated to the ECEF frame at time of reception. This is done iteratively (as described in Section 2.5.1) using broadcasted ephemeris information and the standard algorithm given, e.g., in [124, 69, 118]. $\mathbf{l}_{\mathsf{ant}}$ is the lever-arm from IMU to receiver antenna. Each range bias $\mathrm{b}_{\rho,i}$ is modeled as a 1st-order GM process with a 30-minute correlation time to capture slow varying errors (ionosphere and satellite clock correction residuals and ephemeris errors) [69]. The receiver clock-bias τ_{r} is modeled as a state with dynamics given by (3.99). The tropospheric delay correction $\tau_{\mathrm{T},i,k}(\mathbf{x}_k)$ is computed as in (3.102). As described in Section 3.9.4, the ionospheric delay correction $\tau_{\mathrm{I},i,k}\left(\mathbf{x}_k\right)$ follows the model given in (2.31).

The error-measurement model is similar to (3.22) as

$$\boldsymbol{\delta}\mathbf{y}_{\rho,k} \;=\; \boldsymbol{h}_{\rho,k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - \boldsymbol{h}_{\rho,k}(\hat{\mathbf{x}}_k) + \boldsymbol{\nu}_{\rho,k} \;, \tag{3.121}$$

where for the i^{th} channel

$$h_{\rho,i,k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - h_{\rho,i,k}(\hat{\mathbf{x}}_k) = \delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) + c\left(\delta\tau_{\mathsf{r},k} + \delta\tau_{\mathrm{T},i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right)\right) + \delta\mathrm{b}_{\rho,i,k} \;, \tag{3.122}$$

Note the contribution of the ionospheric delay correction $\tau_{\mathrm{I},i,k}\left(\mathbf{x}_k\right)$ in (3.120) to the error-measurement (3.122) is considered null, as explained in Section 2.5.2.

The error-range $\delta\rho_{i,k}$ is

$$\delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) = \left\|\mathbf{C}^{E_k}_{E(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k})}\mathbf{r}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k}\right) - \hat{\mathbf{r}}^{E}_{\mathsf{ant},k} - \boldsymbol{\delta}\mathbf{r}^{E}_{\mathsf{ant},k}\right\| - \hat{\rho}_{i,k} \;. \tag{3.123}$$

where $\hat{t}_{\mathsf{s},i,k}$ is the estimated time of signal emission and $\delta\Delta t_{i,k}$ is the error in the estimated signal time-of-flight [see (2.16)]. The antenna position error is

$$\delta\mathbf{r}^{E}_{\mathsf{ant},k} = \delta\mathbf{r}^{E}_{k} + \hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\delta\boldsymbol{\theta}^{B}_{k} \;. \tag{3.124}$$

The satellite position term in (3.123) can be expanded resorting to Inertial coordinates and this time definition as

$$\begin{aligned}
\mathbf{C}^{E_k}_{E(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k})}\mathbf{r}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k}\right) &= \mathbf{C}^{E_k}_{I}\mathbf{r}^{I}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k}\right) \\
&\approx \mathbf{C}^{E_k}_{I}\left(\hat{\mathbf{r}}^{I}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) - \delta\Delta t_{i,k}\hat{\mathbf{v}}^{I}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right)\right) &\quad (3.125)\\
&\approx \hat{\mathbf{C}}^{E_k}_{E\left(t_{\mathsf{s},i,k}\right)}\hat{\mathbf{r}}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) - \delta\Delta t_{i,k}\hat{\mathbf{C}}^{E_k}_{E\left(t_{\mathsf{s},i,k}\right)}\left(\hat{\mathbf{v}}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^{E}_{IE}\hat{\mathbf{r}}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right)\right) \;, \\
&&\quad (3.126)
\end{aligned}$$

where (3.125) assumes the satellite has constant inertial velocity within the time scale of the perturbation $\delta\Delta t_{i,k}$.

Plugging (3.126) into (3.123) and realizing from (2.16) that

$$\delta\Delta t_{i,k} = \frac{\delta\rho_{i,k}}{c} \ , \tag{3.127}$$

yields

$$\begin{aligned}
\delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx & \left\| \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) - \frac{\delta\rho_{i,k}}{c} \hat{\mathbf{C}}^{E_k}_{E\left(t_{\mathsf{s},i,k}\right)} \left(\hat{\mathbf{v}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^E_{IE} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) \right) \right. \\
& \left. - \hat{\mathbf{r}}^E_{\mathsf{ant},k} - \boldsymbol{\delta}\mathbf{r}^E_{\mathsf{ant},k} \right\| - \hat{\rho}_{i,k} \\
\approx & \left\| \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) - \hat{\mathbf{r}}^E_{\mathsf{ant},k} \right\| - \hat{\rho}_{i,k} \\
& - \hat{\mathbf{e}}^E_{\rho,i,k}{}^{\mathsf{T}} \left(\frac{\delta\rho_{i,k}}{c} \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \left(\hat{\mathbf{v}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^E_{IE} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) \right) + \boldsymbol{\delta}\mathbf{r}^E_{\mathsf{ant},k} \right) \\
\approx & - \hat{\mathbf{e}}^E_{\rho,i,k}{}^{\mathsf{T}} \left(\frac{\delta\rho_{i,k}}{c} \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \left(\hat{\mathbf{v}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^E_{IE} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) \right) + \boldsymbol{\delta}\mathbf{r}^E_{\mathsf{ant},k} \right) \\
\approx & - \frac{1}{1 + \frac{1}{c} \hat{\mathbf{e}}^E_{\rho,i,k}{}^{\mathsf{T}} \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \left(\hat{\mathbf{v}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^E_{IE} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) \right)} \hat{\mathbf{e}}^E_{\rho,i,k}{}^{\mathsf{T}} \boldsymbol{\delta}\mathbf{r}^E_{\mathsf{ant},k} \ ,
\end{aligned} \tag{3.128}$$

where $\hat{\mathbf{e}}^E_{\rho,i,k}$ is the estimated unit range vector from receiver to satellite i. Expanding the antenna position error term as in (3.124) and renaming the fraction factor gives

$$\delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx -\left(1 + \hat{\beta}_{i,k}\right) \hat{\mathbf{e}}^E_{\rho,i,k}{}^{\mathsf{T}} \left(\boldsymbol{\delta}\mathbf{r}^E_k + \hat{\mathbf{C}}^{E_k}_{B_k} \left[\mathbf{l}^B_{\mathsf{ant}}\times\right] \boldsymbol{\delta\theta}^B_k \right) \ , \tag{3.129}$$

with

$$\hat{\beta}_{i,k} = -\frac{\frac{1}{c} \hat{\mathbf{e}}^E_{\varrho,i,k}{}^{\mathsf{T}} \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \left(\hat{\mathbf{v}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^E_{IE} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) \right)}{1 + \frac{1}{c} \hat{\mathbf{e}}^E_{\varrho,i,k}{}^{\mathsf{T}} \hat{\mathbf{C}}^{E_k}_{E\left(\hat{t}_{\mathsf{s},i,k}\right)} \left(\hat{\mathbf{v}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) + \boldsymbol{\Omega}^E_{IE} \hat{\mathbf{r}}^{E_k}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right) \right)} \ . \tag{3.130}$$

Note that, because the inertial orbital velocity of the emitting satellite is much lower than the speed of light in vacuum, i.e.,

$$\|\mathbf{v}^E_{\mathsf{s},i} + \boldsymbol{\Omega}^E_{IE}\mathbf{r}^E_{\mathsf{s},i}\| \ll c \ , \tag{3.131}$$

the term $\hat{\beta}_{i,k} \ll 1$. In fact, for the typical GPS satellite orbital velocities, $\hat{\beta}_{i,k}$ is in the order of 10^{-5}. Absence of this term carries an error of, at most, millimeter order. While this level of accuracy may be important in space applications such as rendezvous and docking or berthing (e.g., [186]), it is not crucial for the current case; this term is thus neglected. The model then becomes

$$\delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx -\hat{\mathbf{e}}^E_{\rho,i,k}{}^{\mathsf{T}} \left(\boldsymbol{\delta}\mathbf{r}^E_k + \hat{\mathbf{C}}^{E_k}_{B_k} \left[\mathbf{l}^B_{\mathsf{ant}}\times\right] \boldsymbol{\delta\theta}^B_k \right) \ . \tag{3.132}$$

The tropospheric error-delay $\delta\tau_{\mathrm{T},i,k}$ in (3.122) is given by

$$\delta\tau_{\mathrm{T},i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx s_{\mathrm{T},i}\tau_{\mathrm{T},i,k}\left(\hat{\mathbf{x}}_k\right) + \mathbf{h}_{\tau_{\mathrm{T}},i,k}\boldsymbol{\delta}\mathbf{x}_k \ . \tag{3.133}$$

where $\mathbf{h}_{\tau_{\mathrm{T}},i,k}$ is the Jacobian row vector of the tropospheric delay model function $\tau_{\mathrm{T},i,k}$ with respect to the error-state vector, as given in (3.110).

The covariance of the measurement innovation (3.30) is

$$E\left\{\mathbf{z}_{\rho,k}\mathbf{z}_{\rho,k}^{\mathsf{T}}\right\} = \mathbf{H}_{\rho,k}\mathbf{P}_k\mathbf{H}_{\rho,k}^{\mathsf{T}} + c^2\boldsymbol{h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)\mathbf{R}_{s_{\mathrm{T}}}\boldsymbol{h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)^{\mathsf{T}} + \mathbf{R}_{\rho,k} \ , \tag{3.134}$$

where $\boldsymbol{h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)$ is the array of estimated tropospheric delays, and $\mathbf{R}_{s_{\mathrm{T}}}$, the covariance matrix of the tropospheric delay correction model scale-factor, is given by (3.108) or (3.109) (see Section 3.9.3). The Jacobian $\mathbf{H}_{\rho,k}$ has the form

$$\mathbf{H}_{\rho,k} \ = \ \begin{bmatrix}\mathbf{h}_{\rho,1,k}^{\mathsf{T}} & \cdots & \mathbf{h}_{\rho,i,k}^{\mathsf{T}} & \cdots & \mathbf{h}_{\rho,n,k}^{\mathsf{T}}\end{bmatrix}^{\mathsf{T}} \ , \tag{3.135}$$

which has non-null partial derivatives

$$\mathbf{h}_{\rho,i,k}^{\mathrm{r}} \ = \ -\hat{\mathbf{e}}_{\rho,i,k}^{E}{}^{\mathsf{T}} + c\,\mathbf{h}_{\tau_{\mathrm{T}},i,k}^{\mathrm{r}} \ , \qquad \mathrm{h}_{\rho,i,k}^{\tau_{\mathrm{r}}} \ = \ c \ , \tag{3.136}$$

$$\mathbf{h}_{\rho,i,k}^{\theta} \ = \ -\hat{\mathbf{e}}_{\rho,i,k}^{E}{}^{\mathsf{T}}\hat{\mathbf{C}}_{B_k}^{E_k}\left[\mathbf{l}_{\mathsf{ant}}^{B}\times\right] + c\,\mathbf{h}_{\tau_{\mathrm{T}},i,k}^{\theta} \ , \qquad \mathrm{h}_{\rho,i,k}^{\mathrm{b}_{\rho,i}} \ = \ 1 \ . \tag{3.137}$$

$\hat{\mathbf{e}}_{\rho,i,k}^{E}$ is the estimated unit range vector from receiver to satellite i, and $\mathbf{h}_{\tau_{\mathrm{T}},i,k}^{\mathrm{r}}$ and $\mathbf{h}_{\tau_{\mathrm{T}},i,k}^{\theta}$ are the sensitivities of the tropospheric model function $h_{\tau_{\mathrm{T}},i,k}\left(\mathbf{x}_k\right)$ with respect to the error-states $\boldsymbol{\delta}\mathbf{r}_k^E$ and $\boldsymbol{\delta\theta}_k^B$, as given in Section 3.9.3.

3.9.6 Time-differenced carrier phase update

The TDCP measurement set can be written in the form of (3.19)

$$\mathbf{y}_{\Delta\rho,k} \ = \ \boldsymbol{h}_{\Delta\rho,k}(\mathbf{x}_k) + c\Delta\tau_{\mathrm{r},k}\mathbf{1} + \boldsymbol{\nu}_{\Delta\rho,k} \ , \tag{3.138}$$

where $\boldsymbol{\nu}_{\Delta\rho,k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_{\Delta\rho,k}\right)$ holds the single-channel noises. The noise $\boldsymbol{\epsilon}_k$ in (3.19) is here $c\Delta\tau_{\mathrm{r},k}\mathbf{1}$ with $\Delta\tau_{\mathrm{r},k} \sim \mathcal{N}\left(0, \sigma_{\tau_{\mathrm{r}},k-1}^2\right)$. This TDCP clock noise is perfectly correlated to the process noise of τ_{r}, as described in (3.101), which yields the non-null component of (3.20)

$$\mathbf{B}_{\mathrm{w}_{\tau,\mathrm{r}}}^{\epsilon} = E\{\mathrm{w}_{\tau_{\mathrm{r}},k-1}\boldsymbol{\epsilon}_k^{\mathsf{T}}\} = cE\{\mathrm{w}_{\tau_{\mathrm{r}},k-1}\Delta\tau_{\mathrm{r},k}\}\mathbf{1}^{\mathsf{T}} = c\sigma_{\tau_{\mathrm{r}},k-1}^2\mathbf{1}^{\mathsf{T}} \ . \tag{3.139}$$

The state-to-measurement mapping $\boldsymbol{h}_{\Delta\rho,k}(\mathbf{x}_k)$ is for channel i

$$h_{\Delta\rho,i,k}(\mathbf{x}_k) = \rho_{i,k}\left(\mathbf{x}_k\right) - \rho_{i,k-1}\left(\mathbf{x}_k\right) + c\left(\tau_{\mathrm{T},i,k}\left(\mathbf{x}_k\right) - \tau_{\mathrm{T},i,k-1}\left(\mathbf{x}_k\right)\right) + \mathrm{b}_{\Delta\rho,i,k} \ , \tag{3.140}$$

where the current range $\rho_{i,k}\left(\mathbf{x}_k\right)$ is as (3.120), and tropospheric correction $\tau_{\mathrm{T},i,k}\left(\mathbf{x}_k\right)$ as (3.102). The delta-range bias $\mathrm{b}_{\Delta\rho,k}$ of each channel i is modeled as a random walk process, capturing the residual errors in this measurement. The previous epoch range

$$\rho_{i,k-1}\left(\mathbf{x}_k\right) \ = \ \left\|\mathbf{C}_{E(t_{\mathsf{s},i,k-1})}^{E_{k-1}}\mathbf{r}_{\mathsf{s},i}^{E}\left(t_{\mathsf{s},i,k-1}\right) - \mathbf{r}_{\mathsf{ant},k-1,k}^{E_{k-1}}\right\| \ , \tag{3.141}$$

and previous epoch correction $\tau_{\mathrm{T},i,k-1}\left(\mathbf{x}_k\right)$ are computed using a delayed state of GNSS antenna position, $\mathbf{r}^{E_{k-1}}_{\mathsf{ant},k-1,k}$. This is modeled as a constant state (null process noise), that is reset after each measurement update cycle as

$$\left\{\hat{\mathbf{r}}^{E_{k-1}}_{\mathsf{ant},k-1}\right\}_{k^+} \leftarrow \left\{\hat{\mathbf{r}}^{E} + \hat{\mathbf{C}}^{E}_{B}\mathbf{l}^{B}_{\mathsf{ant}}\right\}_{k^+} \ . \tag{3.142}$$

Upon reset, its covariance and cross-covariance (to the remaining states) are set to

$$\left\{\mathbf{P}_{\mathrm{x}\mathrm{r}_{\mathsf{ant}}}\right\}_{k^+} \leftarrow \left\{\mathbf{P}_{\mathrm{xx}}\mathbf{\Phi}^{\mathrm{x}}_{\mathrm{r}_{\mathsf{ant}},0}{}^{\mathsf{T}}\right\}_{k^+} \tag{3.143}$$

$$\left\{\mathbf{P}_{\mathrm{r}_{\mathsf{ant}}\mathrm{r}_{\mathsf{ant}}}\right\}_{k^+} \leftarrow \mathbf{P}_{\mathrm{r}_{\mathsf{ant}},0} \ , \tag{3.144}$$

where $\mathbf{P}_{\mathrm{r}_{\mathsf{ant}},0}$ is a small initial covariance, $\mathbf{P}_{\mathrm{xx}}$ is the covariance matrix of the remaining states, $\mathbf{\Phi}_{\mathrm{r}_{\mathsf{ant}}\mathrm{x},0}$ is sparse with only

$$\mathbf{\Phi}^{\mathrm{r}}_{\mathrm{r}_{\mathsf{ant}},0} = \mathbf{I} \tag{3.145}$$

$$\mathbf{\Phi}^{\theta}_{\mathrm{r}_{\mathsf{ant}},0} = \hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right] \ . \tag{3.146}$$

The error-measurement $\boldsymbol{\delta}\mathbf{y}_{\Delta\rho,k}$ is similar in form to (3.121), being

$$\boldsymbol{\delta}\mathbf{y}_{\Delta\rho,k} \ = \ \boldsymbol{h}_{\Delta\rho,k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - \boldsymbol{h}_{\Delta\rho,k}(\hat{\mathbf{x}}_k) + \mathbf{1}c\Delta\tau_{\mathrm{r},k} + \boldsymbol{\nu}_{\Delta\rho,k} \ . \tag{3.147}$$

being for each channel i

$$h_{\Delta\rho,i,k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - h_{\Delta\rho,i,k}(\hat{\mathbf{x}}_k) = \delta\Delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) + c\delta\Delta\tau_{\mathrm{T},i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) + \delta\mathrm{b}_{\Delta\rho,i,k} \ , \tag{3.148}$$

and

$$\delta\Delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx \delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) + \hat{\mathbf{e}}^{E}_{\rho,i,k-1}{}^{\mathsf{T}}\boldsymbol{\delta}\mathbf{r}^{E_{k-1}}_{\mathsf{ant},k-1} \ , \tag{3.149}$$

$$\delta\Delta\tau_{\mathrm{T},i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx \mathrm{s}_{\mathrm{T},i}\Delta h_{\tau_{\mathrm{T}},i,k}\left(\hat{\mathbf{x}}_k\right) + \mathbf{h}_{\tau_{\mathrm{T}},i,k}\boldsymbol{\delta}\mathbf{x}_k - \mathbf{h}^{\mathrm{r}}_{\tau_{\mathrm{T}},i,k-1}\boldsymbol{\delta}\mathbf{r}^{E_{k-1}}_{\mathsf{ant},k-1} \ . \tag{3.150}$$

where $\delta\rho_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right)$ is as (3.132).

The TDCP measurement innovation covariance is formed accounting for the correlation (3.101), and noting that

$$\mathbf{R}_{\epsilon} = c^2\sigma^2_{\tau_{\mathrm{r}},k-1}\mathbf{1}\mathbf{1}^{\mathsf{T}} \ , \tag{3.151}$$

as

$$E\left\{\mathbf{z}_{\Delta\rho,k}\mathbf{z}^{\mathsf{T}}_{\Delta\rho,k}\right\} = \mathbf{H}_{\Delta\rho,k}\mathbf{P}_k\mathbf{H}^{\mathsf{T}}_{\Delta\rho,k} + c^2\boldsymbol{\Delta h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)\mathbf{R}_{\mathrm{ST}}\boldsymbol{\Delta h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)^{\mathsf{T}} + c^2\sigma^2_{\tau_{\mathrm{r}},k-1}\mathbf{1}\mathbf{1}^{\mathsf{T}} + \mathbf{R}_{\Delta\rho,k} \ , \tag{3.152}$$

where $\boldsymbol{\Delta h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)$ is the array of differences between present and previous tropospheric delays. The covariance matrix of the tropospheric delay correction model scale-factor $\mathbf{R}_{\mathrm{ST}}$, is given by (3.108) or (3.109) (see Section 3.9.3). The Jacobian $\mathbf{H}_{\Delta\rho,k}$ includes the sensitivities of the Tropospheric correction $\boldsymbol{\Delta h}_{\tau_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)$ to the current and previous antenna positions.

The measurement Jacobian matrix is

$$\mathbf{H}_{\Delta\rho,k} \ = \ \left[\mathbf{h}^{\mathsf{T}}_{\Delta\rho,1,k}\cdots\mathbf{h}^{\mathsf{T}}_{\Delta\rho,i,k}\cdots\mathbf{h}^{\mathsf{T}}_{\Delta\rho,n,k}\right]^{\mathsf{T}} \ , \tag{3.153}$$

with non-null partial derivatives

$$\mathbf{h}^{\mathrm{r}}_{\Delta\rho,i,k} = \mathbf{h}^{\mathrm{r}}_{\rho,i,k} \ , \qquad \mathbf{h}^{\mathrm{r}_{\mathsf{ant}}}_{\Delta\rho,i,k} = \hat{\mathbf{e}}^{E}_{\rho,i,k-1}{}^{\mathsf{T}} - c\,\mathbf{h}^{\mathrm{r}}_{\tau_{\mathrm{T}},i,k-1} \ , \tag{3.154}$$

$$\mathbf{h}^{\theta}_{\Delta\rho,i,k} = \mathbf{h}^{\theta}_{\rho,i,k} \ , \qquad \mathrm{h}^{\mathrm{b}_{\Delta\rho,i}}_{\Delta\rho,i,k} \ = \ 1 \ . \tag{3.155}$$

3.10 Performance evaluation

In this section, the navigation filter design described thus far is tested through hardware-in-the-loop simulation under nominal and GNSS outage conditions, and its performance is compared to different designs (e.g., the HNS). Tropospheric correction algorithm options are also compared, and filter consistency is attested.

3.10.1 Trajectory and simulation models

A simulated SHEFEX-2 trajectory (described in Appendix A) is used for tests done in this section. The real GNSS raw measurements are produced by a DLR Phoenix-HD GPS receiver unit (identical to that flown in the SHEFEX-2 vehicle) stimulated by a GNSS emulator (SPIRENT GSS7700) running the true trajectory (i.e., in a hardware in the loop setup). The inertial measurements are generated with the high-fidelity model of the form of that described Section 2.4.3 and fully detailed in Appendix Section C.2. A medium grade sensor profile is used (see Table 2.1). Further results of this system with a different inertial sensor and calibration model have been published in [183, 180].

3.10.2 Tropospheric delay correction

The tropospheric delay correction and compensation mechanism proposed in Section 3.9.3 is here tested.

Figure 3.6 shows the lower ascent velocity estimation of different correction configurations in a filter updated with PR and TDCP. As mentioned above, a SHEFEX-2 trajectory is used. The sensor profile is that of the tactical-grade unit (iMAR iIMU-FCAI-MDS) flown in that mission [160, 161]. Several troposphere corrected schemes are compared to two uncorrected set-ups, which include:

(a) an uncompensated (Figure 3.6a); and

(b) an atmospheric delay compensated one, which underweighs the GNSS measurements (e.g., as in [205]) with a conservative tuning approach similar to that used in the PR-only HNS version flown in SHEFEX-2 [160, 161] (Figure 3.6b).

Unlike the uncorrected and uncompesated set-up (Figure 3.6a), which displays a visibly incoherent behavior, the described conservative tuning is consistent (Figure 3.6b). However, it does not achieve the performance of the corrected configurations.

The tested set-ups using tropospheric delay correction include:

(c) one that neglects correction residuals (Figure 3.6c);

(d) one with covariance compensation through a single scalar scale-factor error (s_{T}), estimated within the filter as a state (Figure 3.6d);

(e) one with covariance compensation through a set of scale-factor errors (one per channel, $s_{\mathrm{T},i}$), all estimated as filter states (Figure 3.6e); and

(f) one with covariance compensation through a single scalar scale-factor error (s_{T}), treated as noise (i.e., without dedicated filter state) (Figure 3.6f).

From the plots shown, it is clear that neglecting the leftover residuals of Tropospheric delay correction also renders the filter incoherent (Figure 3.6c). Among the corrected and compensated schemes, modeling the correction uncertainty as a single (scalar) scale-factor (Figures 3.6d and 3.6f), rather than a set of factors (Figure 3.6e), yields the greatest improvement. Treating this scalar-factor as noise (Figure 3.6f) further decreases the effect on the estimation error and forgoes state augmentation. This last option shall be adopted in the remaining of this and following chapters.

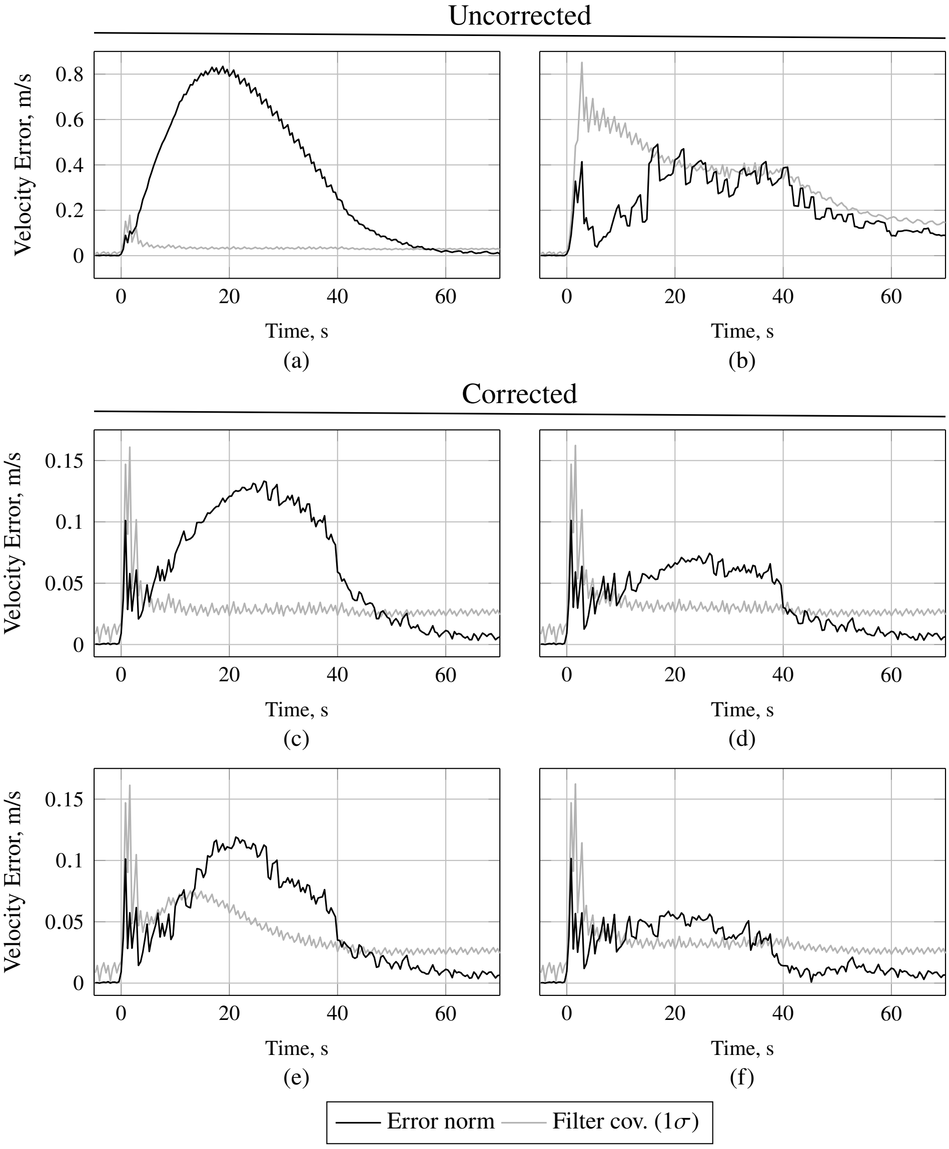


Figure 3.6: Velocity estimation comparison under nominal conditions (SHEFEX-2 trajectory) for different Tropospheric delay correction schemes in the filter updated with PR and TDCP measurements: (a) No correction, no compensation; (b) No correction, compensation through measurement underweighting; (c) Corrected, correction residuals neglected; (d) Corrected, cov. compensation with single scale-factor error, estimated as state; (e) Corrected, cov. compensation with one scale-factor error per channel, estimated as states; and (f) Corrected, cov. compensation with single scale-factor error, treated as noise (no extra state).

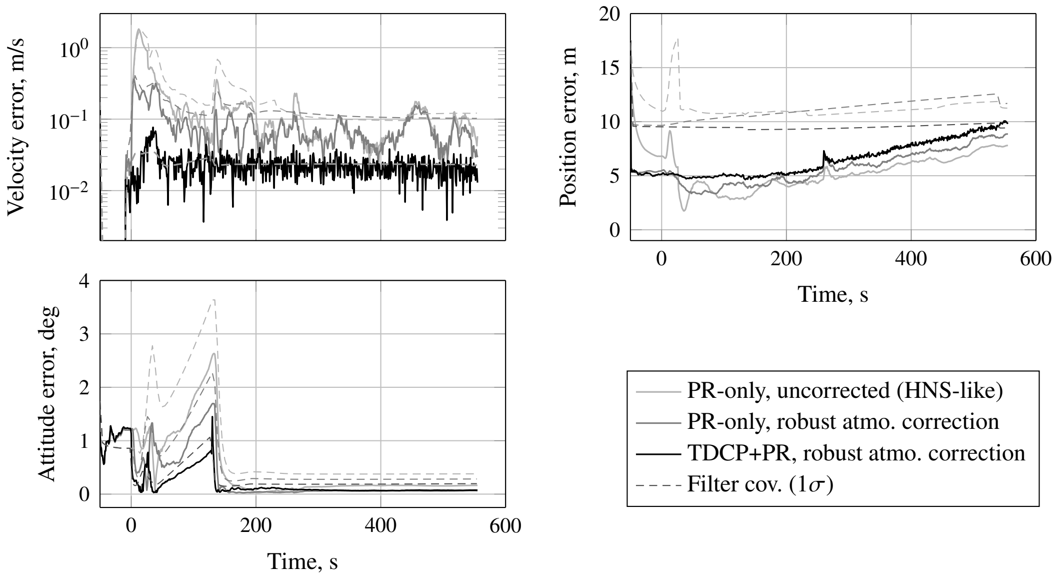


Figure 3.7: Kinematic states estimation error in the TDCP+PR filter under nominal conditions (SHEFEX-2 trajectory) compared to a PR-only filter with and without atmospheric correction

3.10.3 Nominal performance

The nominal performance of the baseline filter under development,

(1) a PR+TDCP filter with atmospheric delay/delay-rate correction [filter (f) in the previous section];

is assessed and compared in this section against:

(2) a PR-only SHEFEX-2 HNS-like filter without atmospheric delay correction; and

(3) a PR-only filter with atmospheric delay correction.

To compensate for (uncorrected) atmospheric delays in the GNSS signals, the HNS-like set-up [(2)] uses a conservative 1σ covariance of the PR measurements scheduled in altitude as [161]: 50 m below 10 km, 10 m for within 10–200 km, and 2 m above 200 km. The other PR-only filter [(3)], on the other hand, uses a robust correction set-up as proposed in Section 3.9.3 (with scalar s_T treated as measurement noise).

The kinematic state errors (velocity, position and attitude) of the above described filters are displayed in Figure 3.7 against their predicted 1σ covariance under the scenario conditions described in Section 3.10.1.

The PR-only filters [(2) and (3) in the above text] show quite similar velocity estimation performance after the lower ascent stage where the tropospheric effects are felt ($t < 60$ s). (Ionospheric disturbances in this sounding rocket flight are smaller than those shown in Section 2.5.2 for the Vega launcher.) Below this point the HNS-like filter performs visibly worse reaching about 2 m/s, 1σ (*vs* 0.4 m/s, 1σ of the corrected PR-only filter). However, it is still consistent due to its conservative tuning which compensates for the uncorrected tropospheric effects. This conservative tuning penalizes assimilation of GNSS-borne information, which deteriorates the estimation of kinematics and IMU states. This can be seen, for instance, in the covariance curves of accelerometer bias and scale-factor states, shown in Figure 3.8.

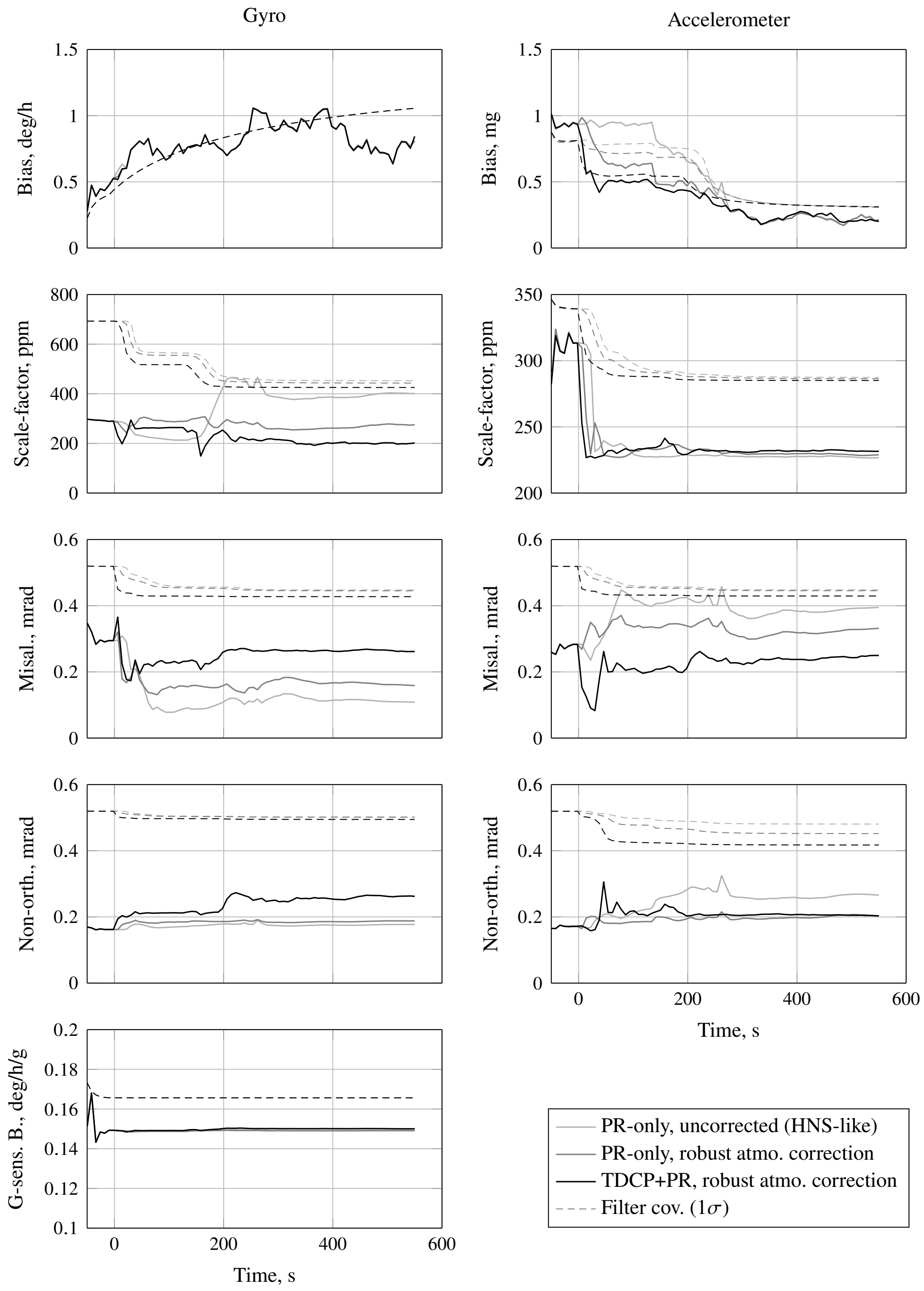


Figure 3.8: Inertial sensor states estimation error norm in the developed TDCP+PR filter under nominal conditions (SHEFEX-2 trajectory) compared to two PR-only filters: with and without atmospheric correction

The addition of TDCP measurements grants an additional significant velocity accuracy improvement, this time throughout (as already seen in Section 2.5.3), down to about 0.03–0.04 m/s, 1σ. Position performance is fairly on-par among all three filters, being mostly set by the PR error profile (about 9–12 m, 1σ). The only differences to be noted are: the higher error of the HNS-like filter during the early stage of flight, and the lower steady-state covariance of the TDCP-updated filter.

Attitude estimation accuracy sees a step-wise improvement, first from the correction of tropospheric effects (and tighter filter tuning) and then from the introduction of TDCP measurements. The main differences in the estimation of this state occur throughout the spun phase ($0 < t < 150$ s), with peaks of 3.7 deg, 2.3 deg, and 1.1 deg for filters (2), (3), and (1), respectively; while the post-burn, steady-state attitude errors ($t > 150$ s) converge to 0.38 deg, 0.28 deg, and 0.19 deg, again respectively.

The improved attitude estimation by the PR+TDCP filter [(1)] stems from the added information from TDCP measurements which also contributes to better estimation of accelerometer bias and gyro scale factor states, as shown in Figure 3.8. Other states as misalignments and non-orthogonalities also display a lower steady-state covariance in this filter, also decreasing attitude error covariance. A deeper investigation of the observability and error impact not only of different IMU error types but also of their different components is done in Chapter 4.

The presented nominal 1σ covariance results for the proposed filter [(1)] clearly fulfill the total position and total velocity performance guidelines set in Section 3.1 (25 m and 0.25 m/s, respectively) and only marginally violate the total attitude performance guideline at post-burn coasting (0.17 deg).

3.10.4 GNSS outage performance

As discussed in Section 2.2, vehicle dynamics can cause loss of lock in the GNSS tracking loops. Events such as separation, ignition, peak dynamic pressure, sonic transition are especially critical [73].

Figure 3.9 compares the estimated kinematics states (position, velocity and attitude) of the developed filter to those of the PR-only configurations described in the previous point and under the same trajectory but with a 30-second GNSS drop-out starting at $t = 20$ s. This outage is approximately centered around the estimated peak dynamic pressure (~35 s) which is believed to have caused one of the largest drops in GNSS signal strength during the real SHEFEX-2 flight (see GPS flight data analysis in Section A.4).

As expected all three filters show drift in all three kinematics quantities during the outage. All filters, however, behave coherently.

Comparing the PR-only set-ups, the atmospheric-corrected filter shows a lower drift in position and velocity (60 m *vs* 145 m, and 4 m/s *vs* 7 m/s, 1σ). As seen in the previous point, the atmospheric corrections allow not only for better estimation of kinematics states but also for faster estimation of IMU uncertainties, when compared to the conservative HNS approach. At the outage onset (i.e., at the start of a dead-reckoning phase) these lower uncertainty levels on both state sets result in lower kinematics divergence. Note that during filter update outage, IMU state error leaks into the kinematics states through the strapdown integration process. As an example, accelerometer scale-factor state covariance (shown in Figure 3.10) is lower on the atmospheric-corrected PR-only

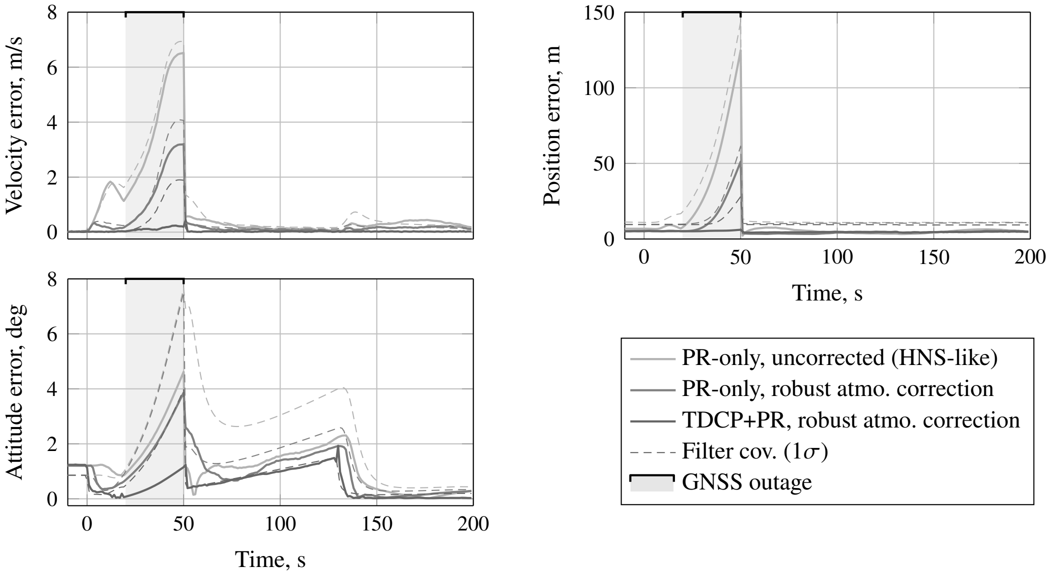


Figure 3.9: Performance of the developed TDCP+PR filter compared to two PR-only filters (with/without atmospheric correction) under GNSS outage

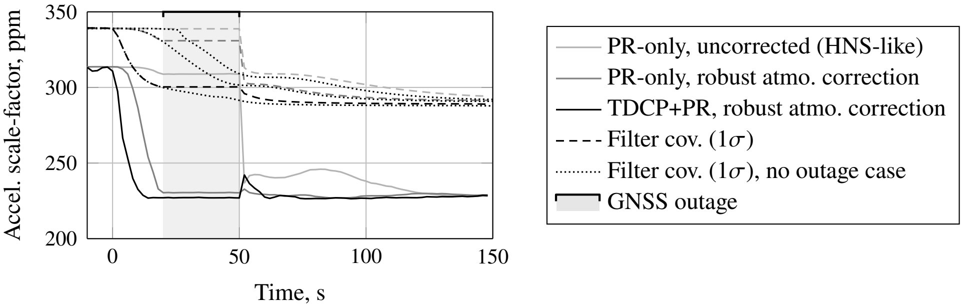


Figure 3.10: Accelerometer scale-factor estimation by the TDCP+PR filter and PR-only filters (with/without atmospheric correction) under GNSS outage

filter at beginning of the GNSS drop-out, compared to the HNS-like filter. Attitude state covariance divergence is on-par between these two filters (7.7 deg, 1σ), which is due to similar uncertainty levels on this state at the drop-out onset (1.1 deg, 1σ). However, the atmospheric-corrected set-up shows a considerably better recovery as GNSS measurements are once again available.

The proposed PR+TDCP configuration displays the lowest kinematic state divergence (25 m, 4 m/s, and 1.9 deg, 1σ), owing to a lower estimation error in kinematics and sensor states at the start of the outage. Notice, for instance, the relatively lower accelerometer scale-factor covariance (in Figure 3.10) in this filter at that instant. Although this IMU state (as all other sensor states) is not updated during the drop-out in any of the filters, visibly lagging with respect to the no-outage covariance levels, it slowly converges to the nominal curves after GNSS reacquisition.

3.10.5 Filter consistency

The consistency of a Kalman filter can be assessed through the statistical properties of the innovation sequence of its measurement updates [13, 38]. In an appropriately modeled and well tuned filter, the innovation vector [as defined in (3.30) or (3.31)] should be a zero-mean, white sequence with covariance $\mathbf{P}_{\mathrm{zz}}$ given by (3.35) (see, e.g., [107]). If these properties hold, then the normalized innovation (norm) squared, given, e.g., by [13], as

$$\mathrm{T}^2_{\mathrm{z},k} = \mathbf{z}_k^{\mathsf{T}} \mathbf{P}^{-1}_{\mathrm{zz},k} \mathbf{z}_k \ , \tag{3.156}$$

follows a χ^2-distribution with a number of DoFs equal to the number of measurements $n_{\mathrm{z},k}$ at epoch k (i.e., the size of $\mathbf{z}_k$). (Note that $\mathrm{T}^2_{\mathrm{z},k}$ and its statistical properties shall be further explored in Chapter 5 upon filter FDIR design.) The time-average of $\mathrm{T}^2_{\mathrm{z},k}/n_{\mathrm{z},k}$, given by

$$\bar{\mathrm{T}}^2_{\mathrm{z},N} = \frac{1}{N} \sum_{k=0}^{N-1} \frac{\mathrm{T}^2_{\mathrm{z},k}}{n_{\mathrm{z},k}} \ , \tag{3.157}$$

where N is the total number of innovation samples included, being the sum of a sequence of χ^2-distributed samples, is therefore itself χ^2-distributed with a number of DoFs given by the sum of the DoFs of all elements in the average, i.e., $\sum_{k=0}^{N-1} n_{\mathrm{z},k}$ (or $N n_{\mathrm{z}}$ if the number of measurements per epoch is constant, as in [13]). $\bar{\mathrm{T}}^2_{\mathrm{z},N}$ asymptotically converges to 1, i.e.,

$$\lim_{N\to\infty} \bar{\mathrm{T}}^2_{\mathrm{z},N} = 1 \ . \tag{3.158}$$

The *null* hypothesis, H_0, that the innovations are indeed zero-mean with covariance consistent with $\mathbf{P}_{\mathrm{zz}}$ is equivalent to the limit (3.158) being valid [13]. H_0 can be checked using a χ^2-*test* [5, referred to as T^2-*test*], being accepted with a confidence level $1-\alpha$ if

$$\bar{\mathrm{T}}^2_{\mathrm{z},N} \in \left[\frac{\chi^2_{\sum n_{\mathrm{z}}}\left(\frac{\alpha}{2}\right)}{\sum_{k=0}^{N-1} n_{\mathrm{z},k}} \ , \ \frac{\chi^2_{\sum n_{\mathrm{z}}}\left(1-\frac{\alpha}{2}\right)}{\sum_{k=0}^{N-1} n_{\mathrm{z},k}} \right] \ , \tag{3.159}$$

where $\chi^2_n(\cdot)$ denotes the quantile function (or inverse cumulative distribution function) of a χ^2-distribution with n DoFs.

Figure 3.11 displays the normalized innovation norm squared $\mathrm{T}^2_{\mathrm{z},k}$ of both PR and TDCP measurement update sets. Shown is also its time-average [given in (3.157)] and the 95% confidence level bounds of the latter [as in (3.159)]. As can be seen, in both plots the time-averaged curve is in accordance with the prescribed confidence bounds.

The whiteness of a zero-mean, scalar innovation sequence can be tested through its sample auto-correlation coefficient

$$\varrho_{\mathrm{z}}(l) = \frac{E\left\{\mathrm{z}_k \mathrm{z}_{k+l}\right\}}{\sqrt{E\left\{\mathrm{z}_k^2\right\} E\left\{\mathrm{z}_{k+l}^2\right\}}} \ , \tag{3.160}$$

where l is the correlation sample lag (or offset). If z is indeed white, then the following holds:

$$\varrho_{\mathrm{z}}(l) = \begin{cases} 1\,, & l = 0 \\ 0\,, & l > 0 \end{cases} \ . \tag{3.161}$$

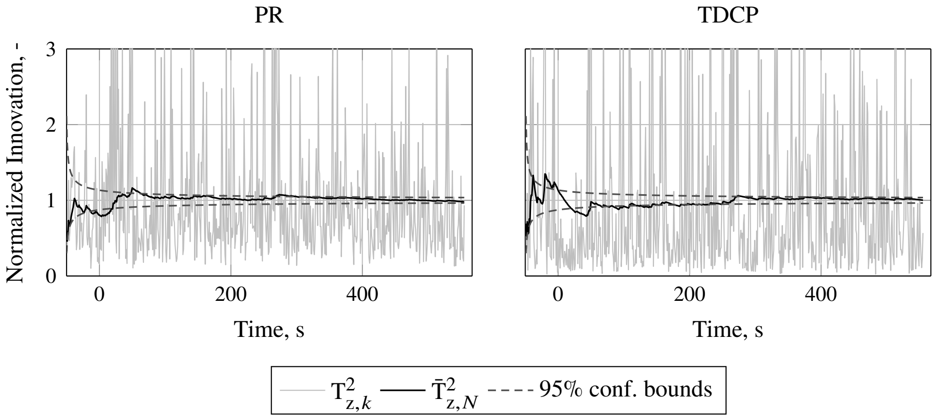


Figure 3.11: Normalized innovation norm squared statistic of the developed PR+TDCP filter under nominal conditions on a SHEFEX-2 trajectory using real GNSS measurements

This autocorrelation profile can be compared to that (numerically) estimated from the innovations sequence to test its whiteness. Another interesting test, proposed by Stoica [165], uses the accumulated autocorrelation over growing offset l to assess whiteness.

For a vector innovations $\mathbf{z}$, the sample auto-correlation coefficient can estimated as

$$\hat{\varrho}_z(l) = \frac{1}{N} \sum_{k=0}^{N-l-1} \left. \frac{\mathbf{z}_k^\mathsf{T} \mathbf{L}_{z,k}^{-\mathsf{T}} \mathbf{L}_{z,k+l}^{-1} \mathbf{z}_{k+l}}{\sqrt{n_z}} \right|_{Y_k \cap Y_{k+l}} , \tag{3.162}$$

where $Y_k \cap Y_{k+l}$ denotes the set of common measurement channels between the epochs k and $k+l$; n_z is the size of such set. $\mathbf{L}_{z,k}$ is the lower triangular Cholesky root of $\mathbf{P}_{zz,k}$, i.e.,

$$\mathbf{P}_{zz} = \mathbf{L}_{z,k} \mathbf{L}_{z,k}^\mathsf{T} . \tag{3.163}$$

For each value of the offset l, only the common measurements of both epochs k and $k+l$ are used to compute the inner-product of normalized innovation vectors in (3.162). As above mentioned, this is denoted by $Y_k \cap Y_{k+l}$.

The sample auto-correlation coefficient estimator (3.162) is similar to that described by Bar-Shalom et al. [13], in that it normalizes the innovation samples multiplying them by their covariance matrix square root inverse. Note, however, that here instead of the sample-computed covariance matrix (as in [13]), the filter predicted matrix $\mathbf{P}_{zz}$ is used (as done in [110]). Furthermore, the estimator (3.162) is of the *biased* type (with N as denominator) which yields lower mean square error when compared to the *unbiased* estimator (with $N-l$ as denominator) [90].

Using the central limit theorem [43], it is possible to show that for large N and zero-mean, white innovations the estimated $\hat{\varrho}_z(l)$ is approximately normally-distributed [13, 38], with zero-mean and variance proportional to $(N-l)/N^2$ [106]. (Note that in [13, 38] the variance of $\hat{\varrho}_z(l)$ is said to be proportional to $1/N$; this is only valid for $N \gg l$.) In fact, because (3.162) decorrelates the innovation vectors by multiplying them by the inverse square root (here Cholesky root) matrix of their covariance matrix, the variance of the estimated $\hat{\varrho}_z(l)$ shall be exactly $(N-l)/N^2$. Note that the factorization of the inner-product of normalized innovations by $\sqrt{n_z}$ renders the variance of $\hat{\varrho}_z(l)$ independent of n_z [13], which is an important feature as n_z may vary throughout the filter

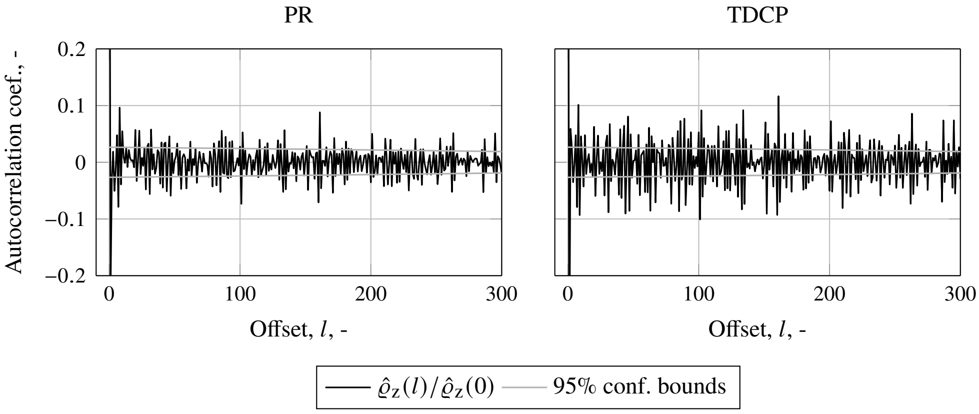


Figure 3.12: Innovation sample autocorrelation coefficient statistic of the developed PR+TDCP filter under nominal conditions on a SHEFEX-2 trajectory using real GNSS measurements

run as different sets of satellites are tracked. However, this also renders $E\left\{\hat{\varrho}_z(0)\right\} \neq 1$. (Using n_z as denominator instead of $\sqrt{n_z}$ would yield $E\left\{\hat{\varrho}_z(0)\right\} = 1$.)

The *null* hypothesis, H_0, that the innovations are indeed white is accepted with a confidence level $1-\alpha$ if

$$\hat{\varrho}_z\left(l\right) \in \left[\frac{\sqrt{N-l}}{N}\mathcal{N}\left(\frac{\alpha}{2}\right),\ \frac{\sqrt{N-l}}{N}\mathcal{N}\left(1-\frac{\alpha}{2}\right)\right], \quad l > 0\,, \tag{3.164}$$

where $\mathcal{N}(\cdot)$ denotes the quantile function (or inverse cumulative distribution function) of the standard normal distribution [i.e., $\mathcal{N}(1,\,0)$].

Figure 3.12 displays the estimated auto-correlation coefficient $\hat{\varrho}_z\left(l\right)$ computed as in (3.162) and factorized by $\hat{\varrho}_z\left(0\right)$ for both PR and TDCP sets and for an offset l up to 300 (i.e., half of the trajectory run-time). The 95% confidence bounds are also shown.

Although the shapes of the autocorrelation curves plotted in Figure 3.12 are as expected from a white sequence [see (3.161)], the variance of the $l > 0$ region is higher than what the 95% bounds predict. This is especially true in the case of TDCP innovations, and is caused by the less-than-white nature of the real GNSS raw measurements used. A way to make the estimated autocorrelation curves agree more closely with the predicted bounds would be to increase the filter's measurement noise covariance matrices. This, however, would severely penalize performance but would also considerably change the innovations mean and covariance, largely altering the results of the χ^2-test of $\bar{\mathrm{T}}^2_{\mathrm{z},N}$ (see above). The current tuning is thus deemed appropriate and the filter sufficiently consistent under the real signals at hand.

4

Robust Reduced Order Filter Design

This chapter departs from the navigation system design of Chapter 3 and develops a set of reduced order filter designs based on Consider Kalman filtering (see Section 2.6.2). The developments in this chapter have been published in [184]. The observability analysis framework here presented has been further used in [141, 149].

4.1 Motivation and concept

The system described in the previous chapter tackles the disturbing effects in inertial and GNSS measurements mostly through intensive modeling and filter augmentation, i.e., including all sensor parameters as filter states. As discussed in Section 2.6.2, this is generally effective but potentially very costly in terms of computational requirements. Alternatively, and as also mentioned in Section 2.6.2, the covariance structure of these uncertain parameters can be accounted for without their explicit estimation through Consider Kalman filtering [38]. The strong asymmetries of launcher dynamics can cause considerable discrepancies in both impact and observability between uncertainty sources and their different components. This chapter introduces a framework for efficient filter state reduction and allocation using *considered*-parameters. Analysis of the impact on the system propagation error of each uncertainty is used to identify potentially neglectable sensor error components, while study of parameter observability within the hybrid filter allows effective selection of estimated or consider parameters.

4.2 Error-state Consider Kalman filter

The error-state Kalman filter of Section 3.4 is here modified to include *consider*-parameters. As before, correlation between measurement and process noises is also included. The presented derivation differs from that found in the literature in that it explicitly separates the filter expressions that depend only on standard-state quantities and those that arise from consider parameter terms. This allows for a modular filter software design where the consider-parameter contributions can be easily enabled and disabled (either in compilation or in operation).

The whole-state system and measurement models are in all similar to those in Section 3.4, but are here written with explicit contributions from a consider-parameter set $\mathbf{c}$ as

$$\mathbf{x}_{k+1} = \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\mathbf{x}_k,\, \mathbf{c}_k) + \mathbf{w}_{\mathrm{x},k} \tag{4.1}$$

$$\mathbf{y}_k = \boldsymbol{h}_k(\mathbf{x}_k,\, \mathbf{c}_k) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \ , \tag{4.2}$$

with $\mathbf{w}_{\mathrm{x},k} \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{Q}_{\mathrm{x},k+1|k}\right)$, $\boldsymbol{\epsilon}_k \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{R}_{\epsilon,k}\right)$ and $\boldsymbol{\nu}_k \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{R}_{\nu,k}\right)$. The consider-parameter set can also be assigned dynamics as

$$\mathbf{c}_{k+1} = \boldsymbol{\phi}_{\mathrm{c},k+1|k}(\mathbf{c}_k) + \mathbf{w}_{\mathrm{c},k} \ , \tag{4.3}$$

with $\mathbf{w}_{\mathrm{c},k} \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{Q}_{\mathrm{c},k+1|k}\right)$.

As in Section 3.4, the measurement noise $\boldsymbol{\epsilon}$ is correlated to the process noise. For generality, this correlation is assumed to exist for both process noise components $\mathbf{w}_\mathrm{x}$ and $\mathbf{w}_\mathrm{c}$. Thus,

$$E\left\{\boldsymbol{\epsilon}_k \mathbf{w}_{\mathrm{x},k-1}^\mathsf{T}\right\} = \mathbf{B}_{\mathrm{x},k}^\mathsf{T} \ , \tag{4.4}$$

$$E\left\{\boldsymbol{\epsilon}_k \mathbf{w}_{\mathrm{c},k-1}^\mathsf{T}\right\} = \mathbf{B}_{\mathrm{c},k}^\mathsf{T} \ . \tag{4.5}$$

The models for the error-state, given by $\boldsymbol{\delta}\mathbf{x}_{k+1} = \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1}$, and error-measurement, given by $\boldsymbol{\delta}\mathbf{y}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k$, are also similar to those in Section 3.4, being now written in terms of the actual consider parameter set $\mathbf{c}$ and its known/assumed value $\hat{\mathbf{c}}$ as

$$\boldsymbol{\delta}\mathbf{x}_{k+1} = \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k,\, \mathbf{c}_k) - \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\hat{\mathbf{x}}_k,\, \hat{\mathbf{c}}_k) + \mathbf{w}_{\mathrm{x},k} \ , \tag{4.6}$$

$$\boldsymbol{\delta}\mathbf{y}_k = \boldsymbol{h}_k(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k,\, \mathbf{c}_k) - \boldsymbol{h}_k(\hat{\mathbf{x}}_k,\, \hat{\mathbf{c}}_k) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \ . \tag{4.7}$$

Error-state estimate propagation can be done either as

$$\boldsymbol{\delta}\hat{\mathbf{x}}_{k+1} = \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\hat{\mathbf{x}}_k,\, \hat{\mathbf{c}}_k) - \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\hat{\mathbf{x}}_k,\, \hat{\mathbf{c}}_k) \ , \tag{4.8}$$

or, with linearized dynamics, as

$$\boldsymbol{\delta}\hat{\mathbf{x}}_{k+1} = \boldsymbol{\Phi}_{\mathrm{xx},k+1|k} \boldsymbol{\delta}\hat{\mathbf{x}}_k \ . \tag{4.9}$$

The filter estimate covariance is propagated as

$$\begin{aligned}\mathbf{P}_{\mathrm{xx},(k+1)^-} &= E\left\{\left(\boldsymbol{\delta}\mathbf{x}_{k+1} - \boldsymbol{\delta}\hat{\mathbf{x}}_{(k+1)^-}\right)\left(\boldsymbol{\delta}\mathbf{x}_{k+1} - \boldsymbol{\delta}\hat{\mathbf{x}}_{(k+1)^-}\right)^\mathsf{T}\right\} \\ &= \mathbf{P}_{\mathrm{xx|x},(k+1)^-} + \mathbf{P}_{\mathrm{xx|c},(k+1)^-} \ ,\end{aligned} \tag{4.10}$$

with the terms due to state-only and to consider-parameter given by

$$\mathbf{P}_{\mathrm{xx|x},(k+1)^-} = \boldsymbol{\Phi}_{\mathrm{xx},k+1|k} \mathbf{P}_{\mathrm{xx},k^+} \boldsymbol{\Phi}_{\mathrm{xx},k+1|k}^\mathsf{T} + \mathbf{Q}_{\mathrm{x},k+1|k} \tag{4.11}$$

$$\mathbf{P}_{\mathrm{xx|c},(k+1)^-} = \boldsymbol{\Phi}_{\mathrm{xc},k+1|k} \mathbf{P}_{\mathrm{xc},k^+}^\mathsf{T} \boldsymbol{\Phi}_{\mathrm{xx},k+1|k}^\mathsf{T} + \left(\boldsymbol{\Phi}_{\mathrm{xx},k+1|k} \mathbf{P}_{\mathrm{xc},k^+} + \boldsymbol{\Phi}_{\mathrm{xc},k+1|k} \mathbf{P}_{\mathrm{cc},k}\right) \boldsymbol{\Phi}_{\mathrm{xc},k+1|k}^\mathsf{T} \ . \tag{4.12}$$

The state covariance propagation terms due to state-only contributions (4.11) are identical to that of the standard filter (3.25).

The cross-covariance between state and considered-parameter is propagated as

$$\begin{aligned}\mathbf{P}_{\mathrm{xc},(k+1)^-} &= E\left\{\left(\boldsymbol{\delta}\mathbf{x}_{k+1} - \boldsymbol{\delta}\hat{\mathbf{x}}_{(k+1)^-}\right)\left(\mathbf{c}_{k+1} - \hat{\mathbf{c}}_{k+1}\right)^{\mathsf{T}}\right\} \\ &= \left(\boldsymbol{\Phi}_{\mathrm{xx},k+1|k}\mathbf{P}_{\mathrm{xc},k^+} + \boldsymbol{\Phi}_{\mathrm{xc},k+1|k}\mathbf{P}_{\mathrm{cc},k^+}\right)\boldsymbol{\Phi}^{\mathsf{T}}_{\mathrm{cc},k+1|k} \ . \end{aligned} \tag{4.13}$$

Similarly, consider-parameter knowledge covariance follows

$$\begin{aligned}\mathbf{P}_{\mathrm{cc},(k+1)^-} &= E\left\{\left(\mathbf{c}_{k+1} - \hat{\mathbf{c}}_{k+1}\right)\left(\mathbf{c}_{k} - \hat{\mathbf{c}}_{k}\right)^{\mathsf{T}}\right\} \\ &= \boldsymbol{\Phi}_{\mathrm{cc},k+1|k}\mathbf{P}_{\mathrm{cc},k^+}\boldsymbol{\Phi}^{\mathsf{T}}_{\mathrm{cc},k+1|k} + \mathbf{Q}_{\mathrm{c},k+1|k} \ . \end{aligned} \tag{4.14}$$

The process transition matrices $\boldsymbol{\Phi}_{\mathrm{xx},k+1|k}$, $\boldsymbol{\Phi}_{\mathrm{xc},k+1|k}$, and $\boldsymbol{\Phi}_{\mathrm{cc},k+1|k}$ are

$$\boldsymbol{\Phi}_{\mathrm{xx},k+1|k} = \left.\frac{\partial \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\mathbf{x}_k, \mathbf{c}_k)}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k^+}, \hat{\mathbf{c}}_k} \ , \tag{4.15}$$

$$\boldsymbol{\Phi}_{\mathrm{xc},k+1|k} = \left.\frac{\partial \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\mathbf{x}_k, \mathbf{c}_k)}{\partial \mathbf{c}_k}\right|_{\hat{\mathbf{x}}_{k^+}, \hat{\mathbf{c}}_k} \ , \quad \text{and} \tag{4.16}$$

$$\boldsymbol{\Phi}_{\mathrm{cc},k+1|k} = \left.\frac{\partial \boldsymbol{\phi}_{\mathrm{c},k+1|k}(\mathbf{c}_k)}{\partial \mathbf{c}_k}\right|_{\hat{\mathbf{c}}_k} \ . \tag{4.17}$$

The *a priori* state is corrected as in (3.27) by

$$\boldsymbol{\delta}\hat{\mathbf{x}}_{k^+} = \boldsymbol{\delta}\hat{\mathbf{x}}_{k^-} + \mathbf{K}_{\mathrm{x},k}\mathbf{z}_k \ , \tag{4.18}$$

with covariance identical to (3.28), i.e.,

$$\begin{aligned}\mathbf{P}_{\mathrm{xx},k^+} &= E\left\{\left(\boldsymbol{\delta}\mathbf{x}_k - \boldsymbol{\delta}\hat{\mathbf{x}}_{k^+}\right)\left(\boldsymbol{\delta}\mathbf{x}_k - \boldsymbol{\delta}\hat{\mathbf{x}}_{k^+}\right)^{\mathsf{T}}\right\} \\ &= \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{P}_{\mathrm{xz},k^-}\mathbf{K}_k^{\mathsf{T}} - \mathbf{K}_k\mathbf{P}^{\mathsf{T}}_{\mathrm{xz},k^-} + \mathbf{K}_k\mathbf{P}_{\mathrm{zz},k}\mathbf{K}_k^{\mathsf{T}} \ . \end{aligned} \tag{4.19}$$

The *a posteriori* state to consider-parameter cross-covariance is

$$\begin{aligned}\mathbf{P}_{\mathrm{xc},k^+} &= E\left\{\left(\boldsymbol{\delta}\mathbf{x}_k - \boldsymbol{\delta}\hat{\mathbf{x}}_{k^+}\right)\left(\mathbf{c}_k - \hat{\mathbf{c}}_k\right)^{\mathsf{T}}\right\} \\ &= \mathbf{P}_{\mathrm{xc},k^-} - \mathbf{K}_k E\left\{\mathbf{z}_k\left(\mathbf{c}_k - \hat{\mathbf{c}}_k\right)^{\mathsf{T}}\right\} \\ &= \mathbf{P}_{\mathrm{xc},k^-} - \mathbf{K}_k\mathbf{P}^{\mathsf{T}}_{\mathrm{cz},k^-} \ . \end{aligned} \tag{4.20}$$

The consider-parameter set is not updated, thus

$$\mathbf{P}_{\mathrm{cc},k^+} = \mathbf{P}_{\mathrm{cc},k^-} \ . \tag{4.21}$$

The Kalman gain is obtained exactly as in (3.29), yielding

$$\mathbf{K}_k = \mathbf{P}_{\mathrm{xz},k^-}\mathbf{P}^{-1}_{\mathrm{zz},k} \ . \tag{4.22}$$

The innovation $\mathbf{z}_k$ is now given as

$$\mathbf{z}_k = \boldsymbol{\delta}\mathbf{y}_k - \boldsymbol{h}_k(\hat{\mathbf{x}}_{k^-} + \boldsymbol{\delta}\hat{\mathbf{x}}_{k^-}, \hat{\mathbf{c}}_k) + \boldsymbol{h}_k(\hat{\mathbf{x}}_{k^-}, \hat{\mathbf{c}}_k) \ , \tag{4.23}$$

which, if linearized, gives

$$\mathbf{z}_k = \boldsymbol{\delta}\mathbf{y}_k - \mathbf{H}_{\mathrm{x},k}\boldsymbol{\delta}\hat{\mathbf{x}}_{k^-} \ , \tag{4.24}$$

and, further linearizing $\boldsymbol{\delta}\mathbf{y}_k$ in (4.7),

$$\mathbf{z}_k = \mathbf{H}_{\mathrm{x},k}(\boldsymbol{\delta}\mathbf{x}_{k^-} - \boldsymbol{\delta}\hat{\mathbf{x}}_{k^-}) + \mathbf{H}_{\mathrm{c},k}(\mathbf{c}_k - \hat{\mathbf{c}}_k) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \ . \tag{4.25}$$

As in the propagation step, the state-to-innovation cross-covariance is also split into state-only and consider-parameter contributions yielding

$$\mathbf{P}_{\mathrm{xz},k} = E\left\{(\boldsymbol{\delta}\mathbf{x}_k - \boldsymbol{\delta}\hat{\mathbf{x}}_k)\,\mathbf{z}_k^\mathsf{T}\right\} = \mathbf{P}_{\mathrm{xz|x},k} + \mathbf{P}_{\mathrm{xz|c},k} \ , \tag{4.26}$$

with

$$\mathbf{P}_{\mathrm{xz|x},k} = \mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}_{\mathrm{x},k}^\mathsf{T} + \mathbf{B}_{\mathrm{x},k} \tag{4.27}$$

$$\mathbf{P}_{\mathrm{xz|c},k} = \mathbf{P}_{\mathrm{xc},k^-}\mathbf{H}_{\mathrm{c},k}^\mathsf{T} \ . \tag{4.28}$$

Likewise, the innovation covariance $\mathbf{P}_{\mathrm{zz},k}$ in (3.28) can be expanded as

$$\mathbf{P}_{\mathrm{zz},k} = E\left\{\mathbf{z}_k\mathbf{z}_k^\mathsf{T}\right\} = \mathbf{P}_{\mathrm{zz|x},k} + \mathbf{P}_{\mathrm{zz|c},k} + \mathbf{R}_{\epsilon,k} + \mathbf{R}_{\nu,k} \ , \tag{4.29}$$

with

$$\mathbf{P}_{\mathrm{zz|x},k} = \mathbf{H}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}_{\mathrm{x},k}^\mathsf{T} + \mathbf{H}_{\mathrm{x},k}\mathbf{B}_{\mathrm{x},k} + \mathbf{B}_{\mathrm{x},k}^\mathsf{T}\mathbf{H}_{\mathrm{x},k}^\mathsf{T} \tag{4.30}$$

$$\mathbf{P}_{\mathrm{zz|c},k} = \mathbf{H}_{\mathrm{c},k}\mathbf{P}_{\mathrm{xc},k^-}^\mathsf{T}\mathbf{H}_{\mathrm{x},k}^\mathsf{T} + \left(\mathbf{H}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xc},k^-} + \mathbf{H}_{\mathrm{c},k}\mathbf{P}_{\mathrm{cc},k}\right)\mathbf{H}_{\mathrm{c},k}^\mathsf{T} + \mathbf{H}_{\mathrm{c},k}\mathbf{B}_{\mathrm{c},k} + \mathbf{B}_{\mathrm{c},k}^\mathsf{T}\mathbf{H}_{\mathrm{c},k}^\mathsf{T} \ . \tag{4.31}$$

The consider-parameter-to-innovation cross-covariance is

$$\mathbf{P}_{\mathrm{cz},k} = E\left\{(\mathbf{c}_k - \hat{\mathbf{c}}_k)\,\mathbf{z}_k^\mathsf{T}\right\} = \mathbf{P}_{\mathrm{xc},k^-}^\mathsf{T}\mathbf{H}_{\mathrm{x},k}^\mathsf{T} + \mathbf{P}_{\mathrm{cc},k^-}\mathbf{H}_{\mathrm{c},k}^\mathsf{T} + \mathbf{B}_{\mathrm{c},k} \ . \tag{4.32}$$

The measurement sensitivity matrices $\mathbf{H}_{\mathrm{x},k}$ and $\mathbf{H}_{\mathrm{c},k}$ are

$$\mathbf{H}_{\mathrm{x},k} = \left.\frac{\partial \boldsymbol{h}_k(\mathbf{x}_k, \mathbf{c}_k)}{\partial \mathbf{x}_k}\right|_{\hat{\mathbf{x}}_{k^-}, \hat{\mathbf{c}}_k} \quad \text{and} \quad \mathbf{H}_{\mathrm{c},k} = \left.\frac{\partial \boldsymbol{h}_k(\mathbf{x}_k, \mathbf{c}_k)}{\partial \mathbf{c}_k}\right|_{\hat{\mathbf{x}}_{k^-}, \hat{\mathbf{c}}_k} \ , \tag{4.33}$$

and the correlation matrices between measurement and process noise as (4.4) and (4.5).

Finally, the state covariance update law in (3.28) can be rewritten as (3.36),

$$\mathbf{P}_{\mathrm{xx},k^+} = \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{K}_k\mathbf{P}_{\mathrm{xz},k^-}^\mathsf{T} \ . \tag{4.34}$$

Exactly as in Section 3.4, step (3.37), the whole-state is updated with the estimated error-state after the correction step (4.18). The filter is then reset.

Figure 4.1 shows the overall filter algorithm data flow. As mentioned in the beginning of this section, the consider-parameter processes are entirely segregated from the main (standard-state) filter. Removal of the (dashed) covariance correction steps yields a regular Kalman filter set-up (cf. Figure 3.4). This new perspective on the Consider Kalman filter scheme provides the basis for a flexible implementation, where, for instance, the two filter parts (standard and consider) can be set up as distinct classes in object oriented programming (e.g., C++), as done in Chapter 6.

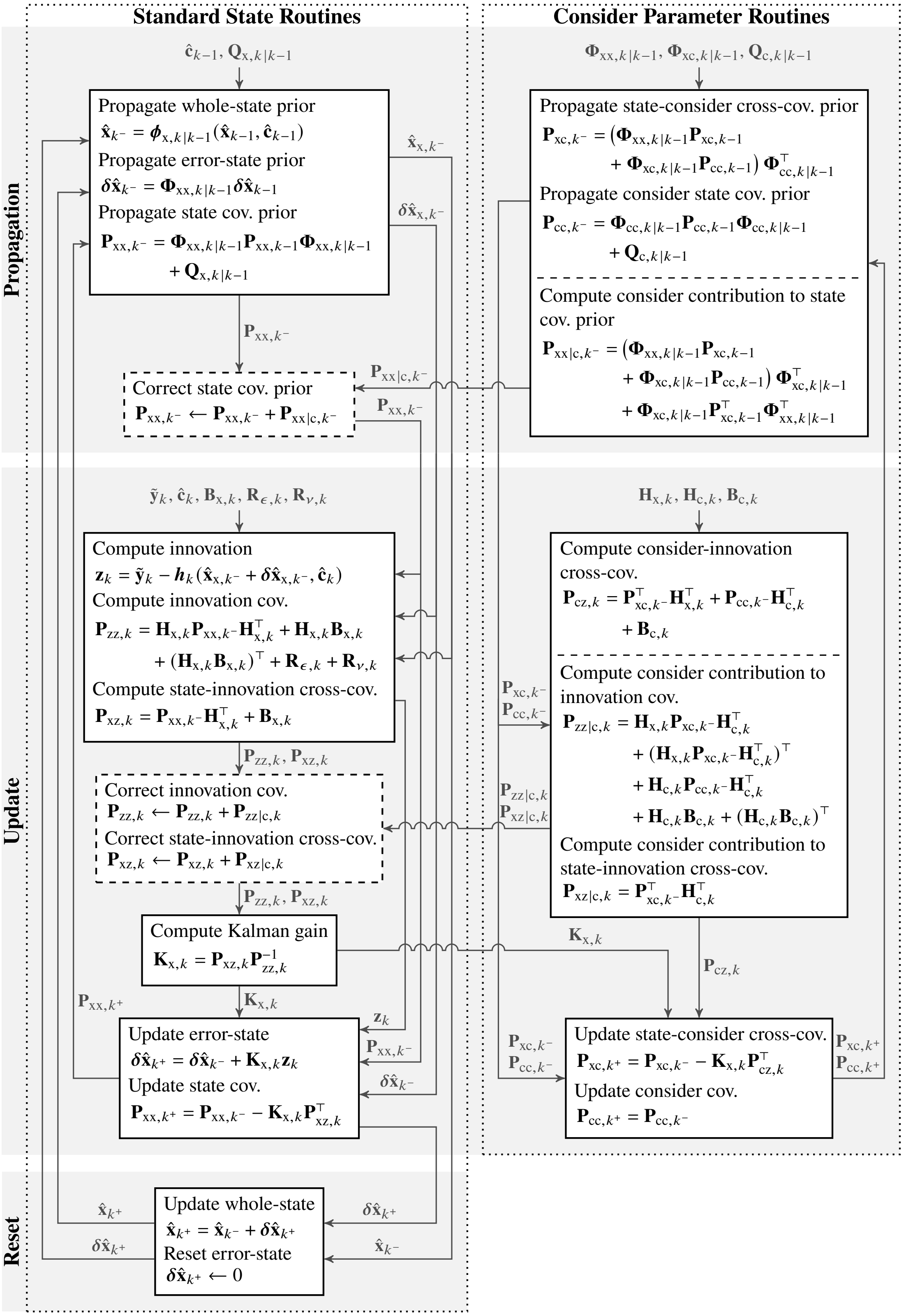

Standard State Routines
Consider Parameter Routines
Propagation
Update
Reset
$\hat{\mathbf{c}}_{k-1}, \mathbf{Q}_{\mathrm{x},k|k-1}$
Propagate whole-state prior
$\hat{\mathbf{x}}_{k^-} = \boldsymbol{\phi}_{\mathrm{x},k|k-1}(\hat{\mathbf{x}}_{k-1}, \hat{\mathbf{c}}_{k-1})$
Propagate error-state prior
$\delta\hat{\mathbf{x}}_{k^-} = \mathbf{\Phi}_{\mathrm{xx},k|k-1}\delta\hat{\mathbf{x}}_{k-1}$
Propagate state cov. prior
$\mathbf{P}_{\mathrm{xx},k^-} = \mathbf{\Phi}_{\mathrm{xx},k|k-1}\mathbf{P}_{\mathrm{xx},k-1}\mathbf{\Phi}_{\mathrm{xx},k|k-1} + \mathbf{Q}_{\mathrm{x},k|k-1}$
$\hat{\mathbf{x}}_{\mathrm{x},k^-}$
$\delta\hat{\mathbf{x}}_{\mathrm{x},k^-}$
$\mathbf{P}_{\mathrm{xx},k^-}$
Correct state cov. prior
$\mathbf{P}_{\mathrm{xx},k^-} \leftarrow \mathbf{P}_{\mathrm{xx},k^-} + \mathbf{P}_{\mathrm{xx|c},k^-}$
$\mathbf{P}_{\mathrm{xx|c},k^-}$
$\mathbf{P}_{\mathrm{xx},k^-}$
$\mathbf{\Phi}_{\mathrm{xx},k|k-1}, \mathbf{\Phi}_{\mathrm{xc},k|k-1}, \mathbf{Q}_{\mathrm{c},k|k-1}$
Propagate state-consider cross-cov. prior
$\mathbf{P}_{\mathrm{xc},k^-} = \left(\mathbf{\Phi}_{\mathrm{xx},k|k-1}\mathbf{P}_{\mathrm{xc},k-1} + \mathbf{\Phi}_{\mathrm{xc},k|k-1}\mathbf{P}_{\mathrm{cc},k-1}\right)\mathbf{\Phi}^{\top}_{\mathrm{cc},k|k-1}$
Propagate consider state cov. prior
$\mathbf{P}_{\mathrm{cc},k^-} = \mathbf{\Phi}_{\mathrm{cc},k|k-1}\mathbf{P}_{\mathrm{cc},k-1}\mathbf{\Phi}_{\mathrm{cc},k|k-1} + \mathbf{Q}_{\mathrm{c},k|k-1}$
Compute consider contribution to state cov. prior
$\mathbf{P}_{\mathrm{xx|c},k^-} = \left(\mathbf{\Phi}_{\mathrm{xx},k|k-1}\mathbf{P}_{\mathrm{xc},k-1} + \mathbf{\Phi}_{\mathrm{xc},k|k-1}\mathbf{P}_{\mathrm{cc},k-1}\right)\mathbf{\Phi}^{\top}_{\mathrm{xc},k|k-1} + \mathbf{\Phi}_{\mathrm{xc},k|k-1}\mathbf{P}^{\top}_{\mathrm{xc},k-1}\mathbf{\Phi}^{\top}_{\mathrm{xx},k|k-1}$
$\tilde{\mathbf{y}}_k, \hat{\mathbf{c}}_k, \mathbf{B}_{\mathrm{x},k}, \mathbf{R}_{\epsilon,k}, \mathbf{R}_{\nu,k}$
Compute innovation
$\mathbf{z}_k = \tilde{\mathbf{y}}_k - \boldsymbol{h}_k(\hat{\mathbf{x}}_{\mathrm{x},k^-} + \delta\hat{\mathbf{x}}_{\mathrm{x},k^-}, \hat{\mathbf{c}}_k)$
Compute innovation cov.
$\mathbf{P}_{\mathrm{zz},k} = \mathbf{H}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}^{\top}_{\mathrm{x},k} + \mathbf{H}_{\mathrm{x},k}\mathbf{B}_{\mathrm{x},k} + (\mathbf{H}_{\mathrm{x},k}\mathbf{B}_{\mathrm{x},k})^{\top} + \mathbf{R}_{\epsilon,k} + \mathbf{R}_{\nu,k}$
Compute state-innovation cross-cov.
$\mathbf{P}_{\mathrm{xz},k} = \mathbf{P}_{\mathrm{xx},k^-}\mathbf{H}^{\top}_{\mathrm{x},k} + \mathbf{B}_{\mathrm{x},k}$
$\mathbf{P}_{\mathrm{zz},k}, \mathbf{P}_{\mathrm{xz},k}$
Correct innovation cov.
$\mathbf{P}_{\mathrm{zz},k} \leftarrow \mathbf{P}_{\mathrm{zz},k} + \mathbf{P}_{\mathrm{zz|c},k}$
Correct state-innovation cross-cov.
$\mathbf{P}_{\mathrm{xz},k} \leftarrow \mathbf{P}_{\mathrm{xz},k} + \mathbf{P}_{\mathrm{xz|c},k}$
$\mathbf{P}_{\mathrm{zz},k}, \mathbf{P}_{\mathrm{xz},k}$
Compute Kalman gain
$\mathbf{K}_{\mathrm{x},k} = \mathbf{P}_{\mathrm{xz},k}\mathbf{P}^{-1}_{\mathrm{zz},k}$
$\mathbf{K}_{\mathrm{x},k}$
$\mathbf{P}_{\mathrm{xx},k^+}$
$\mathbf{z}_k$
$\mathbf{P}_{\mathrm{xx},k^-}$
$\delta\hat{\mathbf{x}}_{k^-}$
Update error-state
$\delta\hat{\mathbf{x}}_{k^+} = \delta\hat{\mathbf{x}}_{k^-} + \mathbf{K}_{\mathrm{x},k}\mathbf{z}_k$
Update state cov.
$\mathbf{P}_{\mathrm{xx},k^+} = \mathbf{P}_{\mathrm{xx},k^-} - \mathbf{K}_{\mathrm{x},k}\mathbf{P}^{\top}_{\mathrm{xz},k}$
$\mathbf{H}_{\mathrm{x},k}, \mathbf{H}_{\mathrm{c},k}, \mathbf{B}_{\mathrm{c},k}$
Compute consider-innovation cross-cov.
$\mathbf{P}_{\mathrm{cz},k} = \mathbf{P}^{\top}_{\mathrm{xc},k^-}\mathbf{H}^{\top}_{\mathrm{x},k} + \mathbf{P}_{\mathrm{cc},k^-}\mathbf{H}^{\top}_{\mathrm{c},k} + \mathbf{B}_{\mathrm{c},k}$
Compute consider contribution to innovation cov.
$\mathbf{P}_{\mathrm{zz|c},k} = \mathbf{H}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xc},k^-}\mathbf{H}^{\top}_{\mathrm{c},k} + (\mathbf{H}_{\mathrm{x},k}\mathbf{P}_{\mathrm{xc},k^-}\mathbf{H}^{\top}_{\mathrm{c},k})^{\top} + \mathbf{H}_{\mathrm{c},k}\mathbf{P}_{\mathrm{cc},k^-}\mathbf{H}^{\top}_{\mathrm{c},k} + \mathbf{H}_{\mathrm{c},k}\mathbf{B}_{\mathrm{c},k} + (\mathbf{H}_{\mathrm{c},k}\mathbf{B}_{\mathrm{c},k})^{\top}$
Compute consider contribution to state-innovation cross-cov.
$\mathbf{P}_{\mathrm{xz|c},k} = \mathbf{P}^{\top}_{\mathrm{xc},k^-}\mathbf{H}^{\top}_{\mathrm{c},k}$
$\mathbf{P}_{\mathrm{xc},k^-}$
$\mathbf{P}_{\mathrm{cc},k^-}$
$\mathbf{P}_{\mathrm{zz|c},k}$
$\mathbf{P}_{\mathrm{xz|c},k}$
$\mathbf{K}_{\mathrm{x},k}$
$\mathbf{P}_{\mathrm{cz},k}$
$\mathbf{P}_{\mathrm{xc},k^-}$
$\mathbf{P}_{\mathrm{cc},k^-}$
Update state-consider cross-cov.
$\mathbf{P}_{\mathrm{xc},k^+} = \mathbf{P}_{\mathrm{xc},k^-} - \mathbf{K}_{\mathrm{x},k}\mathbf{P}^{\top}_{\mathrm{cz},k}$
Update consider cov.
$\mathbf{P}_{\mathrm{cc},k^+} = \mathbf{P}_{\mathrm{cc},k^-}$
$\mathbf{P}_{\mathrm{xc},k^+}$
$\mathbf{P}_{\mathrm{cc},k^+}$
$\hat{\mathbf{x}}_{k^+}$
$\delta\hat{\mathbf{x}}_{k^+}$
Update whole-state
$\hat{\mathbf{x}}_{k^+} = \hat{\mathbf{x}}_{k^-} + \delta\hat{\mathbf{x}}_{k^+}$
Reset error-state
$\delta\hat{\mathbf{x}}_{k^+} \leftarrow 0$
$\delta\hat{\mathbf{x}}_{k^+}$
$\hat{\mathbf{x}}_{k^-}$

Figure 4.1: Data flow diagram of the proposed error-state Consider Kalman filter scheme with separation between standard-state and consider-parameter processes. Note that the consider parameter $\hat{\mathbf{c}}$ is either kept constant or propagated according to (4.3).

4.3 Filter propagation

Filter propagation is done as described in Section 3.7 with the following change: The linearized high-rate whole-state dynamics (3.47) is now

$$\begin{aligned} \delta\mathbf{x}_{j+1} &= \boldsymbol{\phi}_{\mathrm{x},j+1|j}(\hat{\mathbf{x}}_j + \delta\mathbf{x}_j, \mathbf{c}_j) - \boldsymbol{\phi}_{\mathrm{x},j+1|j}(\hat{\mathbf{x}}_j, \hat{\mathbf{c}}_j) + \mathbf{w}_{\mathrm{x},j} && (4.35)\\ &\approx \boldsymbol{\Phi}_{\mathrm{xx},j+1|j}\delta\mathbf{x}_j + \boldsymbol{\Phi}_{\mathrm{xc},j+1|j}\left(\mathbf{c}_j - \hat{\mathbf{c}}_j\right) + \mathbf{w}_{\mathrm{x},j}\ , && (4.36)\end{aligned}$$

$\boldsymbol{\Phi}_{\mathrm{xx},k+1|k}$ and $\mathbf{Q}_{\mathrm{xx},k+1|k}$, with $t_k = t_{j=0}$ and $t_{k+1} = t_{j=N}$, are formed as in (3.48)–(3.49), being $\boldsymbol{\Phi}_{\mathrm{cc},k+1|k}$ and $\mathbf{Q}_{\mathrm{c},k+1|k}$ computed in a similar fashion. The state transition sensitivity to the consider-parameter set $\boldsymbol{\Phi}_{\mathrm{xc},k+1|k}$ is computed recursively as

$$\boldsymbol{\Phi}_{\mathrm{xc},j+1|0} \approx \boldsymbol{\Phi}_{\mathrm{xc},j+1|j}\boldsymbol{\Phi}_{\mathrm{xc},j|0} + \boldsymbol{\Phi}_{\mathrm{xc},j+1|j}\boldsymbol{\Phi}_{\mathrm{cc},j|0}\ . \qquad (4.37)$$

4.4 Parameter impact and observability analysis

The high-order inertial sensor model, (3.43)–(3.44), used for calibration within the filter has a total of 33 parameters (11 uncertainty types, each with 3 components), making it computationally costly in its entirety. In the following points the impact on the propagated kinematics and observability of each parameter of each sensor grade are assessed along several Vega launch trajectories. The resulting insight is used for order reduction and state allocation.

Trajectories of Vega flights VV02, VV03 and VV04 are used in the parametric impact and observability analyses in this section with data covering a period from lift-off (T0) up to T0+900 s (see Appendix B). This includes all manoeuvres up to the end of the first burn of the Vega upper stage. Mostly coasting takes place thereafter.

It is worth noting that the constant thrust direction (in a body-fixed frame) and the higher rotational dynamics around longitudinal axis (spinning motion) result, as shall be shown, in uneven observability and impact between uncertainty directions.

4.4.1 Parameter impact analysis

The impact of each (body-fixed) component of each inertial sensor model parameter, (3.43)–(3.44), on the open-loop inertial integration is estimated through covariance analysis. The trajectories followed are those mentioned in the previous point with initial uncertainties as in Table 2.1. The parameters investigated are bias (on/off + rate/acc. RW), thermal residual, scale-factor (on/off + RW), misalignment and non-orthogonality of both gyro and accelerometer, and the gyro's G-sensitive bias. Figures 4.2, 4.3 and 4.4 display for the low-, medium- and high-grade sensors, respectively, the isolated position and attitude 1σ error norm caused by each component of each sensor parameter. Tables 4.1 and 4.2 present the final values of such attitude and position error, respectively, for each source of each sensor grade together with the total root sum of squares (RSS) per direction, error type and sensor. Angle and velocity RW, modeled as filter process noises, are included for comparison.

In the gyro, longitudinal scale-factor has the largest impact in all grades; in medium and high-grade units it is by far the strongest source, and only in the low-grade sensor are thermal residual and longitudinal G-sensitive bias comparable. Lateral components of G-sensitive bias cause virtually no error in any grade. Impacts of misalignments and non-orthogonalities are well distributed across axes, with similar overall magnitudes. As pointed in [187], this occurs despite the lower non-orthogonality uncertainty (see Table 2.1), stemming from its signal magnification nature. In the accelerometer, due to the mostly longitudinal specific-force, lateral scale-factor components and longitudinal misalignment and non-orthogonality cause negligible error. On/off bias has the largest contribution in all grades. Overall, and as observed in [187], gyro uncertainties dominate the position budget, underlining the importance of its grade.

Table 4.1: Total 1σ attitude error (in deg) for each gyro uncertainty source (per sensor axis and class), under open-loop propagation for 900 s (worst case of three trajectories). Gray-scale between zero and sensor RSS.

	Low				Medium				High			
	x	y	z	RSS	x	y	z	RSS	x	y	z	RSS
Angle RW	0.075	0.075	0.075	0.130	0.005	0.005	0.005	0.009	0.002	0.002	0.002	0.004
Bias (on/off + RW)	0.197	0.199	0.264	0.385	0.040	0.040	0.053	0.077	0.003	0.003	0.004	0.006
Thermal Residual Bias	0.774	0.782	1.046	1.518	0.077	0.078	0.105	0.152	0.002	0.002	0.002	0.003
G-sensitive Bias	0.001	0.001	0.712	0.712	0.000	0.000	0.024	0.024	0.000	0.000	0.002	0.002
Scale-factor (on/off + RW)	0.113	0.087	1.349	1.357	0.045	0.035	0.540	0.543	0.003	0.003	0.040	0.041
Misalignment	0.105	0.078	0.078	0.153	0.031	0.024	0.024	0.046	0.006	0.005	0.005	0.009
Non-orthogonality	0.079	0.091	0.067	0.138	0.024	0.027	0.020	0.041	0.005	0.005	0.004	0.008
RSS	0.821	0.824	1.873	2.205	0.106	0.101	0.554	0.573	0.010	0.009	0.041	0.043

Table 4.2: Total 1σ position error (in km) for each gyro and accel. uncertainty source (per sensor axis and class), under open-loop propagation for 900 s (worst of three trajectories). Gray-scale between zero and sensor RSS.

	Low				Medium				High			
	x	y	z	RSS	x	y	z	RSS	x	y	z	RSS
						Gyro						
Angle RW	2.579	2.822	1.696	4.183	0.172	0.188	0.113	0.279	0.086	0.094	0.057	0.139
Bias (on/off + RW)	1.902	1.861	3.142	4.118	0.367	0.361	0.604	0.794	0.031	0.031	0.052	0.068
Thermal Residual Bias	7.704	7.509	12.760	16.690	0.770	0.751	1.276	1.669	0.015	0.015	0.026	0.033
G-sensitive Bias	0.028	0.016	17.873	17.873	0.001	0.001	0.596	0.596	0.000	0.000	0.060	0.060
Scale-factor (on/off + RW)	4.774	4.917	37.576	38.196	1.909	1.967	15.028	15.276	0.143	0.147	1.127	1.146
Misalignment	5.693	4.794	4.795	8.853	1.708	1.438	1.438	2.656	0.342	0.288	0.288	0.531
Non-orthogonality	3.988	4.176	2.246	6.196	1.196	1.253	0.674	1.859	0.239	0.251	0.135	0.372
Gyro RSS	11.862	11.507	43.989	46.991	2.958	2.870	15.190	15.739	0.451	0.421	1.175	1.327
						Accelerometer						
Velocity RW	1.717	1.640	1.467	2.791	0.172	0.164	0.147	0.279	0.017	0.016	0.015	0.028
Bias (on/off + RW)	4.601	4.778	7.380	9.922	0.998	1.038	1.617	2.166	0.200	0.208	0.323	0.433
Thermal Residual Bias	1.987	2.066	3.218	4.309	0.199	0.207	0.322	0.431	0.020	0.021	0.032	0.043
Scale-factor (on/off + RW)	0.002	0.001	2.456	2.456	0.001	0.001	1.228	1.228	0.000	0.000	0.307	0.307
Misalignment	2.810	2.521	0.005	3.775	0.843	0.756	0.002	1.133	0.169	0.151	0.000	0.227
Non-orthogonality	1.405	1.267	0.003	1.892	0.422	0.380	0.001	0.568	0.084	0.076	0.000	0.114
Accel. RSS	6.159	6.144	8.544	12.194	1.398	1.365	2.061	2.840	0.276	0.269	0.447	0.591

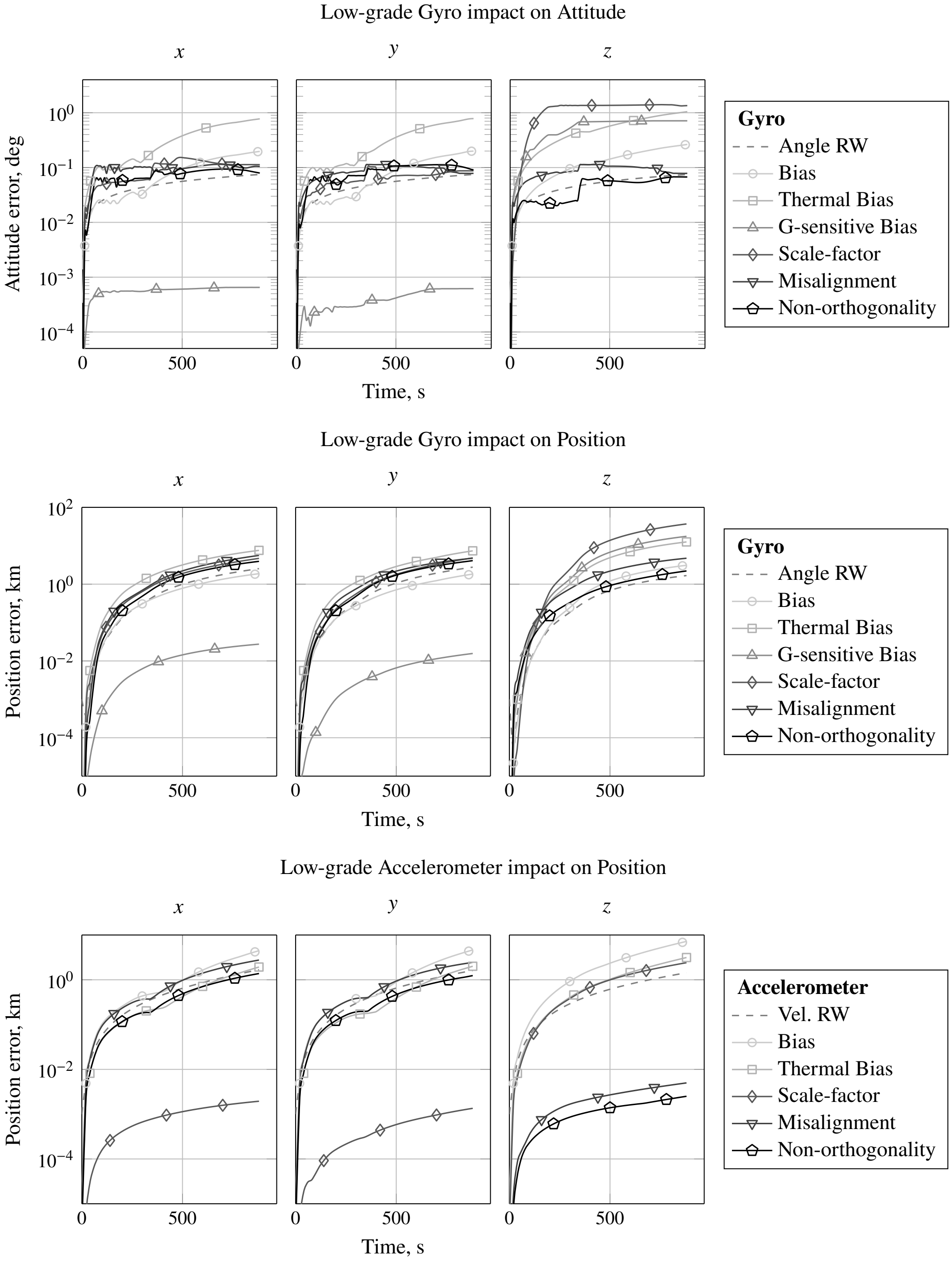


Figure 4.2: Position and attitude 1σ error norm caused by the **Low-grade** Gyro and Accelerometer uncertainties (per source per axis) **in flight** after alignment

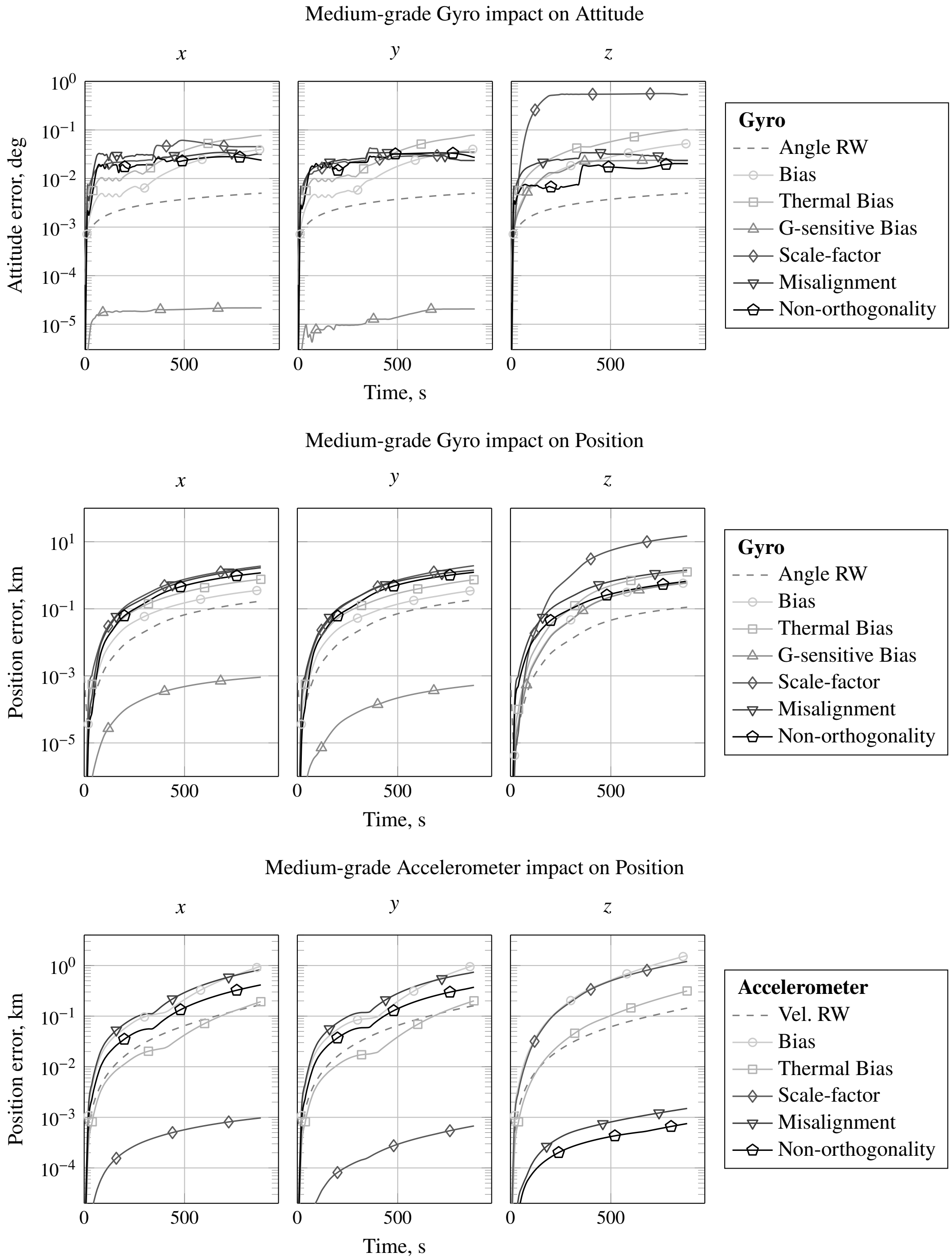


Figure 4.3: Position and attitude 1σ error norm caused by the **Medium-grade** Gyro and Accelerometer uncertainties (per source per axis) **in flight** after alignment

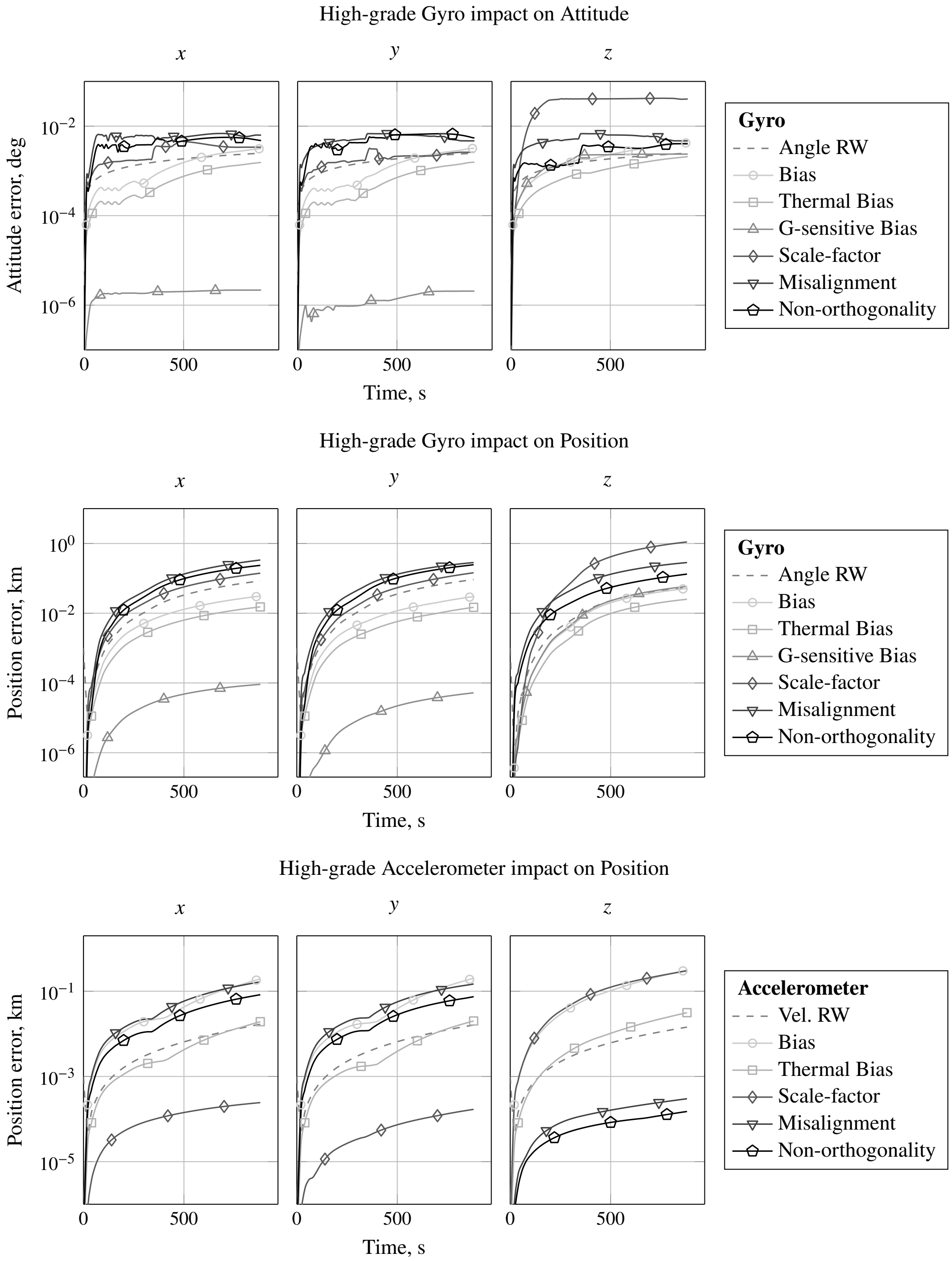


Figure 4.4: Position and attitude 1σ error norm caused by the **High-grade** Gyro and Accelerometer uncertainties (per source per axis) **in flight** after alignment

4.4.2 Observability and the Parametric Cramér-Rao Bound

System observability can be assessed through the rank of the observability matrix [153] or, in time-variant and non-linear systems, of the observability Gramian [88]. This rank test yields a binary result on the observability of the entire system; however, if rank deficient, singular value decomposition of the observability matrix/Gramian can be used to identify the unobservable state combinations [75]. Similarly, the relative magnitude of the Gramian diagonal elements tells of the relative degree of observability of the corresponding states. This, however, does not account for loss of information through process noise. Alternatively, the degree of observability can be assessed through filter covariance evolution [72, 11]: decreasing covariance indicates state observability, while constant or increasing trends reveal weakness or absence thereof. A notable exception occurs in states that are rotated upon propagation, thus having their uncertainty transferred between axes (e.g., attitude error states expressed in body axes). This is however not the case for the sensor states here analysed.

The Parametric Cramér-Rao Bound (PCRB) [167] is used to provide a floor for the achievable filter state covariance $\mathbf{P}$ at each epoch k and thus a measure of the best-case observability along a reference trajectory. This bound is given by

$$\mathbf{P}_k \geq \mathbf{P}_k^\star \,. \tag{4.38}$$

The best-case covariance matrix $\mathbf{P}^\star$ is computed recursively as [18]

$$\mathbf{P}_{k+1}^\star = \mathbf{\Phi}_{k+1|k} \left(\mathbf{P}_k^\star - \mathbf{P}_k^\star \mathbf{H}_k^\mathsf{T} \left(\mathbf{H}_k \mathbf{P}_k^\star \mathbf{H}_k^\mathsf{T} + \mathbf{R}_k \right)^{-1} \mathbf{H}_k \mathbf{P}_k^\star \right) \mathbf{\Phi}_{k+1|k}^\mathsf{T} + \mathbf{Q}_k \tag{4.39}$$

where the matrices $\mathbf{\Phi}_{k+1|k}$ and $\mathbf{H}_k$ are as in Section 3.4 but evaluated at the true state $\mathbf{x}_k^\star$, i.e.,

$$\mathbf{\Phi}_{k+1|k} = \left. \frac{\partial \boldsymbol{\phi}_{\mathrm{x},k+1|k}(\mathbf{x}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k^\star} , \qquad \mathbf{H}_k = \left. \frac{\partial \boldsymbol{h}_k(\mathbf{x}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k^\star} . \tag{4.40}$$

$\mathbf{x}_k^\star$ includes all sensor parameters. $\mathbf{Q}$ and $\mathbf{R}$ are the nominal process and measurement noise covariance matrices.

The Cramér-Rao lower bound is used in [47] for the state observability analysis of a vehicle during entry, descent and landing. In addition, both this covariance metric and the framework here developed to compute it have been used in [141, 149] to estimate the navigation performance of a similar hybrid navigation system intended to fly on a reusability technology demonstrating sounding rocket flight.

4.4.3 Parameter observability analysis

Before take-off the navigation system goes through a fine alignment period on the launch pad. As described in Section 2.4.4, this process uses the static translational and rotational conditions of the system (in an ECEF frame) as a constraint on the navigation filter. The zero velocity and Earth rate update models are as shown in Section 3.8.

Figure 4.5 shows the evolution of the PCRB (as ratio to that of the uncorrected, free-running system) for each inertial sensor state component of the filter model (3.43)–(3.44) for 1800 s of static alignment. Only the static pseudo-measurements mentioned are used during the alignment. Table 4.3 displays the final values as ratios to the (1σ) uncertainty of the unaided free-running system, i.e., that with no measurement updates. For the constant states (without process noise) this is equivalent to the normalization by the initial covariance matrix proposed in [72], and used in [36] in the observability analysis of the alignment of an inertial system. Initial uncertainties are as in Table 2.1.

The alignment period reduces accelerometer on/off bias uncertainty in all grades, especially along the longitudinal direction, having little effect on thermal residual bias. Conversely, gyro thermal residuals of low and medium units are well observed. Given the similar signature of thermal and on/off biases, filter observability is shared between these sources, with the higher uncertainty seeing a larger reduction. The high-grade gyro's thermal residual is relatively small with respect to its on/off bias, so observability is inverted in this case. Longitudinal G-sensitive bias is only reasonably reduced in the low-grade gyro. All remaining states show little or no observability in this phase. Note that observability levels of IMU states tends to reduce with increasing grade.

Figures 4.6, 4.7 and 4.8 display the same PCRB ratios for the low-, medium- and high-grade sensor parameters, respectively, along the trajectories mentioned in the beginning of Section 4.4. Position and velocity measurements of GNSS-like accuracy are assumed available at 1 Hz throughout. Final alignment uncertainties are used as initial condition. Table 4.4 presents, for all sensor grades, the lowest (best-case) final uncertainty ratio between flights.

The plotted trends in Figures 4.6–4.8 show a remarkable concordance between flights; small differences in gyro uncertainties are due to differences in angular rate histories (see Appendix B). The flight dynamics render observable all accelerometer bias components of all grades, further reducing post-alignment uncertainty levels. Accelerometer thermal residuals, which are relatively smaller than on/off biases, are far less observable. Well observed are also the z-components of accelerometer and gyro scale-factors. The former due to the strong longitudinal acceleration, and the latter through the vehicle's spinning motion. Misalignment and non-orthogonality uncertainties are only partially observable.

4.4.4 State allocation

Filter order reduction can be done with support from the results gathered thus far. The basic principles employed are:

- Parameters with little impact are neglected; even if observable, the performance gained by any reduction in uncertainties with little impact is small or negligible.
- The higher impact states are allocated according to their expected observability:
  - Those with reasonable observability are allocated to explicit estimation.
  - Partially or non-observable states are handled as *consider* parameters, as their estimation carries little to no benefit.

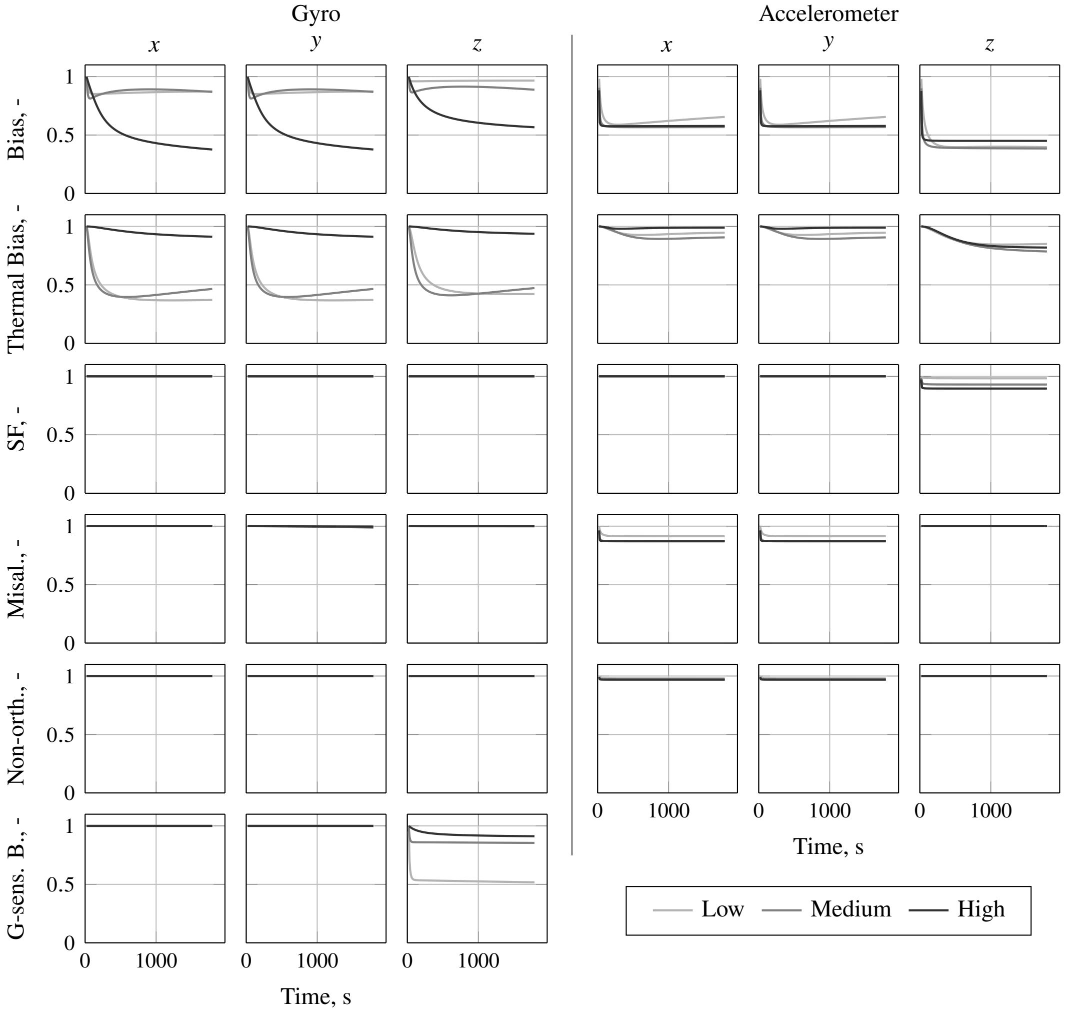


Figure 4.5: Gyro and Accelerometer parametric observability **during alignment** for different sensor grades (Low, Medium and High). Ratio of PCRB to the uncorrected 1σ uncertainty.

Table 4.3: Ratio of PCRB to the uncorrected 1σ uncertainty, for each sensor grade, after 1800 s of static alignment on the launch pad. Darker cell color indicates higher observability.

	Low			Medium			High		
	x	*y*	*z*	*x*	*y*	*z*	*x*	*y*	*z*
				Gyro					
Bias (on/off + RW)	0.873	0.873	0.966	0.868	0.868	0.887	0.376	0.376	0.566
Thermal Residual Bias	0.371	0.371	0.422	0.466	0.466	0.473	0.912	0.912	0.937
G-sensitive Bias	1.000	1.000	0.517	1.000	1.000	0.855	1.000	1.000	0.912
Scale-factor (on/off + RW)	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Misalignment	1.000	0.999	1.000	1.000	0.988	1.000	1.000	0.996	0.999
Non-orthogonality	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
				Accelerometer					
Bias (on/off + RW)	0.655	0.655	0.396	0.568	0.568	0.385	0.578	0.578	0.450
Thermal Residual Bias	0.946	0.946	0.850	0.907	0.906	0.786	0.990	0.990	0.820
Scale-factor (on/off + RW)	1.000	1.000	0.983	1.000	1.000	0.930	1.000	1.000	0.895
Misalignment	0.914	0.914	1.000	0.870	0.870	1.000	0.872	0.872	1.000
Non-orthogonality	0.979	0.979	1.000	0.969	0.969	1.000	0.970	0.970	1.000

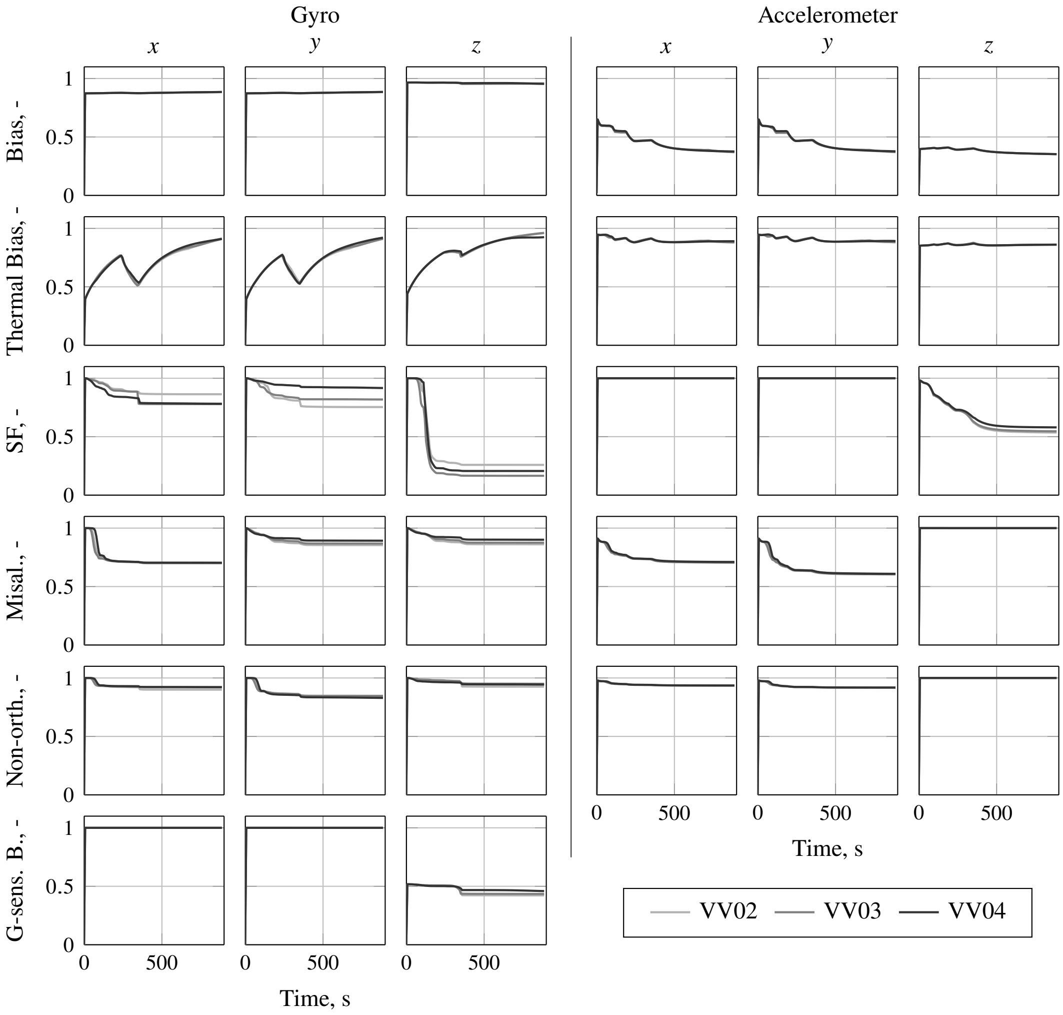


Figure 4.6: Low-grade Gyro and Accelerometer parametric observability **in flight** (three trajectories) after alignment. Ratio of PCRB to the uncorrected 1σ uncertainty.

Table 4.4: Ratio of PCRB to the uncorrected 1σ uncertainty, for each sensor grade, after 900 s of flight. Minimum value over three flights. Darker cell color indicates higher observability.

	Low			Medium			High		
	x	*y*	*z*	*x*	*y*	*z*	*x*	*y*	*z*
				Gyro					
Bias (on/off + RW)	0.884	0.884	0.954	0.881	0.882	0.904	0.384	0.387	0.558
Thermal Residual Bias	0.911	0.912	0.925	0.925	0.927	0.943	0.992	0.993	0.996
G-sensitive Bias	1.000	1.000	0.422	1.000	1.000	0.832	1.000	1.000	0.903
Scale-factor (on/off + RW)	0.779	0.753	0.166	0.413	0.401	0.050	0.781	0.732	0.118
Misalignment	0.699	0.855	0.862	0.507	0.748	0.796	0.524	0.860	0.884
Non-orthogonality	0.900	0.830	0.926	0.823	0.567	0.762	0.833	0.552	0.757
				Accelerometer					
Bias (on/off + RW)	0.369	0.370	0.352	0.162	0.168	0.124	0.102	0.103	0.080
Thermal Residual Bias	0.880	0.882	0.861	0.874	0.890	0.792	0.955	0.959	0.868
Scale-factor (on/off + RW)	1.000	1.000	0.536	1.000	1.000	0.134	0.999	0.999	0.082
Misalignment	0.704	0.604	1.000	0.528	0.509	0.998	0.536	0.511	0.998
Non-orthogonality	0.935	0.917	1.000	0.905	0.902	1.000	0.904	0.900	1.000

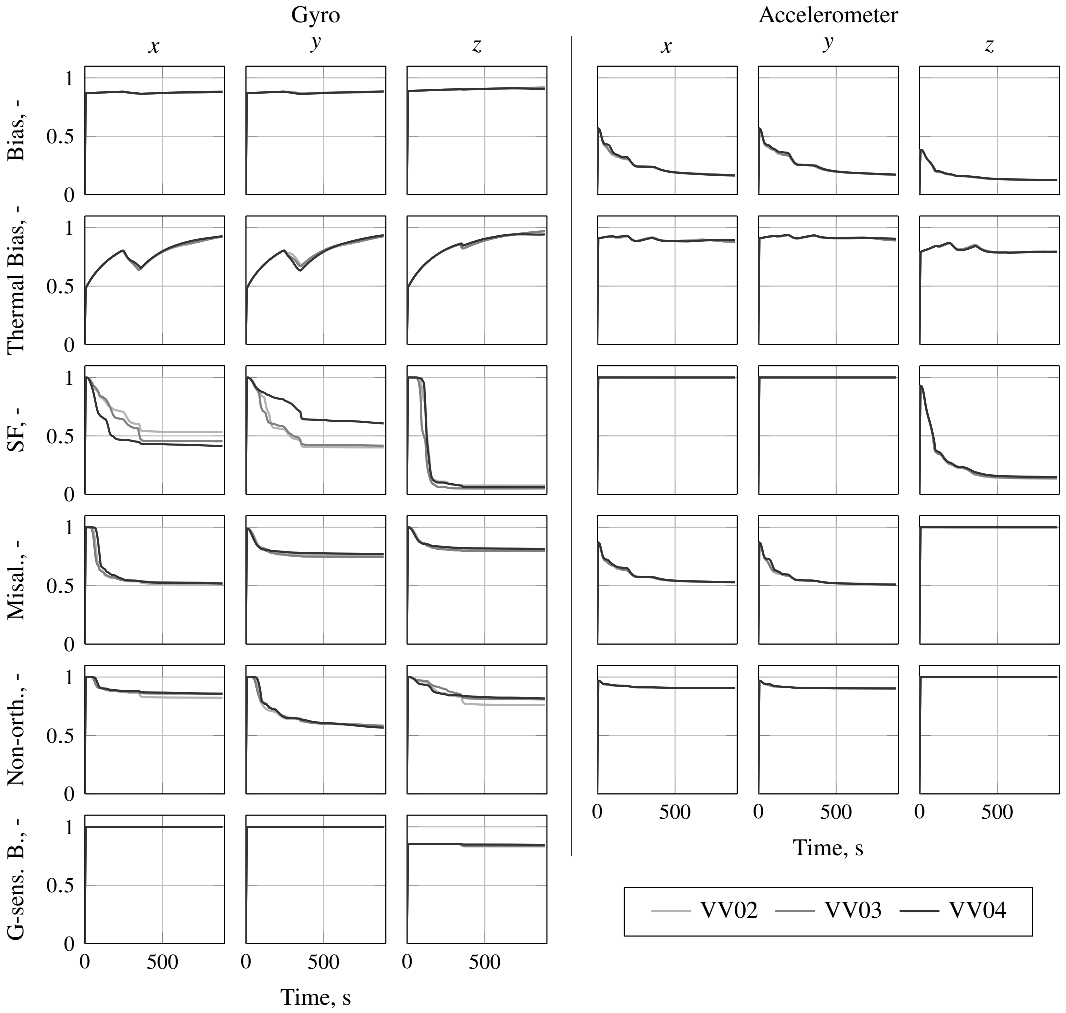


Figure 4.7: Medium-grade Gyro and Accelerometer parametric observability **in flight** (three trajectories) after alignment. Ratio of PCRB to the uncorrected 1σ uncertainty.

Following this logic, the proposed filter model state sets for each grade of gyro and accelerometer are shown in Table 4.5. Note that as partially observable states become considered parameters the observability of states with similar signature (e.g., misalignments and non-orthogonalities) may change, potentially requiring additional design iterations. In addition, as attitude is observed through simultaneous measuring of acceleration in body-fixed (by the accelerometer) and ECEF frames (from consecutive GNSS measurements), observability of gyro states is heavily influenced by accelerometer performance. In practice, this limits accelerometer model state reduction.

Thermal residual bias in the medium and low-grade accelerometers is approximated by a random walk process, which is merged with the random walk component of the on/off bias, thus requiring no additional state nor considered parameter. This approximation, equivalent to shifting the time constant of the GM process to infinity, is valid when the variance of the GM process noise is of the same order of magnitude of the rate RW (bias drift). The prominence of the GM process noise over the rate RW in the medium and low-grade gyros makes this approximation overly conservative in the

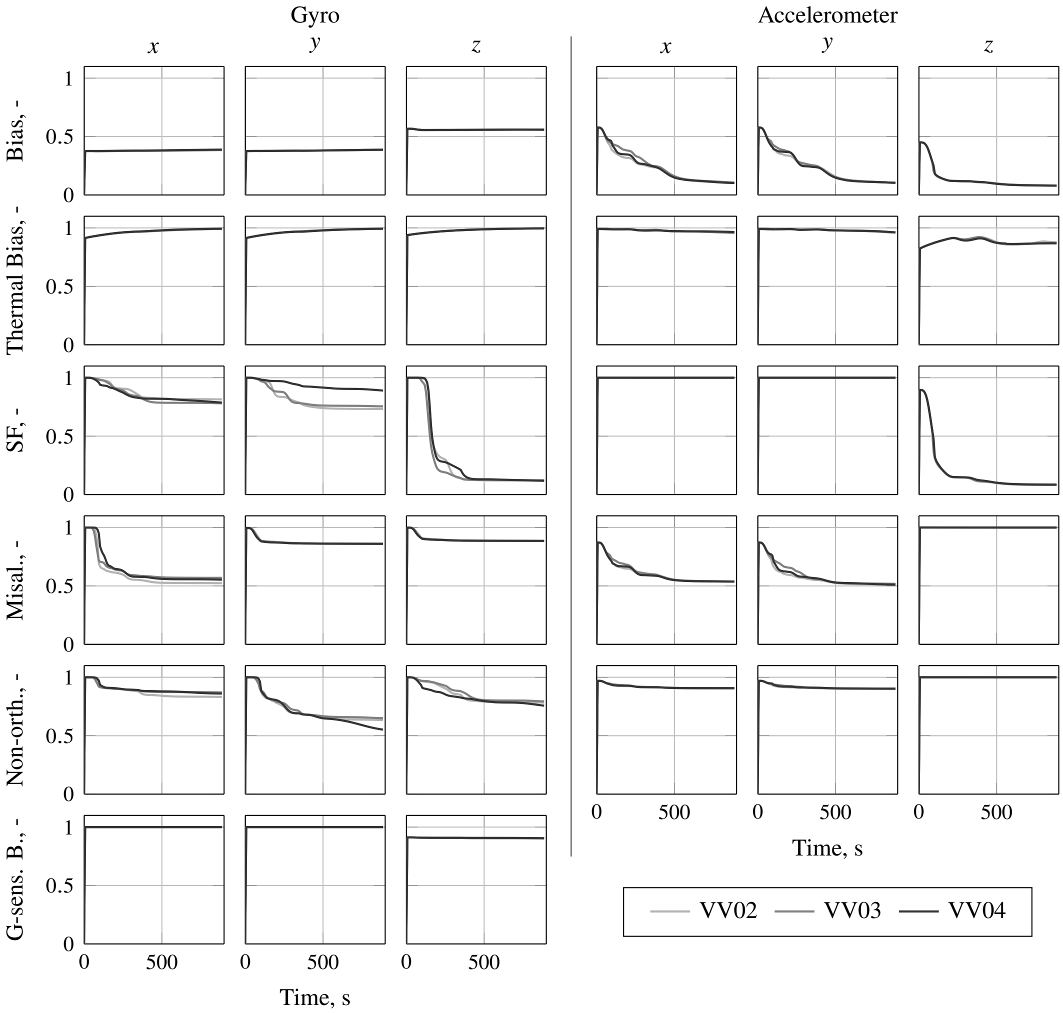


Figure 4.8: High-grade Gyro and Accelerometer parametric observability **in flight** (three trajectories) after alignment. Ratio of PCRB to the uncorrected 1σ uncertainty.

long-term. Their thermal bias is thus explicitly modeled, having as initial condition the on/off bias level; rate RW is neglected. These approximations are easily worked out through Allan Variance analysis of the sensors' noises (see Section C.4).

Differences in parametric impact and observability between sensor grades render different state configurations, especially in gyro model. The high-grade sensor, for instance, has several neglectable sources, requiring fewer states and consider-parameters.

In addition to the inertial sensor states, all GNSS channel biases of pseudorange and TDCP, described in Sections 3.9.5 and 3.9.6, respectively, are handled as *consider* parameters as their observability was found to be low, particularly during flight. The delayed position state used for the TDCP updates is also treated as a *consider* parameter as its *a posteriori* value is unnecessary, being promptly overwritten by a state reset (see Section 3.9.6). The total estimated state counts in Table 4.5 add to the 9 kinematics states (velocity, position, and attitude) and the GNSS model state (receiver clock bias), yielding filter orders of 20, 21, and 19 for the Low-, Medium-, and High-grade configurations, respectively, with 38, 37, and 35 consider parameters.

Table 4.5: Filter IMU model (2.5)–(2.6) state allocation, per sensor grade, as estimated (E), considered (C), or neglected (N)

	Low			Medium			High		
	x	*y*	*z*	*x*	*y*	*z*	*x*	*y*	*z*
	Gyro								
Bias (on/off+RW)	C*	C*	C*	C*	C*	C*	N	N	N
Thermal Residual Bias	C	C	C	C	C	C	N	N	N
G-sensitive Bias	N	N	E	N	N	C	N	N	N
Scale-Factor (on/off+RW)	C	C	E	E	E	E	C	C	E
Misalignment	E	E	C	E	E	C	E	E	C
Non-orthogonality	C	C	C	C	C	C	C	C	C
	Accelerometer								
Bias (on/off+RW)	E	E	E	E	E	E	E	E	E
Thermal Residual Bias	C*	C*	C*	C*	C*	C*	N	N	N
Scale-Factor (on/off+RW)	N	N	E	N	N	E	N	N	E
Misalignment	E	E	N	E	E	N	E	E	N
Non-orthogonality	C	C	N	C	C	N	C	C	N
Total Estimated	10			11			9		
Total Considered	11			10			8		

* Considered by tuning of another state; no consider-parameter slot required

4.4.5 Computational load analysis

Coarse analysis of the number of Floating-Point Operationss (FLOPs) required by the filter algorithm derived in Section 4.2 can be done based on the total count for each type of matrix operation (e.g., [78]). A FLOP is here defined as either one scalar multiplication or one scalar addition. Counts for each matrix operation relevant to the filter implementation are given in Table 4.6.

In the full-order, 70-state filter (with 9 kinematics states, 33 IMU states, a GNSS clock state, 12 pseudorange bias, 12 TDCP bias states, and 3 delayed antenna position states) updated with 12 pseudorange and 12 TDCP measurements, the considering of one state leads to about 0.5% FLOP count saving per propagation + update cycle, while neglecting a single state saves about 2.4%. Following the same analysis process, the reduced-order filter configurations described earlier in the previous section (Table 4.5) result in 48.8%, 48.3% and 54.9% relative FLOP count reductions for the Low-, Medium- and High-class IMU filters, respectively. Complete breakdown of FLOP count per filter and per filter step (see Figure 4.1) is presented in Table 4.7. Note that this analysis accounts for the sparse structure of the system transition matrix $\boldsymbol{\Phi}$ described in (3.50), with diagonal $\boldsymbol{\Phi}_{\mathsf{imu}}^{\mathsf{imu}}$ and $\boldsymbol{\Phi}_{\mathsf{meas}}^{\mathsf{meas}}$ sub-matrices. The matrix inversion in step (4.22) is accomplished through Cholesky decomposition of $\mathbf{P}_{zz}$ followed by forward and backward substitutions [78].

4.5 Performance evaluation

This section tests the proposed navigation system, in its different sensor grade versions, in model- and hardware-in-the-loop set-ups along flight trajectories obtained from real-flight measurements. Performance comparisons are drawn against loosely coupled and inertial-only architectures. The results herein have been partially published in [184].

Table 4.6: FLOP count expressions for several matrix operations [78, 41]

Operation	Expression	FLOP count
Matrix-Matrix product	$\mathbf{AC}$	$ml(2n-1)$
Diagonal Matrix product	$\mathbf{AD}$	mn
Matrix product, symmetric output	$\mathbf{AA}^\top$	$\frac{1}{2}m(m+1)(2n-1)$
Matrix addition	$\mathbf{A}+\mathbf{B}$	mn
Cholesky decomposition	$\mathbf{P}=\mathbf{LL}^\top$	$\frac{1}{3}m^3+\frac{1}{2}m^2+\frac{1}{6}m$
Forward/Backward substitution	$\mathbf{L}^{-1}\ \mathbf{C},\ \mathbf{L}^{-\top}\mathbf{C}$	$n^2 l$
QR factorization	$\mathbf{A}=\mathbf{U}[\mathbf{L}\ \ \mathbf{0}^\top]^\top$	$2mn^2-\frac{2}{3}n^3,\ m\geq n$

Legend: $\mathbf{A},\mathbf{B}\in\mathbb{R}^{m\times n}$, $\mathbf{C}\in\mathbb{R}^{n\times l}$, $\mathbf{D}\in\mathbb{R}^{n\times n}$ diagonal, $\mathbf{L}\in\mathbb{R}^{m\times m}$ lower triangular, $\mathbf{P}\in\mathbb{R}^{m\times m}$ positive definite, $\mathbf{U}\in\mathbb{R}^{m\times m}$ orthogonal

Table 4.7: FLOP count for each step of each filter algorithm implementation: full order, low grade, medium grade and high grade IMU versions

Step	Output	Expression	FLOP count			
			Full	Low	Medium	High
	Propagation					
Propagate error-state	$\delta\hat{\mathbf{x}}_{k+1}$	(4.9)	808	344	363	325
Propagate state cov. prior	$\mathbf{P}_{\mathrm{xx}\mid\mathrm{x},(k+1)^-}$	(4.11)	65298	10070	10997	9183
Propagate state-consider cross-cov. prior	$\mathbf{P}_{\mathrm{xc},(k+1)^-}$	(4.13)	0	15358	15394	13230
Propagate consider state cov. prior	$\mathbf{P}_{\mathrm{cc},(k+1)^-}$	(4.14)	0	2332	2235	2053
Consider contribution to state cov. prior	$\mathbf{P}_{\mathrm{xx}\mid\mathrm{c},(k+1)^-}$	(4.12)	0	13275	12258	8928
Correct state cov. prior	$\mathbf{P}_{\mathrm{xx},(k+1)^-}$	(4.10)	0	180	189	171
Total propagation			66106	41559	41436	33890
	Update					
Innovation	$\mathbf{z}_k$	(4.24)	3360	960	1008	912
Innovation cov.	$\mathbf{P}_{\mathrm{zz}\mid\mathrm{x},k}$	(4.30)	80928	23328	24480	22176
State-innovation cross-cov.	$\mathbf{P}_{\mathrm{xz}\mid\mathrm{x},k}$	(4.27)	235200	19200	21168	17328
Consider-innovation cross-cov.	$\mathbf{P}_{\mathrm{cz},k}$	(4.32)	0	105792	103008	90720
Consider contribution to innovation cov.	$\mathbf{P}_{\mathrm{zz}\mid\mathrm{c},k}$	(4.31)	0	43776	42624	40320
Consider contribution to state-innovation cross-cov.	$\mathbf{P}_{\mathrm{xz}\mid\mathrm{c},k}$	(4.28)	0	36000	36792	31464
Correct innovation cov.	$\mathbf{P}_{\mathrm{zz},k}$	(4.29)	0	288	288	288
Correct state-innovation cross-cov.	$\mathbf{P}_{\mathrm{xz},k}$	(4.26)	0	480	504	456
Kalman gain	$\mathbf{K}_{\mathrm{x},k}$	(4.22)	41620	12820	13396	12244
Update error-state	$\delta\hat{\mathbf{x}}_{k^+}$	(4.18)	3360	960	1008	912
Update state cov.	$\mathbf{P}_{\mathrm{xx},k^+}$	(4.34)	235200	19200	21168	17328
Update state-consider cross-cov.	$\mathbf{P}_{\mathrm{xc},k^+}$	(4.20)	0	36480	37296	31920
Update consider cov.	$\mathbf{P}_{\mathrm{cc},k^+}$	(4.21)	0	0	0	0
Total update			599668	299284	302740	266068
Total (% savings to Full)			665774 –	340843 **(48.8)**	344176 **(48.3)**	299958 **(54.9)**

4.5.1 Trajectory and simulation models

The trajectory used in the tests to follow corresponds to the Vega VV02 flight (see Appendix B). Smoothed real-flight data is used to produce inertial and GNSS measurements. The GNSS raw measurements and receiver solutions are either produced by a simulation model of the DLR Phoenix-HD GPS receiver or by the receiver itself fed by a GNSS emulator (SPIRENT GSS7700) running the true trajectory. The inertial measurements are simulated through a high-fidelity model of the form of that described Section 2.4.3 and introduced in detail in Appendix Section C.2.

4.5.2 Nominal performance

Full-order filter

Filter performance and general behavior is this time assessed through Monte Carlo (MC) simulation. An alignment period of 500 sec precedes the (VV02) launch, with GNSS measurement updates starting 50 sec before lift-off.

Figure 4.9 shows the MC root mean square (RMS) and maximum error of the designed baseline full-order filters for each inertial sensor grade. The filter predicted 1σ and 3σ covariance bounds are shown for comparison. The agreement between both RMS and filter predicted 1σ bound is good, validating the filter model structure in the face of high-fidelity inertial and GNSS signals. The velocity errors in the medium and low-grade versions oscillate visibly, largely due to the high accelerometer velocity RW of these units. Position error, mostly set by the GNSS positioning accuracy, is nearly identical in all three versions. Attitude performance differs considerably between grades, as expected. The slight difference between the MC RMS and filter 1σ in the attitude of the low-grade filter stems from the (neglected) gyro scale-factor non-linearity, which is relatively high in this sensor class. The MC maximum error also approaches well the filter 3σ bounds in all states and all grades, further proving the robustness of the baseline design. Note that, under this trajectory, the designed system (in all grades) satisfies the total position and velocity performance guidelines set in Section 2.2. Clearly, the lower dynamics of the larger vehicle (cf. Figure 1.1) contribute to a significantly easier attitude estimation (cf. Figure 3.7) with only the low-grade set-up violating the performance guideline for this state (set in Section 2.2).

In Appendix E further plots for this scenario are shown with kinematics state and inertial sensor state MC error evolutions per axis. As for the kinematics states, the inertial sensor state predicted and actual error trends indicate coherent behavior.

Reduced-order filter

As for the full-order, the reduced order designs are evaluated in Monte Carlo simulation. Figure 4.10 shows, for each inertial grade, the RMS and maximum kinematics errors along all runs compared to the corresponding predicted 1σ and 3σ filter covariance bounds. Once again, the agreement between predicted and actual error bounds is good, revealing coherent filter behavior. Note that the more conservative nature of the consider filter makes the attitude mismatch previously seen in the low-grade full-order filter (Figure 4.9) mostly vanish. Further plots for this test case can be found in

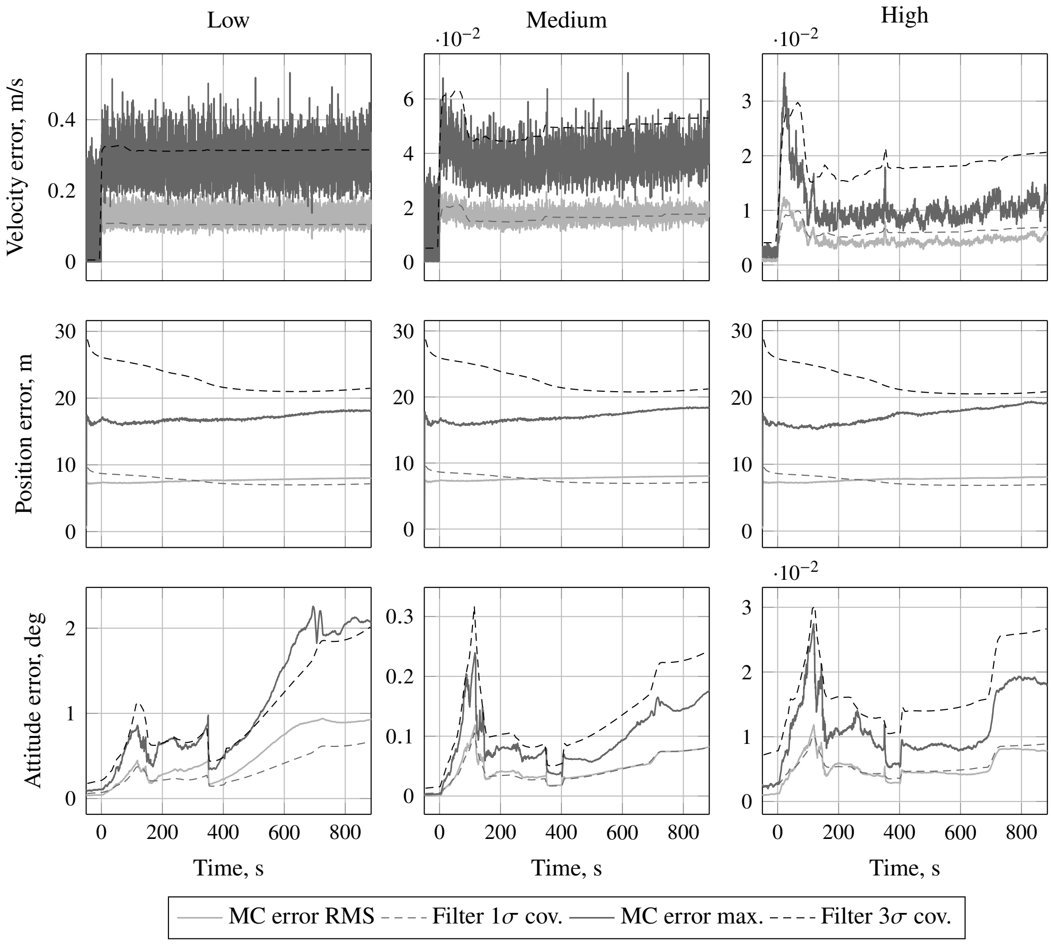


Figure 4.9: Total kinematics state error in the **full-order** filters of each inertial sensor grade. MC RMS and maximum bounds *vs* predicted 1σ and 3σ filter cov.. Nominal conditions along a Vega VV02 trajectory.

Appendix E, showing kinematics and inertial sensor state MC error evolutions.

In Figure 4.11, the predicted 1σ bounds for the kinematics states of the reduced-order filters are compared to the corresponding full-order Parametric Cramér-Rao Bound (PCRB) (see Section 4.4.2). For all grades, velocity error closely approaches the PCRB. In terms of position, a gap can be seen between reduced-order covariance and PCRB. This arises from *considering* the GNSS channel biases. In attitude, performance approaches the lower bound with only a small offset attributed to the gyro non-orthogonalities (not explicitly estimated). Avoiding explicit estimation of gyro and accelerometer misalignments (i.e., *considering* them), as in [187], would free four state slots but cause a ~50% error increase in both attitude (throughout) and velocity (during engine burns). Overall, the proposed designs follow closely the *best-case* bound, demonstrating the effectiveness of the order reduction.

As an additional demonstration of the value of the achieved reduced-order design, Figure 4.12 shows the MC error bounds of a medium-grade reduced-order design similar to that described in Table 4.5 but with all consider-parameters neglected. It estimates gyro scale factor (all axes) and x- and y-axis misalignment, and accelerometer bias (all

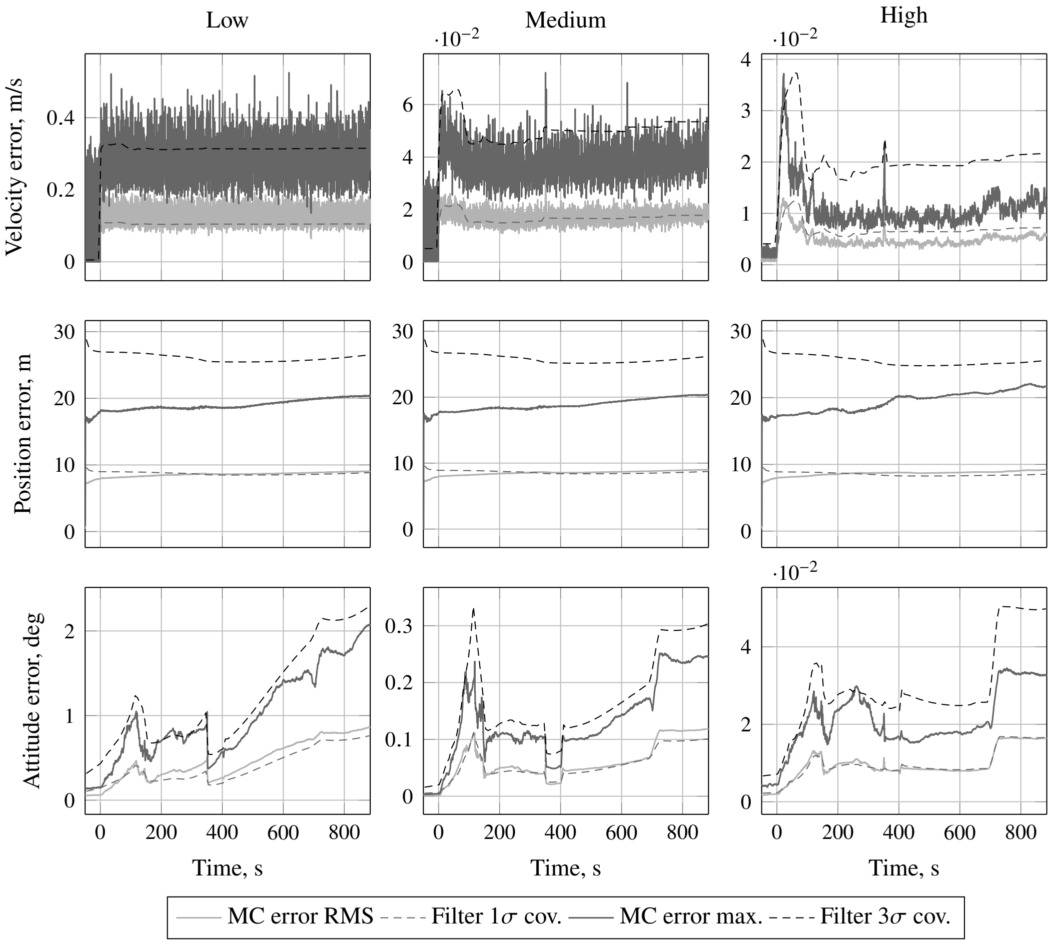


Figure 4.10: Total kinematics state error in **reduced-order** filters (all inertial sensor grades). MC RMS and maximum bounds *vs* predicted 1σ and 3σ filter cov.. Nominal conditions on Vega VV02 trajectory.

axes), z-axis scale-factor and x- and y-axis misalignment. Comparatively to the reduced-order order *consider* design, it neglects gyro biases (on/off+RW, thermal residual and G-sensitive), gyro z-axis misalignment, all gyro non-orthogonalities, accelerometer thermal residual bias and accelerometer non-orthogonalities. The plotted curves for position and especially attitude error show a severely incoherent behavior.

4.5.3 GNSS outage performance and comparison to loose coupling

The harsh launch environment can cause GNSS receiver tracking loops to lose lock, leading to partial or complete signal drop-out. This has been observed in all Vega flights carrying a GNSS receiver (VV02–VV11) upon 1st-stage and fairing separations.

In this section the estimation performance of the designed medium-grade reduced-order tightly-coupled system is compared to that of an equivalent loosely-coupled scheme under partial outages with 1, 2 and 3 satellites tracked. Figure 4.13 shows the kinematics states error and covariance under such condition on a Vega VV02 trajectory.

The partial outages last for 60 s starting at T0+125 s. This test uses GNSS raw

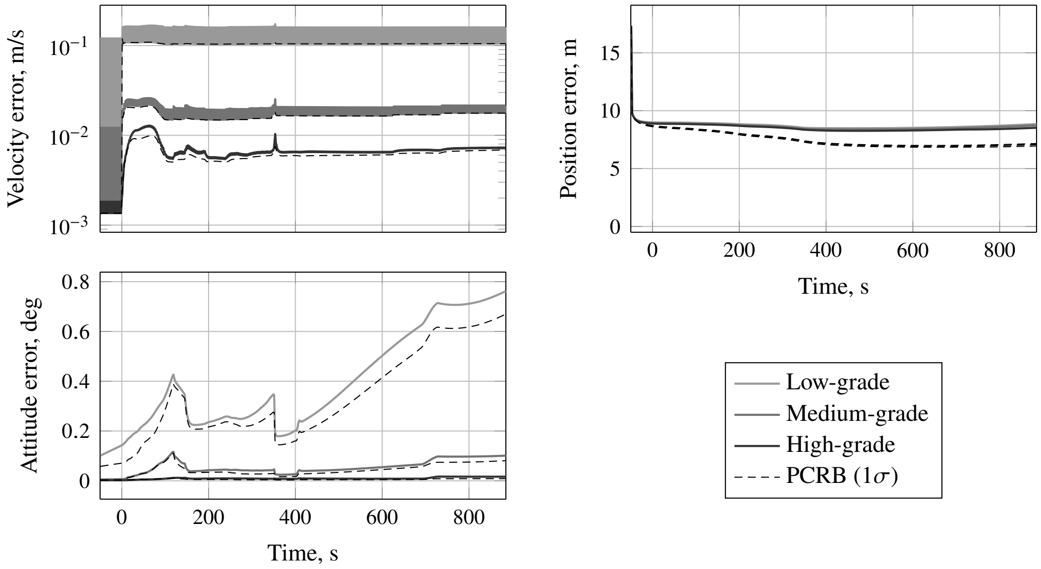


Figure 4.11: Reduced-order filter cov. *vs* full-order equivalent PCRB for each inertial sensor grade

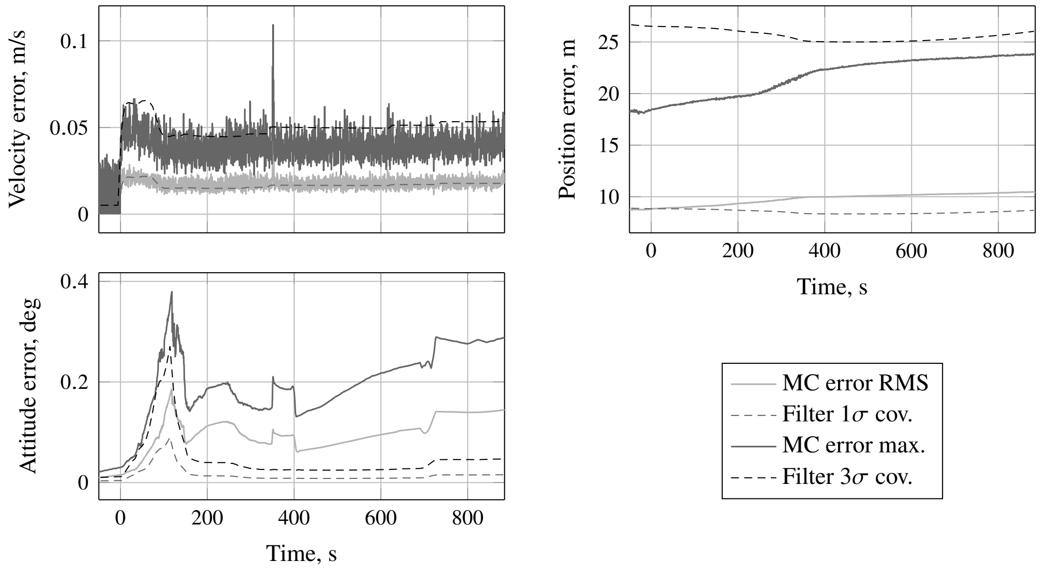


Figure 4.12: Total kinematics state error of a filter similar to the medium-grade reduced-order design (see Table 4.5) but neglecting all consider parameters. MC RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Nominal conditions along a Vega VV02 trajectory.

measurements and receiver solutions produced by the DLR Phoenix GPS receiver fed by a GNSS emulator (SPIRENT GSS7700) running the true trajectory. Inertial measurements are simulated as in the previous section.

The loosely coupled filter is updated with receiver position and velocity, which require the tracking of at least four satellites signals. The partial outages simulated therefore result in complete absence of GNSS solution. This leads to a stronger divergence of the loose coupling during this period in comparison to the tight scheme, especially under 2 and 3 tracked satellites. Differences are particularly flagrant in velocity and position, with the unaided loose filter reaching about 1.5 m/s and 37 m (1σ) of error. (Although not displayed for brevity, the tightly coupled set-up reaches a similar divergence under complete outage.) Attitude, as only partially observable, is less affected by the different outages.

As the receiver at hand uses carrier-phase-smoothed pseudorange and pseudorange-rate measurements to produce its navigation solution, the nominal position and velocity errors of the loose set-up are (almost) on par with those of the proposed filter. However, although model-based Tropospheric correction inside the receiver has been enabled in the loose set-up tested, a 0.2-m/s velocity error hump is visible during early ascent (T0+0 s to T0+70 s) reaching far beyond filter predicted covariance (detail in Figure 4.13). This is caused by the leftover correction residuals, which in the designed system are compensated as described in Section 3.9.3. Corroborating the early results in Figure 3.6, this shows that neglecting such residuals clearly compromises filter coherence.

4.5.4 Long-term performance and inertial-only solution

A long-term comparison between the designed filter on its medium-grade version and medium and high-grade inertial-only solutions is presented in Figure 4.14. The inertial and GNSS measurements are produced as in the preceding point. The VV02 trajectory now extends up to Proba-V delivery at T0+3327 s (refer to Chapter B for the altitude profile). During high-altitude coasting phases typical inertial platforms resort to a null-specific-force approximation, foregoing accelerometer data to avoid sensor error (e.g., [17]). In contrast, the hybrid system, having GNSS information available throughout (delivery altitude, ~820 km, is well below GNSS constellation) maintains accelerometer-based propagation to ensure continuous sensor calibration.

The hybrid system position and velocity performances are fairly constant throughout the flight, being bounded by the GNSS information accuracy. This is inline with the hybrid navigation behavior reported by Boulade et al. [23] for an Ariane-V scenario. The achieved accuracy on the translational states represents an improvement of more than 3 orders of magnitude with respect to the inertial propagation with the same sensor and 2 with respect to the grade above (high-grade). In terms of attitude, the aided system departs from its inertial equivalent and approaches that of the high-grade propagation during the first set of engine burns (T0<650 s). During the coasting that follows attitude drifts (about 0.5 deg/h) due to gyro bias to a peak of 0.36 deg (1σ) at the 2nd AVUM (Attitude Vernier Upper Module) ignition (T0+3185). This burn grants a realignment to about 0.16 deg at payload delivery. The high-grade sensor does not drift nearly as much, providing 0.045 deg at delivery. The Delta-V equivalent to the 1σ navigation errors at delivery are 20.59 m/s and 2.29 m/s for the medium and high-grade inertial

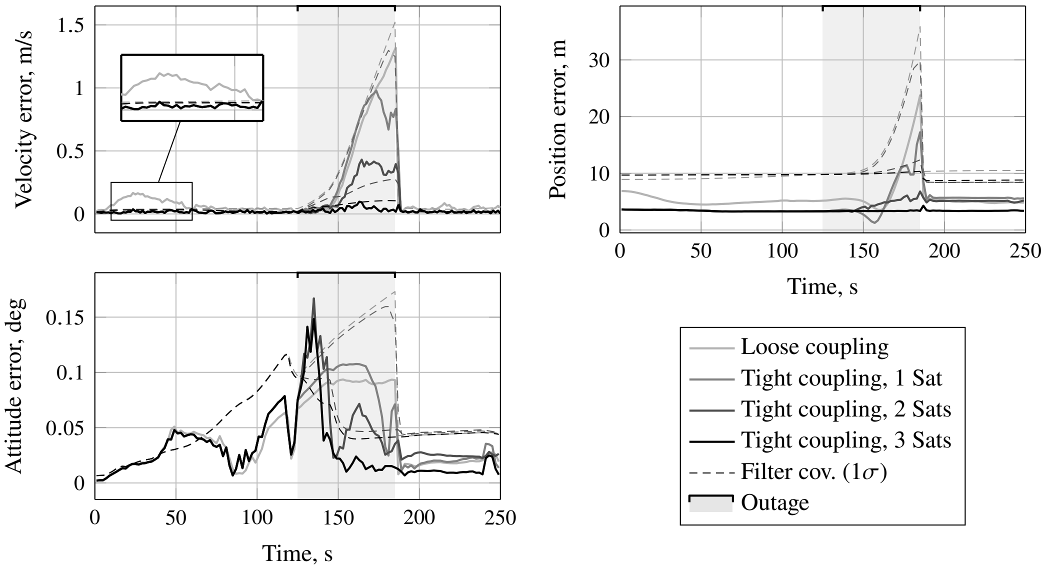


Figure 4.13: Single run performance of medium-grade tightly coupled filter compared to the equivalent loose integration under partial outages with 1, 2 and 3 satellites still tracked

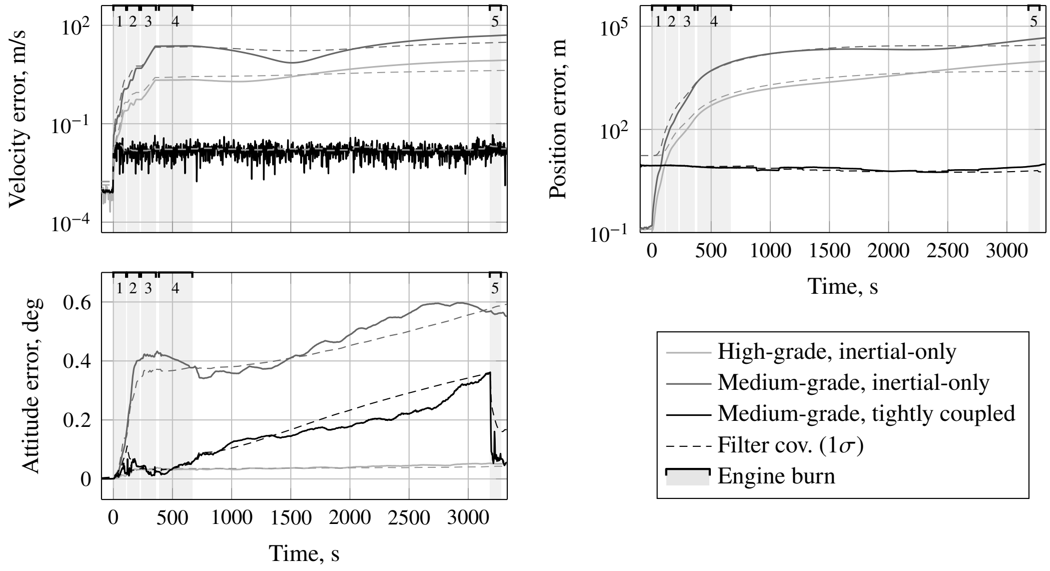


Figure 4.14: Single run comparison between tightly coupled system with medium-grade inertial sensor and the inertial-only architectures based on medium and high-grade sensors. Engine burns: 1-P80, 2-Z23, 3-Z9, 4-AVUM1, 5-AVUM2.

solutions, respectively, and 0.02 m/s for the hybrid system. These values are obtained with pseudoinverse of Gauss's variational equations (see, e.g., [15, 26]) applied to the final navigation covariance matrix written in terms of Keplerian elements. Guidance and control errors are not accounted for.

4.5.5 Discussion

As discussed in Section 1.1, trajectory maintenance and orbital delivery dispersions are strongly influenced by navigation performance. To compensate, launcher inertial-only navigation systems employ accurate, costly sensors. The results presented in this chapter show considerable improvements of the hybridized solution over an inertial-only one. Recall from Section 2.2 that new GNSS-based navigation systems for launchers should, at least, match the performance of the currently used inertial-only systems. While the hybrid system in all inertial sensor grades far outperforms the tested high-grade inertial-only solution in terms of position and velocity, the attitude accuracy of the medium-grade-based hybrid system approaches that of this inertial-only setup during engine burns. This is when accurate pointing is most needed [102]. Attainable delivery orbit accuracy, and its share from navigation, can only be determined through closed-loop analysis of the full GNC chain, which is beyond the scope of this work. The cost savings associated with an eventual downgrading of the inertial sensor may still far compensate the additional cost of the GNSS receiver. In fact, as demonstrated in the OCAM-G initiative (which included the Phoenix-HD), commercial receivers without space hardening and with only minor software changes can be suitable for launcher navigation [25, 73]. While GNSS may be vulnerable to external disturbances, the inertial data in the hybrid set-up ensure solution continuity, as shown in Section 4.5.3.

Concerning the type of hybrid architecture, tight and loose couplings yield comparable nominal performances (see Section 4.5.3). System complexity is also on a par: both may use the same class of GNSS receiver (most commercial units output raw data), computing unit, and inertial sensor. Software-wise, the tight system is slightly more complex as it requires a GNSS satellite ephemeris solution routine (e.g., [69]) and range-based filter update models (see Section 3.9). The simpler GNSS solution (loose) updates are described in Section D.1. Treating GNSS channel biases (tight coupling) and receiver solution biases (loose coupling) as consider parameters yields a filter GNSS model order difference of a single state (receiver clock bias). Considerable differences, however, arise in robustness and availability; the several advantages of the tight architecture have already been explained in Section 2.3.1, the most prominent for the present application being the availability at reduced constellation (see Section 4.5.3).

5 Fault Detection, Isolation and Recovery Design

This chapter presents the Fault Detection, Isolation and Recovery (FDIR) algorithm designed for the navigation system under development. The top-level model of the algorithm here described has been published in [181, 182].

5.1 Motivation and design guidelines

As mentioned in Section 2.2, stringent integrity requirements are commonly applied to launch vehicle navigation system design. Fault detection and contingency is, in many cases, accomplished through redundancy of inertial units. Although effective, this usually represents significant additional cost and mass.

The use of GNSS data in a hybrid navigation scheme provides some degree of information redundancy, which can be used to continuously monitor the health of the filter and its measurement sources [69]. However, GNSS receiver measurements occasionally contain outliers of their own; single-channel failures, phase cycle-slips and SV/receiver clock discontinuities are all possible occurrences [20]. These failures (or anomalies) may be of fleeting or permanent nature, and of variable severity. Additionally, due to high vehicle dynamics, loss of GNSS signal is rather likely during certain phases of launch. During such outages the navigation relies solely on inertial measurements, working in a dead-reckoning. When GNSS sensing is resumed the system must evaluate both: if the new measurements are sane; and if the kinematics states propagated by the strapdown have diverged beyond predicted bounds during the outage. Failure to identify any one of these anomalies might lead to severe navigation system accuracy degradation or even long-term filter divergence.

Consider, as an example, a tightly coupled hybrid navigation system using PR measurements to correct a high-rate strapdown inertial algorithm in a Kalman filter configuration. To prevent against GNSS errors, the system promptly rejects (very) large

PR measurement innovation outliers. (This is a common strategy used, for instance, in [161, 147].) Assume that, in flight, a GNSS signal outage occurs during which the system undergoes violent shocks, vibration and other dynamic effects not considered in the filter design. As a result, at GNSS signal reacquisition, the real kinematics state error is far beyond the filter-predicted 1σ bounds. The large measurement innovations of all PR channels caused by this filter inconsistency are then interpreted as GNSS outliers and promptly rejected. The same happens in subsequent epochs, preventing any real use of GNSS measurements thereafter. and ultimately causing permanent filter divergence and near-complete navigation failure.

This severe example is surprisingly plausible if such contingencies are not accounted for. FDIR strategies are then extremely important not only for maintaining system accuracy under large disturbances but also to prevent system failure in case of severe inconsistencies.

The design of these safeguard schemes, however, faces two important trade-offs in the case of real-time, highly dynamic applications. These are:

- Real-time capability sets a strict ceiling on computational burden, generally excluding complex/intensive, but often more effective and robust, FDIR schemes;
- Measurements are scarce and carry valuable information, therefore failure detection thresholds should be high enough so that false-alarm rates do not compromise availability, but not too high as to allow significant events to sieve through.

Finally, and inline with the latter point, the design of a multiple-test or test-cascade approach should strive to minimize of paths (and their likelihood) leading to inconclusive fault identification results.

5.2 Failure modes

In [20] an intensive list of possible GNSS/inertial failure modes is compiled, varying in likelihood and magnitude. In an effort to maintain the proposed FDIR scheme simple, the most important failures are grouped into three different signatures:

1. large single GNSS channel blunder;
2. large GNSS channel common mode bias discontinuity;
3. large filter kinematics state divergence with respect to GNSS solution.

These failure signatures cover GNSS faults such as SV clock discontinuity/reset, phase cycle-slip on TDCP observables, receiver channel malfunction, and receiver clock (bias or drift) discontinuity. On the INS side, severe kinematics solution drift (with respect to predicted filter covariance) caused for instance by extreme vehicle dynamics (vibration, shock and coning/sculling) is also covered by the above set.

5.3 Preliminaries

The following points describe the basics of the FDIR approach proposed for the navigation system at hand.

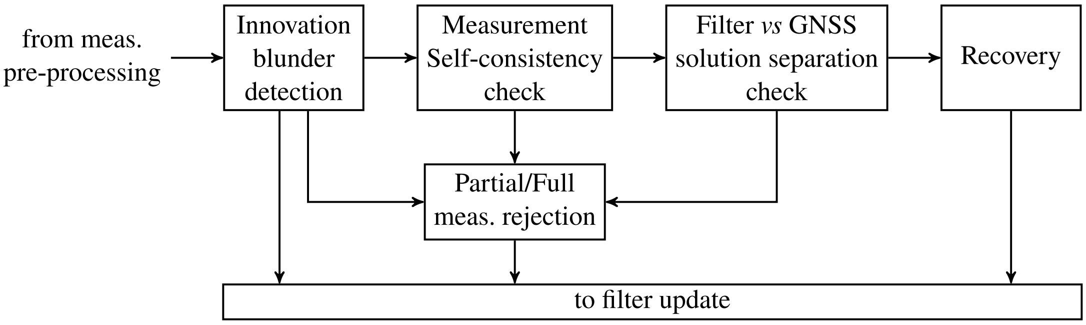

from meas.
pre-processing
Innovation blunder detection
Measurement Self-consistency check
Filter *vs* GNSS solution separation check
Recovery
Partial/Full meas. rejection
to filter update

Figure 5.1: Top-level FDIR concept

5.3.1 FDIR strategy

Given the design trade-offs stated in the introductory section and the considered failure modes presented in the previous point, a strategy to detect faults, identify their cause (as much as possible) and recover the system may follow the steps depicted on Figure 5.1. There, a cascade of three detection/identification steps is used, which includes an innovation test, a measurement mutual consistency check, and a filter state *vs* GNSS-only solution separation check. These have the potential to detect, and even identify, different anomalies. Innovation is sensitive to all considered failures, thus being the strongest detection source and the first quantity to be tested. However, it can only be used as identifier in the case of single channel failure (where only one innovation component violates the threshold). Mutual inconsistency of the GNSS channel measurements can indicate that one or more channels have oddly high errors. This can serve as way to distinguish between receiver channel faults and filter *vs* GNSS offset. The latter failure is confirmed by evaluating the actual separation against its predicted covariance level. In case a wide gap is found, recovery of the filter should be performed to fix its covariance limits.

This general strategy is applicable to PR, PRR and TDCP observables.

5.3.2 Measurement innovation

The general model of measurement innovation is written in (3.30) as difference between the actual measurement and the predicted one, as

$$\mathbf{z}_k = \tilde{\mathbf{y}}_k - \hat{\mathbf{y}}_k = \boldsymbol{h}(\hat{\mathbf{x}} + \delta\mathbf{x}) - \boldsymbol{h}(\hat{\mathbf{x}} + \delta\hat{\mathbf{x}}) + \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k \ , \tag{5.1}$$

which, linearizing the measurement function $\boldsymbol{h}(\mathbf{x})$ around the current filter estimate $\hat{\mathbf{x}}$, gives

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{e}_k + \boldsymbol{\upsilon}_k \tag{5.2}$$

where $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} - \delta\hat{\mathbf{x}} = \delta\mathbf{x} - \delta\hat{\mathbf{x}}$ is the filter estimate error which is normally distributed with $\mathcal{N}(\mathbf{0}, \mathbf{P}_k)$, and $\boldsymbol{\upsilon}_k = \boldsymbol{\epsilon}_k + \boldsymbol{\nu}_k$ is the total measurement noise, distributed with $\mathcal{N}\left(\mathbf{0}, \mathbf{R}_k = \mathbf{R}_{\epsilon,k} + \mathbf{R}_{\nu,k}\right)$. The innovation covariance, given in (3.35), is here written in a simpler form as

$$\mathbf{P}_{zz,k} = E\left\{\mathbf{z}_k\mathbf{z}_k^{\mathsf{T}}\right\} = \mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^{\mathsf{T}} + \mathbf{H}_k\mathbf{B}_k + \mathbf{B}_k^{\mathsf{T}}\mathbf{H}_k^{\mathsf{T}} + \mathbf{R}_k \ . \tag{5.3}$$

5.3.3 Failure models

Single GNSS channel blunder

If one of the active GNSS channels is assumed to have failed, the innovation vector can be written as

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{e}_k + \mathbf{m}_i \Delta_{i,k} + \boldsymbol{\upsilon}_k \ , \tag{5.4}$$

where $\mathbf{m}_i$ is a unit vector in the direction i, and $\Delta_{i,k}$ is the blunder magnitude.

GNSS receiver clock discontinuity

A receiver clock bias discontinuity results in a sudden large change of common mode bias in pseudorange and integrated carrier phase observables. The same happens to Pseudorange-rate under a receiver clock drift discontinuity. The innovation model under these cases is

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{e}_k + \mathbf{1} \Delta_{\mathsf{clk},k} + \boldsymbol{\upsilon}_k \ , \tag{5.5}$$

where $\mathbf{1}$ is a column vector of ones, and $\Delta_{\mathsf{clk},k}$ is the magnitude of the receiver clock discontinuity (bias/drift/delta-bias).

Kinematics state divergence

Given the harsh dynamics of launch it is not unlikely that over a period of no GNSS measurements the strapdown integration error on the kinematics states within the filter grows larger than what the propagated covariance would indicate. This could be caused by unmodeled extreme vibrations, shock, or even malign motions such as coning/sculling. If this *failure* is not properly handled and resolved, the new (and sane) GNSS measurements might be erroneously under-weighted or even rejected leading to slow convergence or even failure to converge at all.

This type of state perturbation can be included in the innovation model as

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{e}_k + \mathbf{H}_{\mathsf{kin},k} \boldsymbol{\Delta}_{\mathsf{kin},k} + \boldsymbol{\upsilon}_k \ , \tag{5.6}$$

where $\boldsymbol{\Delta}_{\mathsf{kin},k}$ is a large offset in the kinematics state set estimate, being $\mathbf{H}_{\mathsf{kin},k}$ the sensitivity of the measurement model to this state set.

5.3.4 Statistics tests

Innovation test

As described in Section 3.10.5, the normalized innovation (norm) squared defined, e.g., by [13], as

$$\mathrm{T}^2_{\mathsf{z},k} = \mathbf{z}_k^\mathsf{T} \mathbf{P}^{-1}_{\mathsf{zz},k} \mathbf{z}_k \ , \tag{5.7}$$

which follows a n_k-DoF χ^2-distribution (n_k being the number of measurements at epoch k) is a powerful indicator of filter consistency. It is, therefore, commonly used to detect measurement/model faults; however, it cannot point to the origin of the fault.

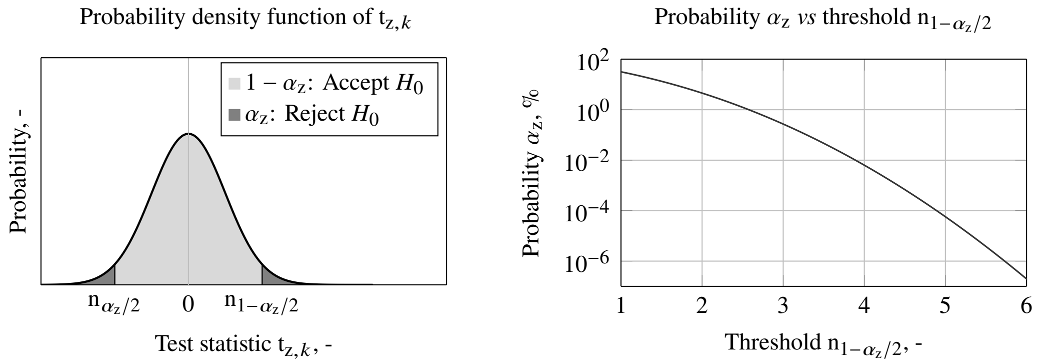


Figure 5.2: Null-hypothesis acceptance threshold $\mathrm{n}_{1-\alpha_{\mathrm{z}}/2}$, probability density function of a normalized innovation component $\mathrm{t}_{\mathrm{z},k}$ and probability of false alarm α_{z}

For more insight, the innovation of each measurement channel can be normalized separately as in [130] as

$$\mathrm{t}_{\mathrm{z},i,k} = \frac{\left(\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_i\right)^{\mathsf{T}}}{\left\|\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_i\right\|}\,\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{z}_k \ , \tag{5.8}$$

where $\mathbf{m}_i$ is a column vector of zeros with a '1' in position i, and $\mathbf{L}_{\mathrm{z},k}$ is the lower triangular square-root of $\mathbf{P}_{\mathrm{zz},k}$ obtained by Cholesky factorization as

$$\mathbf{P}_{\mathrm{zz},k} = \mathbf{L}_{\mathrm{z},k}\mathbf{L}_{\mathrm{z},k}^{\mathsf{T}} \ . \tag{5.9}$$

Note that $\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{z}_k$ and $\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_i$ can be computed efficiently through forward substitution, avoiding the need for explicit inverse matrix calculation.

Because it operates on the entire innovation vector to obtain each normalized component, accounting for all correlations, this method is referred to as *batch* (innovation) *normalization*. Under nominal conditions, $\mathrm{t}_{\mathrm{z},i,k}$ follows a standard normal distribution (null mean, unit variance). This quantity can be tested against two statistical hypotheses: H_0, the null-hypothesis, in which it is unbiased; and H_λ, the alternate-hypothesis, in which it is biased by a quantity λ_{z}. Under these hypotheses each $\mathrm{t}_{\mathrm{z},i,k}$ is distributed as

$$\mathrm{t}_{\mathrm{z},i,k}\big|_{H_0} \sim \mathcal{N}\left(0,1\right) \tag{5.10}$$

$$\mathrm{t}_{\mathrm{z},i,k}\big|_{H_\lambda} \sim \mathcal{N}\left(\lambda_{\mathrm{z}},1\right) \ , \tag{5.11}$$

where λ_{z} is also known as the noncentrality parameter. The null-hypothesis is rejected if

$$|\mathrm{t}_{\mathrm{z},i,k}| > \mathrm{n}_{1-\alpha_{\mathrm{z}}/2} \ , \tag{5.12}$$

where $\mathrm{n}_{1-\alpha_{\mathrm{z}}/2}$ is the quantile $1-\alpha_{\mathrm{z}}/2$. The probability α_{z}, denoted *significance level* of the test, is a tuning parameter of the algorithm. Figure 5.2 depicts the $\alpha_{\mathrm{z}}/2$ tails within the Probability Density Function (PDF) of $\mathrm{t}_{\mathrm{z},k}$ and relates α_{z} with the selected threshold $\mathrm{n}_{1-\alpha_{\mathrm{z}}/2}$.

Another way of obtaining the normalized innovation components is to assume a sequential-update-like normalization scheme, as in [69]. Each scalar innovation is divided by its predicted covariance, not accounting for cross-couplings, as

$$\mathrm{t}_{\mathrm{z},i,k} = \frac{\mathrm{z}_{i,k}}{\sqrt{\mathbf{m}_i^{\mathsf{T}} \mathbf{P}_{\mathrm{zz},k} \mathbf{m}_i}} = \frac{\mathrm{z}_{i,k}}{\sqrt{\mathbf{h}_{i,k}^{\mathsf{T}} \mathbf{P}_k \mathbf{h}_{i,k} + \mathbf{h}_{i,k}^{\mathsf{T}} \mathbf{b}_{i,k} + \mathbf{b}_{i,k}^{\mathsf{T}} \mathbf{h}_{i,k} + \mathbf{m}_i^{\mathsf{T}} \mathbf{R}_k \mathbf{m}_i}} \;, \tag{5.13}$$

where $\mathbf{h}_{i,k}^{\mathsf{T}}$ is the i^{th} row of $\mathbf{H}_k$ and $\mathbf{b}_{i,k}$ is the i^{th} column of $\mathbf{B}_k$. This method, referred to as *sequential normalization*, is computationally lighter than (5.8); however, in case of strong innovation couplings, may yield very different results.

GNSS-only least-squares residual test

In classical RAIM schemes where generally a filtered *a priori* state solution is not available, faults are detected by testing the consistency among channel measurements [30]. A possible test-quantity for this evaluation is the squared norm of the normalized residuals of the Least Squares (LS) solution. The LS residual vector is

$$\boldsymbol{\varepsilon}_k = \left(\mathbf{I} - \bar{\mathbf{H}}_k \mathbf{G}_k\right) \mathbf{y}_k \;, \tag{5.14}$$

where $\mathbf{y}_k$ is the measurement vector and the measurement sensitivity matrix is

$$\bar{\mathbf{H}}_k = \begin{bmatrix} \mathbf{H}_{\mathsf{kin},k} & \mathbf{1} \end{bmatrix} \;, \tag{5.15}$$

and the LS solution mapping matrix is

$$\mathbf{G}_k = \left(\bar{\mathbf{H}}_k^{\mathsf{T}} \mathbf{R}_k^{-1} \bar{\mathbf{H}}_k\right)^{-1} \bar{\mathbf{H}}_k^{\mathsf{T}} \mathbf{R}_k^{-1} \;. \tag{5.16}$$

The reduced measurement sensitivity matrix $\bar{\mathbf{H}}_k$ includes only the sensitivities with respect to a reduced set of kinematics states (three components of position, velocity or displacement) and to a receiver clock state (bias, drift or delta-bias). For the LS problem to have a solution the number of observations n_k must be greater than the number of states of the least-squares m_{LS} (which is usually four).

Because

$$\left(\mathbf{I} - \bar{\mathbf{H}}_k \mathbf{G}_k\right) \bar{\mathbf{H}}_k = \mathbf{0} \,, \tag{5.17}$$

the LS residual vector $\boldsymbol{\varepsilon}_k$ can actually be obtained from the innovation vector as

$$\boldsymbol{\varepsilon}_k = \left(\mathbf{I} - \bar{\mathbf{H}}_k \mathbf{G}_k\right) \mathbf{z}_k = \left(\mathbf{I} - \bar{\mathbf{H}}_k \mathbf{G}_k\right) \left(\mathbf{H}_k \mathbf{e}_k + \boldsymbol{v}_k\right) \approx \left(\mathbf{I} - \bar{\mathbf{H}}_k \mathbf{G}_k\right) \boldsymbol{v}_k \;, \tag{5.18}$$

being (approximately) blind to the noise in the *a priori* estimate, $\mathbf{e}_k$. This makes the computation of $\boldsymbol{\varepsilon}_k$ through (5.14) virtually equal to computing it iteratively from an initial condition different from the *a priori* filter estimate.

As shown in [57], the sum of squares of the normalized (weighted) residuals, given by

$$\mathrm{T}_{\varepsilon,k}^2 = \boldsymbol{\varepsilon}_k^{\mathsf{T}} \mathbf{R}_k^{-1} \boldsymbol{\varepsilon}_k \;, \tag{5.19}$$

follows a χ^2-distribution with $n_k - m_{\mathsf{LS}}$ DoFs, where m_{LS} is the number of states in the sought solution (which will be seen to be 4 regardless of the GNSS observable). Hence, $n_k - m_{\mathsf{LS}}$ represents the degree of redundancy of the measurement vector.

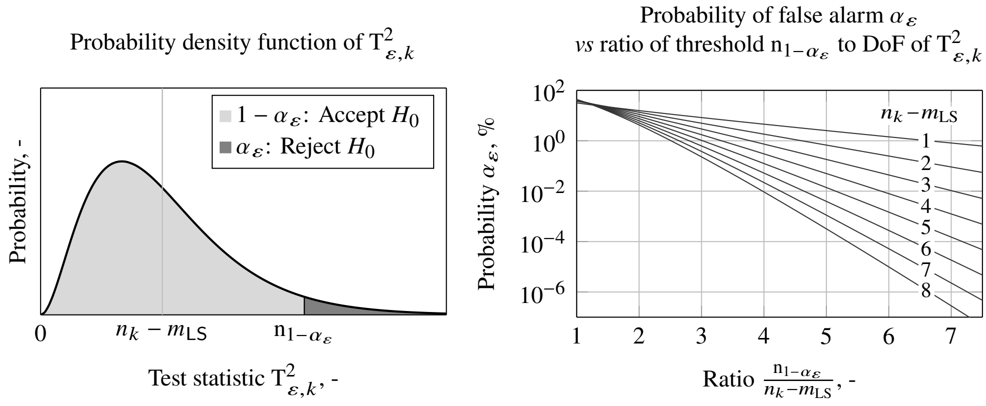


Figure 5.3: Depiction of probability of false alarm α_ε and relation to acceptance threshold $\mathrm{n}_{1-\alpha_\varepsilon}$

The test statistics in this case, under the same hypotheses H_0 and H_λ, is distributed

$$\mathrm{T}^2_{\varepsilon,k}\Big|_{H_0} \sim \chi^2\left(n_k - m_{\mathsf{LS}},\, 0\right) \tag{5.20}$$

$$\mathrm{T}^2_{\varepsilon,k}\Big|_{H_\lambda} \sim \chi^2\left(n_k - m_{\mathsf{LS}},\, \lambda^2_\varepsilon\right)\ , \tag{5.21}$$

where λ^2_ε is the noncentrality parameter of the noncentral χ^2-distribution. Again, the null-hypothesis is rejected if $\mathrm{T}^2_{\varepsilon,k} > \mathrm{n}_{1-\alpha_\varepsilon}$, where $\mathrm{n}_{1-\alpha_\varepsilon}$ is the $1-\alpha_\varepsilon$ quantile, as shown in Figure 5.3 for a 6-DoF χ^2-distribution. Figure 5.3 also shows the relation between the threshold $\mathrm{n}_{1-\alpha_\varepsilon}$ (here divided by the number of DoFs of the distribution) and the false alarm probability α_ε.

An alternative, and often more *economic*, way of computing $\mathrm{T}^2_{\varepsilon,k}$ is

$$\mathrm{T}^2_{\varepsilon,k} = \mathbf{z}_k^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-\mathsf{T}}\mathbf{U}_{2,k}\mathbf{U}_{2,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\mathbf{z}_k, \tag{5.22}$$

where $\mathbf{L}_{\upsilon,k}$ is obtained through the Cholesky decomposition of $\mathbf{R}_k$,

$$\mathbf{R}_k = \mathbf{L}_{\upsilon,k}\mathbf{L}_{\upsilon,k}^{\mathsf{T}} \tag{5.23}$$

or, in the case of uncorrelated measurements, computed simply taking the square root of the diagonal elements of $\mathbf{R}_k$. $\mathbf{U}_{2,k}$ arises from the QR decomposition of $\mathbf{L}_{\upsilon,k}^{-1}\bar{\mathbf{H}}_k$ as

$$\mathbf{L}_{\upsilon,k}^{-1}\bar{\mathbf{H}}_k = \begin{bmatrix}\mathbf{U}_{1,k} & \mathbf{U}_{2,k}\end{bmatrix}\begin{bmatrix}\mathbf{V}_k^{\mathsf{T}} \\ \mathbf{0}\end{bmatrix}\ . \tag{5.24}$$

Expression (5.22) is obtained from (5.19) noting that $\mathbf{U}_{1,k}$ and $\mathbf{U}_{2,k}$ are mutually orthogonal unitary matrices, i.e.,

$$\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{U}_{1,k} = \mathbf{U}_{2,k}^{\mathsf{T}}\mathbf{U}_{2,k} = \mathbf{I}\ , \qquad \mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{U}_{2,k} = \mathbf{0}\ . \tag{5.25}$$

Detection of multiple channel blunders can be done through iterative application of test (5.20)–(5.21) on successively smaller measurements sets: upon test violation, the measurement with the largest normalized innovation magnitude is evicted until a valid set is achieved, a maximum number evictions is reached, or not enough measurements

Table 5.1: Least-squares estimated offsets (filter with respect to GNSS-only) for different GNSS observable models

Measurement	$\boldsymbol{\Delta}_{\mathsf{kin}}$	Δ_{clk}
PR	position	clock bias
PRR	velocity	clock drift
TDCP	displacement	clock bias change

are left to perform the LS residual test. Note, however, that if the normalized innovations used for sorting and eviction of channels have been produced with a batch method, renormalization is necessary upon each set reduction (i.e., each channel eviction) due to the couplings between components in this normalization method (e.g., [169]).

GNSS-only least-squares solution test

Assuming that GNSS single-channel faults have been ruled out by the consistency check, the LS solution can be used to quantify either how much the kinematics state propagation has diverged, or how much the receiver clock model (bias/drift/delta-bias) has “jumped” with respect to the previous GNSS epoch.

The combined fault model from (5.5) and (5.6) is

$$\mathbf{z}_k = \mathbf{H}_{\mathsf{kin},k}\boldsymbol{\Delta}_{\mathsf{kin},k} + \mathbf{1}\Delta_{\mathsf{clk},k} + \mathbf{H}_k\mathbf{e}_k + \boldsymbol{\upsilon}_k \ , \tag{5.26}$$

where the actual meaning of $\boldsymbol{\Delta}_{\mathsf{kin},k}$ and $\Delta_{\mathsf{clk},k}$ depend on the GNSS observable being tested. Table 5.1 shows the estimated offset quantities for each of the considered observable types: PR, PR-rate and TDCP.

The LS estimate is[1]

$$\begin{bmatrix} \hat{\boldsymbol{\Delta}}_{\mathsf{kin},k} \\ \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix} = \mathbf{G}_k\mathbf{z}_k \ . \tag{5.27}$$

Having performed a QR decomposition of $\mathbf{L}_{\upsilon,k}^{-1}\bar{\mathbf{H}}_k$, the gain $\mathbf{G}_k$ can be computed as

$$\mathbf{G}_k = \mathbf{V}_k^{-\mathsf{T}}\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1} \ . \tag{5.28}$$

The covariance of the solution is

$$\begin{aligned} \mathbf{P}_{\Delta\Delta,k} &= E\left\{\begin{bmatrix} \hat{\boldsymbol{\Delta}}_{\mathsf{x},k} \\ \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\Delta}}_{\mathsf{x},k}^{\mathsf{T}} & \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix}\right\} = E\left\{\mathbf{G}_k\mathbf{z}_k\mathbf{z}_k^{\mathsf{T}}\mathbf{G}_k^{\mathsf{T}}\right\} = \mathbf{G}_k\mathbf{P}_{\mathsf{zz},k}\mathbf{G}_k^{\mathsf{T}} \\ &= \mathbf{V}_k^{-\mathsf{T}}\left(\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\mathbf{P}_{\mathsf{zz},k}\mathbf{L}_{\upsilon,k}^{-\mathsf{T}}\mathbf{U}_{1,k}\right)\mathbf{V}_k^{-1} \qquad (5.29) \\ &= \mathbf{V}_k^{-\mathsf{T}}\mathbf{L}_{\Delta,k}\mathbf{L}_{\Delta,k}^{\mathsf{T}}\mathbf{V}_k^{-1} \ , \qquad (5.30) \end{aligned}$$

where $\mathbf{L}_{\Delta,k}$ is the lower triangular square-root of the bracketed term in (5.29), obtained through Cholesky factorization of $\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\mathbf{P}_{\mathsf{zz},k}\mathbf{L}_{\upsilon,k}^{-\mathsf{T}}\mathbf{U}_{1,k}$ or by QR decomposition of $\mathbf{L}_{\mathsf{z},k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-\mathsf{T}}\mathbf{U}_{1,k}$. Note that, if the state contributions not included in the reduced sensitivity

[1] Note that, as it is formulated, (5.27) is really an estimate of $\boldsymbol{\Delta}_{\mathsf{kin},k}$ plus the elements of $\mathbf{e}_k$ corresponding to the non-null sensitivities in $\mathbf{H}_{\mathsf{x},k}$.

matrix $\bar{\mathbf{H}}_k$, in (5.16), are deemed negligible to $\mathbf{P}_{\mathrm{zz},k}$ (e.g., small pseudorange bias), then (5.29) can be simplified expanding $\mathbf{P}_{\mathrm{zz},k}$ and noting that $\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\bar{\mathbf{H}}_k = \mathbf{V}_k^{\mathsf{T}}$.

The overall LS result can be tested using the sum of squares of the normalized solution

$$\mathrm{T}_{\Delta,k}^2 = \begin{bmatrix} \hat{\mathbf{\Delta}}_{\mathsf{kin},k}^{\mathsf{T}} & \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix} \mathbf{P}_{\Delta\Delta,k}^{-1} \begin{bmatrix} \hat{\mathbf{\Delta}}_{\mathsf{kin},k} \\ \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix} , \tag{5.31}$$

which should follow a χ^2-distribution with 4 DoFs. However, once again the intention is to pinpoint the state (or states) that fall out of the predicted covariance bounds. So, each solution component is individually normalized as

$$\mathrm{t}_{\Delta,l,k} = \frac{\left(\mathbf{L}_{\Delta,k}^{-1}\mathbf{V}_k^{\mathsf{T}}\mathbf{m}_l\right)^{\mathsf{T}}}{\left\|\mathbf{L}_{\Delta,k}^{-1}\mathbf{V}_k^{\mathsf{T}}\mathbf{m}_l\right\|} \mathbf{L}_{\Delta,k}^{-1}\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\mathbf{z}_k , \tag{5.32}$$

where $\mathbf{m}_l$ is a unit vector in the direction l (from 1 to 4). Through forward substitution the calculation of the explicit inverses of $\mathbf{L}_{\Delta,k}$ and $\mathbf{L}_{\upsilon,k}$ is avoided. Each $\mathrm{t}_{\Delta,j,k}$ shall follow a standard normal distribution. The test statics for $\mathrm{t}_{\Delta,j,k}$ are then distributed according to the null- and alternate-hypotheses

$$\mathrm{t}_{\Delta,l,k}\big|_{H_0} \sim \mathcal{N}(0,1) \tag{5.33}$$

$$\mathrm{t}_{\Delta,l,k}\big|_{H_\lambda} \sim \mathcal{N}\left(\lambda_{\Delta,l},1\right) , \tag{5.34}$$

where the noncentrality parameter $\lambda_{\Delta,l}$ depend on the significance level $\alpha_{\Delta,l}$. The false alarm probability relates to the threshold as shown in Figure 5.2 for the innovation test.

The simpler sequentially computed normalized LS solution $\mathrm{t}_{\Delta,l,k}$ [akin to (5.13)] is

$$\mathrm{t}_{\Delta,l,k} = \frac{\mathbf{m}_l^{\mathsf{T}}\mathbf{G}_k\mathbf{z}_k}{\sqrt{\mathbf{m}_l^{\mathsf{T}}\mathbf{P}_{\Delta\Delta,k}\mathbf{m}_l}} . \tag{5.35}$$

5.3.5 Recovery

In case the null-hypothesis is rejected for one of the LS estimates, a fault is declared on the corresponding filter state. This is to be interpreted as incoherent filter kinematics state divergence and/or receiver clock bias discontinuity. Despite the different fault origins, the recovery strategy for these two failure types is identical: covariance injection to the affected states, followed by a normal filter update step. The value of this delta-covariance to be added shall be enough to bring the estimated offset back within filter covariance bounds, i.e.,

$$\frac{1}{\mathbf{m}_l^{\mathsf{T}}\mathbf{P}_{\Delta,k}\mathbf{m}_l + \Delta\sigma_{l,k}^2}\left(\mathbf{m}_l^{\mathsf{T}}\begin{bmatrix} \hat{\mathbf{\Delta}}_{\mathsf{kin},k} \\ \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix}\right)^2 = \mathrm{s}^2 \quad \text{with} \quad 0 < \mathrm{s}^2 \leq 1 , \tag{5.36}$$

which, rearranged, gives

$$\Delta\sigma_{l,k}^2 = \frac{1}{\mathrm{s}^2}\left(\mathbf{m}_l^{\mathsf{T}}\begin{bmatrix} \hat{\mathbf{\Delta}}_{\mathsf{kin},k} \\ \hat{\Delta}_{\mathsf{clk},k} \end{bmatrix}\right)^2 - \mathbf{m}_l^{\mathsf{T}}\mathbf{P}_{\Delta,k}\mathbf{m}_l , \tag{5.37}$$

where s^2 can be seen as a safety margin parameter which is unity in the least conservative case. The filter covariance is then recovered adding $\Delta\sigma_{l,k}^2$ to the corresponding state diagonal element of $\mathbf{P}_k$.

5.4 Implementation

In this section the basic FDIR principles and strategy presented in the preceding section are applied to the hybrid navigation design of the previous chapters.

5.4.1 Overall scheme

Starting at a higher level, Figure 5.4 shows the overall FDIR logic flow. After measurement collection, two Fault Detection and Isolation (FDI) routines parallelly process the PR and TDCP measurements. The results of this are then analyzed together and, if issued by any of the two routines, recovery is performed. Filter update follows.

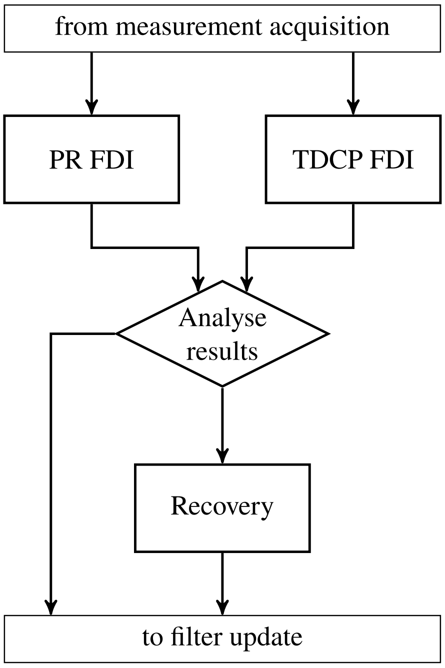


Figure 5.4: Integration of fault detection with recovery

5.4.2 Fault detection and isolation

Figure 5.5 shows the FDI process, which is applied to PR and TDCP measurements separately. The algorithm performs the following steps:

1. Begin;
2. Current innovations are normalized (yielding $t_{z,i}$) and tested against a threshold:
   (a) If no violations occur, all measurements are flagged valid, FDI exists and measurement update proceeds;

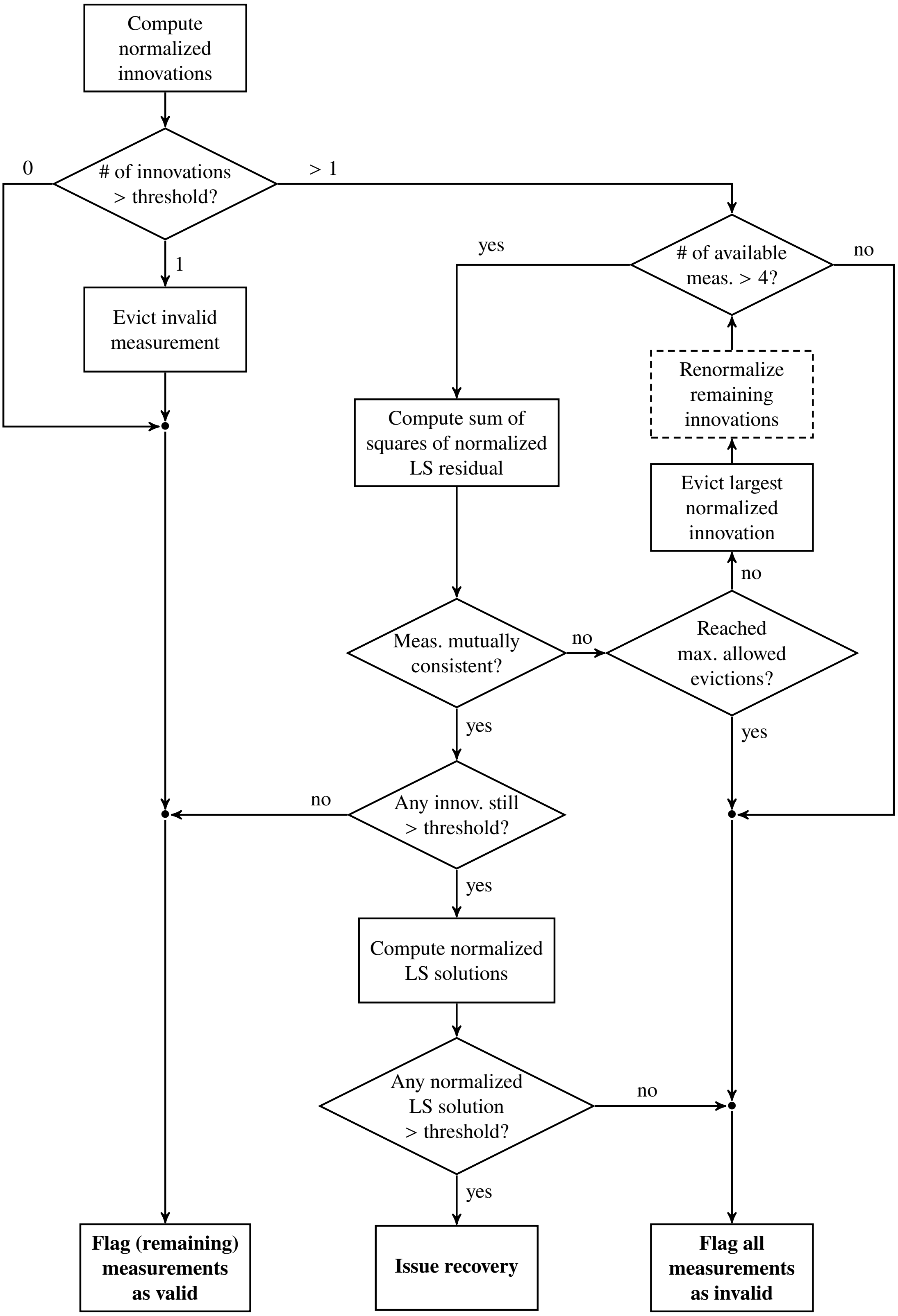

Compute normalized innovations
# of innovations > threshold?
0
> 1
1
Evict invalid measurement
# of available meas. > 4?
yes
no
Renormalize remaining innovations
Compute sum of squares of normalized LS residual
Evict largest normalized innovation
no
Meas. mutually consistent?
no
Reached max. allowed evictions?
yes
yes
Any innov. still > threshold?
no
yes
Compute normalized LS solutions
Any normalized LS solution > threshold?
no
yes
Flag (remaining) measurements as valid
Issue recovery
Flag all measurements as invalid

Figure 5.5: Fault detection and isolation diagram

   (b) If one innovation component fails this test it is eliminated from the set, FDI exists and measurement update proceeds;
   (c) If more than one fail, then inter-channel consistency testing is issued.

3. If only four or fewer measurements are available, all measurements are flagged invalid, FDI exists and update is skipped;

4. Inter-channel consistency is tested through sum of squares of normalized LS residual ($\mathrm{T}_{\varepsilon}^{2}$):

   (a) If the measurement set is not consistent, the largest normalized innovation (absolute-value-wise) is evicted
   (b) The reduced innovation set is renormalized (only if batch normalization is used);
   (c) Inter-channel consistency is retested; this loop goes on until one of the following is satisfied:
      i. a four-element or smaller set is achieved;
      ii. the number of evicted innovation components has achieved the maximum allowed; or
      iii. a consistent measurement set is reached;

      In the first two cases all measurements are flagged invalid and update is skipped, and in the latter one the remaining normalized innovations are re-checked;
   (d) If none of the remaining normalized innovations violates the threshold, all are flagged valid, FDI exists and measurement update proceeds;
   (e) Otherwise, solution separation test is issued;

5. Normalized LS solutions (i.e., $\mathrm{t}_{\Delta,j}$) are tested:

   (a) If any kinematics state component or receiver clock LS state violation occurs, recovery is issued;
   (b) Otherwise, an inconclusive result was reached and all measurements are flagged invalid, FDI exists and update is skipped;

6. End.

Naturally, prior to this FDI process, GNSS message checksums and individual channel status flags are used as early validity indicators.

As shall be shown in Section 5.5 upon statistical analysis of the signals within the two FDI chains, the PR and TDCP measurements have markedly different noise profiles. Because of this the innovation normalization process of both chains is different:

- Due to the strong effect of common-mode (receiver clock) noise on TDCP, the FDI chain for this measurement type includes testing for both batch- and sequentially-normalized innovations [(5.8) and (5.13), respectively]. This, as will be shown, allows for effective detection of both common-mode and single-channel failures;

- As shall be seen in Section 5.5.1, because the PR measurements are far less coupled, only sequential normalization [as in (5.13)] is used in the innovations testing in this FDI chain.

Different between the two chains is also the state set (in Table 5.1) of the LS used to test for both measurement self-consistency and filter-state/receiver-clock failures:

- The LS formulation in the PR chain uses ECEF position and receiver clock bias as states; while
- The state set of the LS in the TDCP chain includes displacement in ECEF (i.e., the difference between current position and that at the previous GNSS epoch) and receiver clock noise (i.e., clock bias change between current and previous GNSS epochs).

The normalization of the LS solutions in both chains (for the t_{Δ}-test) is done using the sequential method only [as in (5.35)]. This choice reflects the low coupling between different components of the LS solution and will be justified in Section 5.5.

5.4.3 Recovery

As shown in Figure 5.4, recovery is reached whenever kinematics state and/or receiver clock anomalies are identified in either of the two FDI chains (PR and TDCP). In such case, and as detailed in Section 5.3.5, additional covariance may be injected into the affected states according to (5.37).

While the LS solutions for the PR model are perturbations in quantities directly described by filter states (position and clock bias, as shown in Table 5.1), for the TDCP this is not so. The LS solutions in this case are displacement (or position change) and receiver clock noise (or clock bias change). Therefore, in case of failure, the covariance increments given in terms of these quantities by the TDCP FDI chain need to be translated into usable filter covariance values. Under clock noise blunder, the issued covariance amount $\Delta\sigma_{\Delta,\mathrm{clk}}$ is added not only to the noise term $\mathbf{R}_{\epsilon}$ in the TDCP measurement covariance model (3.152), but also to the filter clock bias state covariance, and to the correlation term between the two, $\mathbf{B}^{\epsilon}_{\mathrm{w}_{\tau,\mathrm{r}}}$ [defined in (3.139)]. This is intended to correct the identified missing uncertainty in all instances of modeled receiver clock noise in the filter during the last GNSS epoch.

In the case of a kinematics state failure, the current (real) position covariance can be approximated from previous epoch state and process noise covariances as

$$\mathbf{P}_{\mathrm{r},k} \approx \mathbf{P}_{\mathrm{r},k-1} + \Delta t \mathbf{P}_{\mathrm{v},k-1} + \mathbf{Q}_{\mathrm{r},k-1|k} + \frac{\Delta t^2}{3}\left(\mathbf{Q}_{\mathrm{v},k-1|k} + \mathbf{Q}_{\Delta\mathrm{v},k-1|k}\right) + \mathbf{\Delta P}_{\mathrm{r},k-1|k} \ , \qquad (5.38)$$

where $\mathbf{P}_{\mathrm{v},k-1}$ is the velocity covariance at the beginning of the interval, and $\mathbf{Q}_{\mathrm{r},k|k+1}$ and $\mathbf{Q}_{\mathrm{v},k|k+1}$ are the process noise covariance (due to numerical integration error) accumulated over the interval Δt in the position and velocity states, respectively, and $\mathbf{Q}_{\Delta\mathrm{v},k|k+1}$ is the accumulated inertial measurement uncertainty over such interval. The term $\mathbf{\Delta P}_{\mathrm{r},k|k+1}$ represents the "missing" position uncertainty that would have rendered the new set of GNSS measurements sane (i.e., acceptable to the FDI chains) had the filter accounted for it during the last interval Δt, from t_{k-1} to t_k. This quantity approximated as a diagonal matrix and estimated by the recovery expression (5.37), i.e.

$$\boldsymbol{\Delta}\mathbf{P}_{\mathrm{r},k-1|k} \approx \mathsf{diag}\left(\boldsymbol{\Delta\sigma}^2_{\mathsf{kin},k}\right) . \tag{5.39}$$

In the TDCP chain, and as above mentioned, the FDI LS state includes displacement and not position. The covariance of this displacement during the last GNSS epoch can be approximated as the uncertainty gained by $\mathbf{P}_\mathrm{r}$ during that period, i.e.,

$$E\left\{\Delta\mathbf{r}_{k|k-1}\Delta\mathbf{r}^{\mathsf{T}}_{k|k-1}\right\} \approx \Delta t\mathbf{P}_{\mathrm{v},k-1} + \mathbf{Q}_{\mathrm{r},k-1|k} + \frac{\Delta t^2}{3}\left(\mathbf{Q}_{\mathrm{v},k-1|k} + \mathbf{Q}_{\Delta\mathrm{v},k-1|k}\right) + \boldsymbol{\Delta}\mathbf{P}_{\mathrm{r},k-1|k} , \tag{5.40}$$

which yields a similar dependence on the term $\boldsymbol{\Delta}\mathbf{P}_{\mathrm{r},k|k+1}$. Note that in the above approximations all rotations have been neglected.

Because position and velocity are propagated together in the strapdown process, an insufficient filter position covariance implies an equally lacking velocity covariance. From (5.38), it is possible to relate $\boldsymbol{\Delta}\mathbf{P}_{\mathrm{r},k|k+1}$ to a “missing” velocity covariance term $\boldsymbol{\Delta}\mathbf{P}_{\mathrm{v},k|k+1}$ as

$$\boldsymbol{\Delta}\mathbf{P}_{\mathrm{r},k|k+1} = \frac{\Delta t^2}{3}\boldsymbol{\Delta}\mathbf{P}_{\mathrm{v},k|k+1} . \tag{5.41}$$

The filter position and velocity covariance terms are thus recovered together as

$$\mathbf{P}_{\mathrm{r},k} \leftarrow \mathbf{P}_{\mathrm{r},k} + \mathsf{diag}\left(\boldsymbol{\Delta\sigma}^2_{\mathsf{kin},k}\right) \tag{5.42}$$

$$\mathbf{P}_{\mathrm{v},k} \leftarrow \mathbf{P}_{\mathrm{v},k} + \frac{3}{\Delta t^2}\,\mathsf{diag}\left(\boldsymbol{\Delta\sigma}^2_{\mathsf{kin},k}\right) \tag{5.43}$$

In case of recovery from a GNSS outage, Δt refers to the duration of the signal gap. In such case, TDCP updates are not performed due to lack of carrier phase measurements for the (directly) preceding GNSS epoch.

Shall a failure be detected on both FDI streams, PR and TDCP, the applied covariance injection is the most conservative of the prescribed values, i.e.,

$$\Delta\sigma^2_{l,k} = \mathsf{max}\left(\left\{\Delta\sigma^2_{l,k}\right\}_{\mathsf{PR-FDI}},\ \left\{\Delta\sigma^2_{l,k}\right\}_{\mathsf{TDCP-FDI}}\right) . \tag{5.44}$$

Divergence between filter and GNSS-only solutions is most likely to occur during periods of satellite navigation outage. Upon reacquisition, in case of incoherent offset between filter and (consistent) GNSS solution, recovery is key to allow a smooth and prompt convergence of the filter back to a coherent state. However, if such divergence is detected during continuous GNSS coverage, and to avoid impulsive reaction to a one-time event, the application of recovery measures is delayed by one measurement epoch: the current update is canceled and in the following epoch recovery is applied if the failure remains. This process is depicted in Figure 5.6, where the previous epoch FDIR result is checked before recovery is applied.

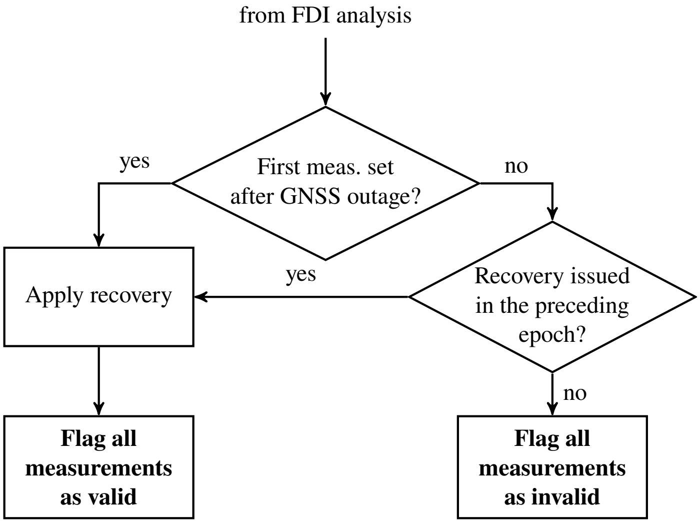

from FDI analysis
First meas. set after GNSS outage?
yes
no
Apply recovery
yes
Recovery issued in the preceding epoch?
no
Flag all measurements as valid
Flag all measurements as invalid

Figure 5.6: Recovery logic diagram

5.5 Verification and threshold selection

This section describes the test statistics validation and threshold selection using static GPS data collected using a DLR Phoenix-HD receiver stimulated by a GSS7700 SPIRENT GNSS signal emulator. During about 1000 s, 8 SVs are continuously tracked by the receiver and their corresponding raw measurements are collected at 1 Hz. This data is then fed to the navigation algorithm which, unless otherwise stated, is the reduced-order filter with Medium-grade sensor (see Section 4.4.4).

5.5.1 Signal profile verification

The above described nominal, fault-free scenario is used to verify the statistical properties of the fault detection test quantities for the developed filter. These quantities are listed with their statistical description in Table 5.2. Note that the normalized innovation norm squared ($\mathrm{T}^2_{\mathrm{z},k}$) is here included for analysis purposes only; it does not figure in the FDI algorithm described in the preceding sections.

Table 5.2: Stochastic distribution of the quantities in the FDIR algorithm and its analysis

Quantity	Symbol	Definition	Nominal distribution
Normalized innovation norm squared	$\mathrm{T}^2_{\mathrm{z},k}$	(5.7)	$\chi^2\left(n_k, 0\right)$
Normalized innovation	$\mathrm{t}_{\mathrm{z},i,k}$	(5.8), (5.13)	$\mathcal{N}\left(0, 1\right)$
Sum of squares of normalized LS residual	$\mathrm{T}^2_{\varepsilon,k}$	(5.19), (5.22)	$\chi^2\left(n_k - m_{\mathrm{LS}}, 0\right)$
Normalized LS solution norm squared	$\mathrm{T}^2_{\Delta,k}$	(5.31)	$\chi^2\left(m_{\mathrm{LS}}, 0\right)$
Normalized LS solution	$\mathrm{t}_{\Delta,l,k}$	(5.32), (5.35)	$\mathcal{N}\left(0, 1\right)$

Starting the verification at a high level, Figure 5.7 shows the normalized innovation norm squared for the PR and TDCP measurement sets in the reduced-order filter. The distributions of these signals are there also compared to the full-order filter equivalents and the theoretical nominal distributions of both (given in Table 5.2). While the

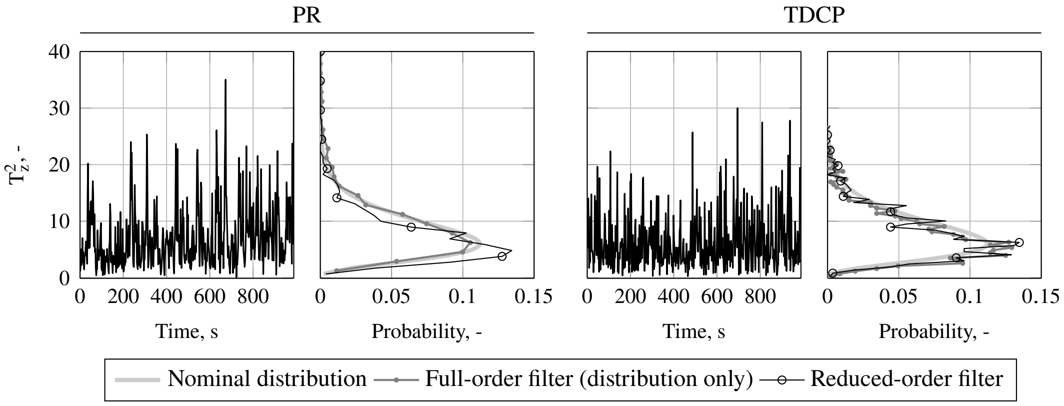


Figure 5.7: Normalized innovation norm squared statistic T_z^2 of the reduced-order filter in static test with real GNSS data

plotted distribution curves of both reduced-, and full-order filters approach the nominal one, that of the full-order estimator comes visibly closer to it, especially for the PR innovations. This, as shall be discussed below, is caused by the considering (i.e., not explicitly estimating) of PR and TDCP channel biases in the reduced-order filter.

The first-line fault detection in the conceived run-time algorithm is the normalized innovation test. As explained Section 5.3.4, the two methods considered for innovation component normalization are: batch and sequential. The sequential method simply divides the innovation components by the square-root of their self-covariance (i.e., of their corresponding innovations covariance matrix diagonal element), while the batch normalization uses the inverse of the innovations covariance matrix (or the inverse of its triangular Cholesky square-root) to produce (approximately) decoupled normalized individual components.

Figure 5.8 shows the normalized PR innovations for each of the 8 active channels computed using both of these methods. The distributions of these normalized signals are also displayed along with their nominal distributions (as in Table 5.2). The distributions of the equivalent signals obtained with a full-order filter (as the baseline design of Chapter 3) are shown for comparison. The same quantities for the TDCP measurements are displayed in Figure 5.9.

The signals resulting from both normalization methods on both measurement types follow approximately the nominal standard normal distributions, with RMS standard deviations of 0.901 and 0.928 for the batch and sequentially normalizations, respectively, in the PR innovations case and 0.974 and 1.073 in the case of TDCP innovations. While for PR both batch and sequential methods produce similar normalization results, for TDCP innovations this is not the case: note how the sequentially-normalized signals (and their distributions) look remarkably alike across channels. This is caused by the fact that in the receiver setup used the TDCP common-mode noise (receiver clock noise) is far larger than the individual channel noises. The batch-normalization effectively mitigates this effect. Evidence of this is given in Figure 5.10, where the distribution of the sum of squares (channel-wise) of the normalized innovations (i.e., $\sum_{i=1}^{n_k} t_{z,i,k}^2$) obtained with each method is compared to the nominal one (an 8-DoF χ^2). As can be seen the batch-normalization produces a distribution much closer to the nominal than the

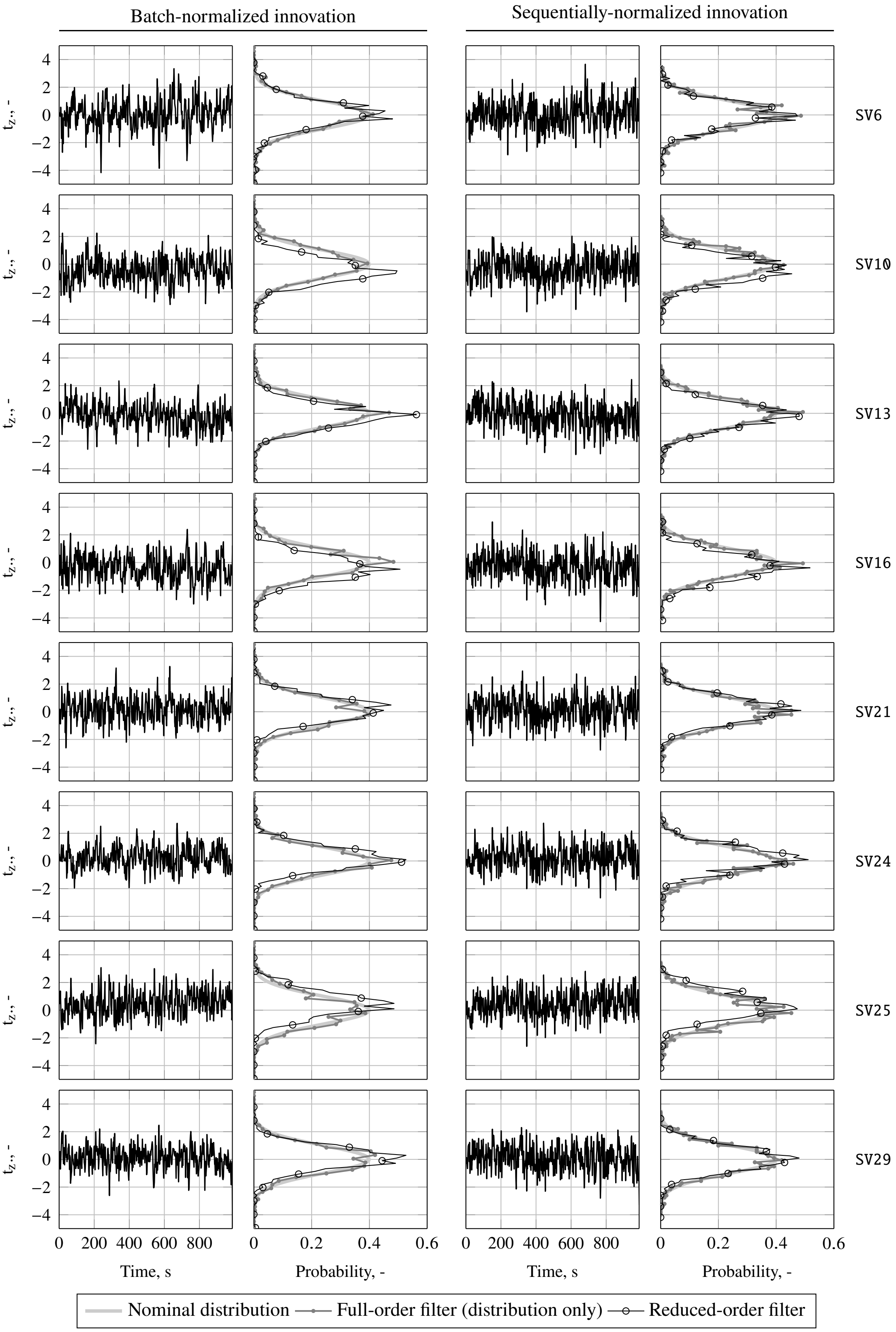


Figure 5.8: Time history and distribution of normalized PR innovations of the reduced-order filter under static real GNSS data and comparison to nominal and full-order filter equivalent distributions

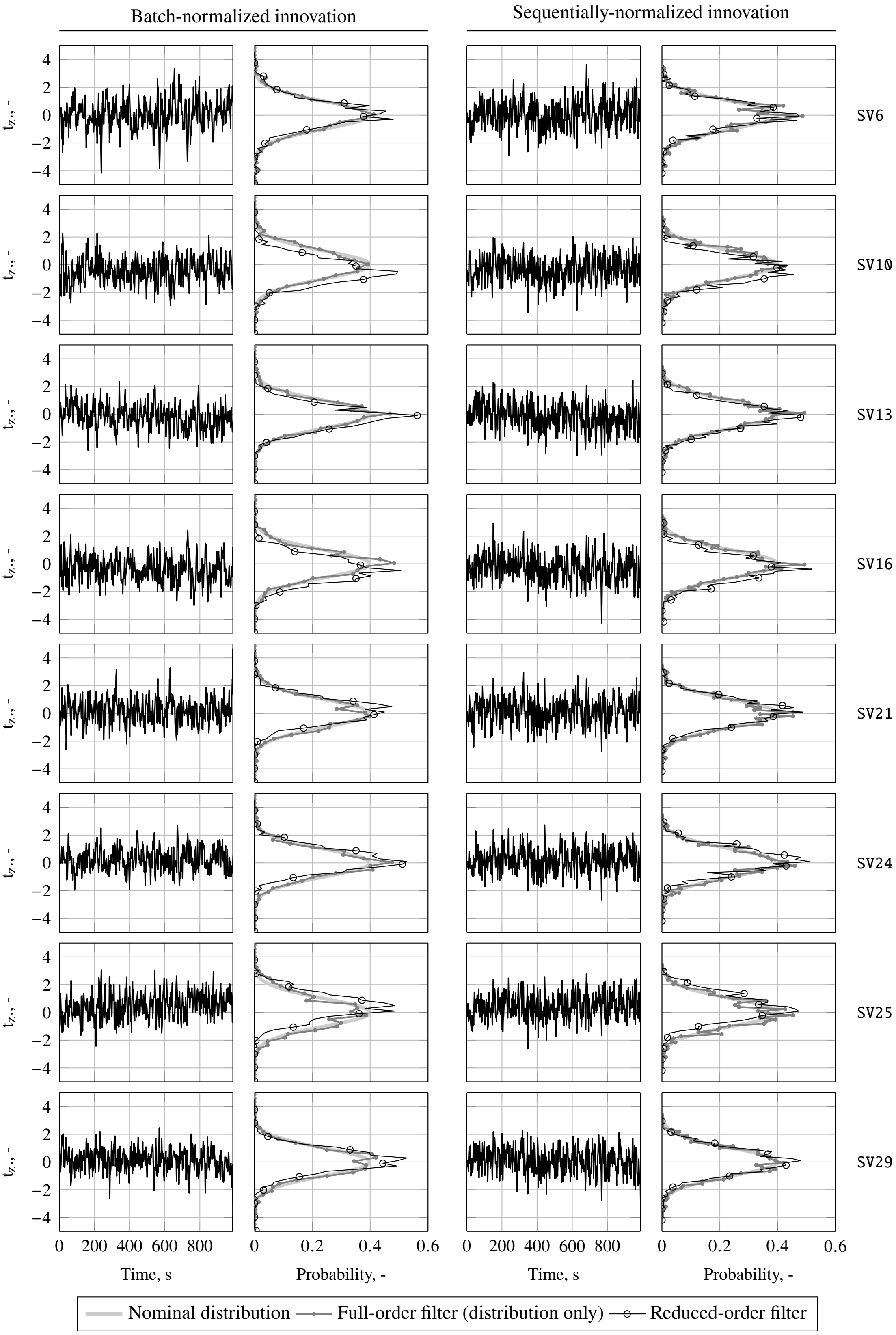


Figure 5.9: Time history and distribution of normalized TDCP innovations of the reduced filter under static real GNSS data and comparison to nominal and full-order filter equivalent distributions

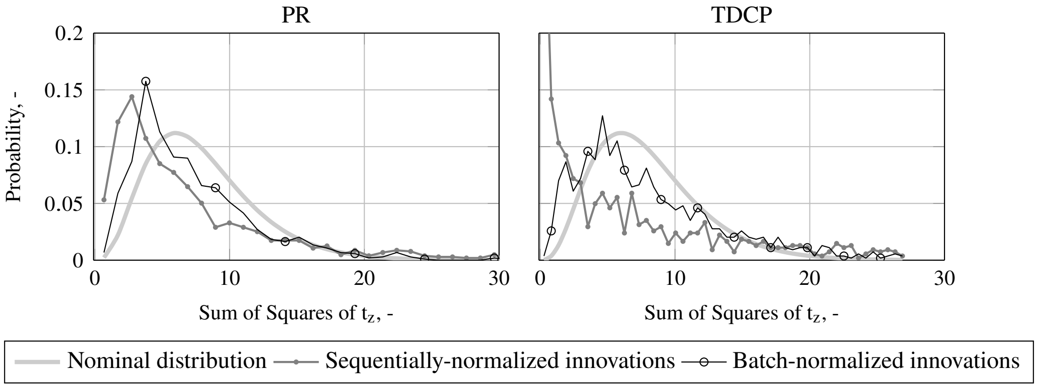


Figure 5.10: Distributions of the sum of squares of normalized innovations of PR and TDCP (of Figures 5.8 and 5.9, respectively)

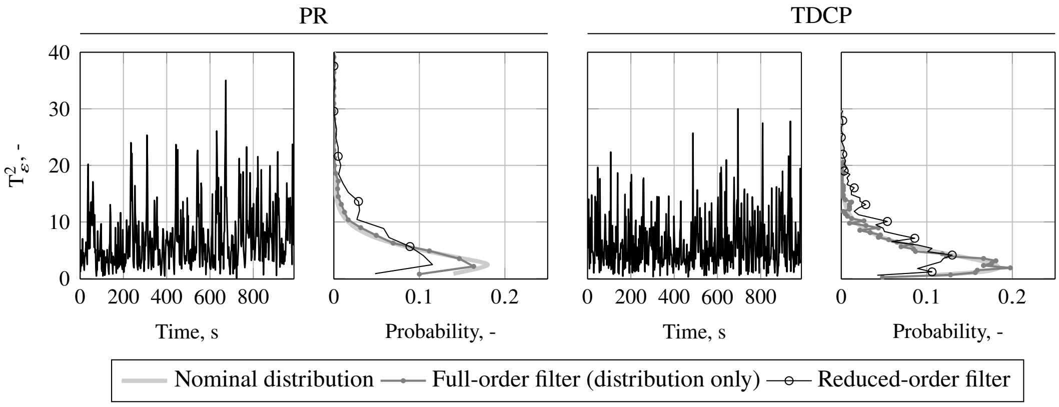


Figure 5.11: Time history and distribution of LS normalized residual norm squared (PR and TDCP) for the reduced filter under static real GNSS data and comparison to nominal and full-order filter equivalent distributions

sequential method. Indeed, the distribution obtained with the later method approaches the DoF χ^2 far closer, again, evidencing the strong coupling between channels. The distributions of the same signals on the PR set (also shown in Figure 5.10) are fairly similar, with, as expected, that of the batch-normalized signal approaching better the nominal distribution (also an 8-DoF χ^2). Finally, comparing now the individual normalized innovation distributions of the reduced-order filter with those of the full-order filter, it is clear that those of the former are generally slightly off center (i.e., biased), with RMS means of 0.307 and 0.247 for batch and sequentially normalized PR innovations, respectively, and 0.270 and 0.002 in the case of TDCP innovations. The higher RMS mean of the batch method, particularly for TDCP measurements, is explained by the partial elimination of common-mode (clock) effects which leaves evidenced the individual channel biases. Such biases are caused by the considering of both PR and TDCP channel biases in the reduced-order filter. As it will be shown further in this section, the slightly biased innovations shall not hinder the FDIR effectiveness.

Figure 5.11 shows, for PR and TDCP sets, the measurement-only LS normalized residual norm squared [as defined in (5.19)–(5.22)]. The distribution of these signals and their full-order filter equivalents are also displayed and compared to their nominal

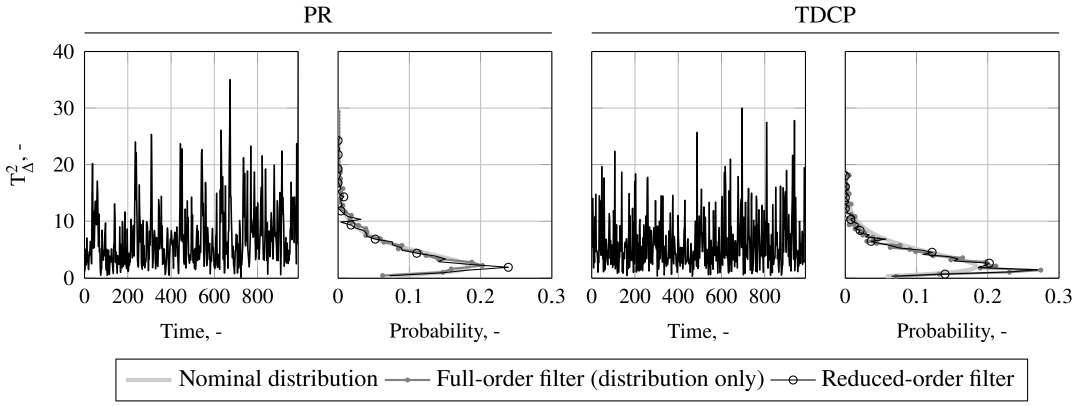


Figure 5.12: Time history and distribution of LS normalized solution norm squared (PR and TDCP) in the reduced filter under static real GNSS data and comparison to nominal and full-order filter equivalent distributions

distribution (a 4-DoF χ^2 in this case). As can be seen, the reduced-order distributions do not follow the nominal one nearly as well as their full-order counterparts; their shape is narrower at lower values of T_ε with a slightly thicker tail at higher values. This effect, again mainly due to the considering of GNSS channel biases, could be ameliorated through changes in the tuning, which would however degrade the statistical profile of the first-line $t_{z,i}$-tests. An obvious result of this distribution difference is a higher false alarm rate of the T_ε-test. This is discussed further in Section 5.7.

The LS normalized solution norm squared (in Figure 5.12), however, does not show the same degree of mismatch. In fact, it displays adequate distributions with only a visible peak veering away from the nominal at very low signal values. This is seen for both PR and TDCP measurement sets, and is more pronounced for the TDCP set and the reduced-order filter.

The individual normalized LS solutions, presented in Figure 5.13 for PR and TDCP measurements, also display distributions that approximate fairly well the standard normal for both filter orders and both normalization methods (batch and sequential). Slight departures from the nominal are, however, visible in the distributions of x and y components of the TDCP LS normalized solutions of both filters. The distributions of both filters nearly match in all components and measurement types.

Finally, and as done in the analysis of the normalized innovations, the sum of squares of the individual normalized components of each normalization strategy is compared in Figure 5.14 for PR and TDCP. This time both methods yield very similar distributions (and both acceptably close to the nominal, 4-DoF χ^2), evidencing a low level of cross-coupling between LS solution components.

5.5.2 Threshold selection and detection limits

Having defined the FDI strategy and validated the nominal statistic profiles of the signals therein, proper selection of the test statistics thresholds is due. In addition to setting the false alarm levels, these thresholds define the blunder detection limits of the FDI scheme in what is known as internal reliability of the system [171, 130].

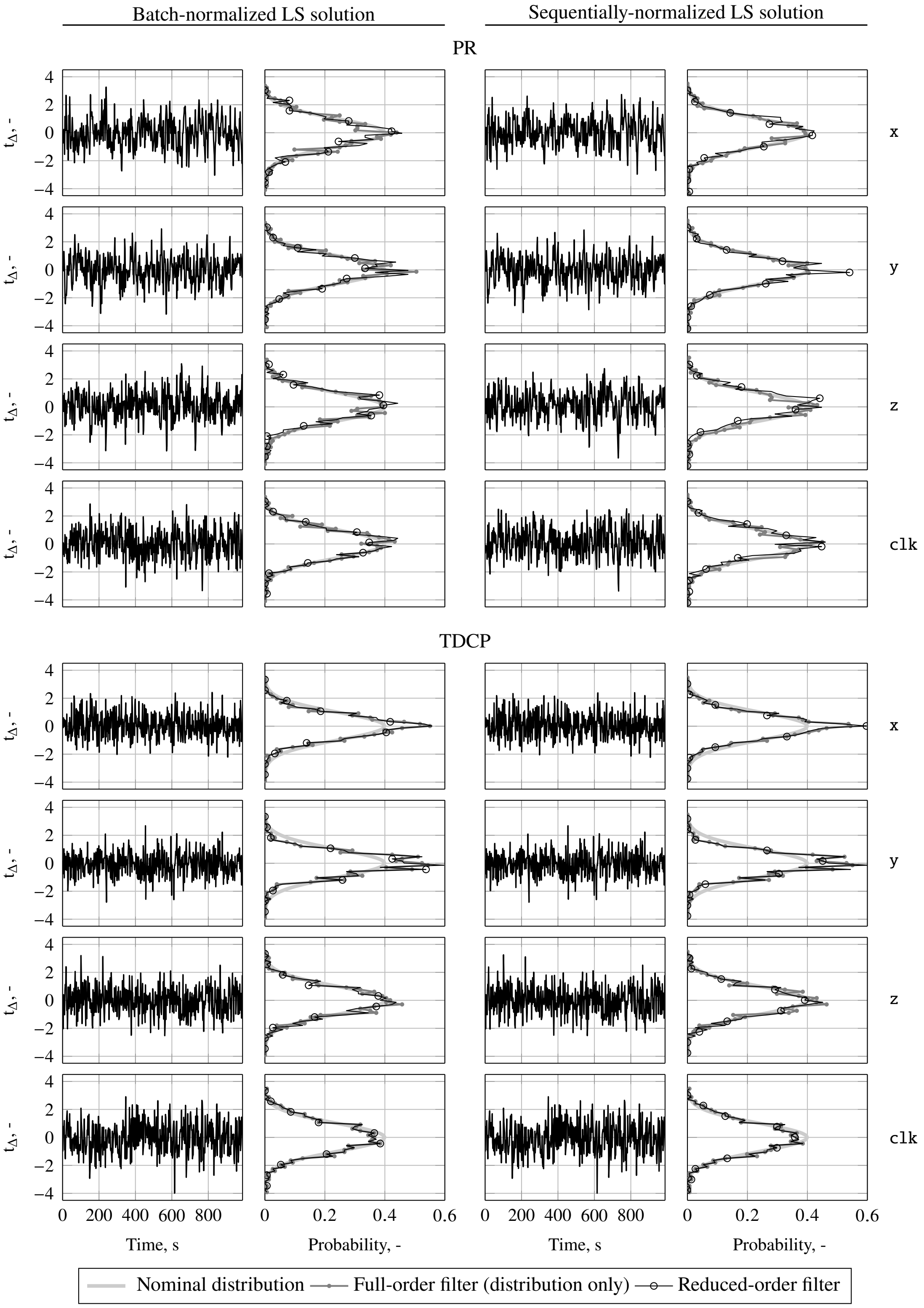


Figure 5.13: Time history and distribution of LS normalized solutions in PR and TDCP FDI chains of the reduced filter under static real GNSS data and comparison to nominal and full-order filter equivalent distributions

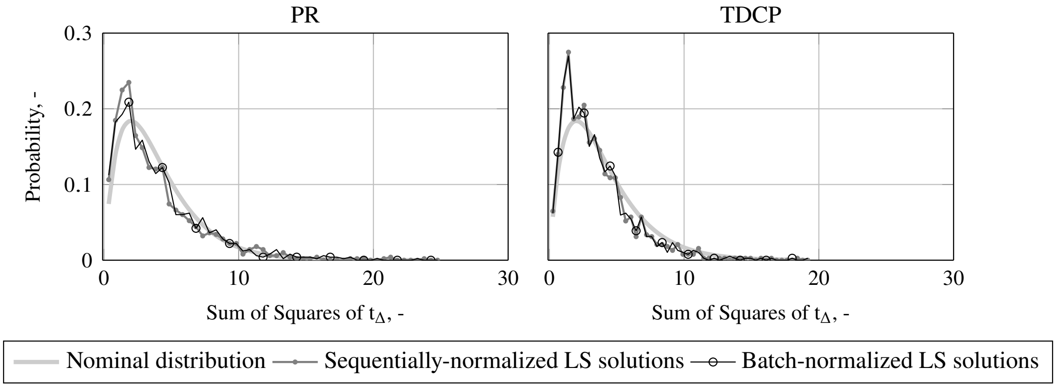


Figure 5.14: Distributions of the sum of squares of LS normalized solutions in the PR and TDCP FDI chains (of Figure 5.13)

Innovation test threshold

As previously mentioned, in nominal conditions the innovation test quantities $t_{z,i,k}$ [in (5.8) and (5.13)] follow standard normal distributions. The choice of threshold $n_{1-\alpha_z/2}$ of the test (5.10)–(5.11) dictates the probability of false alarm (type I error) as depicted in Figure 5.2. False alarm levels of 0.1–0.001% are common practice. Here, $n_{1-\alpha_z/2}$ is selected as 5 (σ) yielding

$$\alpha_z \approx 5.7\text{e}{-5}\ \% \ , \tag{5.45}$$

i.e., virtually no false detections.

With the threshold set, it is possible to evaluate the resulting detection limits by computing the Minimum Detectable (MD) blunders. This can be easily done by inverting (5.8) or (5.13), depending on the normalization in use, to obtain the blunder magnitudes that render the test statistics $t_{z,i,k}$ equal to the threshold $n_{1-\alpha_z/2}$. However, blunders of such magnitude are only detectable with a 50% chance; which can hardly be considered “detectable”. The meaning of Minimum Detectable should then be redefined by fixing the probability, β, of overlooking a blunder, in a so-called *type II error*. Typical values for this probability range from 1 to 20%. Given this setting, the value of the non-centrality parameter λ_z of the biased hypothesis (5.11) is

$$\lambda_z = n_{1-\alpha/2} + n_{1-\beta} \ . \tag{5.46}$$

This can be seen graphically in Figure 5.15.

The minimum detectable blunder is then computed making $t_{z,i,k} = \lambda_z$. For a single channel fault it is given by

$$\Delta_{i,k}^{\mathsf{MD}} = \frac{\lambda_z}{\left\| \mathbf{L}_{z,k}^{-1} \mathbf{m}_i \right\|} \ , \tag{5.47}$$

for the batch normalized innovation (5.8) and

$$\Delta_{i,k}^{\mathsf{MD}} = \lambda_z \sqrt{\mathbf{m}_i^{\mathsf{T}} \mathbf{P}_{zz,k} \mathbf{m}_i} \ , \tag{5.48}$$

for the sequentially normalized innovation (5.13). Note that the former is obtained realizing that the diagonal elements of the inverse of a triangular matrix equal the inverse of the diagonal elements of the original matrix.

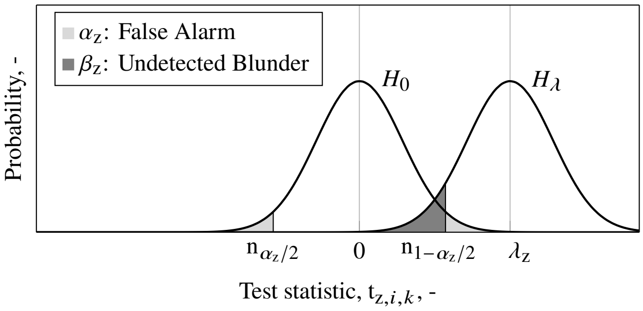


Figure 5.15: Probability density function of the normalized innovation statistic $t_{z,i,k}$ under both unbiased and biased conditions

For better interpretation, the blunder $\Delta_{i,k}^{\mathsf{MD}}$ can be given in terms of the channel noise covariance as

$$\Delta_{i,k}^{\mathsf{MD}} = \mathrm{B}_{i,k}^{\mathsf{MD}} \sqrt{\mathbf{m}_i^{\mathsf{T}} \mathbf{R}_{\nu,k} \mathbf{m}_i} \ , \tag{5.49}$$

where $\mathrm{B}_{i,k}^{\mathsf{MD}}$ is the blunder magnitude to channel noise ratio of the i^{th} channel.

Concerning GNSS receiver clock faults, as described in Section 5.4, a minimum of two innovation component test violations should occur for a correct detection (not identification) of a clock bias discontinuity (or clock noise blunder). The detection limit $\Delta_{\mathsf{clk},k}^{\mathsf{MD}}$ for this failure can be computed as the second smallest (absolute-value-wise) $\Delta_{\mathsf{clk},i,k}^{\mathsf{MD}}$, which is the minimum clock fault detectable by the statistic test of the i^{th} innovation component given as

$$\Delta_{\mathsf{clk},i,k}^{\mathsf{MD}} = \frac{\left\| \mathbf{L}_{\mathrm{z},k}^{-1} \mathbf{m}_i \right\| \lambda_{\mathrm{z}}}{\left(\mathbf{L}_{\mathrm{z},k}^{-1} \mathbf{m}_i \right)^{\mathsf{T}} \mathbf{L}_{\mathrm{z},k}^{-1} \mathbf{1}} \ , \tag{5.50}$$

for the batch normalization and

$$\Delta_{\mathsf{clk},i,k}^{\mathsf{MD}} = \lambda_{\mathrm{z}} \sqrt{\mathbf{m}_i^{\mathsf{T}} \mathbf{P}_{\mathrm{zz},k} \mathbf{m}_i} \ , \tag{5.51}$$

for the sequential normalization. **1** is a column vector of *ones* the size of the innovations vector. The value of $\Delta_{\mathsf{clk},k}^{\mathsf{MD}}$ can be given as a ratio to the receiver clock bias state covariance as

$$\Delta_{\mathsf{clk},k}^{\mathsf{MD}} = \mathrm{B}_{\mathsf{clk},k}^{\mathsf{MD}} \sqrt{\sigma_{\mathsf{clk},k}^{2}} \ . \tag{5.52}$$

For the case of kinematics state divergence a slightly different, yet important question can be posed: what is the largest kinematics state offset (norm-wise) that is undetected with the selected threshold? Again, recall that correct detection of this type of failure requires at least two innovation components to fail the first-line test [(5.10)–(5.11)]. The array of normalized innovation components $\mathbf{t}_{\mathrm{z},k}$ in the event of a maximum undetected kinematics state fault $\boldsymbol{\Delta}_{\mathrm{x},k}^{\mathrm{MU}}$ [see fault model (5.6)] given by

$$\mathbf{t}_{\mathrm{z},k} = \lambda_{\mathrm{z}} \left(\begin{bmatrix} \frac{\left(\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_1\right)^{\mathsf{T}}}{\left\|\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_1\right\|} \\ \vdots \\ \frac{\left(\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_{n_k}\right)^{\mathsf{T}}}{\left\|\mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{m}_{n_k}\right\|} \end{bmatrix} \mathbf{L}_{\mathrm{z},k}^{-1}\mathbf{H}_{\mathrm{kin},k} \right) \boldsymbol{\Delta}_{\mathrm{kin},k}^{\mathrm{MU}} , \tag{5.53}$$

for the batch normalization and

$$\mathbf{t}_{\mathrm{z},k} = \lambda_{\mathrm{z}} \left(\begin{bmatrix} \frac{\mathbf{m}_1^{\mathsf{T}}}{\sqrt{\mathbf{m}_1^{\mathsf{T}}\mathbf{P}_{\mathrm{zz},k}\mathbf{m}_1}} \\ \vdots \\ \frac{\mathbf{m}_{n_k}^{\mathsf{T}}}{\sqrt{\mathbf{m}_{n_k}^{\mathsf{T}}\mathbf{P}_{\mathrm{zz},k}\mathbf{m}_{n_k}}} \end{bmatrix} \mathbf{H}_{\mathrm{kin},k} \right) \boldsymbol{\Delta}_{\mathrm{kin},k}^{\mathrm{MU}} , \tag{5.54}$$

for the sequential normalization.

The vector $\boldsymbol{\Delta}_{\mathrm{kin},k}^{\mathrm{MU}}$ for each of these cases lies along the direction of minimum sensitivity of (5.53) or (5.54). This direction is parallel to the right-singular vector corresponding to the lowest singular value (absolute-value-wise) of the compound matrix term in curved brackets in (5.53) and (5.54). The maximum undetectable kinematics state offset is that that (along the direction just mentioned) brings two components of $\mathbf{t}_{\mathrm{z},k}$ to or above the test threshold. Again, for a better understanding of the detection levels, this value is factorized with the total position filter state covariance as

$$\left\|\boldsymbol{\Delta}_{\mathrm{kin},k}^{\mathrm{MU}}\right\| = \mathrm{B}_{\mathrm{kin},k}^{\mathrm{MU}} \sqrt{\sum_{l=1}^{3} \sigma_{\mathrm{kin},l,k}^{2}} . \tag{5.55}$$

Figure 5.16 shows the lowest minimum detectable single-channel blunder (among all channels), the minimum detectable clock discontinuity, and the maximum undetectable kinematics state offset for both PR and TDCP measurements. The values for these quantities obtained with batch and sequential normalization methods are compared, and different levels of β_{z} (1%, 10%, and 50%) are also confronted.

It is clear from the plots corresponding to both PR and TDCP that the batch normalization provides higher sensitivity to single-channel measurement blunders. This feature, explained by the detrending of the innovations, is particularly evident in the TDCP case, where the common-mode (clock) noise is the dominant measurement perturbation. In this case, detrending is necessary if single-channel events in the scale of the single-channel noise are to be detected. For the PR measurement fault detection the advantage of the batch normalization is relatively small.

One crucial feature of the batch normalization is that it renders the $\mathrm{t}_{\mathrm{z},i}$ statistic insensitive to common-mode events. Again, this is the most evident for the TDCP, rendering sequential normalization necessary for clock failure detection in this measurement type.

Similarly to the single-channel blunders, the maximum undetectable kinematics state offset is lower in the case of batch normalized innovations. The difference between the two normalization schemes is, however, similar in both measurement types.

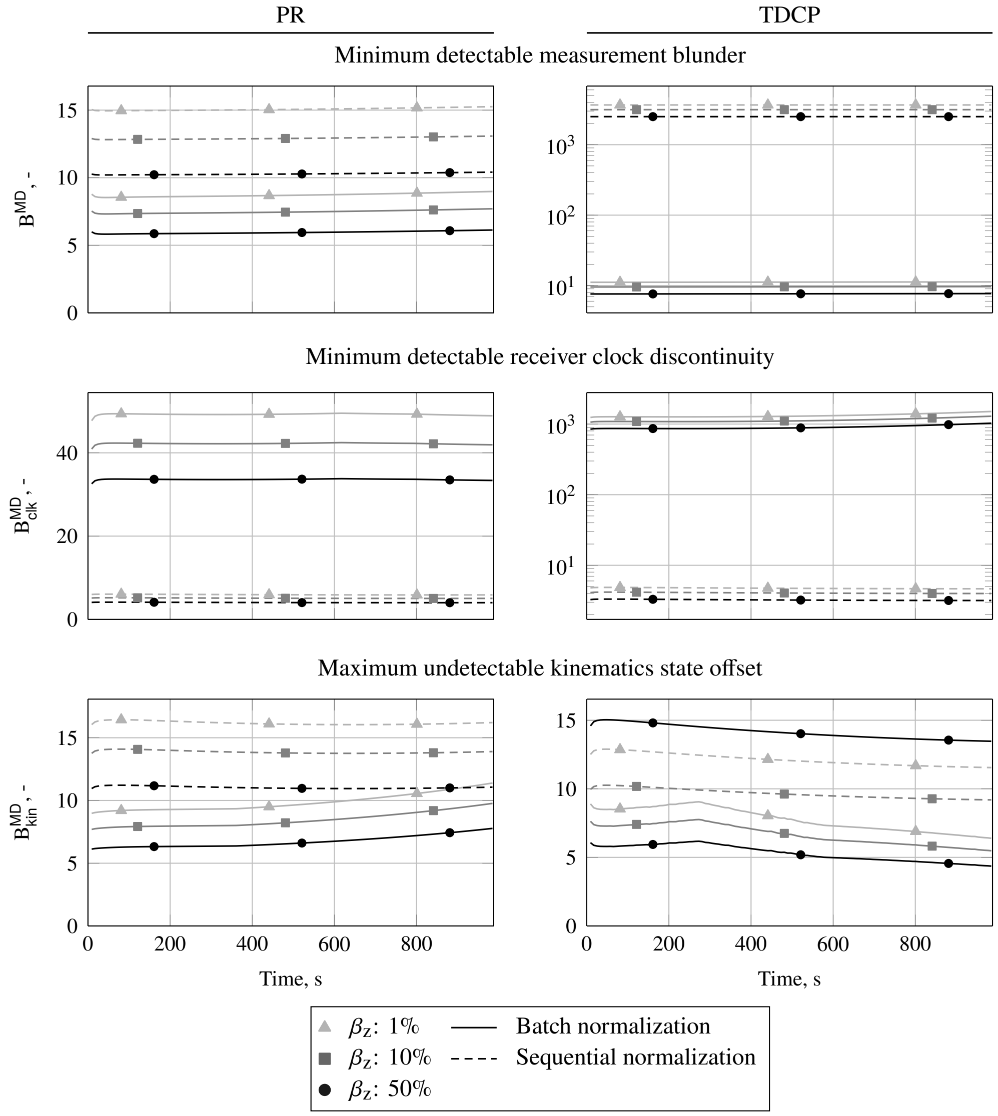


Figure 5.16: Detection limits of the normalized innovation test (batch and sequential methods) for all considered failure modes of PR and TDCP, and different probabilities of missed detection (β_z)

GNSS-only LS residual test threshold

In nominal conditions the LS residual (or inter-channel coherence) test statistic, $T^2_{\varepsilon,k}$, is χ^2-distributed as described in (5.20). Threshold $n_{1-\alpha_\varepsilon}$ sets the false alarm rate of this test (see Figure 5.3). Because this test is here used for identification and not detection, a false alarm and a missed-detection are actually failed identification events. α_ε is the probability of false alarm, i.e., of wrongly declaring a channel inconsistency.

A level of

$$\alpha_\varepsilon = 2\% \tag{5.56}$$

is selected as a trade-off between false alarm rate and coherence with the minimum detection limits of the innovation test. A low threshold renders the consistency test more sensitive, capturing a greater part of the single-channel events that trigger the

Table 5.3: Thresholds of 2% false alarm for χ^2-distributions of several DoFs

$n_k - m_{\mathsf{LS}}$	1	2	3	4	5	6	7	8
$\mathrm{n}_{1-\alpha_\varepsilon}$	5.41	7.82	9.84	11.67	13.39	15.03	16.62	18.17

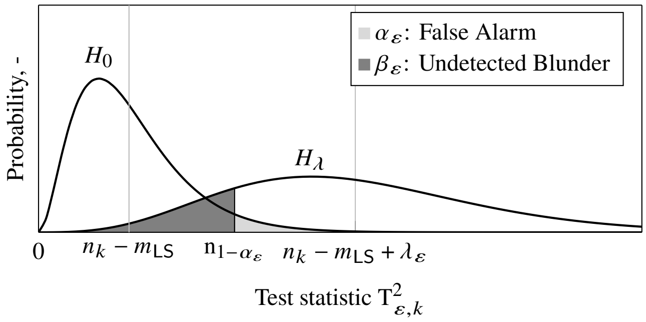


Figure 5.17: Probability density function of the LS residual squared norm test statistic $\mathrm{T}^2_{\varepsilon,k}$ under both unbiased and biased conditions

innovation test; but it also increases the chance of false positive inconsistency results. The chosen α_ε translates to the thresholds shown in Table 5.3 for χ^2-distributions of DoF from 1 to 8 (i.e., for a number of measurements from 5 to 12).

Under single-channel failure the test statistic has a non-central χ^2-distribution [hypothesis (5.21)] as shown in Figure 5.17 for a 6-DoF distribution.

It is possible to relate the size of an hypothetical single-channel blunder with the resulting non-centrality λ_ε. Plugging the perturbed model (5.4) to (5.22) gives

$$\mathrm{T}^2_{\varepsilon,k} = \left(\mathbf{m}_i \Delta_{i,k} + \boldsymbol{\upsilon}_k\right)^{\mathsf{T}} \mathbf{L}_{\upsilon,k}^{-\mathsf{T}} \mathbf{U}_{2,k} \mathbf{U}_{2,k}^{\mathsf{T}} \mathbf{L}_{\upsilon,k}^{-1} \left(\mathbf{m}_i \Delta_{i,k} + \boldsymbol{\upsilon}_k\right) \ . \tag{5.57}$$

Note that the approximate orthogonality of the innovation model term $\mathbf{H}_k \mathbf{e}_k$ [in (5.4)] to $\mathbf{U}_{2,k}^{\mathsf{T}} \mathbf{L}_{\upsilon,k}$ [shown in (5.18)] makes it vanish. Taking the expected value of $\mathrm{T}^2_{\varepsilon,k}$ yields

$$\begin{aligned} E\left\{\mathrm{T}^2_{\varepsilon,k}\right\} &= \mathsf{tr}\left(\mathbf{U}_{2,k}^{\mathsf{T}} \mathbf{L}_{\upsilon,k}^{-1} E\left\{\left(\mathbf{m}_i \Delta_{i,k} + \boldsymbol{\upsilon}_k\right)\left(\mathbf{m}_i \Delta_{i,k} + \boldsymbol{\upsilon}_k\right)^{\mathsf{T}}\right\} \mathbf{L}_{\upsilon,k}^{-\mathsf{T}} \mathbf{U}_{2,k}\right) \\ &= \mathsf{tr}\left(\mathbf{U}_{2,k}^{\mathsf{T}} \mathbf{L}_{\upsilon,k}^{-1} \left(\mathbf{R}_k + \Delta_{i,k}^2 \mathbf{m}_i \mathbf{m}_i^{\mathsf{T}}\right) \mathbf{L}_{\upsilon,k}^{-\mathsf{T}} \mathbf{U}_{2,k}\right) \\ &= \left(n_k - m_{\mathsf{LS}}\right) \ + \ \underbrace{\Delta_{i,k}^2 \mathbf{m}_i^{\mathsf{T}} \mathbf{L}_{\upsilon,k}^{-\mathsf{T}} \mathbf{U}_{2,k} \mathbf{U}_{2,k}^{\mathsf{T}} \mathbf{L}_{\upsilon,k}^{-1} \mathbf{m}_i}_{\lambda_\varepsilon} \ , \end{aligned} \tag{5.58}$$

isolating the non-centrality term λ_ε. And, if $\mathbf{R}_k$ is diagonal, this term is reduced to

$$\lambda_\varepsilon = \frac{\Delta_{i,k}^2}{\mathbf{m}_i^{\mathsf{T}} \mathbf{R}_k \mathbf{m}_i} \mathbf{m}_i^{\mathsf{T}} \mathbf{U}_{2,k} \mathbf{U}_{2,k}^{\mathsf{T}} \mathbf{m}_i \ . \tag{5.59}$$

The trace of $\mathbf{U}_{2,k} \mathbf{U}_{2,k}^{\mathsf{T}}$ is equal to the number of DoFs of the distribution (i.e., $n_k - m_{\mathsf{LS}}$) [57], and its diagonal elements range from 0 to $(n_k - m_{\mathsf{LS}})$, depending on the measurement geometry, averaging $\left(n_k - m_{\mathsf{LS}}\right) / n_k$. A low value of one of these elements is associated with low redundancy on the direction of that particular measurement, and thus with a low fault detectability along that axis.

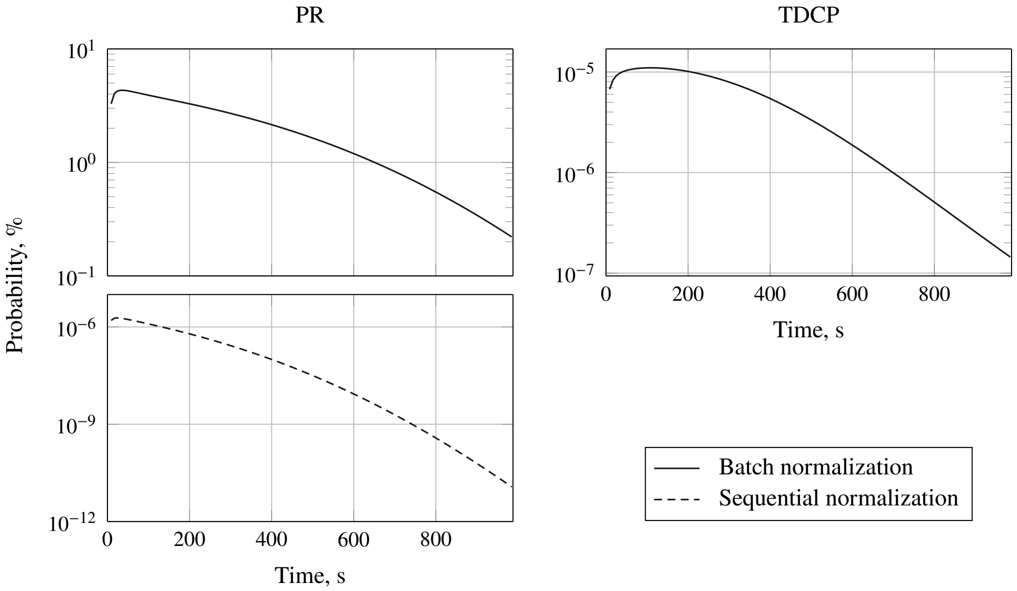


Figure 5.18: Maximum probability of missed detection β_ε (type II error) by the LS residual test of a blunder of magnitude equal to the minimum detection level ($\beta_\mathrm{z} = 1\%$) of the innovation test

Expression (5.58) [or (5.59)] allows for the direct evaluation of detection performance of the T^2_ε-test under minimum detection limits of the innovation test. Note that the T^2_ε-test is only performed under multiple innovation violations.

Figure 5.18 shows, for PR and TDCP and for both batch and sequential innovation normalization methods, the maximum probability of type II error β_ε under single-channel blunder of magnitude equal to the MD blunder of the innovation test ($\beta_\mathrm{z} = 1\%$, Figure 5.16). Note that the minimum detectable blunders by the sequentially normalized test statistic are high enough to yield near 100% chance of detection by the LS residual squared norm test. (In the TDCP case this is virtually 100%, being thus not worth displaying.) The minimum detectable blunders of the batch normalized strategy are considerably lower, making them harder to detect by the T^2_ε test. Indeed, the maximum probability of missed detection for the PR set reaches nearly 5%. On the TDCP side, because the minimum detectable blunders (even those of the batch normalized innovation test) are slightly higher, their probability of missed detection by this LS residuals test is far lower ($\lesssim 10^{-5}\%$). This generally high sensitivity of the LS residual test is key to avoid innovation triggers from yielding inconclusive or incorrect FDI results. Naturally, situations of deficient SV-geometry may lead to higher type II error probabilities under blunders in channels with low redundancy.

GNSS-only LS solution test threshold

As seen in Section 5.3.4, the LS normalized solution components $\mathrm{t}_{\Delta,j,k}$ follow standard normal distributions. Once again, defining the test threshold $\mathrm{n}_{1-\alpha_\Delta/2}$ will set the significance level of the test, α_Δ. This time a the limit is chosen as $\mathrm{n}_{1-\alpha_\Delta/2} = 3$ (i.e, 3 σ), which yields

$$\alpha_\Delta \approx 0.3\ \% \ , \tag{5.60}$$

to prevent inconclusive outcomes while maintaining a high confidence level (~ 99.6%).

In case of a kinematics state or receiver clock fault the test statistics affected becomes biased as described by hypothesis H_λ in (5.34). (See Figure 5.15 for a graphical illustration of the shift in the statistical profile.) The mean of this non-centred normal distribution can be given in terms of the offset/discontinuity as

$$\lambda_{\Delta,k} = E\left\{\mathrm{t}_{\Delta,k}\right\} = \frac{\left(\mathbf{L}_{\Delta,k}^{-1}\mathbf{V}_k^{\mathsf{T}}\mathbf{m}_l\right)^{\mathsf{T}}}{\left\|\mathbf{L}_{\Delta,k}^{-1}\mathbf{V}_k^{\mathsf{T}}\mathbf{m}_j\right\|}\mathbf{L}_{\Delta,k}^{-1}\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\bar{\mathbf{H}}_k\mathbf{m}_l\mathbf{m}_l^{\mathsf{T}}\begin{bmatrix}\boldsymbol{\Delta}_{\mathsf{kin},k}\\ \Delta_{\mathsf{clk},k}\end{bmatrix} \tag{5.61}$$

$$= \left\|\mathbf{L}_{\Delta,k}^{-1}\mathbf{V}_k^{\mathsf{T}}\mathbf{m}_l\right\|\mathbf{m}_l^{\mathsf{T}}\begin{bmatrix}\boldsymbol{\Delta}_{\mathsf{kin},k}\\ \Delta_{\mathsf{clk},k}\end{bmatrix}, \tag{5.62}$$

for the batch normalization of the LS solution, and

$$\lambda_{\Delta,k} = E\left\{\mathrm{t}_{\Delta,k}\right\} = \frac{\mathbf{m}_l^{\mathsf{T}}\mathbf{V}_k^{-\mathsf{T}}\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}}{\sqrt{\mathbf{m}_l^{\mathsf{T}}\mathbf{P}_{\Delta\Delta,k}\mathbf{m}_l}}\bar{\mathbf{H}}_k\mathbf{m}_l\mathbf{m}_l^{\mathsf{T}}\begin{bmatrix}\boldsymbol{\Delta}_{\mathsf{kin},k}\\ \Delta_{\mathsf{clk},k}\end{bmatrix} \tag{5.63}$$

$$= \frac{1}{\sqrt{\mathbf{m}_l^{\mathsf{T}}\mathbf{P}_{\Delta\Delta,k}\mathbf{m}_l}}\mathbf{m}_l^{\mathsf{T}}\begin{bmatrix}\boldsymbol{\Delta}_{\mathsf{kin},k}\\ \Delta_{\mathsf{clk},k}\end{bmatrix}, \tag{5.64}$$

for the sequential normalization of the solution. Both simplified expressions are obtained noting that $\mathbf{U}_{1,k}^{\mathsf{T}}\mathbf{L}_{\upsilon,k}^{-1}\bar{\mathbf{H}}_k = \mathbf{V}_k^{\mathsf{T}}$.

Using these expressions the minimum detectable (or identifiable, in this case) strapdown offset and clock discontinuity can be estimated. As λ_z, the non-centrality λ_Δ is computed based on the threshold $\mathrm{n}_{1-\alpha_\Delta/2}$ and on a minimum detection level β_Δ. Figure 5.19 compares the minimum detectable levels (type II error probability of 1%) of each component of the LS solution test against the minimum detectable equivalent by the innovation test for PR and TDCP. In both tests, results from batch and sequential normalization methods are displayed. The values are factorized as done in (5.52) and (5.55) (albeit here each kinematics state component is factorized separately).

As can be seen, in all solution components the batch normalized LS statistic results in detection levels either lower than (in the PR case) or nearly identical to (TDCP case) the sequentially normalized one. This contrasts with the results for the normalized innovation test, where large differences in detection levels arise between normalization methods. This justifies the use of only sequential normalization in the LS solution test.

Finally, in all cases the minimum detection limit of the LS solution test for each normalization method (batch or sequential) is either close to or lower than that of the innovation test with the same method. This reveals a higher identification sensitivity with respect to detection which is key to the minimization of inconclusive FDI results.

5.6 Computational load analysis

To evaluate the computational load required by the developed FDIR scheme, FLOP count analysis, similar to that done in Section 4.4.5 for the different filter versions, can be performed. The FDI and recovery schemes are as described in Sections 5.4.2 and 5.4.3, respectively, and as shown in Figures 5.5 and 5.6, also respectively. The

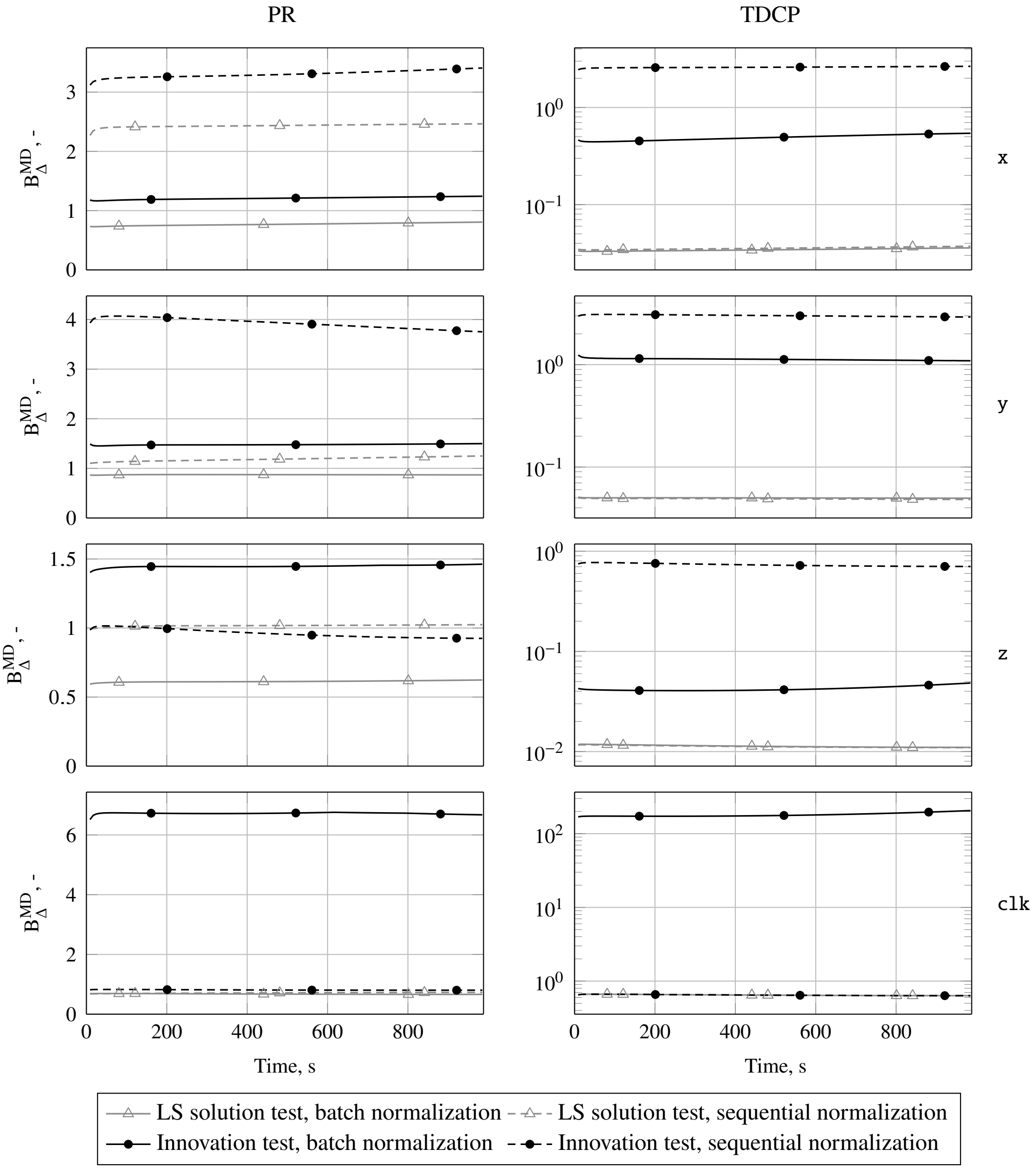


Figure 5.19: Minimum detection levels ($\beta = 1\%$) of all components of the least-squares solution test (PR and TDCP) compared to the equivalent minimum detection levels of the innovation test

FLOP counts for each matrix/vector operation are as listed in Table 4.6, and compared to the FLOP count of one update cycle of the medium-grade IMU filter in Table 4.7.

The total FLOP counts for each test within the PR and TDCP FDI chains, and for the subsequent recovery step are displayed in Table 5.4 under correct detection/recovery of different blunder occurrences. The numbers shown are for 12 active channels, and the total counts assume that any blunders affect both PR and TDCP simultaneously. Note that these assumptions lead to a worst case operations count.

The displayed FLOP counts show a very low nominal (i.e., no failure) overhead of around 1% of the filter update cycle burden. The sequential-only normalization method used by the PR FDI is far lighter than the batch scheme used by the TDCP

Table 5.4: Single-call FLOP counts of the FDI chains (PR and TDCP) under different blunders

Step	FLOP count				
	No failure	Single-channel blunder	Double-channel blunder	Kinematics blunder	Clock blunder
			PR chain		
Innovation test (t_z)	24	24	24	24	24
LS residual test (T_ε^2)	0	0	4777	1910	1910
LS solution test (t_Δ)	0	0	0	3652	3652
Total PR chain	24	24	4801	5586	5586
			TDCP chain		
Innovation test (t_z)	3134	3134	3134	3134	3134
LS residual test (T_ε^2)	0	0	9125	1910	1910
LS solution test (t_Δ)	0	0	0	3652	3652
Total TDCP chain	3134	3134	12259	8696	8696
			Recovery		
Recovery cov. ($\Delta\sigma^2$)	0	0	0	24	4
Total (% of filter update count*)	3158 **(1.04)**	3158 **(1.04)**	17060 **(5.63)**	14306 **(4.73)**	14286 **(4.72)**

* Of the medium-grade IMU filter version in Table 4.7

chain. Under a contingency, the maximum load is still relatively low (at about 5.6% under two simultaneous receiver channel blunders). This occurrence yields the worst case count for the TDCP FDI chain due to the innovation renormalization (by batch method) required in every iteration of the LS residual test (see diagram in Figure 5.5). In the PR FDI chain, worst case count happens for the receiver clock and filter kinematics state failures, as could be expected. Finally, recovery is rare and relatively light, with that from kinematics state fault having considerably higher load (as all components of position and velocity are recovered in this case *versus* a single receiver clock bias).

5.7 Performance evaluation

After verification and tuning of the FDIR design in static state, testing is done in a flight scenario with real GNSS data.

5.7.1 Trajectory and simulation models

The trajectory followed is a smoothed version of the real flight Vega VV04 trajectory (see Appendix B). The real GNSS data is produced, as before, by a DLR Phoenix-HD receiver fed by a GNSS emulator (SPIRENT GSS7700). The inertial measurements are simulated through the high-fidelity model described in Section 2.4.3 and derived in Appendix Section C.2. A medium grade sensor profile is used (see Table 2.1).

5.7.2 Detection and identification

The first test runs the FDI algorithm on the reduced-order filter innovations under no injected blunders. To assess the false alarm rates of all detection and identification levels

Table 5.5: False alarm rates of all tests under no injected blunders

Occurrence	PR	TDCP
False alarm of channel blunder (t_z test)	0 of 9998 (0%)	0 of 9982 (0%)
False alarm of channel set inconsistency (T^2_ε test)	20 of 993 (2.01%)	67 of 992 (6.75%)
False alarm of LS solution blunder (t_Δ test)	9 of 3972 (0.23%)	7 of 3968 (0.18%)

Table 5.6: Single and double channel blunder ($\beta_z = 1\%$) detection results

Occurrence	Single channel blunder		Double channel blunder	
	PR	TDCP	PR	TDCP
Correct detection	992 (99.2%)	991 (99.1%)	990 (99.0%)	804 (80.4%)
Partial detection	–	–	9 (0.9%)	60 (6.0%)
Missed detection	8 (0.8%)	9 (0.9%)	1 (0.1%)	136 (13.6%)

(normalized innovations, LS residuals, and LS solution tests) each test is performed at every measurement epoch, i.e., the conditional cascade of testing is removed.

Table 5.5 displays the obtained false alarm occurrences and their respective rates. Note that the number of times each test is perform depends on the number of measurement sets available and on the number of active channels on each set The number of PR and TDCP measurement sets differ by one due to the impossibility to form a TDCP measurement on the very first GNSS delivery epoch (two successive carrier phase measurements are needed to form a TDCP sample). The normalized innovation (t_z) test retrieved no false alarms for both measurement types. This is expected given the 5σ threshold selected (nominal false alarm porbability of 5.7e-5%, see Section 5.5.2). The LS residual norm squared (T^2_ε) test also yield a false alarm rate closely matching the nominal selected level ($\alpha_\varepsilon = 2\%$, Section 5.5.2) for the PR measurement chain. For the TDCP measurements, however, the resulting rate (6.75%) clearly exceeded this nominal value. This had been anticipated during the signal profiling (with real GNSS data) in Section 5.5.1, where the probability distribution of the computed T^2_ε spread wider than its nominal distribution curve. Interestingly, although this distribution mismatch had been seen in the static data also for PR, the current flight data does not suffer from it. The LS solution (t_Δ) test resulted in false alarm rates close (though slightly lower than) the design value ($\alpha_\Delta = 0.3\%$, Section 5.5.2) for both measurement chains.

The detection of GNSS channel blunders is tested by randomly adding 1000 failure events to the measurement data, 50 at a time over 20 FDI runs. This test is performed separately to PR and TDCP sets and for both single- and double-occurring blunder events (50 singles or 50 pairs per run). The blunders have a magnitude of 99% nominal detectability (i.e., $\beta_z = 1\%$). Table 5.6 displays the overall detection results.

The obtained single-blunder detection rate matches the expected level of detectability of the generated outliers in both PR and TDCP chains (i.e., around 99%). The double-blunder detection showed results inline with the expected in the PR chain. For the TDCP data set, however, detection proved far more difficult, with much higher rates of partially detected and undetected blunder pairs. The reason for this lies in the strong correlation between channels caused by the receiver clock noise, which requires the use of batch-normalized innovations in the detection of channel blunders (as shown in Section 5.5.2). This normalization removes the common-mode error but spills the effect of a channel blunder to other components of t_z. Under a single blunder the

Table 5.7: Receiver clock and kinematics state blunders (β_z = 1%) detection results

Occurrence	Receiver clock blunder		Kinematics state blunder	
	PR	TDCP	PR	TDCP
Correct detection	979 (97.9%)	936 (93.6%)	981 (98.1%)	932 (93.2%)
Missed detection	21 (2.1%)	64 (6.4%)	19 (1.9%)	68 (6.8%)
Inconclusive identification	0 (0%)	0 (0%)	0 (0%)	0 (0%)

normalized innovation component corresponding to the offending channel will (in the great majority of cases) still hold the highest value among the t_z set; however, in the case of multiple outliers their combined effect on sane normalized components can cause one of these to hold the maximum t_z of the set. Under this scenario the detection algorithm, which iteratively eliminates the highest (absolute-value-wise) normalized innovation, is unable to converge to the correct outcome. A way to overcome this would be to perform the measurement consistency test under elimination of each innovation separately, as it is commonly done in traditional GNSS RAIM schemes [30]. The fully/partially successful detection rate obtained by the current algorithm (about 86%) is still acceptable given the low likelihood of a double-channel failure.

Finally, receiver clock and filter kinematics state failure detections are tested by randomly adding 1000 artificial blunders (of each failure type) to both PR and TDCP data sets as done above for the channel blunder detection test. Results are shown in Table 5.7. Again, the outliers (99% detectability, i.e., β_z = 1%) are split between 20 runs (50 per run per type and per chain). The kinematics state blunders are maximum undetectable blunders [see definition in Section 5.5.2, expressions (5.53)–(5.54)], and thus potentially affect all three Cartesian components of the LS solution space.

To be successfully detected and identified by the proposed algorithm, a failure of this type has to be detected on at least two components of the normalized innovations test, then test negative for inter-channel inconsistency (i.e., T_ε^2 or LS residuals test), and then finally cause a violation in the correct component of the LS solution test. In this test cascade the evaluation with the highest level of missed detection is the T_ε^2, at 2%. (Note that, in this case, a missed detection is a violation of the test threshold and thus equivalent to a false alarm.) Indeed, this is the missed detection level obtained by the PR FDI chain reported in Table 5.7 for both failure types. The TDCP data generates a lower success level under the two failures due to the statistical distribution mismatch of the T_ε^2 quantitiy in this data set (as above explained). This level roughly matches the false alarm rate of the T_ε^2 test in Table 5.5. In none of the four cases an inconclusive identification was reached, i.e., a type II error of the LS solution test occurred.

5.7.3 Recovery

The recovery strategy described in Sections 5.3.5 and 5.4.3 can be tested by artificially inducing severe receiver clock and inertial errors in the measurement data.

Figure 5.20 shows the kinematics state estimation (velocity, position, and attitude) of the proposed reduced order filter with and without FDIR in the case of a GNSS receiver clock jump with a magnitude 50 times higher than the minimum detectable blunder of the PR chain (99% detectability, see Figure 5.16). As in the previous section, the GNSS data is produced by a Phoenix-HD receiver stimulated with signals

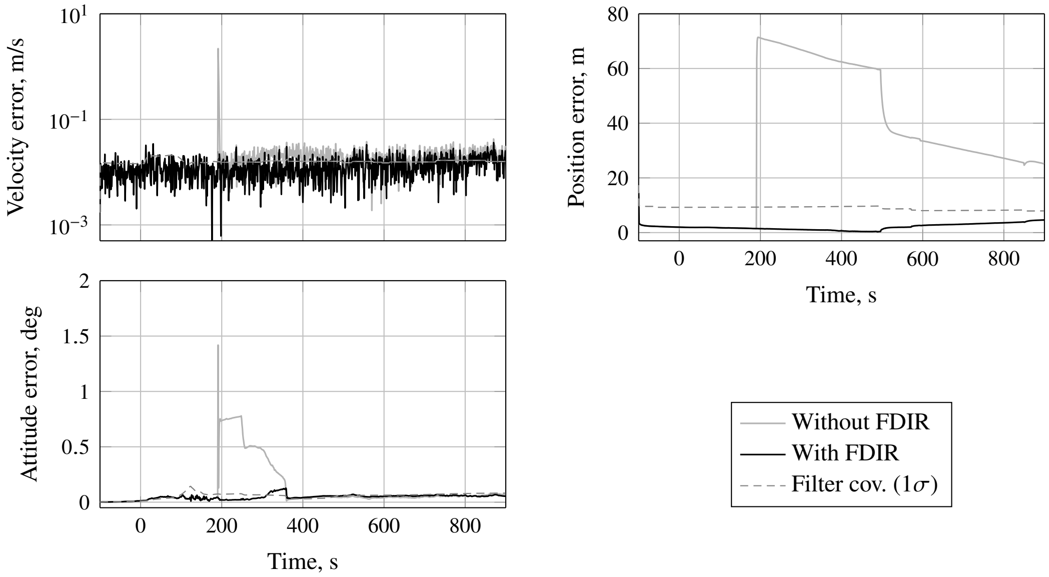


Figure 5.20: Kinematics state estimation of the medium-grade reduced-order filter with and without FDIR under severe GNSS receiver clock bias jump on a Vega VV04 trajectory

emulating a Vega VV04 trajectory. Comparing the estimation error curves of both filter configurations, the clock jump can be clearly seen at around 200 s in that without FDIR. Velocity, position and attitude errors peak to about 2 m/s, 70 m and 1.4 deg, respectively. While, velocity shows only small effects after the jump, attitude and especially position take far longer to converge to under the filter-predicted 1σ bound and effectively to the (coherent) error levels displayed by the FDIR-enabled filter set-up. In fact, the filter with FDIR shows seemingly no effects from the clock failure.

The same type of testing can be used to evaluate the active recovery from a filter kinematics state failure. As described in Section 5.3.3, this type of failure can be cause, for instance, by sudden malign dynamics (e.g., vibration and shock) or by sudden changes in the error profile of the inertial sensor (due, e.g., to such strong dynamics). The effects of these events are far worse in the absence of aiding by GNSS measurements. Therefore, to test the response to a severe scenario, a sudden change of IMU bias and scale-factor levels (by a 10σ magnitude) is temporarily and artificially induced during a very prolonged GNSS outage (2 minutes). The estimation errors of the proposed filter with and without FDIR is displayed in Figure 5.21. As can be seen, the effect is indeed rather severe, with estimation errors in velocity and position rising about a decade above the filter-predicted covariance during the outage. Upon reacquisition of GNSS measurements the failure is detected and active recovery is performed by the FDIR-enabled set-up. Promptly the error curves of this filter return to within coherent levels. Contrarily, those of the FDIR-disabled configuration wander far from 1σ bounds only slowly converging. This behavior can be better understood looking at the estimated accelerometer bias and scale-factor states in Figure 5.22. A significant amount of the large innovation at GNSS reacquisition is channeled by the FDIR-disabled filter into these IMU states due to insufficient uncertainty in the main kinematics states (mainly, velocity and position), severely misestimating these quantities. This in turn perturbs the strapdown integration from that point on, delaying kinematics convergence. In the FDIR-enabled filter the estimation of these IMU states is relatively unperturbed, attesting to the success of the covariance injection by the recovery scheme proposed.

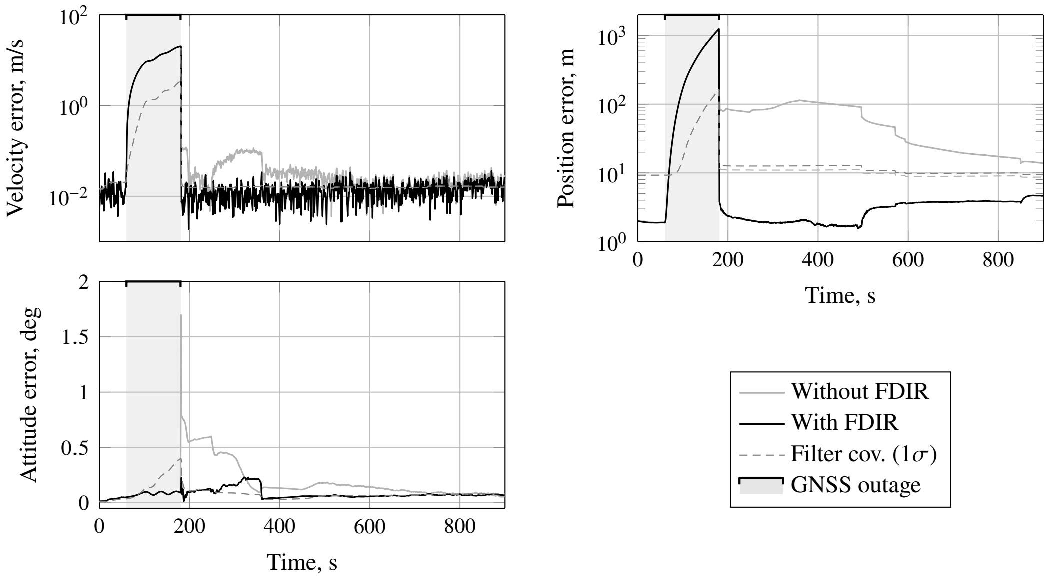


Figure 5.21: Kinematics state estimation of the medium-grade reduced-order filter with and without FDIR under severe IMU error change during GNSS outage on a Vega VV04 trajectory

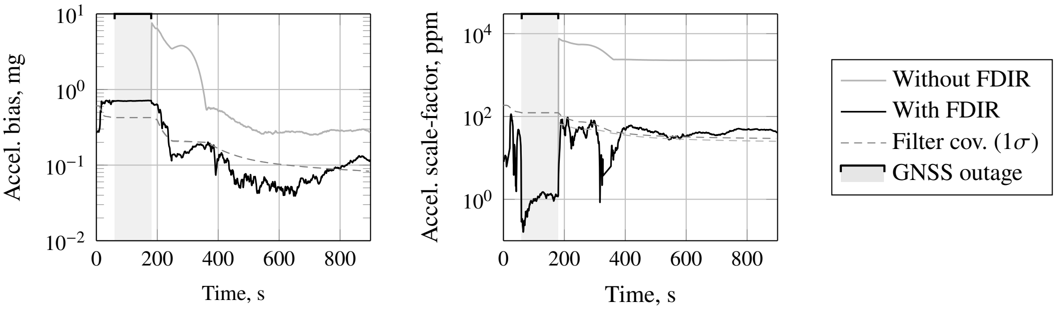


Figure 5.22: Accelerometer state estimation with and without FDIR under severe IMU error change during GNSS outage

6

Navigation Software

Having demonstrated the features and performance of the proposed navigation system design thus far in a MATLAB-code-based implementation (see Chapters 3 and 4), in this chapter a real-time-capable implementation of this system is developed and tested.

6.1 Motivation

Aiming at raising the readiness level of the developed system, the algorithm proposed in the previous chapters is here rewritten in real-time capable software form using C/C++ code. Important real-time system features such as GNSS measurement output latency and Navigation Computer (NC) internal clock error are also considered, modeled in the simulation environment, and tackled within the filter design. Furthermore, evidence of real-time capability is drawn from computational complexity analysis and comparison to the flight-proven HNS design [161, 164].

6.2 Dealing with measurement latency

As alluded to in Section 2.3.3, GNSS measurements are output with significant latency with respect to the time instant which they describe. Latencies of up to 500 ms are common [157], which under the high velocity of a launcher easily produce hundreds of meters of error. This latency can be dealt with within the filter in several ways:

1. Using the state transition matrix pertaining to the elapsed latency period (from pulse per second (PPS) output – regarded as instantaneous – to GNSS measurement output) the past state instance can be mapped within the measurement update equation (see, e.g., [38] and a more general realization in [12, 153]);

2. In a simpler approach, extrapolating the delayed measurement to the time of assimilation using the estimated state change during the latency period [97, 9];

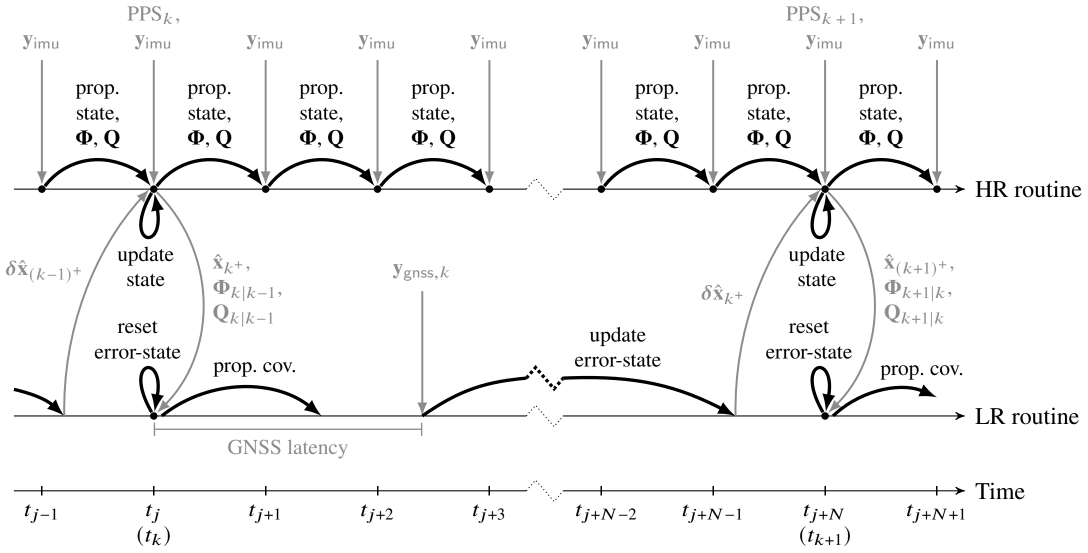

PPS$_k$,
y_{imu}
prop. state, $\mathbf{\Phi}$, $\mathbf{Q}$
PPS$_{k+1}$,
HR routine
$\delta\hat{\mathbf{x}}_{(k-1)^+}$
update state
reset error-state
$\hat{\mathbf{x}}_{k^+}$, $\mathbf{\Phi}_{k|k-1}$, $\mathbf{Q}_{k|k-1}$
prop. cov.
$y_{gnss,k}$
update error-state
$\delta\hat{\mathbf{x}}_{k^+}$
$\hat{\mathbf{x}}_{(k+1)^+}$, $\mathbf{\Phi}_{k+1|k}$, $\mathbf{Q}_{k+1|k}$
LR routine
GNSS latency
Time
t_{j-1}
t_j (t_k)
t_{j+1}
t_{j+2}
t_{j+3}
t_{j+N-2}
t_{j+N-1}
t_{j+N} (t_{k+1})
t_{j+N+1}

Figure 6.1: Timing diagram of implemented delayed error-state filter (loosely adapted from [159])

3. Augmenting the filter with a delayed state portion referring to the time of GNSS measurement collection (approximated by the time of PPS reception) allows the measurement update to happen at a later stage with a the regular structure [4, 16];

4. Delaying the entire error-state filter (state and covariance) by an amount greater than the maximum expected GNSS output latency allows the update to be done once the measurement is available (using a copy of the whole-state at the PPS time) and as if it were on time; the updated error-state can then be fast-forwarded with the elapsed transition matrix and used to correct the real-time (non-delayed) whole-state propagation (strapdown process, in this case) [159].

While all the approaches above can result in optimal filters for the problem at hand, they all carry varying levels of increased computational burden with respect to a latency-neglecting filter. The last solution (4.) requires the least extra effort both implementation- and computation-wise. This is similar to the method used by the HNS: error-state filter delayed by one low rate (LR) cycle (1 s) with respect to the near real-time high rate (HR) thread that performs the whole-state (strapdown) propagation.

Figure 6.1 depicts the time sequence of the selected dual-rate, delayed filter approach. Starting at t_k, upon reception of a PPS trigger from the GNSS receiver, the HR strapdown routine passes the up-to-date whole-state $\hat{\mathbf{x}}_{k^+}$ to the LR filter routine together with the state transition matrix $\mathbf{\Phi}_{k|k-1}$ and system process noise covariance matrix $\mathbf{Q}_{k|k-1}$ relative to the elapsed period from the previous to the current PPS. The error-state filter has its state reset (to zero) and its covariance propagated ($\mathbf{P}_{(k-1)^+}$ to $\mathbf{P}_{k^-}$). When the GNSS measurement belonging to t_k is received, the error-state filter is updated, yielding $\delta\hat{\mathbf{x}}_{k^+}$, $\mathbf{P}_{k^+}$. The updated error-state $\delta\hat{\mathbf{x}}_{k^+}$ is then fed back to the strapdown process where at the next PPS epoch (t_{k+1}) the propagated whole state $\hat{\mathbf{x}}_{(k+1)^-}$ is updated as

$$\hat{\mathbf{x}}_{(k+1)^+} = \hat{\mathbf{x}}_{(k+1)^-} + \mathbf{\Phi}_{k+1|k}\delta\hat{\mathbf{x}}_{k^+} \; . \tag{6.1}$$

The error-state is then reset,

$$\delta\hat{\mathbf{x}}_{k^+} \leftarrow \mathbf{0} \; , \tag{6.2}$$

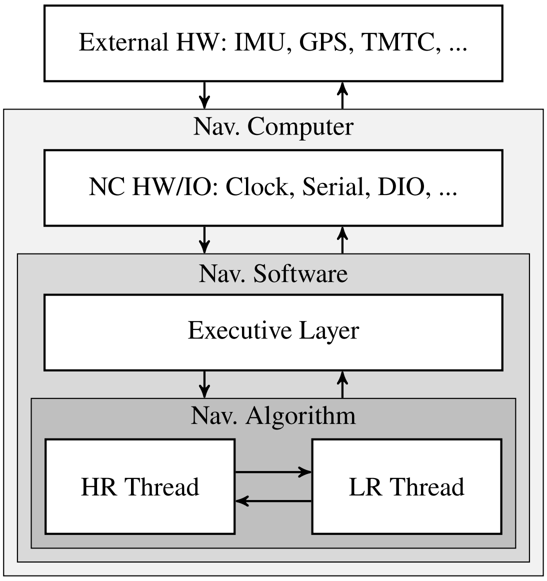


Figure 6.2: Software top-level diagram within navigation system

closing the cycle. The above mentioned state transition and process noise covariance matrices are integrated within the HR routine as described in Sections 3.7 and 4.3 [expressions (3.48)–(3.54), and (4.37)]. Both are reset upon delivery to the LR routine. Note that the depicted relative lengths of the filter covariance propagation and measurement update processes are arbitrary, and only intended to convey independence with respect to the length of the HR cycle.

The delayed filter just described is identical to that proposed in [159] and differs from the method used within the HNS (see, e.g., [161]) in that it forward-propagates the updated error-state before adding it to the *a priori* whole-state [as in (6.1)].

6.3 Software architecture and implementation

As mentioned in the beginning of this chapter, the implemented real-time software is based on the HNS flight code developed for its SHEFEX-2 version (see, e.g., [161, 160]). This code is written in C and C++ languages and is composed of two main blocks: the executive layer and the navigation algorithm. Figure 6.2 depicts these software components and their interface with other navigation computer functions (e.g., clocks and hardware interfaces) and with external hardware such as sensors, and Telemetry and Telecommand (TMTC) systems.

The executive thread manages a series of low-level tasks such as interfacing with sensors and TMTC via Serial connections (RS422), and with trigger signals through digital input/output (DIO). It also interfaces with the NC clock, recording time-stamps of all messages and triggers (sent and received), and saving all raw serial data to the NC hard drive and external mass memory units.

The navigation algorithm is composed of a HR thread that runs whenever a new IMU message is received, and a LR thread that runs asynchronously at 50 Hz. The tasks performed by each of these threads as well as communication between them is further explained below, in Section 6.3.2.

While the executive layer is essentially unchanged with respect to the HNS code version, the navigation algorithm (i.e., HR/LR threads) has been heavily modified to accommodate the algorithms proposed in this work. In addition to changes in the top layer of these threads (described in Section 6.3.2), further changes include:

- a Consider Kalman Filter (KF) library, implementing the general filter presented in Chapter 4 in a modular fashion;
- new update methods for PRs and TDCPs measurements including an atmospheric correction model library (with methods as described in Chapter 3, and for static pseudo-measurements (zero velocity (ZV) and Earth rate (ER));
- a FDIR library, implementing the scheme proposed in Chapter 5;
- new strapdown integration (as described in Chapter 3), coordinate frame, and gravity model libraries;
- new GPS time conversion and ephemeris solution libraries; and
- new matrix, vector and quaternion libraries.

6.3.1 Navigation modes and commands

The navigation software has a set of different modes that define the behavior of the program by enabling or disabling different functionalities. Table 6.1 describes all modes defined within the code. As above mentioned, the executive layer is responsible for mode management. Mode transition is done in two ways: automatically, when a certain condition is fulfilled; or manually, through external command. Figure 6.3 shows the mode transition diagram. The typical mode sequence of a flight scenario is described in [160] and is as follows:

1. At start-up the navigation software is in `INITIALIZATION` mode;
2. After setting the required parameters (by external command) and upon external `RESET` command, it transitions to `RESET` mode;
3. Once reset is complete the software automatically switches to `ALIGNMENT` mode;
4. It remains in `ALIGNMENT` mode (meant for on-ground operation) until an external mode transition command is received or until a thrust acceleration is sensed, upon which the software switches to `FLIGHT` mode;
5. When in `FLIGHT` if the vehicle dynamics exceeds the range of the IMU (i.e., when the sensor saturates), the software goes into `SAFE` mode;
6. Before power to system is shut off the system is commanded to `SHUTDOWN`.

In addition to the modes used in the flight sequence above, a `TEST` mode (see Table 6.1) can be used on-ground during integration and testing.

The implemented software commands, sent by the Electrical Ground Support Equipment (EGSE) to the NC, are listed in Table 6.2.

Table 6.1: Navigation software modes [160]

Mode	Description
INITIALIZATION	All initialization tasks are performed (initializes memory blocks, defines variables and computes initial values) before the software can transition to RESET;
RESET	The navigation filter kinematics whole-state values and covariances are (re)set via external command;
ALIGNMENT	The navigation filter runs normally, performing static updates (ZV and ER) in addition to the GNSS updates to allow alignment while on the launch pad;
FLIGHT	The navigation filter runs normally, performing only GNSS updates;
SAFE	Navigation algorithm stops computing navigation solutions, mass memory data storage is stopped and mass memory units are dismounted to protect the data saved thus far;
SHUTDOWN	Program memory is freed and software exits;
TEST	Housekeeping filter data delivery is stopped and all messages to and from the sensors are relayed to the EGSE.

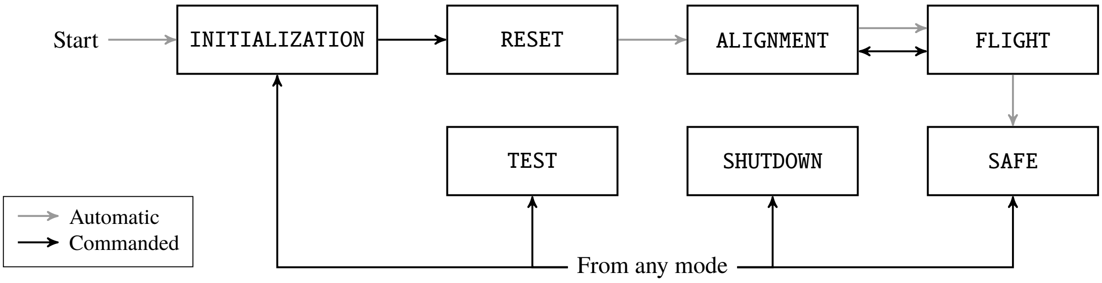


Figure 6.3: Software mode sequence diagram (adapted from [160])

Table 6.2: Navigation Software commands [160]

Command	Description
SETPARAMS	Delivers and sets several filter parameters such as initial IMU and GPS whole-state values and covariances, IMU alignment matrices, static measurement covariances, and GPS antenna to IMU lever arm;
RESET	Delivers and resets the navigation filter kinematics whole-state values and covariances;
Mode transition	Commands a mode transition (see Figure 6.3 for the actual *commandable* transitions).

6.3.2 High rate and Low rate threads

As mentioned at the start of Section 6.3, the navigation algorithm is distributed in two threads: a high rate (HR) one, and a low rate (LR) one. The HR is responsible for the strapdown propagation (integration of kinematics whole-states: position, velocity and attitude), and for the integration of the state transition matrices (see Section 6.2). To this end, it processes the IMU raw samples (correcting for the estimated error quantities: bias, scale-factor, etc.) and maintains the calibration of both the NC and IMU clocks against the GNSS PPS signal. It runs whenever an IMU measurement is available (400 Hz) and is thus called by the executive layer, which manages the DIO and triggering system. The LR thread runs independently at 50 Hz, but only performing meaningful calculations when new HR data is made available to it (once per second upon reception of PPS trigger), or when new GNSS measurement data is received (also once per second). This thread is responsible for the LR filter tasks described in Section 6.2, including covariance propagation, measurement update, and error-state reset. The following points describe the flow of each of these threads in more detail.

HR

The algorithm performed by the HR thread is depicted in Figure 6.4. It begins by buffering the latest trigger times and any available GNSS message. It then processes all newly received commands and requested mode transitions, it processes the latest IMU sample and calibrates the NC and IMU clocks against the latest PPS trigger. If a `RESET` command has been received, the whole-states are reset (the kinematics states to the received values, the remaining to their hard-coded origin), and a request is made for the reinitialization of the error-state side of the filter (on the LR thread). The mode is then set to `ALIGNMENT`. If no `RESET` commands has been received and if the current mode is `ALIGNMENT`, `FLIGHT` or `SAFE`, the strapdown solution is propagated using the processed IMU measurement [as in (3.1)–(3.2) and (3.9)–(3.17)], and the elapsed system transition matrix is integrated [as in (3.48), (3.51)–(3.54), and (4.37)]. If a new PPS trigger has been received, the state propagation is synchronized with the GNSS time, the latest updated error-state sample is fed back to the whole-state set [as in (6.1)], and this freshly updated whole-state is sent to the LR thread. Finally, regardless of the mode, command or reception of a trigger, the navigation solution and housekeeping output package is written and and approved for output.

LR

The LR thread algorithm diagram is shown in Figure 6.5. As above mentioned, the LR threads runs relatively frequently as to quickly react to two events: issuing of new HR-to-LR data; and reception of a new valid GNSS measurement. In the first case, the error-state filter is reinitialized if a request for filter reinitialization has been issued by the HR thread, otherwise normal procedure ensues: the error-state vector is reset [as in (6.2)] and the filter covariance matrix is propagated [as in (4.10)–(4.14)] using the state transition matrix provided by the HR routine. Then, if working in `ALIGNMENT` mode, ZV and ER static filter updates are performed (as described in Sections 3.8.1 and 3.8.2, respectively). The updated error-state is then sent to the HR thread. In the

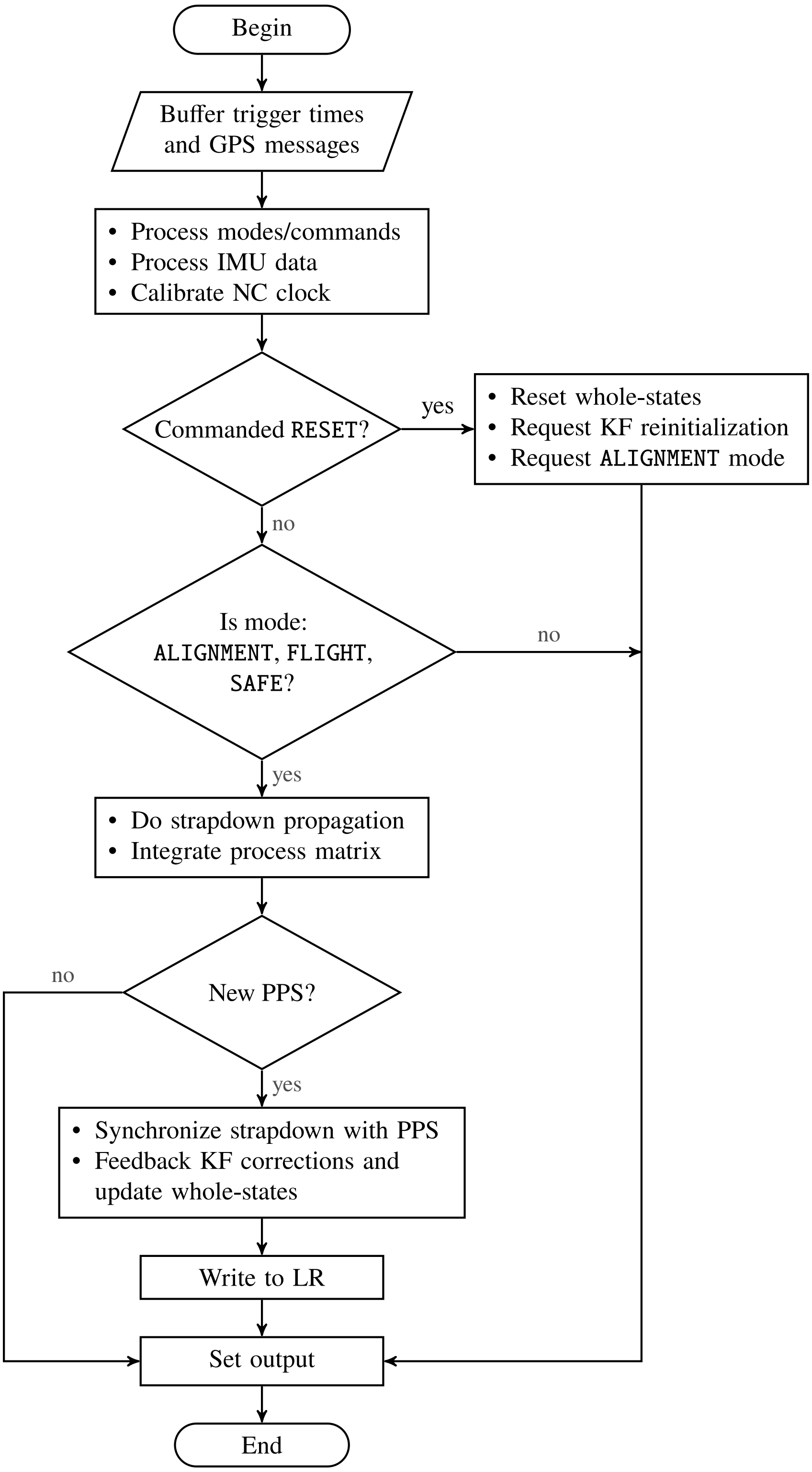

Begin
Buffer trigger times and GPS messages
• Process modes/commands
• Process IMU data
• Calibrate NC clock
Commanded RESET?
yes
• Reset whole-states
• Request KF reinitialization
• Request ALIGNMENT mode
no
Is mode: ALIGNMENT, FLIGHT, SAFE?
no
yes
• Do strapdown propagation
• Integrate process matrix
New PPS?
no
yes
• Synchronize strapdown with PPS
• Feedback KF corrections and update whole-states
Write to LR
Set output
End

Figure 6.4: HR thread flow diagram (loosely adapted from [160])

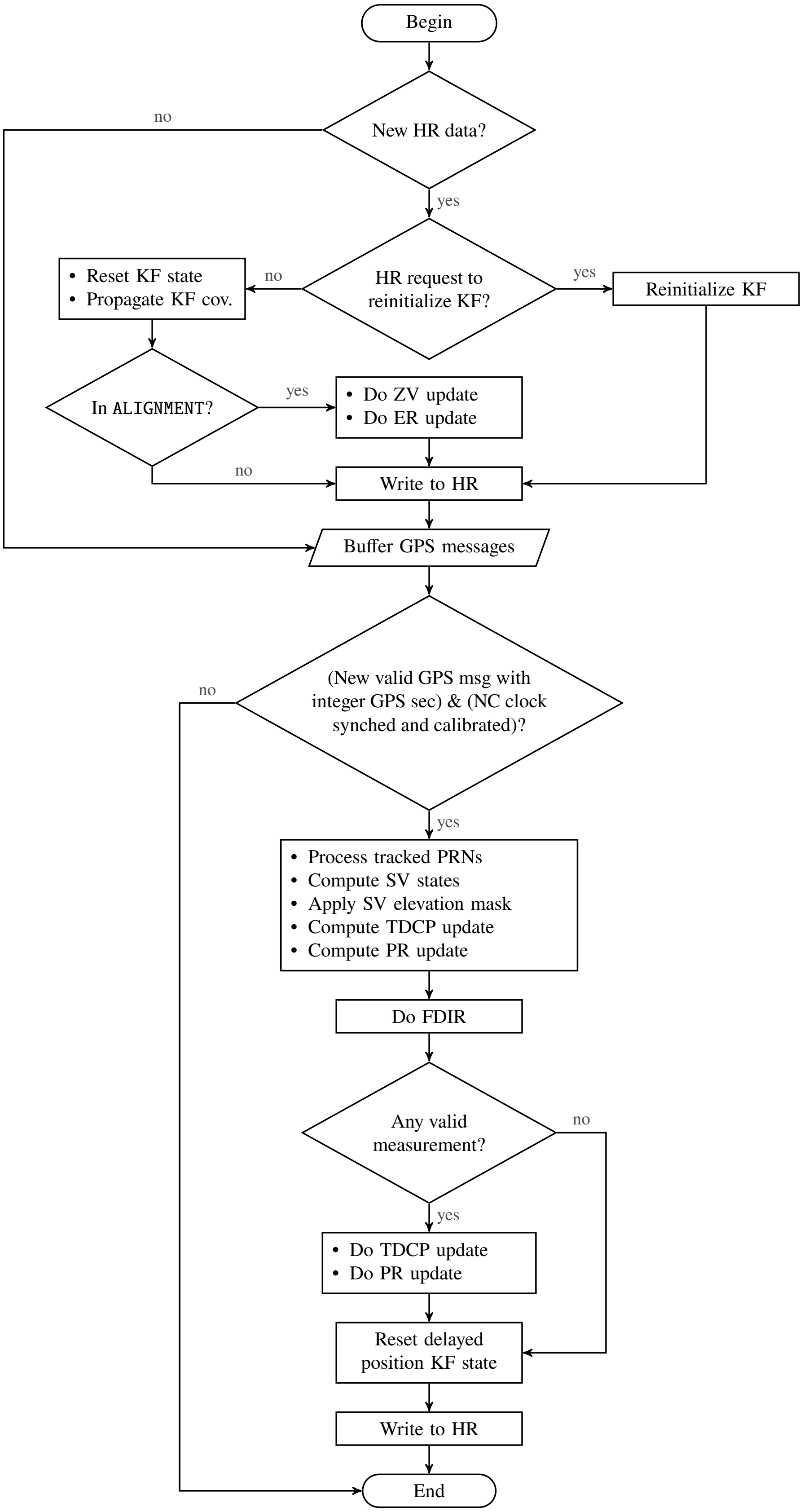


Figure 6.5: LR thread flow diagram (loosely adapted from [160])

case of a new valid GNSS measurement, and if the NC clock has already been synched and calibrated against the GNSS PPS signal, the measurement update process follows: The reported tracked PRN set is processed (e.g., channel bias states are reset if the PRN on that channel has changed); The positions of tracked satellites are computed (as in [124, 69, 118]) and, with that, an elevation mask is applied (i.e., satellites too close to the apparent horizon are evicted from the valid measurement set); The TDCP and PR update calculations are then performed (as described in Sections 3.9.5 and 3.9.6, respectively) including tropospheric and ionospheric corrections (Sections 3.9.3 and 3.9.4, respectively), but actual update is deferred; FDIR algorithm is run (according to the scheme described in Section 5.4); If a measurement subset is deemed valid, filter update of sane TDCP and PR samples is done; Delayed antenna position state is then reset [as in (3.142)–(3.144)] and the updated error-states sent to the HR thread.

6.4 Computational load analysis

Section 4.4.5 shows that the proposed reduced-order 19-21-state Consider KF design results in significant computational burden reduction with respect to the 70-state full-order filter configuration. However, although smaller in order with respect to the filter of the HNS (33 states), the reduced-order filter here developed still needs to handle a large amount of consider parameters (35 to 38, see Section 4.4.4), demanding an effort which, as seen in Section 4.4.5, is far from negligible. Moreover, the introduction of TDCP measurements (unused by the HNS) also requires additional computation.

Coarse FLOP count analysis, similar to that in Section 4.4.5, can show how these two design choices affect the overall computational load of the filter, and how it compares to that of the original HNS. Table 6.3 breaks down the FLOP count for the proposed reduced-order robust estimator (medium-grade version, see Section 4.4.4), for its full-order equivalent, and for the HNS filter (SHEFEX-2 version [161]). The HR and LR steps are separately accounted, assuming a 1-second operation, i.e., 400 HR steps and a single LR step. The kinematics whole-state propagation (position, velocity and attitude), performed by the strapdown algorithm, is not counted. FDIR operations are as in Table 5.4. The HNS performs a simple, sequential PR innovations test [161].

The results show that the HR tasks (integration of state transition matrices) dominate the overall load, amounting to 4–14 times that of the LR tasks. The large state vector of the full-order filter (i.e., without consider-states) with respect to the HNS more than doubles the overall computational load required (137.2% increase). The proposed reduced-order set-up (see Chapter 4) cuts added load significantly to around 59.6%.

The computational burden analysis data reported in [161] for the HNS filter, revealed that the HR thread load requirement is considerably more stringent (due to the higher rate of this routine) than that of the LR thread; the HR tasks used a worst-case 11.8% of the allocated processing time, while the LR used only 0.6%. (Conveniently, the majority of extra load, with respect the HNS, required by the proposed filter falls upon the LR tasks.) The estimated additional 44.9% and 260.2% loads on HR and LR threads of the new navigation algorithm, respectively, shall use only a very small fraction of the remaining wide load margins estimated for the original software on the HNS computer hardware, thus posing no challenge to the real-time capability of the software.

Table 6.3: FLOP count for the implemented dual-rate (HR: 400 Hz –LR: 1 Hz) filter in its full- and reduced-order versions (medium-grade IMU) compared to that of the HNS

Step	Output	Expression	FLOP Count		
			HNS	Full	Reduced
Propagation, HR					
Integrate kin-kin transition matrix	$\mathbf{\Phi}^{\mathsf{x_{kin}}}_{\mathsf{x_{kin}},k+1\|k}$	(3.51)	550800	550800	550800
Integrate kin-imu state transition matrix	$\mathbf{\Phi}^{\mathsf{x_{imu}}}_{\mathsf{x_{kin}},k+1\|k}$	(3.52)	820800	2257200	752400
Integrate kin-imu consider transition matrix	$\mathbf{\Phi}^{\mathsf{c_{imu}}}_{\mathsf{x_{kin}},k+1\|k}$	(4.37)	0	0	684000
Total propagation HR			1371600	2808000	1987200
Propagation, LR					
Propagate error-state	$\boldsymbol{\delta\hat{\mathbf{x}}}_{(k+1)^-}$	(4.9)	393	808	363
Propagate state cov. prior	$\mathbf{P}_{\mathsf{xx\|x},(k+1)^-}$	(4.11)	16683	65298	10997
Propagate state-consider cross-cov. prior	$\mathbf{P}_{\mathsf{xc},(k+1)^-}$	(4.13)	0	0	15394
Propagate consider state cov. prior	$\mathbf{P}_{\mathsf{cc},(k+1)^-}$	(4.14)	0	0	2235
Consider contribution to state cov. prior	$\mathbf{P}_{\mathsf{xx\|c},(k+1)^-}$	(4.12)	0	0	12258
Correct state cov. prior	$\mathbf{P}_{\mathsf{xx},(k+1)^-}$	(4.10)	0	0	189
Total propagation LR			17076	66106	41436
Update, LR					
Innovation	$\mathbf{z}_k$	(4.24)	792	3360	1008
Innovation cov.	$\mathbf{P}_{\mathsf{zz\|x},k}$	(4.30)	19152	80928	24480
State-innovation cross-cov.	$\mathbf{P}_{\mathsf{xz\|x},k}$	(4.27)	26136	235200	21168
Consider-innovation cross-cov.	$\mathbf{P}_{\mathsf{cz},k}$	(4.32)	0	0	103008
Consider contribution to innovation cov.	$\mathbf{P}_{\mathsf{zz\|c},k}$	(4.31)	0	0	42624
Consider contribution to state-innovation cross-cov.	$\mathbf{P}_{\mathsf{xz\|c},k}$	(4.28)	0	0	36792
Correct innovation cov.	$\mathbf{P}_{\mathsf{zz},k}$	(4.29)	0	0	288
Correct state-innovation cross-cov.	$\mathbf{P}_{\mathsf{xz},k}$	(4.26)	0	0	504
Kalman gain	$\mathbf{K}_{\mathsf{x},k}$	(4.22)	10154	41620	13396
Update error-state	$\boldsymbol{\delta\hat{\mathbf{x}}}_{k^+}$	(4.18)	792	3360	1008
Update whole-state	$\hat{\mathbf{x}}_{k^+}$	(3.37)	33	70	21
Update state cov.	$\mathbf{P}_{\mathsf{xx},k^+}$	(4.34)	26136	235200	21168
Update state-consider cross-cov.	$\mathbf{P}_{\mathsf{xc},k^+}$	(4.20)	0	0	37296
Update consider cov.	$\mathbf{P}_{\mathsf{cc},k^+}$	(4.21)	0	0	0
Total update LR			83195	599738	302761
FDIR, LR					
Total FDIR LR		(see Table 5.4)	12	17060	17060
Total HR (% increase from HNS)			1371600 –	2808000 **(104.7)**	1987200 **(44.9)**
Total LR (% increase from HNS)			100283 –	682904 **(581.0)**	361257 **(260.2)**
Total (% increase from HNS)			1471883 –	3490904 **(137.2)**	2348457 **(59.6)**

6.5 Performance evaluation

In this section, the implemented navigation software is tested in a software-in-the-loop set-up using both simulated and real GNSS data.

6.5.1 Trajectory and simulation models

As in the previous chapter, the trajectory used is that of the Vega VV04 flight (see Appendix B). The inertial measurements correspond to a medium grade sensor profile (see Table 2.1) and are generated by a high-fidelity model as described in Section 2.4.3 and derived in Appendix Section C.2. The simulated GNSS measurements are produced by a simulation model of the DLR Phoenix-HD GPS receiver and the real ones are output by the said unit stimulated by a GNSS emulator (SPIRENT GSS7700).

6.5.2 Results

To test the navigation real-time software, the C/C++ code is wrapped in an S-Function block inside MATLAB Simulink simulation environment. Figure 6.6 depicts the software-in-the-loop (SIL) simulation set-up. The true trajectory data is used by the sensor models (IMU and GNSS) to produce simulated measurements that are, in turn, fed to the navigation computer block. This block contains not only the mentioned navigation software S-Function but also a model of the navigation computer clock and triggering system (including, e.g., clock bias and scale-factor). The software executive layer is here reduced to an input/output managing code within the S-Function wrapper. To simulate GNSS receiver latency a 200 ms delay has been placed between the GNSS model output and the navigation computer block. The PPS signal and the IMU data output are considered *real-time*.

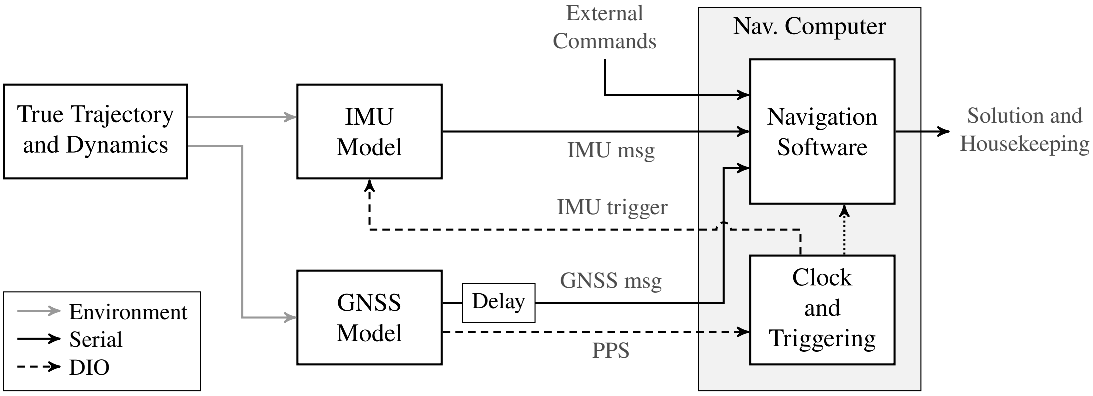


Figure 6.6: Software-in-the-loop simulation architecture (adapted from [162])

Figures 6.7 and 6.8 show the Monte Carlo simulation results of the real-time navigation software within the described set-up. Monte Carlo campaign RMS is compared to filter predicted 1σ covariance, and campaign maximum curve is compared to 3σ covariance. The trajectory followed is that of Vega flight VV04 (see Appendix B), the IMU model is that described in Appendix C (and used in the tests in Chapters 3 and 4, and the GNSS is a simulation model of the Phoenix-HD. The plotted campaign error

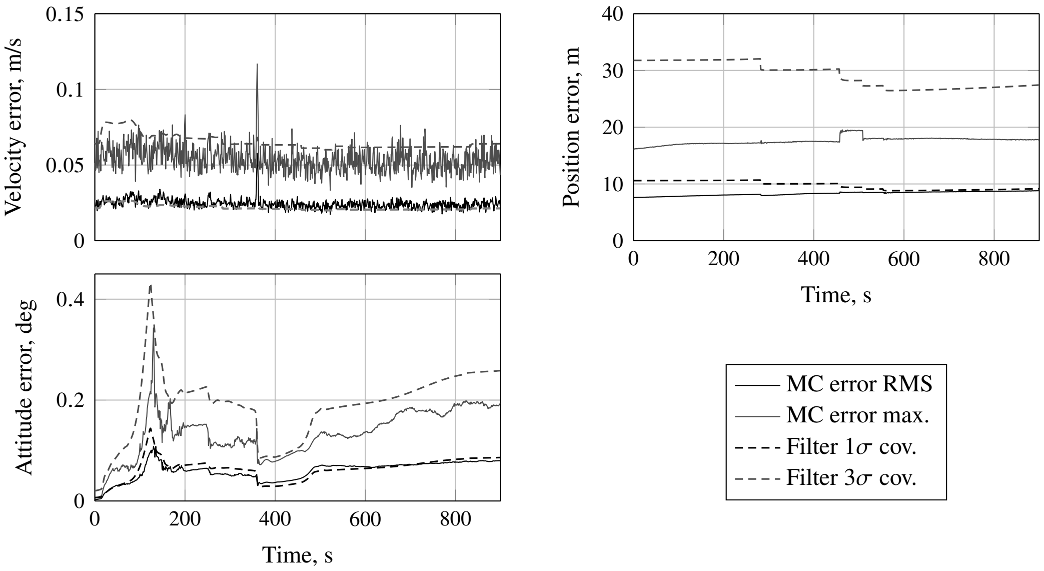


Figure 6.7: MC campaign RMS and max. kinematics state errors of the real-time navigation software on a Vega VV04 trajectory

RMS curves are close to the predicted covariance for all states (kinematics in Figure 6.7 and IMU states in Figure 6.8), attesting to the coherent behavior of the filter.

To test the software under real GNSS data, the GNSS model in the SIL set-up (Figure 6.6) is replaced by a data source block looking-up recorded real GNSS data produced by the Phoenix-HD receiver fed by a SPIRENT GNSS signal emulator (as done in previous chapters). The above mentioned delay on the GNSS output remains. The estimation results of a single-run in this configuration are shown in Figures 6.9 and 6.10 for the kinematics and IMU states, respectively, and compared to those obtained with the non-real-time (MATLAB-code) implementation of the navigation filter. The trajectory is again that of Vega VV04 flight and the IMU model is unchanged. The overlapping/proximity of filter error and predicted covariance of the two filters shows that the C/C++ implementation is successful in maintaining the performance under stricter, more representative conditions. The small difference between covariances especially in velocity is due to a retuning meant to compensate for NC clock error.

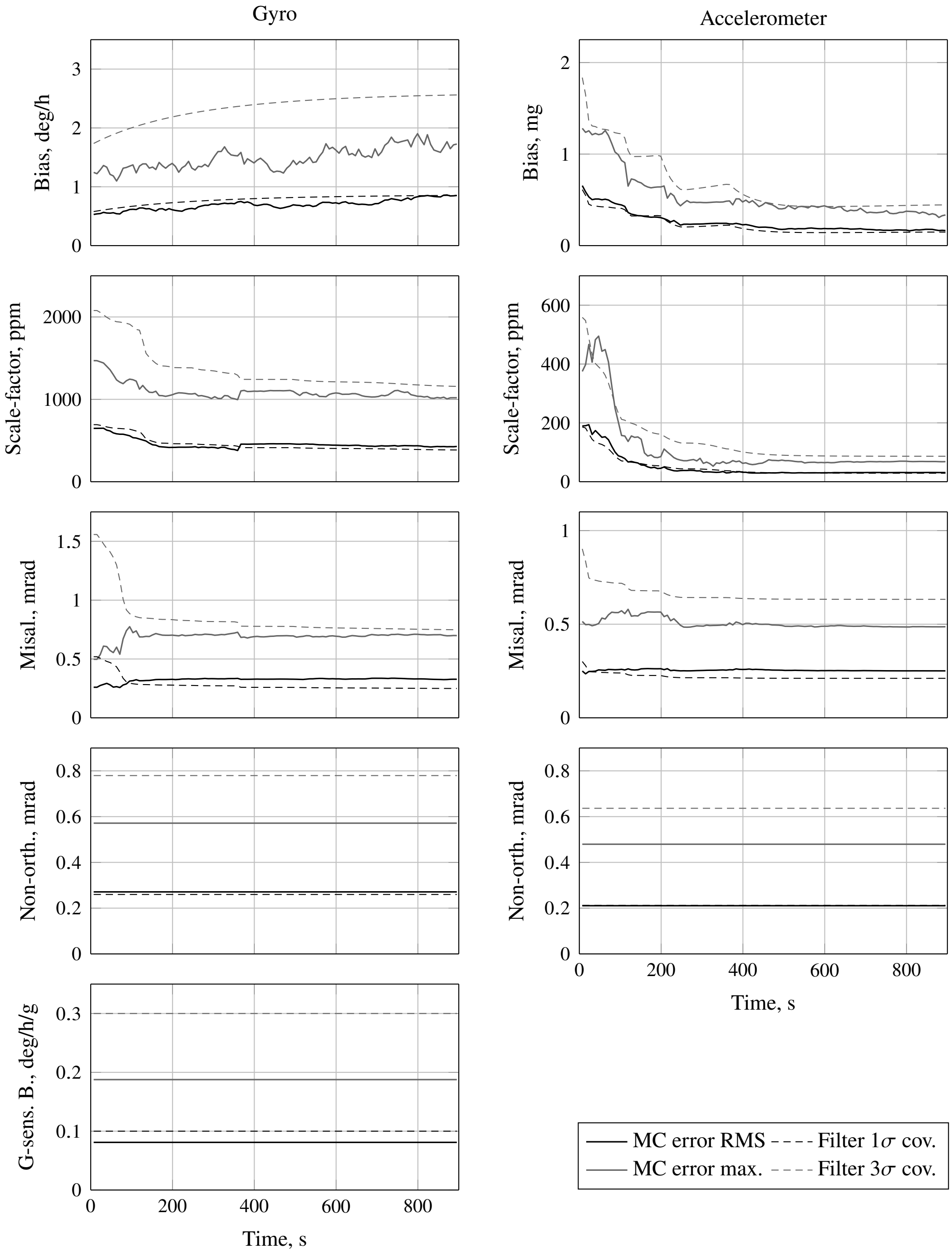


Figure 6.8: MC campaign RMS and max. inertial sensor state errors of the real-time navigation software on a Vega VV04 trajectory

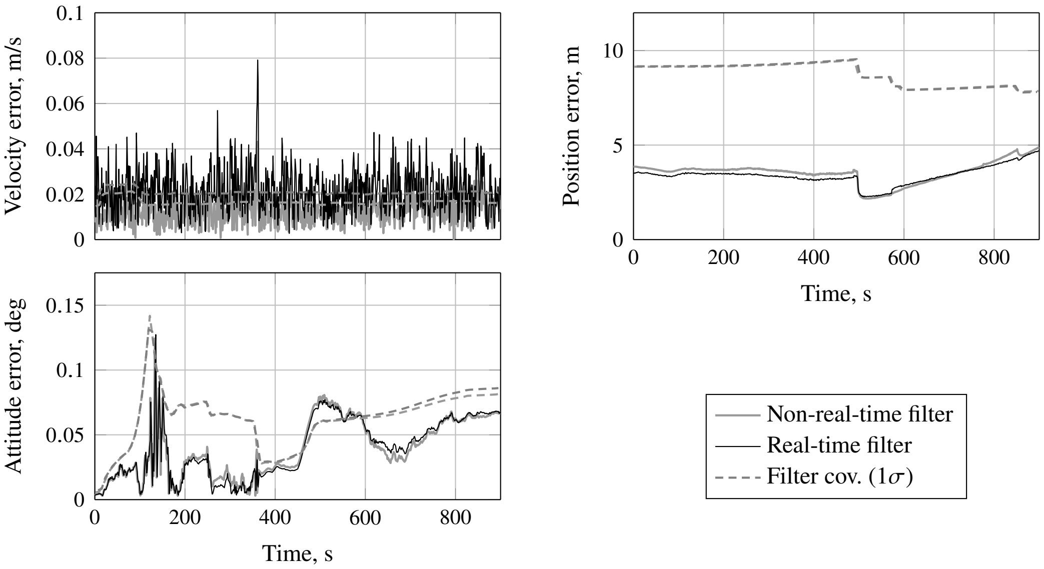


Figure 6.9: Kinematics state error comparison of real-time and non-real-time navigation implementations under real GNSS data of a Vega VV04 trajectory

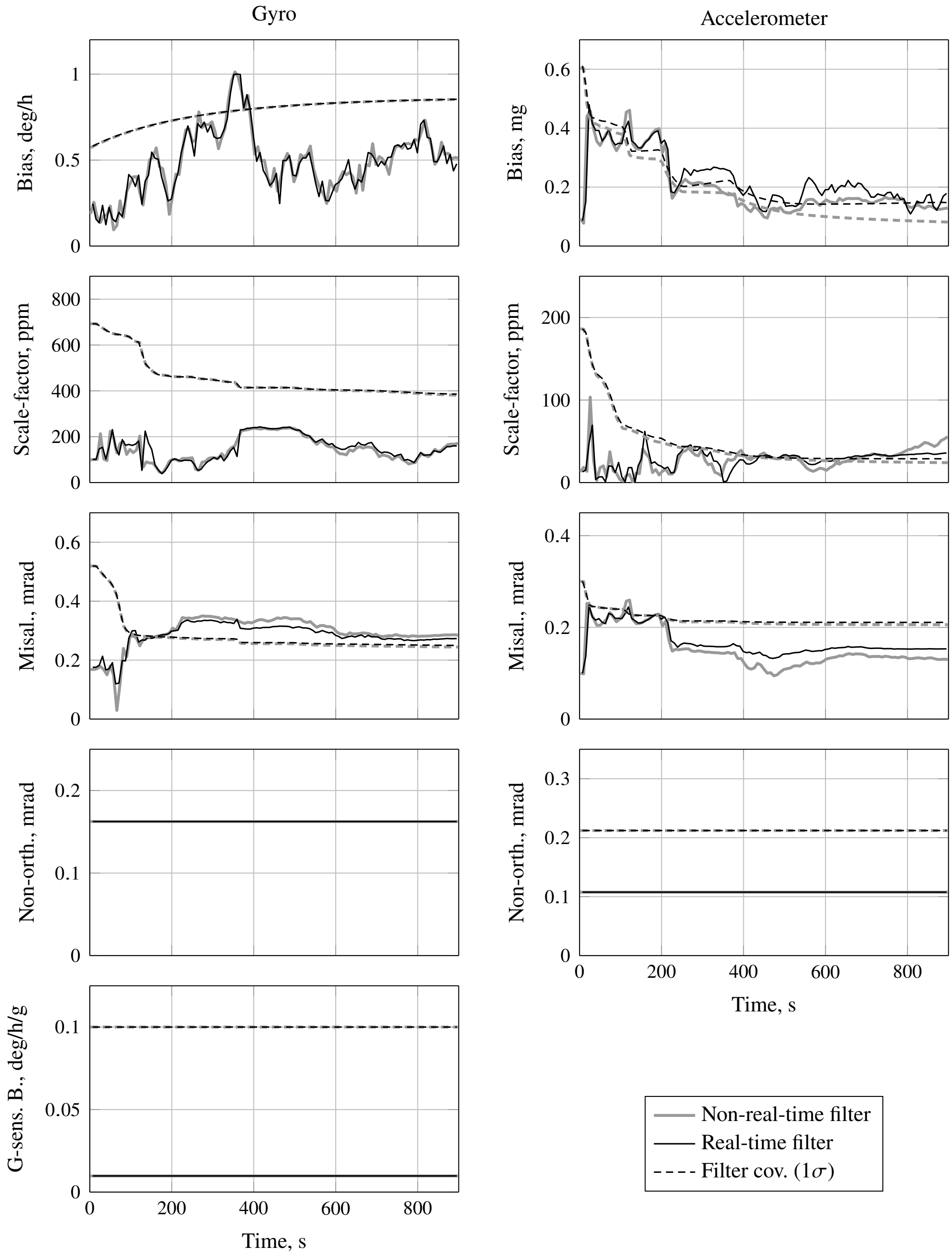


Figure 6.10: Inertial state error comparison of real-time and non-real-time navigation implementations under real GNSS data of a Vega VV04 trajectory

7

Conclusion and Ways Forward

7.1 Conclusion

Inertial-only navigation systems, used by most launchers, strongly limit launch mission definition and execution. GNSS measurements can, in many instances, be a remedy for the unbounded inertial dead-reckoning. Indeed, several GNSS-based set-ups, both commercial and experimental, have flown on board of launch vehicles. Despite advantageous, design and implementation of such hybrid, GNSS-inertial navigation systems face multiple challenges.

The work described in this thesis starts with an investigation on the key architectural elements and design aspects of hybrid GNSS-inertial navigation in the context of space transportation. Survey of architectures with varying coupling depths has shown that the tightly coupled set-up offers a good compromise between flexibility and robustness, and ease of realization, namely with COTS components. Deeper in the design, a closed-loop-type scheme, with inertial propagation corrected by online estimates, prevents unbounded filter state growth and allows in-flight inertial calibration, effectively improving robustness and performance during eventual free propagation. Modular designs, with separate strapdown and fusion algorithm routines, allow for parallelization and offer an simple way of coping with GNSS output latency.

Concerning the inertial components of the system, strapdown propagation algorithm order and rate are crucial in highly dynamic applications. Integration rate reduction can be achieved using higher-order compensation algorithms; however, only mild reductions seem possible in the launcher application at hand. On the inertial sensor side, use of a lower grade unit may require additional states in the fusion algorithm to account for higher than first-order bias models, axis misalignments, and G-sensitivities. Coarse covariance analysis in a Vega scenario has demonstrated that the fusion of GNSS and inertial data not only restricts position and velocity state errors to GNSS levels, but also grants attitude observability during engine burns. This yields an improvement in attitude estimation of about one IMU grade with respect to a inertial-only set-up.

Among GNSS outputs, position, velocity, pseudorange (PR), pseudorange-rate (PRR) and carrier phase measurements have been investigated. Formed from carrier phases, time-differenced carrier phase (TDCP) pseudo-measurements are free from phase ambiguities and provide lower-noise velocity information than PR or PRR. Error-wise, receiver clock is an important source. Modeling of this error as a state in the estimator is preferable to measurement cross-differencing as it allows raw GNSS measurements with as few as a single measurement to used. Atmospheric effects (tropospheric and ionospheric), another important GNSS error source, can severely affect velocity-based GNSS measurements due to the high climb speed of the launcher and the increased apparent elevation-rate of the tracked satellites. Atmospheric model-based corrections, even coarse ones, greatly improve velocity estimation; however, correction residuals, especially of tropospheric delay, need to be accounted for in the measurement covariance model. Finer covariance analysis of hybrid set-ups with different sets of the GNSS measurements has shown the benefits of using velocity-based measurements as support to position-based ones, not only in terms of velocity estimation but also in attitude. Among the tested options, the PR and TDCP combination yielded the lowest estimation error covariances. Estimator-wise, given the moderate non-linearities in the strapdown propagation equations and in the range-based models, more complex algorithms than the EKF are generally unnecessary.

Based on the results of the architectural element review and analysis, a baseline navigation system has been designed. This is a tightly coupled, TDCP- and PR-updated, hybrid system; using a EKF-like, error-state filter and a full-rate strapdown scheme set up modularly and in closed-loop. Analysis of receiver clock error affecting both TDCP and PR in the receiver used informed the development of a reduced-order model. Model-based tropospheric and ionospheric delay correction has been implemented and a robustness scheme against tropospheric correction residuals has been proposed. Testing of different configurations allowed for the selection of a tropospheric residual compensation scheme that does not require state augmentation, but is still able to maintain filter coherence with the low-fidelity correction model. For robustness against lower grade inertial sensor errors, the filter uses a full-order inertial sensor calibration model. Performance testing of the overall full-order system has shown robustness against high-fidelity IMU and GNSS models. Comparison to a HNS-like and PR-only configurations shows significant performance improvements, while filter consistency has been evaluated through innovation signal average and whiteness measures.

With the goal of reducing the computational footprint of the conceived baseline estimator while maintaining its robustness, a Consider Kalman filter framework has been implemented. The generic consider-state estimator algorithm has been written in a novel manner, clearly separating standard (estimated) state and consider-state processes, yielding a entirely modular design. In addition, it has been derived including error-states and correlation between process and measurement noise. With this framework, a state reduction process has been proposed based on extensive parametric impact and observability analysis using Parametric Cramér-Rao bounds. Streamlining of inertial sensor and GNSS filter models has been performed and the resulting designs, tailored to Vega launcher dynamics and to different inertial sensor grades, have been estimated to require about half of the FLOP count of the full-order filter. Covariance testing has yielded remarkably similar performance to the full-order equivalent Parametric Cramér-

Rao bound, while Monte Carlo testing has shown filter consistency and maintained robustness against full-fidelity sensor models. Comparison to a loose coupling scheme under partial outage (i.e., less than four tracked satellites) has evidenced the ability of the designed, tightly coupled system to use deficient satellite sets, minimizing performance loss. Long-term accuracy of the designed filter (medium-grade version) has been compared to that of purely-inertial solutions with equivalent and higher grade inertial sensors. Improvements of more than three orders of magnitude in position and velocity estimation have been observed relative to the inertial-only solution of the same medium-grade sensor and of two orders of magnitude over that of the higher grade sensor. Attitude performance of the same medium-grade hybrid system has been shown to converge to that of the higher-grade inertial system during propelled phases. Overall, results have suggested the possibility of a one-grade reduction in the inertial sensor with the proposed navigation hybridization (i.e., high to medium grade), with potential significant cost savings.

Beyond robustness, a fault-detection, isolation and recovery scheme has been conceived for the developed navigation system. The proposed novel method combines traditional RAIM (GNSS measurement-based) techniques with filter innovation statistics monitoring, being capable of detecting and excluding single/multiple GNSS channel blunders, receiver common-mode faults (e.g., clock discontinuities) and filter kinematics state faults (e.g., excessive inertial propagation divergence). Filter recovery schemes for the latter two scenarios have also been created. Verification has been achieved through extensive statistical analysis under real, static GNSS data, and threshold selection has been done in a sound, analytical process based on resulting detection levels. The worst case required computational load analysis has been estimated as only 5.6% of the total update load of the medium-grade robust filter. Testing of the FDI part of the algorithm under real GNSS data of a Vega trajectory has shown false-alarm and detection levels consistent with the tuning targets. Recovery has been tested for the same GNSS data and under both severe common-mode GNSS fault and severe filter kinematics state divergence revealing remarkable improvements over the unprotected filter set-up.

With the goal of raising the readiness level of the system, a real-time capable, software implementation of the designed navigation algorithm has been written based on the HNS flight code. In addition to the robustness features tested in previous chapters, and like the HNS, this implementation tolerates typical GNSS output latencies by delaying the error-state filter processes (thus also the measurement updates) with respect to the near-real-time strapdown propagation process. Rotation of the delayed error-state to the actual time of feedback improves upon the method used by the HNS. Computational load analysis of the proposed implementation has revealed a modest increase of the reduced-order robust filter (of 59.6%) with respect to the original HNS code. The additional burden does not threat the comfortable load margins held by the original HNS software on a real-time computer, and therefore the real-time capability of the code. Navigation software validation has been done through Monte Carlo testing on a Vega flight trajectory and under simulated NC clock errors and GNSS latency. Finally, comparison with the non-real-time, MATLAB-code implementation under real-GNSS data has revealed only minor deviations.

7.2 Ways forward

The work developed in this thesis can be carried on in multiple directions. The following points summarize several possibilities.

- The proposed navigation concept has been taken from early, coarse covariance analysis to C/C++ navigation software; the natural next step would be to test it on a real-time, flight-representative computer, in a processor-in-the-loop set-up. Given the opportunity, flight testing the system, for instance, as experimental payload on a sounding rocket flight (as, e.g., in [24]) or on an orbital launch vehicle (as done, e.g., by the OCAM-G initiative on Ariane-V [73]) would further raise its readiness level.

- As mentioned in Section 2.6.3, strict system reliability requirements are often met through sensor redundancy; particularly, when working with lower grade/cost units this option may be an attractive one. Including additional inertial sensors (or axes) in redundant configurations is not uncommon. The general inertial sensor model framework derived in Section C.3 can be used to combine all available single-axis measurements and distill them to a three dimensional (Cartesian) space. This prevents filter state augmentation while maintaining rigorous stochastic description of the overall inertial sensor errors. Furthermore, as being considered for the ReFEx experiment [149], GNSS receiver redundancy is also a possibility worth pursuing. In addition to increased reliability, a multi-receiver configuration may allow (if set up with multiple antennas) direct attitude observability, or some DoFs thereof (as shown, e.g., in [52, 32, 177, 136] for different applications and receiver configurations).

- Reusability has quickly become one of the most (if not the most) researched topics within LV technology [155, 87, 134, 34]. As shown in [146], typical inertial navigation means cannot provide enough solution accuracy to enable safe return and landing. The additional dynamics in a return trajectory leg (e.g., kick-back manoeuvre, deceleration burns, re-entry braking) may yield significant levels of attitude observability in a GNSS/inertial system configuration, allowing re-convergence of this state caused by coasting and eventual vehicle spinning during ascent. The attainable attitude accuracy improvement and whether it is sufficient for safe approach and landing can be studied using covariance analysis tools, namely the PCRB (as done in [149, 141] for ReFEx, a horizontally landing reusable LV experiment). If GNSS aiding is deemed insufficient of providing the final accuracy levels required, the system design proposed in this work can be extended to include additional sensors, such as sun sensors, laser/radar altimeters, radio-frequency pseudolites, or differential GNSS.

A

SHEFEX-2 Sounding Rocket

A.1 Mission and vehicle

SHEFEX-2, the second DLR SHarp Edge Flight EXperiment sounding rocket mission, was carried out on June 22, 2012, launched from Andøya Rocket Range, Norway, by the Mobile Rocket Base (MORABA, Mobile Raketenbasis) of DLR, and aimed to validate models and technologies used for hypersonic and space transportation [195, 194, 193, 174, 172]. The experimental payload was mounted on a modified 2-stage Brazilian VS-40 rocket, composed of an S-40 and an S-44 solid propellant motors. A picture of the launch and a graphical impression of the vehicle can be seen in Figure A.1.

A.2 Flight trajectory

The SHEFEX-2 was conceived to test models and technologies in a hypersonic flight regime. For this purpose the trajectory was designed to reach Mach 11 (~ 3 km/s) for up to 45 seconds [194].

Figure A.2 shows a simulated SHEFEX-2 trajectory in terms altitude, total velocity, total specific-force acceleration, and total angular rate. The 2-stage (2-burn) rocket is spin-stabilized during the entire burn sequence at about 750 deg/s (~ 2 rev/s). The sub-orbital, sounding-rocket-like trajectory has apogee at about 244 km and a maximum specific-force acceleration of around 72 m/s^2 (~ 7 g).

The real flight trajectory, shown in Figure A.3 with event timeline described on Table A.1, had a slightly lower apogee (177 km) and achieved a Mach number of about 9 (~ 2.8 km/s) during re-entry [193]. The spin-rate of the rocket during ascent was also slightly lower than in the simulated trajectory (~ 560 deg/s *vs* ~ 750 deg/s). This difference together with a slightly different IMU location in the experimental payload (with respect to the spin-axis) caused the difference in specific-force acceleration around the transversal Body axes (x and y).

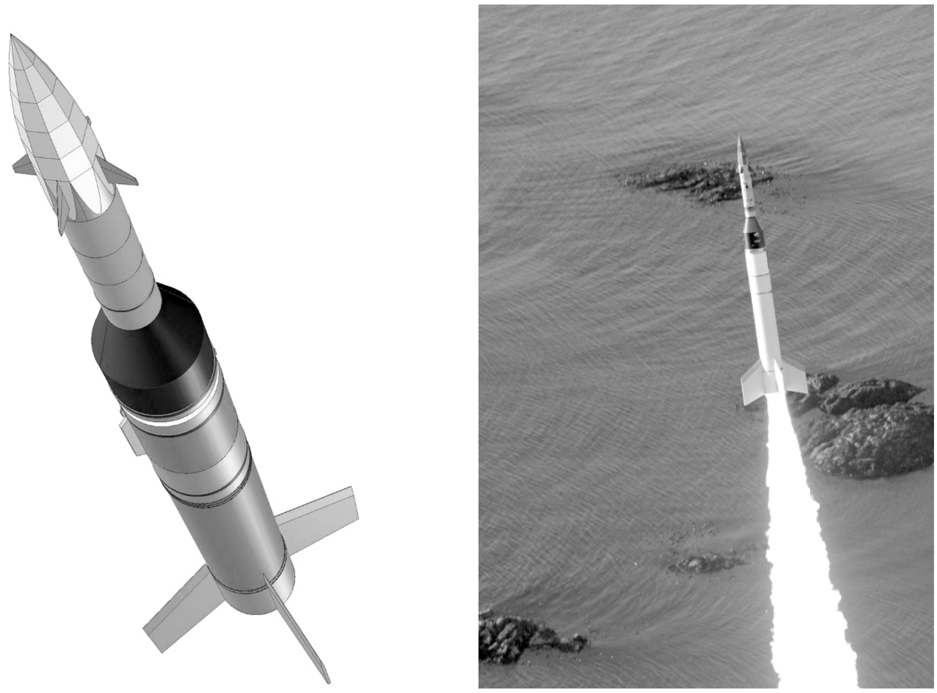


Figure A.1: SHEFEX-2 graphical depiction and picture of launch (adapted from [193])

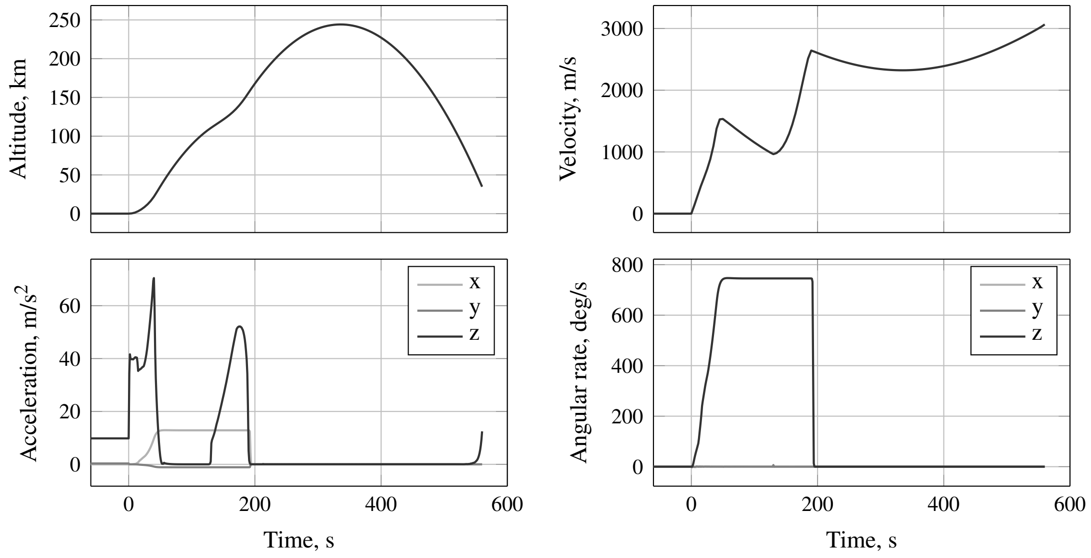


Figure A.2: SHEFEX-2 simulated trajectory (downsampled); acceleration and angular in Body coordinates (see Figure 2.5)

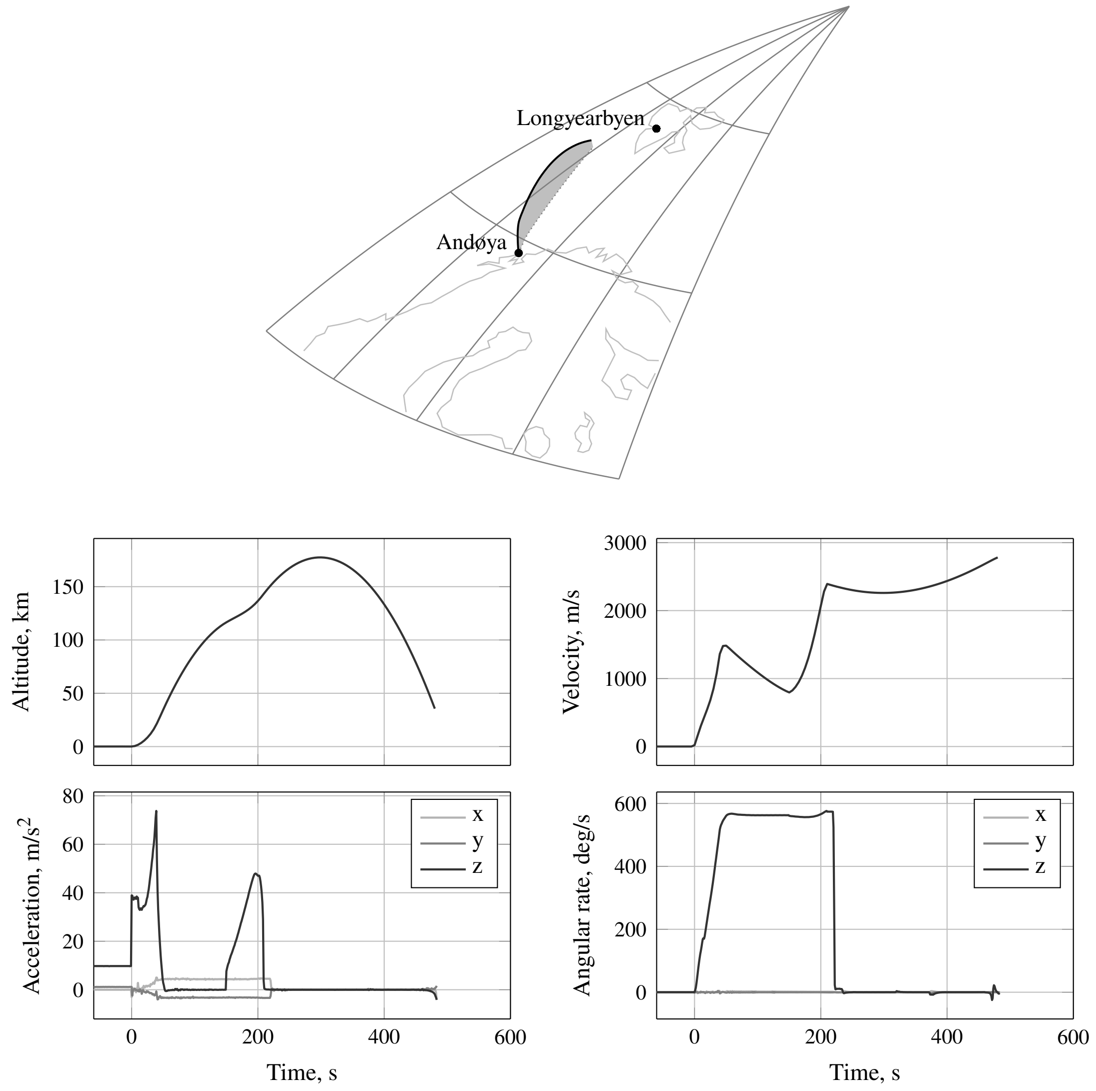


Figure A.3: SHEFEX-2 real flight trajectory (downsampled); acceleration and angular in Body coordinates (see Figure 2.5)

Table A.1: SHEFEX-2 real flight event timeline [164]

Event	Flight time, s
1st stage ignition	0
1st stage burnout	55
Maximum spin rate	60
1st stage separation	90
2nd stage ignition	150
2nd stage burnout	213
Yo-yo de-spin	220
Fairing release	225
2nd stage separation	230
Reorientation for STR	237
STR activated	269
2nd reorientation for STR	320
Reorientation for entry	372
Loss of telemetry	485

A.3 The Hybrid Navigation System

Developed by the DLR, the Hybrid Navigation System (HNS) was conceived as a primary navigation for launcher and sounding rocket applications [173, 174, 172], and has been flown on-board SHEFEX-2 vehicle as part of the experimental payload [160]. This system includes a tactical-grade COTS 3-axis IMU (iMAR iIMU-FCAI-MDS) [84], and COTS-based DLR-developed GPS receiver [104] and Star Tracker (STR) [125, 142]. It is based on a tightly-coupled, modular, closed-loop architecture (see definitions in Section 2.3), using an error-state EKF, akin to that derived in Section 3.4, to fuse strapdown propagated kinematics (driven by high-rate IMU measurements) with low-rate GPS pseudoranges and, when available, Star Tracker (STR) attitude fixes [161, 160].

Thorough analysis of the HNS estimation performance on the SHEFEX-2 flight is done in [164, 163, 160], where trajectory reconstruction of the collected flight data uses a UKF, forward-backward Unscented Rauch-Tung-Striebel Smoother (see, e.g., [143]), and improved filter models and tuning. The results revealed consistent filter behavior throughout the flight. However, the post-flight estimated HNS attitude accuracy just before atmospheric re-entry (~ 450 s), and without information from the STR, which was not available, was about 3 deg [163] (which is slightly better than the 5 deg indicated by the first post-processing efforts [164, 160]). This far exceeded the required 0.17 deg [174, 173]. (Note that the HNS had complied with this requirement in the pre-flight software-in-the-loop [160] and hardware-in-the-loop tests [161] even without STR information.) The proposed explanation for the discrepancy blames the vibration experienced during launch, which was higher than expected. Indeed, as reported in [163], when (post-flight) the HNS prototype was fed the flight-collected, vibration-laden sensor data, it yielded a level of performance identical to that in flight.

Although not further explained or analyzed in the cited post-processing works, it is possible to reason that high-frequency vibration (e.g., during engine burns, sonic transition and dynamic pressure peak) would have been rectified by the IMU (due to its finite bandwidth) causing an additional null/low-frequency component bias. To this rectification could also have contributed the sensor scale-factor nonlinearity. The temporary nature of this effect would have produced erroneous estimates of some IMU error states (e.g., bias and scale-factor) which would then have remained once the vibrations had subsided and the observability window of such error states (granted by vehicle dynamics) had *closed*, corrupting the attitude state estimates from then on. However possible, this chain of events would have caused severe inconsistencies between estimation error and covariance, which, as above mentioned, the post-processing did not reveal [163].

During the present study, it was found that the SHEFEX-2 simulated trajectory used for the HNS development and testing (e.g., in [162, 161, 160]) was not fully representative of a real rocket flight in that the specific-force acceleration vector (during engine burns) was severely misaligned with the vehicle longitudinal (and spin) axis. In fact, the angle between the thrust vector and the vehicle Body z-axis (see Figure 2.5) would slowly grow from lift-off and arrive to as much as 56 deg during the second burn ($t \in [130, 195]$ s). This resulted from the way this trajectory was generated: through integration of specific-force and angular velocity reference profiles defined in Inertial and Body frame (see frame definitions in Section 2.4.1), respectively, without fixing of the specific-force vector to the Body z-axis. (This process is partially described in [162].) The resulting motion rendered observable the attitude around roll-axis, explaining the optimistic attitude estimation levels obtained by the HNS in its development phase [161], and their discrepancy to the subsequent results in-flight. Recall that attitude observability in a hybrid GNSS/inertial navigation system stems from simultaneous measurement of acceleration information in Body axes, by the IMU, and in ECEF axes, by the GNSS receiver (indirectly, through consecutive measurements of velocity and/or position). Strictly, only two attitude DoFs are observable, as the angle around the thrust axis cannot be resolved in the manner just described. In practice, because the attitude error in the absence of sensor error is fix in inertial space [which is easily verifiable by writing the attitude error propagation law (3.59) in inertial frame, I], and because the thrust direction changes between (and to a lower degree during) engine burns, the roll angle actually becomes slightly observable over time.

Correction of the original simulated trajectory by introduction of a lateral specific-force regulator in the trajectory integration routine results in the profiles displayed in Figure A.2, which closely resemble those of the actual flight (cf. Figure A.3). This corrected trajectory is the one used in the analyses and tests in Chapters 2 and 3.

Figure A.4 compares the attitude estimation (1σ) covariance of a filter identical to that of the HNS (in its SHEFEX-2 configuration without STR) under the original simulated SHEFEX-2 trajectory, its corrected version (above described, and shown in Figure A.2), and the vibration-laden real flight trajectory (Figure A.3). The tuning and initial conditions match those in [161, 163, 160]. The results show that attitude performance under the corrected simulated trajectory is much worse than under the original one, both during rocket spin ($t \in [0, 195]$ s) and high-altitude coasting ($t > 230$ s). In fact, under the corrected trajectory the filter outcome comes far closer to that under the real trajectory. Note that the covariance curve here obtained with real trajectory data is almost identical to the actual HNS in-flight covariance in [163] (which could not use STR), while that here shown for the original simulated trajectory is almost identical to that of the HNS software-in-the-loop testing (using the same trajectory) up to STR turn-on ($t < 283$ s) [160]. These results do not support the hypothesis, proposed in [163, 164, 160], that vibration originated the attitude estimation accuracy discrepancy between pre-flight tests and in-flight run; instead, they strongly suggest that the optimistic attitude observability granted by the spin-axis/thrust-axis misalignment in the original simulated trajectory caused the mismatch.

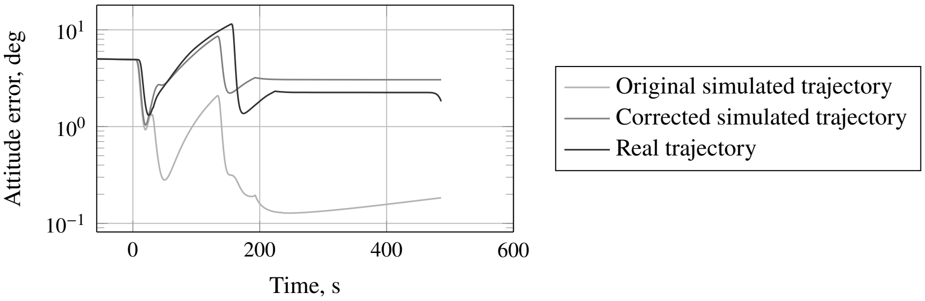


Figure A.4: Attitude (1σ) covariance of a filter identical to the HNS (SHEFEX-2 version, without STR) under SHEFEX-2 trajectories: real (Figure A.3), simulated original, and simulated corrected (Figure A.2)

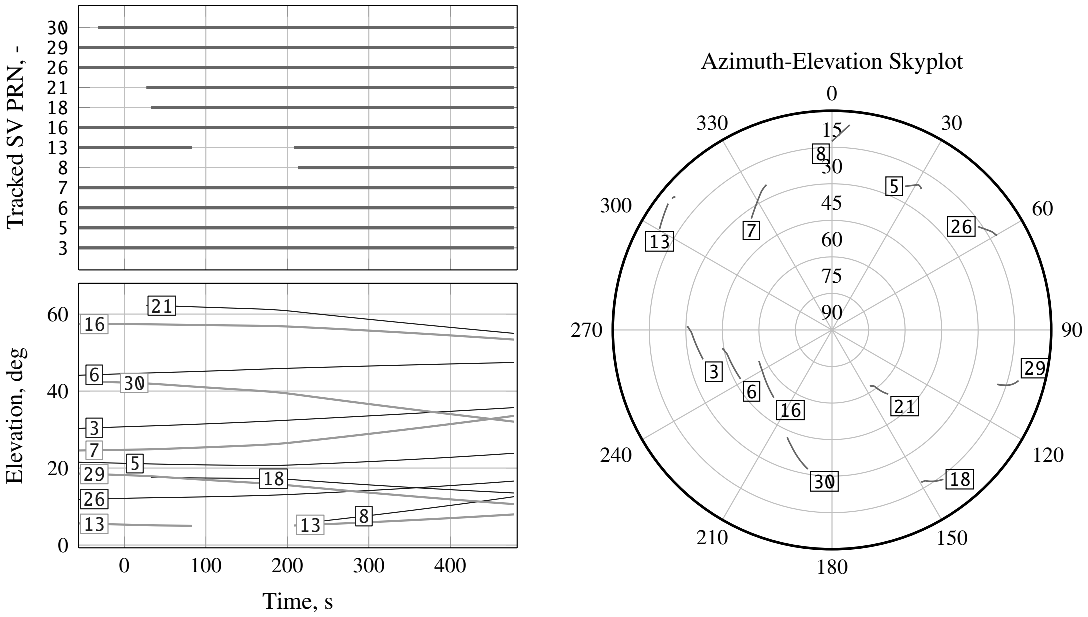


Figure A.5: Profile, elevation and azimuth of GPS satellite tracking of payload Phoenix receiver 2 during SHEFEX-2 flight

A.4 GPS flight data analysis

Two identical Phoenix-HD GPS receiver units flew on board of the SHEFEX-2 experimental vehicle: one, the main receiver, provided only a PPS signal and raw pseudorange measurements to the on-board navigation system (HNS); and a second (here referred to as *receiver 2*), carried as experimental payload but connected to the same wrap-around antenna, had its entire raw data streaming sent to ground through the telemetry channel. For its completeness, the raw GPS data from this latter unit is here analysed.

Figure A.5 shows the satellites tracked by the 12-channel receiver 2 along with their apparent elevation and *sky* trajectory (azimuth *vs* elevation). At any given point during the plotted timeline no fewer than 8 satellites are tracked (9 during flight, $t > 0$ s). While the elevation of these satellites is, in general, medium to low (only 3 satellites are higher than 45 deg during flight) due to the high latitude of the flight (69–76 deg), their scattered azimuth distribution grants reasonably low Geometric Dilution of Precision (GDOP) (see definition, e.g., in [92]), as shown in Figure A.6.

Several sudden changes in signal strength can be seen in Figure A.7, which displays the RMS (across all active channels) of the Carrier-to-Noise-density ratio (C/N_0) differences between consecutive epochs. These peaks, generally in pairs (i.e., C/N_0 drop followed by recovery), happen at around (cf. Table A.1):

0–5 s	– lift-off;
35–41 s	– peak dynamic pressure;
221–245 s	– vehicle de-spin, fairing release and re-pointing manoeuvre;
374–383 s	– re-pointing manoeuvre.

This shows the vulnerability of the signal tracking under high dynamics; the first two events nearly caused total signal outage. The very last peaks in Figure A.7 ($t > 465$ s) were caused by atmospheric re-entry which preceded loss of telemetry.

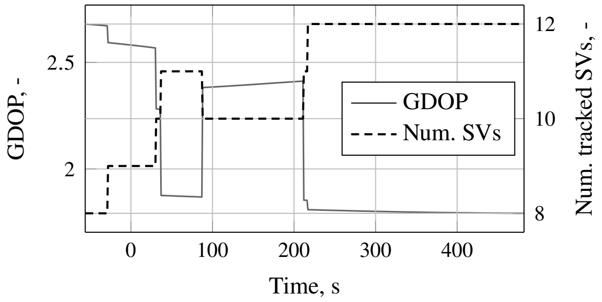


Figure A.6: GDOP *vs* number of tracked satellites of receiver 2 during SHEFEX-2 flight

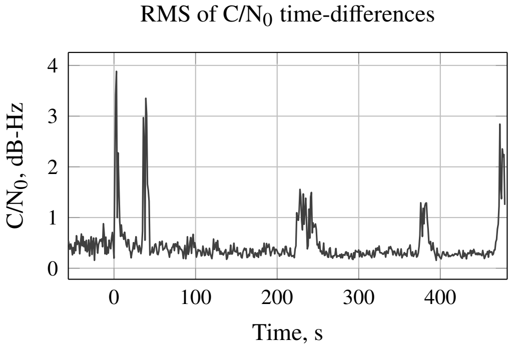


Figure A.7: RMS of C/N$_0$ differences between consecutive epochs of the GPS signals tracked by receiver 2 during SHEFEX-2 flight

The following points analyse the estimated error on both the raw outputs (pseudorange, pseudorange-rate and integrated carrier-phase) and receiver-computed solutions (position and velocity). This is done using high reference the best reconstructed trajectory available for the SHEFEX-2 flight: that obtained in [163], and briefly referred to in Section A.3. The receiver's internal tropospheric corrections, enabled in the SHEFEX-2 receiver configuration, are applied only to smoothed PRs (smoothed with carrier phase differences [112]) and not to the raw observables collected [117, 116]. Tropospheric errors, along with Ionosphere-induced ones, are thus here estimated, analysed and corrected. The raw GPS data and receiver navigation solution available were collected at 1 Hz and 10 Hz, respectively.

Pseudorange and pseudorange-rate

The estimated pseudorange error history for each of the 12 SVs tracked by the equally many channels is presented in Figure A.8. The total estimated error per SV is shown after correction for GPS satellite clock and relativistic effects using ephemeris data for the day and time of flight (see correction model, e.g., in [124]). The tropospheric delays displayed are estimated through a modified UNB3 model [99]. Ionospheric delays are computed using the altitude-dependent VTEC model described in Section 2.5.2 using a surface VTEC of 15 TECU (value retrieved from IONOLAB [71, 150] using a JPL IONEX product for the Ny-Alesund station, Svalbard, Norway and date and time of flight). The common-mode error, to which the receiver clock offset is the main contributor, is also shown. The PR residual errors are those remaining after correction

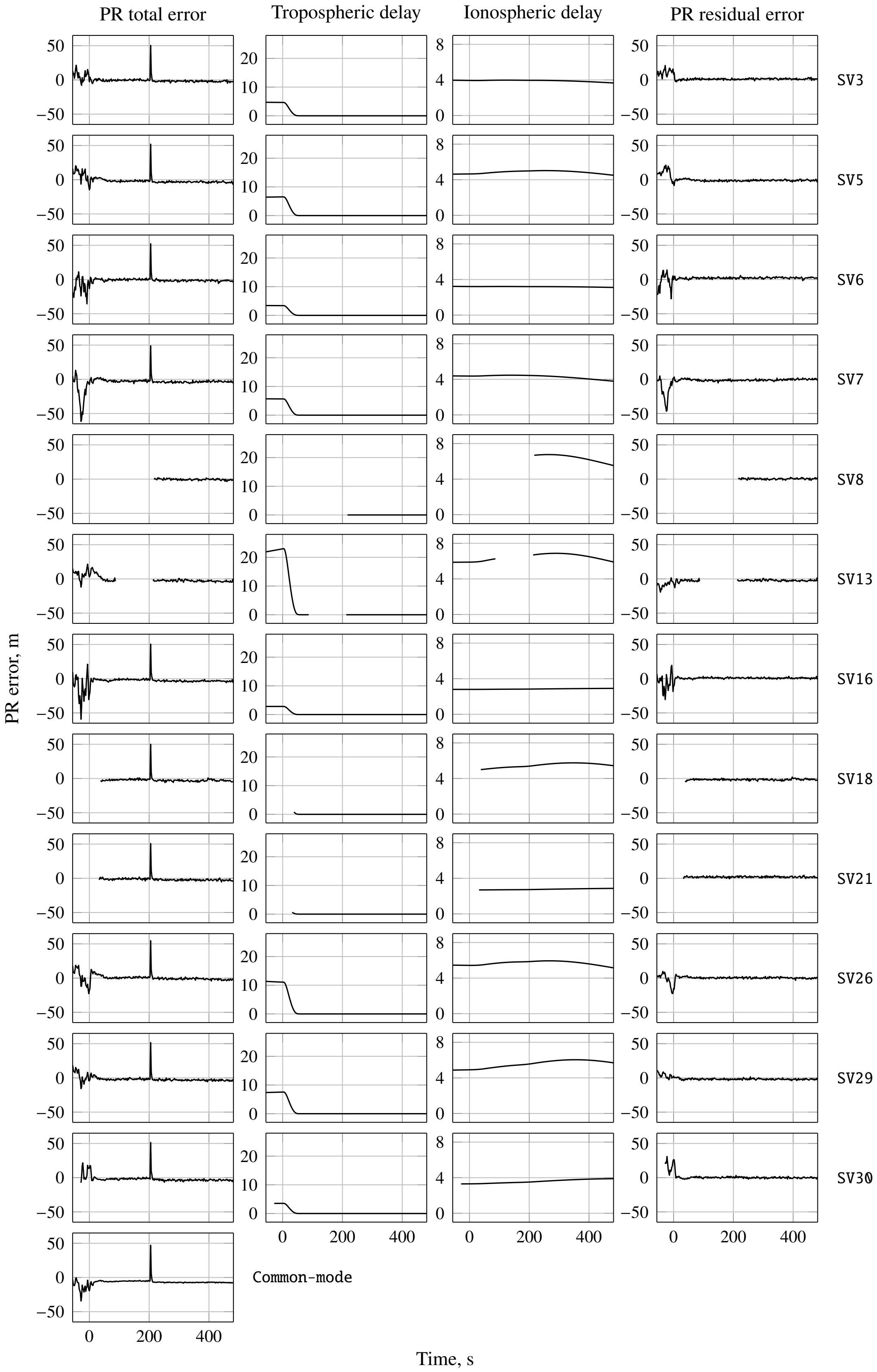


Figure A.8: Estimated PR errors on 12 channels tracked by receiver 2 on the SHEFEX-2 flight

for the error sources just mentioned. These leftovers still include not only residuals of the error sources corrected for, but also multipath, hardware errors and noise, as well as potential errors of the filtered reference trajectory.

The high errors visible before the flight phase ($t < 0$ s) in both raw and corrected residuals are attributed to multipath, reaching up to 50 m in a highly erratic manner. These vanish shortly after lift-off. Tropospheric delays, also significant during pre-flight and early ascent, also vanish quickly with the altitude gain, as the vehicle traverses and eventually leaves the denser part of the Earth's atmosphere. Note that, as expected, tropospheric delays are much higher in lower elevation SVs, reaching up to 23 m for SV13 (~5 deg elevation). The ionospheric delay shows a much lower rate of change and, as the tropospheric effect, is more severe for lower elevation SVs.

At around $t = 205$ s a peak of about 50 m in the total error of all channels (and consequently in the common-mode) can be seen. This coincides with the reacquisition of a low elevation satellite, SV13. One possible explanation is that the receiver may have used a measurement from this newly reacquired SV in the code clock bias estimate before full lock had actually been achieved. This premature measurement would have contaminated all pseudoranges and carrier phases through the shared clock model. Given the common-mode nature of the disturbance, it was filtered out by the least-squares algorithm and did not affect the receiver solution (position and velocity), as shall be seen. Indeed, PR post-correction residuals show no sign of this occurrence.

The corrected signals behave noise-like, especially after early ascent ($t > 60$ s), having a standard deviation around 1 m and a mean of 0.2–3 m (absolute value).

Pseudorange-rate errors, shown in Figure A.9 for all tracked SVs, are estimated in a similar way to those of the pseudorange observables described above. Total errors are corrected for SV clock and relativistic effects (as above described), while tropospheric and ionospheric delay rates are now computed using time-derivatives of the delay models used above. Note that, as explained in Section 2.5.2, ionosphere advances the carrier phase, meaning that the ionospheric delay rate correction is to be added (and not subtracted) to the Doppler signals. Again, common-mode error is also evaluated and displayed, and so is the residual after correction for these error sources.

The most evident feature is the very high magnitude of the errors affecting all channels (400–500 m/s). Being strictly a common-mode effect (as can be seen), this error is attributed to receiver clock drift. Comparing such common-mode curve with that of the PR measurement set in Figure A.8, it becomes clear that the two are disjoint: the estimated receiver clock drift does not follow the time-derivative of the estimated receiver clock delay. Indeed, this is alluded to by the makers of the sensor in [112] when describing the Orion receiver, the predecessor to the Phoenix line of sensors. As explained, the receiver uses an internal clock model to estimate and keep correct time (affecting PR measurements), while Doppler (PRR) measurements are corrupted by frequency error from the free-running oscillator (caused, e.g, by temperature excursions). Note that this decoupling also explains the absence of the high error peak observed in all PRs at $t = 205$ s.

Tropospheric effects are again stronger in lower elevation SVs, manifesting mostly after launch during the early ascent. This is caused by the quickly decreasing residual atmosphere above the receiver. When uncorrected, this hump-like effect (which reaches -0.8 m/s in SV13), severely corrupts the receiver velocity solution (as shall be seen).

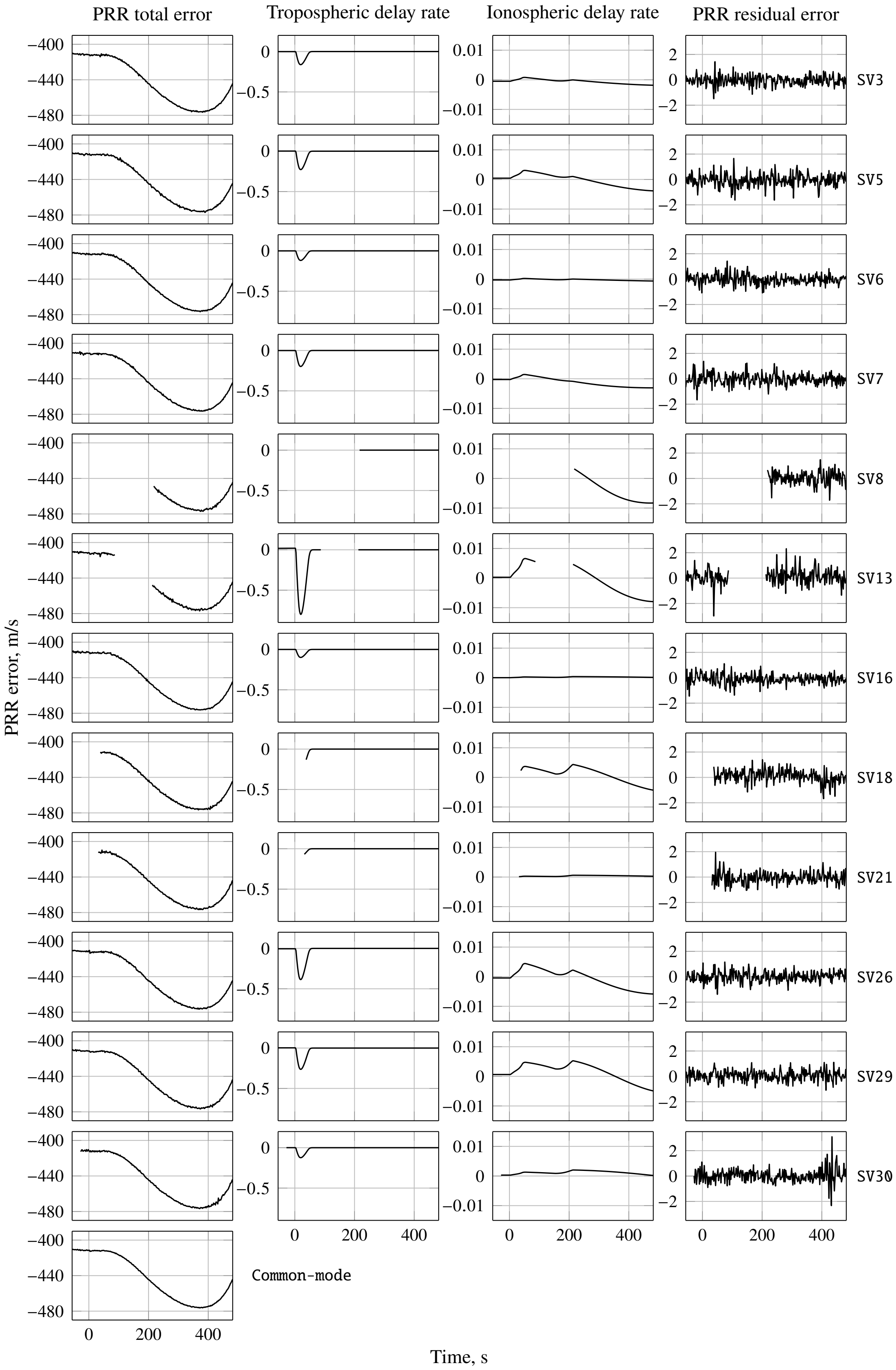


Figure A.9: Estimated PRR errors on 12 channels tracked by receiver 2 on the SHEFEX-2 flight

Given the slow varying ionospheric delays in Figure A.8, ionospheric delay rates are relatively low (e.g., up to ~ 0.007 m/s for SV13). The post-correction residuals show only small bias component (0.02–0.1 m/s) and the standard deviation of their noise-like evolution (0.4–0.6 m/s, 1σ) does not visibly change according to the flight phase (e.g., between on-ground, spin, slew, or de-spun coasting), revealing, e.g., that PRRs are not visibly affected by multipath during the -on-ground period.

Time-differenced carrier phase

The last raw observable analysed is TDCP. As explained in Section 2.5.1, this pseudo-measurement is obtained by differencing integrated carrier phase measurements of consecutive epochs, being thus proportional to range increment during one GPS time period. The total estimated TDCP error (again without SV clock and relativistic effects) along with tropospheric, ionospheric, common-mode and post-correction TDCP errors are shown in Figure A.10 for all tracked SVs. Note that troposphere and ionosphere effects are now delta-delays (i.e., delta-ranges). As noted above in the PRR analysis, ionospheric effects in carrier phase have opposite sign the equivalent effects in PR.

As for PRs, large on-ground errors (up to 10 m) are evident. This time, however, these are almost entirely common-mode, hailing from the receiver clock model (which is presumably driven by the multipath-laden PR measurements). This reveals that TDCPs, like PRRs, are fairly unaffected by multipath errors. Visible in all TDCP channels is also the peak (or better, its time-difference) observed in the PR raw error evolutions ($\sim t = 205$ s), again evidencing the receiver clock model shared with PR.

Estimated tropospheric and ionospheric delay differences are almost identical to the estimated delay rates displayed in Figure A.9, which is to be expected given the slow change of such delays with respect to the TDCP time differencing period (1 s).

The post-correction errors (also plotted) show, in some channels, residuals of the tropospheric delta-range correction (e.g., SV7, SV13, SV29 and SV30 for $0 < t < 60$ s). The remaining parts of the corrected signals show closer resemblance to noise. Note, however, that the error displayed for the on-ground period (0.002–0.004 m, 1σ) is much lower than that during post-tropospheric flight (0.05–0.08 m, 1σ). This may be due to leftover noise in the flight portion of the reference trajectory (the on-ground portion was forcefully made static), suggesting that the residual error in this observable in flight is lower than that shown.

Position and velocity

The estimated error in the receiver navigation solution, i.e., ECEF position and velocity is plotted in Figure A.11. As described in [112, 116] for an earlier version of the DLR-developed receiver (Orion), the navigation solution uses smoothed PRs and carrier-phase-derived (thus also smoothed) range-rates. These are both corrected for tropospheric effects.

Large errors in position (up to ~ 50 m) during the pre-launch phase are due to the multipath delays in the PRs measurements used by the receiver to compute this solution. Velocity solutions are also affected by multipath, however, to a lower degree. A hump of about 1 m/s in velocity error in z-axis (ECEF) is visible during the early ascent

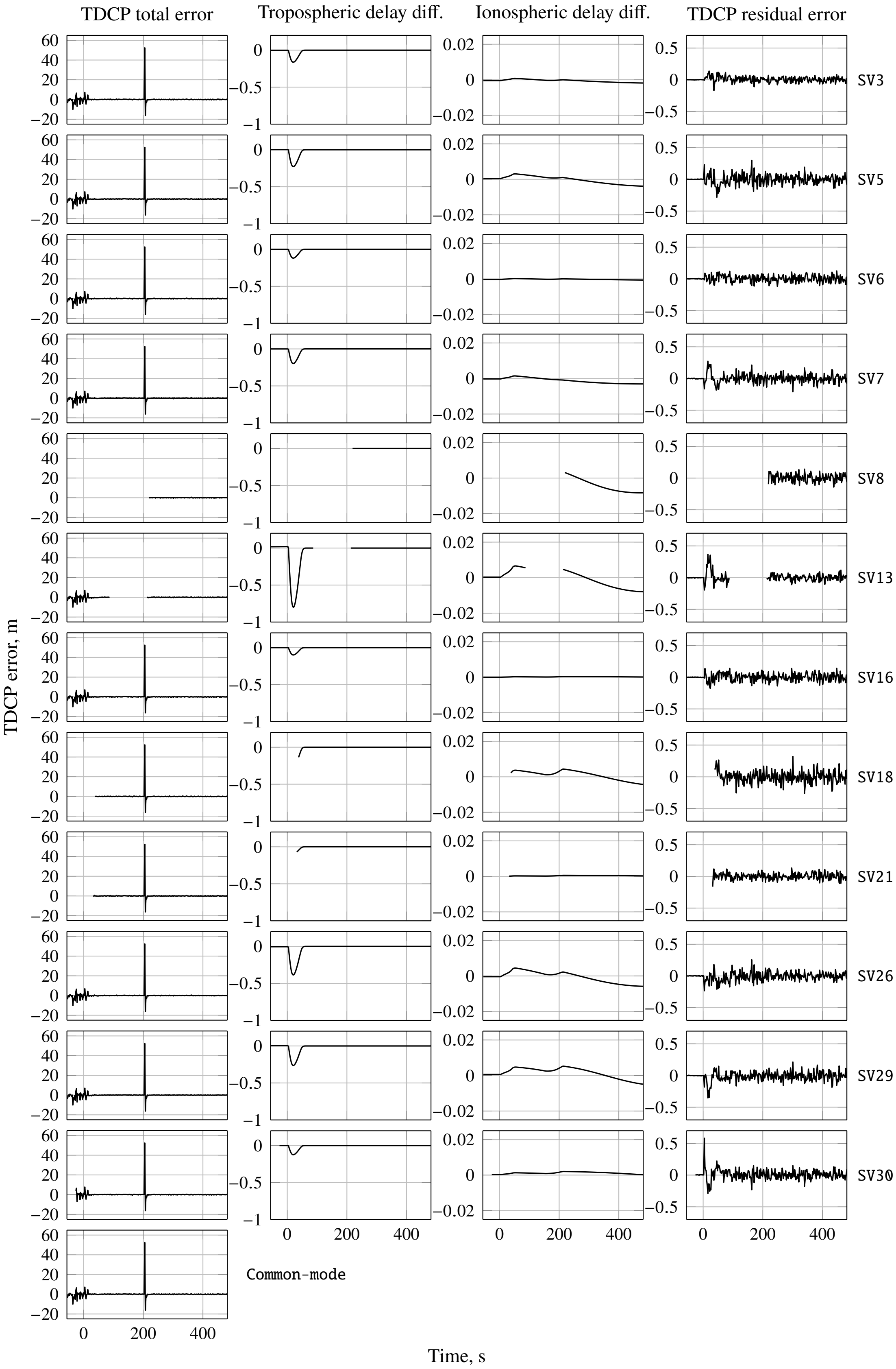


Figure A.10: Estimated TDCP errors on 12 channels tracked by receiver 2 on the SHEFEX-2 flight

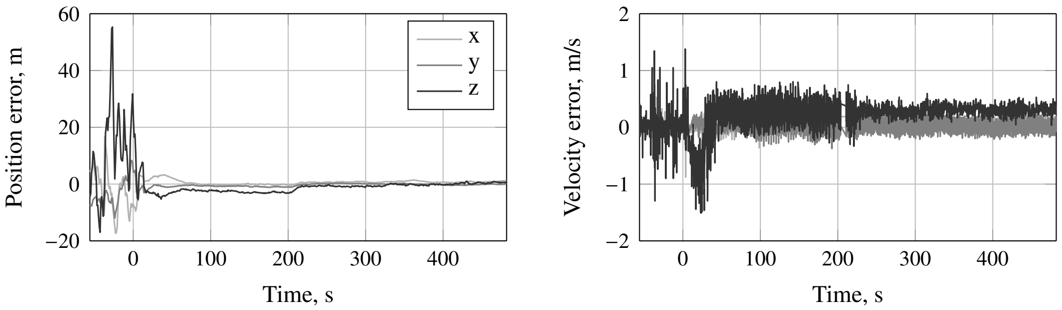


Figure A.11: Receiver 2 position and velocity error (ECEF) during SHEFEX-2 flight

($0 < t < 60$ s). This is due to the fact that the receiver does not apply tropospheric corrections (even if enabled) to the smoothed range-rates which it uses to produce the navigation fix [117, 116]; only smoothed PRs are corrected. Indeed, the slight hump also seen in the position solution at the same period yields from the residual error of the simple tropospheric model used by the device.

Notice that the large peak observed in both PR and TDCP raw measurements at around $t = 260$ s is absent from the navigation solution error history. As aforementioned, this is explained by the strict common-mode nature of this disturbance, which is thus orthogonal to the LS position solution space.

When the tropospheric effects subside ($t > 60$ s), position solutions show standard deviations within 0.2–0.5 m and offsets of up to 3 m, while velocity estimates have standard deviations within 0.1–0.28 m/s and offsets of up to 0.22 m/s. The higher Dilution of Precision (DOP) (about three times higher) along the z-axis of ECEF (see Figure A.12), which is approximately parallel to the local vertical direction near the north pole, explains the larger error (both offset and standard deviation) along this axis.

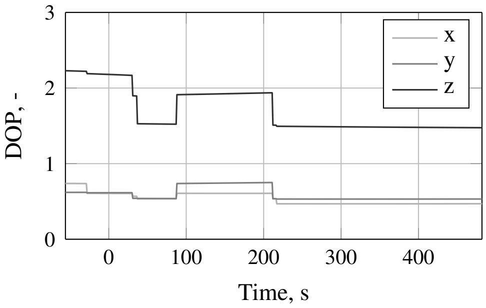


Figure A.12: Receiver 2 solution DOP (in ECEF axes) during SHEFEX-2 flight

B
Vega Launch Vehicle

B.1 Vehicle

Vega is a small expendable European launcher. It is developed by ELV and ESA and is operated by Arianespace from Guiana Space Center, Kourou. The vehicle uses 4-stages and is capable of delivering a payload of up to 1.5 tons to Sun-Synchronous Orbit (SSO) [191]. Its first three stages (P80, Z23 and Z9) use solid propellant motors while the AVUM upper stage has a restartable liquid-fuel motor. It had its maiden flight in February 2012. Figure B.1 shows a graphical depiction of the launcher and a picture of it at the VV02 launch.

B.2 VV02–VV04 mission trajectories

Trajectories of Vega flights VV02, VV03 and VV04 are used in this study. These missions' dates and outcomes were the following:

> **VV02**, launched on May 7th, 2013, delivered Proba-V, an ESA, QinetiQ Space Belgium built Earth observation satellite [22], to a 820 km-high, 98.7 deg-inclined SSO, and VNREDSat-1 and ESTCube-1 to a 665 km-high, 98.1 deg-inclined SSO [188, 6];
>
> **VV03**, launched on April 30th, 2014, delivered KazEOSat-1 (DZZ-HR), Kazakhstan's first Earth observation satellite [120], to a 750 km-high, 98.5 deg-inclined SSO [189, 7];
>
> **VV04**, launched on February 11th, 2015, put the Intermediate eXperimental Vehicle (IXV), an ESA's re-entry experimental vehicle [185], on a 416×76 km, 5.4 deg-inclined LEO suborbital (intercontinental) trajectory [190, 192].

Figure B.1: Vega graphical depiction and picture of VV02 launch. Source: ESA (`https://www.esa.int`), Picture IDs: 195334 (left), 291104 (right), ©ESA.

Figure B.2 shows the ECEF trajectories of VV02 and VV04 flights (VV03 is not depicted due to its closeness to VV02) and the altitude and velocity profiles of all the three trajectories. Note that VV02 (and VV03) and VV04 launch towards very different directions. In terms of altitude and velocity, VV02 and VV03, both targeting similar SSO orbits, have close evolutions, while VV04 differs significantly, especially in altitude. Real-flight measured specific-force acceleration and angular rate for each of the three flights are shown in Figure B.3 from lift-off (T0) up to T0+900 s. This interval includes all manoeuvres up to the end of the first burn of the fourth (and upper) stage, AVUM. After this point mostly coasting takes place. VV02 and VV03 show quite similar acceleration evolutions; VV04 has a somewhat weaker (though longer) third (Z9) and fourth (AVUM) burns. In all three flights between lift-off and T0+200 s the vehicle spins at 7–11 deg/s around its longitudinal axis (Body z-axis).

An extended VV02 trajectory profile was obtained by combining real-flight data for T0–T0+900 s with pre-flight simulated data for T0+900–T0+3327 s (i.e., up to the release of Proba-V satellite). The ECEF path as well altitude and velocity profiles of this extended trajectory are shown in Figure B.4.

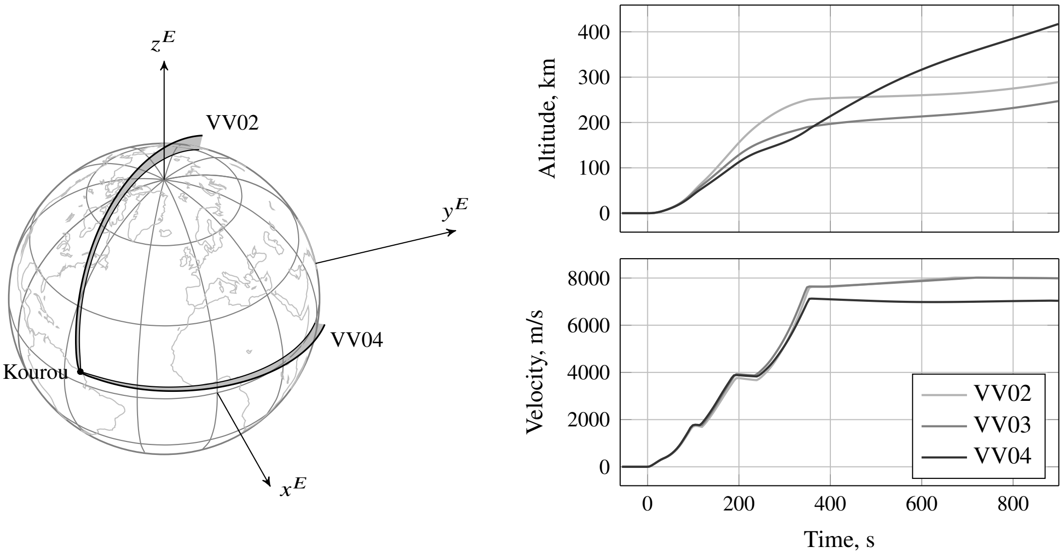


Figure B.2: VV02, VV03 and VV04 Vega trajectories: altitude, velocity and ECEF path (VV02 and VV04 only)

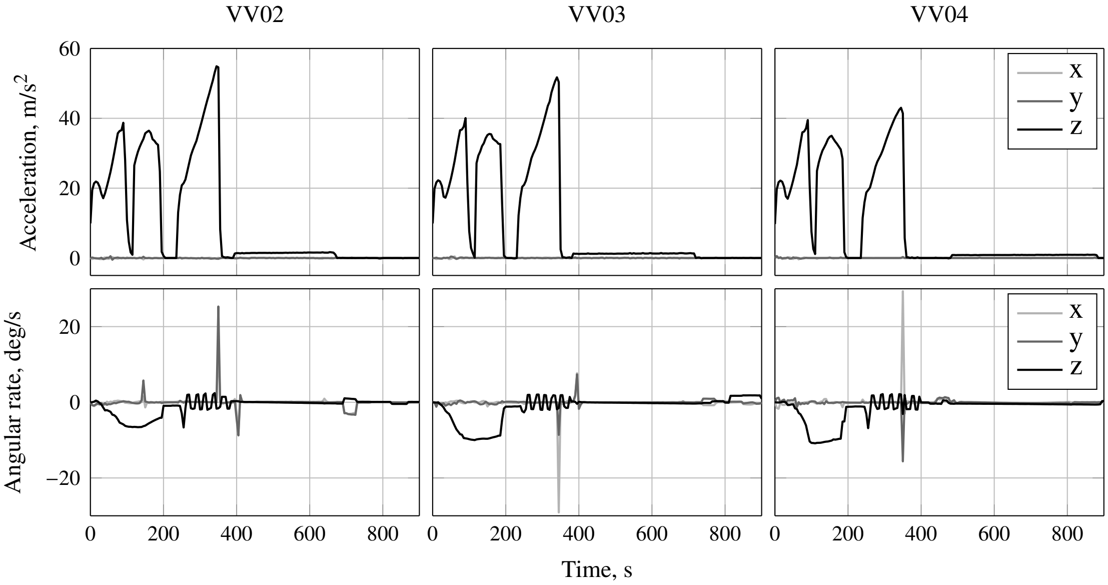


Figure B.3: Specific-force acceleration and angular rate measurement profiles (in Body axes) of VV02, VV03 and VV04 Vega flights

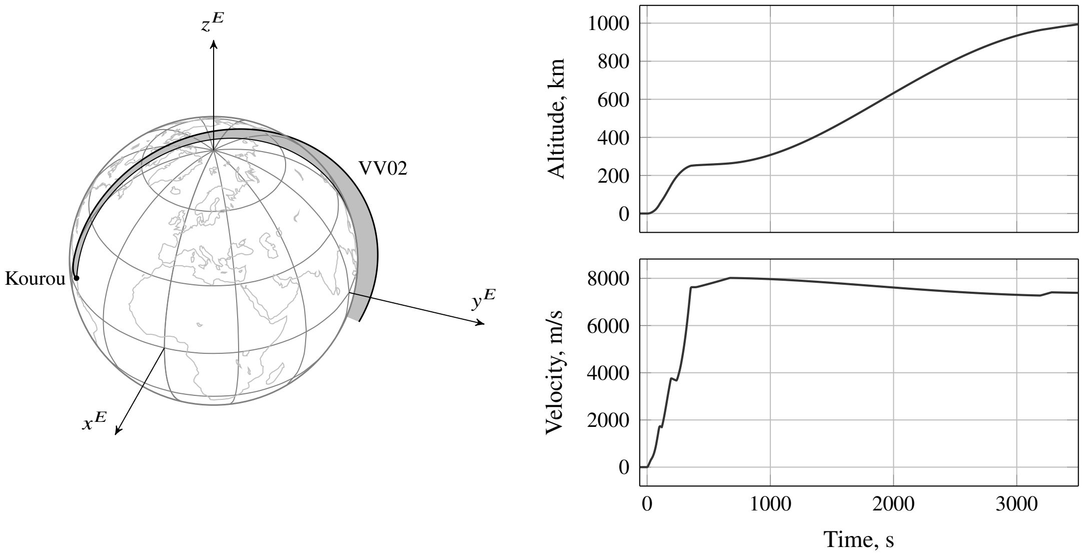


Figure B.4: Extended VV02 Vega trajectory: altitude, velocity and ECEF path

C

IMU Models

As evidenced in Chapter 2, inertial measurements are crucial for launch vehicle navigation. This appendix describes the general n-axis IMU model used herein for high-fidelity simulation and the restructuring and reduction of this model for parametric estimation purposes.

C.1 IMU n-axes model

Inertial Measurement Units generally contain multiple single-axis gyroscopes and accelerometers arranged in diverse geometries. While the most common set-ups use three single-axis sensors of each kind in an orthogonal fashion, instances of redundant configurations are certainly not rare, especially in aeronautics and space applications (e.g., [48, 50, 58, 63]).

In the case of strapdown inertial units (i.e., ungimbaled), the measurements are collected and retrieved in a sensor-fixed (thus, if rigidly mounted, also Body-fixed) reference frame. Such inertial measurements are expressed in one of two ways: accelerations and rotation rates; or time-step-integrations of the two, i.e., increments of velocity and attitude. The latter is used in this work. In such case these measurements are

$$\boldsymbol{\Delta}\tilde{\boldsymbol{\theta}}_k^B = \int_{t_k}^{t_{k+1}} \tilde{\omega}_{IB}^B \ \mathsf{d}t \tag{C.1}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{v}}_k^B = \int_{t_k}^{t_{k+1}} \tilde{\mathbf{a}}_{\mathsf{sf}}^B \ \mathsf{d}t \ , \tag{C.2}$$

where $\tilde{\mathbf{a}}_{\mathsf{sf}}^B$ and $\tilde{\omega}_{IB}^B$ are the specific-force acceleration and absolute angular velocity felt by the sensors.

For an IMU with n single-axis accelerometers and n single-axis gyroscopes the total acceleration and absolute angular rate felt by the sensor sets can given by

$$\tilde{\boldsymbol{\omega}}_{IB}^{B} = \hat{\mathbf{G}}_{\mathsf{g}}^{B} \sum_{i=1}^{n} \hat{\mathbf{e}}_{\mathsf{g},i}^{B} \, \tilde{\omega}_i \tag{C.3}$$

$$\tilde{\mathbf{a}}_{\mathsf{sf}}^{B} = \hat{\mathbf{G}}_{\mathsf{a}}^{B} \sum_{i=1}^{n} \hat{\mathbf{e}}_{\mathsf{a},i}^{B} \, \tilde{\mathrm{a}}_{\mathsf{sf},i} \ , \tag{C.4}$$

where $\tilde{\mathrm{a}}_{\mathsf{sf},i}$ is the measurement of the i^{th} accelerometer and ω_i is the rate measured by the i^{th} gyroscope; $\hat{\mathbf{e}}_{\mathsf{a},i}^{B}$ is the measured (known) unit direction of the i^{th} accelerometer in B frame, $\hat{\mathbf{e}}_{\mathsf{g},i}^{B}$ is the equivalent for the i^{th} gyroscope. The constant weight matrices $\hat{\mathbf{G}}_{\mathsf{a}}^{B}$ and $\hat{\mathbf{G}}_{\mathsf{g}}^{B}$ can be obtained, for instance, through a simple least-squares law, being then

$$\hat{\mathbf{G}}_{\mathsf{g}}^{B} = \left(\hat{\mathbf{U}}_{\mathsf{g}}^{B} \hat{\mathbf{U}}_{\mathsf{g}}^{B\,\mathsf{T}}\right)^{-1} \quad \text{and} \quad \hat{\mathbf{G}}_{\mathsf{a}}^{B} = \left(\hat{\mathbf{U}}_{\mathsf{a}}^{B} \hat{\mathbf{U}}_{\mathsf{a}}^{B\,\mathsf{T}}\right)^{-1} \ , \tag{C.5}$$

with sensor mapping matrices

$$\hat{\mathbf{U}}_{\mathsf{g}}^{B} = \left[\hat{\mathbf{e}}_{\mathsf{g},1}^{B} \quad \hat{\mathbf{e}}_{\mathsf{g},2}^{B} \quad \cdots \quad \hat{\mathbf{e}}_{\mathsf{g},n}^{B}\right] \quad \text{and} \quad \hat{\mathbf{U}}_{\mathsf{a}}^{B} = \left[\hat{\mathbf{e}}_{\mathsf{a},1}^{B} \quad \hat{\mathbf{e}}_{\mathsf{a},2}^{B} \quad \cdots \quad \hat{\mathbf{e}}_{\mathsf{a},n}^{B}\right] \ . \tag{C.6}$$

Note that for $\hat{\mathbf{G}}$ to be defined $\hat{\mathbf{U}}$ has to have rank equal to 3, i.e., the set of single-axis sensors have to span all Eucledean dimensions. Note also that in the present chapter a unit with the same number of gyro and accelerometer axes is assumed; departure from this assumption is however straightforward.

C.2 IMU model for simulation

As described in Section 2.4.2, the IMU measurements are subjected to numerous errors and disturbances. These can have multiple causes and be of deterministic or stochastic nature. In order to realistically generate IMU signals such effects, errors and noises must be simulated. This section presents the high-fidelity models used to simulate such effects on each of the single-axis sensor of the n-axes unit considered in the previous section.

C.2.1 Single-axis models

The measurement of each of the single-axis accelerometers and gyroscopes can be simulated as shown in Figures C.1 and C.2, respectively. Note that these sensors provide incremental (or integrated) measurements.

The models depicted include the following components, which when not specified are valid for both gyro and accelerometer models:

- **Axis misalignment $\boldsymbol{\vartheta}_i^B$** modeled as a random constant and applied as

$$\mathbf{e}_i^B = \left(\mathbf{I} + \left[\boldsymbol{\vartheta}_i^B \times\right]\right) \hat{\mathbf{e}}_i^B \ , \tag{C.7}$$

  where $\mathbf{e}_i^B$ and $\hat{\mathbf{e}}_i^B$ are the actual and known sensor axes, respectively.

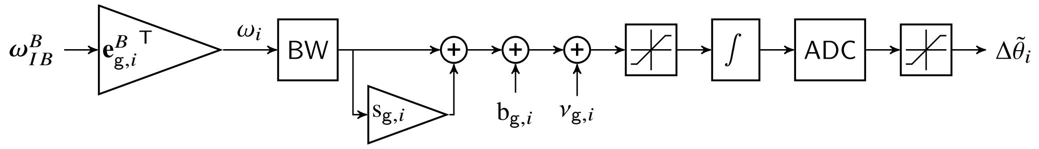

ω_{IB}^{B}
$\mathbf{e}_{\mathrm{g},i}^{B\,\mathsf{T}}$
ω_i
BW
$\mathrm{s}_{\mathrm{g},i}$
$\mathrm{b}_{\mathrm{g},i}$
$\nu_{\mathrm{g},i}$
$\int$
ADC
$\Delta\tilde{\theta}_i$

Figure C.1: Single-axis gyroscope model

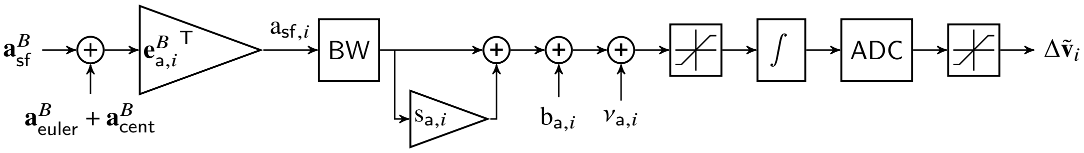

$\mathbf{a}_{\mathrm{sf}}^{B}$
$\mathbf{a}_{\mathrm{euler}}^{B} + \mathbf{a}_{\mathrm{cent}}^{B}$
$\mathbf{e}_{\mathrm{a},i}^{B\,\mathsf{T}}$
$\mathrm{a}_{\mathrm{sf},i}$
BW
$\mathrm{s}_{\mathrm{a},i}$
$\mathrm{b}_{\mathrm{a},i}$
$\nu_{\mathrm{a},i}$
$\int$
ADC
$\Delta\tilde{\mathbf{v}}_i$

Figure C.2: Single-axis accelerometer model

- **Bandwidth** (BW) dynamics, modeled as a first-order filter

$$\mathrm{BW}_i(\mathrm{s}) = \frac{2\pi \mathrm{f}_i}{2\pi \mathrm{f}_i + \mathrm{s}} , \tag{C.8}$$

  where f_i is the 3 dB cut-off bandwidth frequency.

- **Scale-factor** error $\mathrm{s}_{i,j}$ computed as

$$\mathrm{s}_{i,j} = \mathrm{s}_{\mathrm{RW},i,j} + \underbrace{\mathrm{s}_{\mathrm{G,g},i,j}}_{\text{gyro only}} + \mathrm{s}_{\mathrm{NL},i,j} \tag{C.9}$$

  with:

  - turn-on term and random-walk components

$$\mathrm{s}_{\mathrm{RW},i,j} = \mathrm{s}_{\mathrm{RW},i,j-1} + \sqrt{\Delta t_j}\,\sigma_{\mathrm{SFRW},i}\,\mathrm{w}_{\mathrm{RW},i,j} , \tag{C.10}$$

    where $\mathrm{s}_{\mathrm{RW},i,j=0}$ is the sensor turn-on scale-factor, Δt_j is the sample time, $\sigma_{\mathrm{SFRW},i}$ is the scale-factor random-walk standard deviation (also referred to as *scale-factor drift*), and $\mathrm{w}_{\mathrm{WN},i,j}$ is a standard normally distributed sample;

  - in the gyro, an acceleration (G-) sensitive component

$$\mathrm{s}_{\mathrm{G,g},i,j} = \mathrm{m}_{\mathrm{s,g},i}\,\mathrm{a}_{\mathrm{sf},i,j} \tag{C.11}$$

    where the scale-factor G-sensitivity parameter $\mathrm{m}_{\mathrm{s,g},i}$ is modeled as random constant; and

  - a non-linearity term

$$\mathrm{s}_{\mathrm{NL},i,j} = \mathrm{NL}_i(\omega_{i,j} \ \text{or} \ \mathrm{a}_{\mathrm{sf},i,j}) \tag{C.12}$$

    where the non-linearity model function $\mathrm{NL}_i(\,\cdot\,)$ is a set of 3rd-order splines that piece-wise interpolate evenly-spaced, randomly-distributed (and time-wise constant) non-linearity samples along the entirety of the sensor's measurement range; this model is discussed in more detail in Section C.2.2.

- **Bias** error $\mathrm{b}_{i,j}$ as

$$\mathrm{b}_{i,j} = \mathrm{b}_{\mathrm{RW},i,j} + \mathrm{b}_{\mathrm{EC},i,j} + \underbrace{\mathrm{b}_{\mathrm{G,g},i,j}}_{\text{gyro only}} \,, \tag{C.13}$$

  including:

  - a turn-on and random-walk components modeled similarly to (C.10) as

$$\mathrm{b}_{\mathrm{RW},i,j} = \mathrm{b}_{\mathrm{RW},i,j-1} + \sqrt{\Delta t_j}\,\sigma_{\mathrm{K},i}\,\mathrm{w}_{\mathrm{RW},i,j} \,, \tag{C.14}$$

    where $\mathrm{b}_{\mathrm{RW},i,j=0}$ is the sensor turn-on bias, $\sigma_{\mathrm{K},i}$ is the rate/acceleration random-walk (gyro/accelerometer) standard deviation (e.g., obtained by static testing and Allan variance analysis, see Section C.4), and $\mathrm{w}_{\mathrm{RW},i,j} \sim \mathcal{N}(0,1)$;

  - an exponentially correlated (EC) or colored component caused, for instance, by thermal effects within the sensor [187] and modeled as a discrete-time 1st-order Gauss-Markov (GM) [31]

$$\mathrm{b}_{\mathrm{EC},i,j} = e^{-\beta_{\mathrm{EC},i}\Delta t_j}\mathrm{b}_{\mathrm{EC},i,j-1} + \sigma_{\mathrm{EC},i,\infty}\sqrt{1-e^{-2\beta_{\mathrm{EC},i}\Delta t_j}}\,\mathrm{w}_{\mathrm{EC},i,j} \,, \tag{C.15}$$

    where $\beta_{\mathrm{EC},i}$ is the inverse of the process correlation time, $\mathrm{w}_{\mathrm{EC},i,j} \sim \mathcal{N}(0,1)$, and $\sigma_{\mathrm{EC},i,\infty}$ is the standard deviation of the GM process (see Section C.4 for the Allan variance profile of this process); if the initial state of the process $\mathrm{b}_{\mathrm{EC},i,0}$ is distributed with a standard deviation equal to $\sigma_{\mathrm{EC},\infty}$ then the process is said to be stationary, otherwise, the process standard deviation will tend exponentially to $\sigma_{\mathrm{EC},\infty}$; and

  - in the gyroscope model, an acceleration sensitive term (G-sensitivity) modeled as depending only on the acceleration component parallel to the unit axis (approximation which is valid even for MEMS-grade sensors [10]) as

$$\mathrm{b}_{\mathrm{G,g},i,j} = \mathrm{m}^{B}_{\mathrm{b,g},i}\,\mathrm{a}_{\mathrm{sf},i,j} \tag{C.16}$$

    where the bias G-sensitivity parameter $\mathrm{m}_{\mathrm{B,g},i}$ is modeled as random constant;

- **Measurement noise** v_i, including:

  - a white Gaussian noise term, also known as angle or velocity RW in the gyro and accelerometer, respectively, which are obtained as

$$v_{\mathrm{WN},i,j} = \sqrt{\Delta t_j}\sigma_{\mathrm{N},i}\mathrm{w}_{\mathrm{WN},i,j} \,, \tag{C.17}$$

    where $\sigma_{\mathrm{N},i}$ is the angle/velocity RW standard deviation (e.g., obtained through static testing and Allan variance analysis, see Section C.4), and $\mathrm{w}_{\mathrm{WN},i,j} \sim \mathcal{N}(0,1)$; and

  - a flicker noise (also known as pink noise or bias instability) component produced (offline) through inverse Fast Fourier Transform of an $1/f$ spectrum.

- **Measurement saturation**, which depends on the measurement principle of the sensorn and imposed on the sensor analog signal;

- **Quantization noise** which arises from discretization or analog-to-digital conversion of the analog measurement with the unit's least significant bit value, also known as output resolution; and
- **Output saturation**, determined by the digital format of the output, imposed on the sensor digital signal.

The accelerometer measurement takes into account the centripetal $\mathbf{a}_{\text{cent},i}$ and Euler $\mathbf{a}_{\text{euler},i}$ terms caused by the lever-arm of the sensor measurement point with respect to the origin of the measurement unit. These are given by

$$\mathbf{a}^{B}_{\text{cent},i} = \left(\boldsymbol{\Omega}^{B}_{IB}\right)^{2} \mathbf{l}^{B}_{\text{a},i} \tag{C.18}$$

$$\mathbf{a}^{B}_{\text{euler},i} = \dot{\boldsymbol{\Omega}}^{B}_{IB}\mathbf{l}^{B}_{\text{a},i} \ , \tag{C.19}$$

where $\mathbf{l}^{B}_{\text{a},i}$ is the lever arm from the center of B frame to the measurement point of sensor i, in B coordinates, and $\boldsymbol{\Omega}^{B}_{IB}$ is the skew-symmetric matrix of $\boldsymbol{\omega}^{B}_{IB}$.

C.2.2 Scale-factor non-linearity model

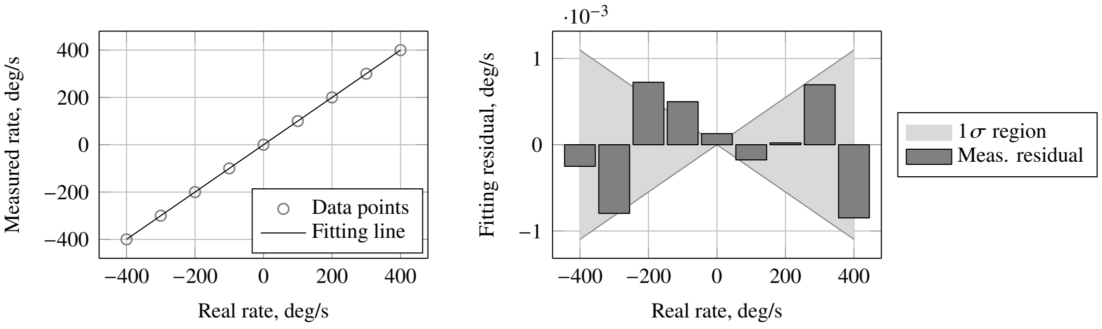
The scale-factor non-linearity is an important source of error in medium/low grade inertial sensors. A common way to model this error source is through a quadratic trend departing from the linear scale-factor curve. This is, in general, an over-simplistic description of this error source. The following text describes an alternative non-linearity model. Gyro is used as an example; accelerometer modeling is identical.

Figure C.3a shows a set of sparsely collected data from a FOG sensor stimulated by a rotation table at different, equally spaced, rates. Figure C.3b zooms-in and displays the linear fitting residuals, i.e., the scale-factor non-linearity. From these residuals a 1σ area is computed modeling it as linearly increasing with the measured rate.

The non-linearity value is then given as

$$\sigma_{\text{NL,g},i} = \text{std}\left\{\frac{\epsilon(\omega_i)}{\omega_i}\right\} \tag{C.20}$$

where $\text{std}\{\bullet\}$ denotes the standard deviation, and $\epsilon(\omega_i)$ is the residual for the i$^{\text{th}}$ input rate sample ω_i. The value of $\sigma_{\text{NL,g},i}$ should be numerically computed with data covering the sensor's full measurement range.



(a) Data points and fitting

(b) Fitting residuals

Figure C.3: Linear scale factor fitting of experimental points from a FOG gyroscope

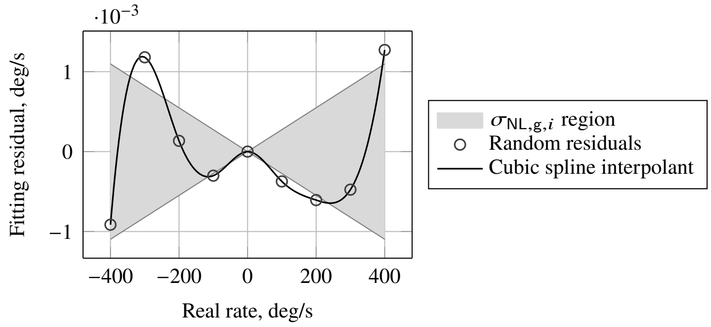


Figure C.4: Generated non-linearity residuals

The standard deviation (C.20) can then be used to simulate the scale-factor non-linearity profile of the sensor. This profile is generated by bridging a set of randomly generated, equally-spaced (covering the entire sensor range) residuals normally-distributed with $\sigma_{\mathsf{NL},g,i}$ (from sensor testing results or datasheet values) with cubic splines. The non-linearity perturbation at each epoch is obtained through interpolation of the complete set of such splines according to the instantaneous undisturbed physical measurement. Figure C.4 shows an example of random residuals and interpolating splines.

C.3 IMU model for estimation

The IMU simulation model presented in the previous section features numerous non-linear effects and error sources. This prevents its direct application within an estimation filter as that described in Chapter 3. Moreover, for units with several redundant axes (i.e., $n > 3$) the number of potential states needed to estimate the sensor uncertainties scales accordingly, increasing the computing cost of the filter.

In this section a simpler and linear model is presented which can be used in the estimation algorithm at hand. This model also collapses the perturbations of an n-axis sensor configuration to 3-dimensional space using the method described in Section C.1.

C.3.1 Single-axis models

Considering an n-axis IMU, the measurement of each single-axis sensor can be reduced to the form

$$\Delta\tilde{\theta}_i = \left(\hat{\mathbf{e}}^B_{g,i} + \mathrm{s}_{g,i}\hat{\mathbf{e}}^B_{g,i} + \hat{\mathbf{e}}^B_{g,i} \times \boldsymbol{\vartheta}^B_{g,i}\right)^{\mathsf{T}} \boldsymbol{\Delta\theta}^B + \mathrm{m}_{g,i}\hat{\mathbf{e}}^B_{g,i}{}^{\mathsf{T}} \boldsymbol{\Delta}\mathbf{v}^B_{\mathsf{sf}} + \Delta t \mathrm{b}_{g,i} + \nu_{g,i} \ , \tag{C.21}$$

$$\Delta\tilde{\mathrm{v}}_{\mathsf{sf},i} = \left(\hat{\mathbf{e}}^B_{a,i} + \mathrm{s}_{a,i}\hat{\mathbf{e}}^B_{a,i} + \hat{\mathbf{e}}^B_{a,i} \times \boldsymbol{\vartheta}^B_{a,i}\right)^{\mathsf{T}} \left(\boldsymbol{\Delta}\mathbf{v}^B_{\mathsf{sf}} + \boldsymbol{\Delta}\mathbf{v}^B_{\mathsf{cent},i} + \boldsymbol{\Delta}\mathbf{v}^B_{\mathsf{euler},i}\right) + \Delta t \mathrm{b}_{a,i} + \nu_{a,i} \tag{C.22}$$

where $\mathrm{s}_{g,i}$ and $\mathrm{s}_{a,i}$ are the scale-factor errors of the i^{th} accelerometer and gyro, respectively, $\mathrm{b}_{g,i}$ and $\mathrm{b}_{a,i}$ are their biases, and $\nu_{g,i}$ and $\nu_{a,i}$ are their noise components. $\mathrm{m}_{g,i}$ is the gyro G-sensitivity. $\boldsymbol{\vartheta}^B_{g,i}$ and $\boldsymbol{\vartheta}^B_{a,i}$ are the misalignment angles of the *known* sensor unit directions, $\hat{\mathbf{e}}^B_{g,i}$ and $\hat{\mathbf{e}}^B_{a,i}$, with respect to the *real* ones, $\mathbf{e}^B_{g,i}$ and $\mathbf{e}^B_{a,i}$, according to relation (C.7). $\Delta\mathbf{v}^B_{\mathsf{cent},i}$ and $\Delta\mathbf{v}^B_{\mathsf{euler},i}$ are approximated from (C.18)–(C.19) as

$$\Delta\mathbf{v}^B_{\mathsf{cent},i} \approx \frac{1}{\Delta t}\,\Delta\boldsymbol{\theta}^B \times \Delta\boldsymbol{\theta}^B \times \mathbf{l}^B_{\mathsf{a},i} \tag{C.23}$$

$$\Delta\mathbf{v}^B_{\mathsf{euler},i} \approx \frac{1}{\Delta t}\,\Delta\dot{\boldsymbol{\theta}}^B \times \mathbf{l}^B_{\mathsf{a},i}\ . \tag{C.24}$$

C.3.2 Collapsed n-axes models

Combining the uncertainty models (C.22)-(C.21) with the measurement summation schemes (C.4)-(C.3) yields

$$\begin{aligned}\Delta\tilde{\boldsymbol{\theta}}^B = {} & \hat{\mathbf{G}}^B_{\mathsf{g}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{g},i}\left(\hat{\mathbf{e}}^B_{\mathsf{g},i} + \mathrm{s}_{\mathsf{g},i}\hat{\mathbf{e}}^B_{\mathsf{g},i} + \hat{\mathbf{e}}^B_{\mathsf{g},i} \times \boldsymbol{\vartheta}^B_{\mathsf{g},i}\right)^{\mathsf{T}} \Delta\boldsymbol{\theta}^B + \hat{\mathbf{G}}^B_{\mathsf{g}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{g},i}\mathrm{m}_{\mathsf{g},i}\hat{\mathbf{e}}^{B\ \mathsf{T}}_{\mathsf{g},i}\,\Delta\mathbf{v}^B_{\mathsf{sf}} \\ & + \Delta t\hat{\mathbf{G}}^B_{\mathsf{g}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{g},i}\mathsf{b}_{\mathsf{g},i} + \hat{\mathbf{G}}^B_{\mathsf{g}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{g},i}\nu_{\mathsf{g},i}\end{aligned} \tag{C.25}$$

$$\begin{aligned}\Delta\tilde{\mathbf{v}}^B_{\mathsf{sf}} = {} & \hat{\mathbf{G}}^B_{\mathsf{a}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{a},i}\left(\hat{\mathbf{e}}^B_{\mathsf{a},i} + \mathrm{s}_{\mathsf{a},i}\hat{\mathbf{e}}^B_{\mathsf{a},i} + \hat{\mathbf{e}}^B_{\mathsf{a},i} \times \boldsymbol{\vartheta}^B_{\mathsf{a},i}\right)^{\mathsf{T}} \left(\Delta\mathbf{v}^B_{\mathsf{sf}} + \Delta\mathbf{v}^B_{\mathsf{cent},i} + \Delta\mathbf{v}^B_{\mathsf{euler},i}\right) \\ & + \Delta t\hat{\mathbf{G}}^B_{\mathsf{a}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{a},i}\mathsf{b}_{\mathsf{a},i} + \hat{\mathbf{G}}^B_{\mathsf{a}} \sum_{i=1}^{n} \hat{\mathbf{e}}^B_{\mathsf{a},i}\nu_{\mathsf{a},i}\end{aligned} \tag{C.26}$$

where $\hat{\mathbf{G}}^B_{\mathsf{g}}$ and $\hat{\mathbf{G}}^B_{\mathsf{a}}$ are as in (C.5).

The model (C.25)–(C.26) has a total number of uncertainties (i.e., potential estimator states) of $11n$, i.e., 6 per gyro axis plus 5 per accelerometer axis. This can be reduced to $9n$ as each misalignment $\boldsymbol{\vartheta}^B_i$ of each axis can be modeled as a 2-element angle instead of a 3-element one, as a rotation around $\hat{\mathbf{e}}^B_i$ yields null error.

Although the $9n$-uncertainty model (C.25)–(C.26) is in a form suitable for integration within a state estimator, it can, depending on the number of axis n, potentially yield a cumbersomely high number of states. This can be reduced by redefining the overall sensor uncertainties around the three orthogonal directions of a sensor-fixed (here B) coordinate frame instead of along the $2n$ sensing axes. In units with triads of single-axis sensors arranged orthogonally this process is unnecessary. In such case, $\hat{\mathbf{G}}^B$ and $\sum_{i=1}^{n} \hat{\mathbf{e}}^B_i(\hat{\mathbf{e}}^B_i)^{\mathsf{T}}$ are both identity matrices.

Scale-factor, non-orthogonality and misalignment

In the general case, the 3×3 matrix terms multiplying the real inertial quantities $\Delta\boldsymbol{\theta}^B$ and $\Delta\mathbf{v}^B_{\mathsf{sf}}$ in (C.26)–(C.25) can be decomposed into diagonal, antisymmetric (or skew-symmetric) and symmetric matrix parts, as

$$\hat{\mathbf{G}}^B \sum_{i=1}^{n} \hat{\mathbf{e}}^B_i\left(\hat{\mathbf{e}}^B_i + \mathrm{s}_i\hat{\mathbf{e}}^B_i + \hat{\mathbf{e}}^B_i \times \boldsymbol{\vartheta}^B_i\right)^{\mathsf{T}} = \mathbf{I} + \sum_{i=1}^{n} \mathbf{A}_{\mathsf{diag},i} + \sum_{i=1}^{n} \mathbf{A}_{\mathsf{sym},i} + \sum_{i=1}^{n} \mathbf{A}_{\mathsf{skew},i}\ , \tag{C.27}$$

where the matrices $\mathbf{A}_{\mathsf{diag}}$, $\mathbf{A}_{\mathsf{sym}}$, and $\mathbf{A}_{\mathsf{skew}}$ are diagonal, null-diagonal symmetric, and null-diagonal skew-symmetric. It is easily shown that the identity matrix in (C.27) arises as

$$\hat{\mathbf{G}}^B \sum_{i=1}^{n} \hat{\mathbf{e}}_i^B \hat{\mathbf{e}}_i^{B^\mathsf{T}} = \hat{\mathbf{G}}^B \hat{\mathbf{U}}^B \hat{\mathbf{U}}^{B^\mathsf{T}} = \mathbf{I}\,. \tag{C.28}$$

The diagonal-symmetric-antisymmetric matrix decomposition is obtained as

$$\mathbf{A} = \mathbf{A}_{\mathsf{diag}} + \underbrace{\tfrac{1}{2}\left(\mathbf{A} + \mathbf{A}^\mathsf{T} - 2\mathbf{A}_{\mathsf{diag}}\right)}_{\mathbf{A}_{\mathsf{sym}}} + \underbrace{\tfrac{1}{2}\left(\mathbf{A} - \mathbf{A}^\mathsf{T}\right)}_{\mathbf{A}_{\mathsf{skew}}} \tag{C.29}$$

where $\mathbf{A}_{\mathsf{diag}}$ is a diagonal matrix with the same diagonal elements of $\mathbf{A}$.

The decomposition (C.27) yields a typical multiplicative uncertainty set: scale-factor (diagonal matrix), non-orthogonality (symmetric matrix) and axes-misalignment (skew matrix) as

$$\sum_{i=1}^{n} \mathbf{A}_{\mathsf{diag},i} = \mathsf{diag}(\mathbf{s}^B) = \begin{bmatrix} \mathrm{s}_x & 0 & 0 \\ 0 & \mathrm{s}_y & 0 \\ 0 & 0 & \mathrm{s}_z \end{bmatrix}, \tag{C.30}$$

$$\sum_{i=1}^{n} \mathbf{A}_{\mathsf{sym},i} = \boldsymbol{S}(\boldsymbol{\eta}^B) = \begin{bmatrix} 0 & \eta_z & \eta_y \\ \eta_z & 0 & \eta_x \\ \eta_y & \eta_x & 0 \end{bmatrix}, \quad \text{and} \tag{C.31}$$

$$\sum_{i=1}^{n} \mathbf{A}_{\mathsf{skew},i} = \left[\boldsymbol{\vartheta}^B \times\right] = \begin{bmatrix} 0 & -\vartheta_z & \vartheta_y \\ \vartheta_z & 0 & -\vartheta_x \\ -\vartheta_y & \vartheta_x & 0 \end{bmatrix}, \tag{C.32}$$

where the scale-factor $\mathbf{s}^B$, non-orthogonality $\boldsymbol{\eta}^B$ and misalignment $\boldsymbol{\vartheta}^B$ are 3-element vectors expressed in B frame. Note that the sum pure diagonal matrices is a diagonal matrix, just as the sum of symmetric matrices is a symmetric matrix and the sum of antisymmetric matrices is antisymmetric.

Decomposing the left-hand side term in (C.27) for each single-axis i, through (C.29), as to obtain all $\mathbf{A}_{\mathsf{diag},i}$, $\mathbf{A}_{\mathsf{sym},i}$, and $\mathbf{A}_{\mathsf{skew},i}$ as matrix functions of s_i and $\boldsymbol{\vartheta}_i^B$ yields, through inversion of (C.30)–(C.32), the relationships between overall scale-factor $\mathbf{s}^B$, non-orthogonality $\boldsymbol{\eta}^B$ and misalignment $\boldsymbol{\vartheta}^B$ and the scale-factor s_i and misalignment $\boldsymbol{\vartheta}_i^B$ of each single-axis sensor. These are

$$\mathbf{s}^B = \sum_{i=1}^{n} \begin{bmatrix} \{\hat{\mathbf{W}}_i^B\}_{1,1} \\ \{\hat{\mathbf{W}}_i^B\}_{2,2} \\ \{\hat{\mathbf{W}}_i^B\}_{3,3} \end{bmatrix} \mathrm{s}_i + \sum_{i=1}^{n} \begin{bmatrix} 0 & -\{\hat{\mathbf{W}}_i^B\}_{1,3} & \{\hat{\mathbf{W}}_i^B\}_{1,2} \\ \{\hat{\mathbf{W}}_i^B\}_{2,3} & 0 & -\{\hat{\mathbf{W}}_i^B\}_{2,1} \\ -\{\hat{\mathbf{W}}_i^B\}_{3,2} & \{\hat{\mathbf{W}}_i^B\}_{3,1} & 0 \end{bmatrix} \boldsymbol{\vartheta}_i^B\,, \tag{C.33}$$

$$\begin{aligned} \boldsymbol{\eta}^B = {} & \tfrac{1}{2} \sum_{i=1}^{n} \begin{bmatrix} \{\hat{\mathbf{W}}_i^B\}_{2,3} + \{\hat{\mathbf{W}}_i^B\}_{3,2} \\ \{\hat{\mathbf{W}}_i^B\}_{1,3} + \{\hat{\mathbf{W}}_i^B\}_{3,1} \\ \{\hat{\mathbf{W}}_i^B\}_{1,2} + \{\hat{\mathbf{W}}_i^B\}_{2,1} \end{bmatrix} \mathrm{s}_i \\ & + \tfrac{1}{2} \sum_{i=1}^{n} \begin{bmatrix} \{\hat{\mathbf{W}}_i^B\}_{3,3} - \{\hat{\mathbf{W}}_i^B\}_{2,2} & \{\hat{\mathbf{W}}_i^B\}_{2,1} & -\{\hat{\mathbf{W}}_i^B\}_{3,1} \\ -\{\hat{\mathbf{W}}_i^B\}_{1,2} & \{\hat{\mathbf{W}}_i^B\}_{1,1} - \{\hat{\mathbf{W}}_i^B\}_{3,3} & \{\hat{\mathbf{W}}_i^B\}_{3,2} \\ \{\hat{\mathbf{W}}_i^B\}_{1,3} & -\{\hat{\mathbf{W}}_i^B\}_{2,3} & \{\hat{\mathbf{W}}_i^B\}_{2,2} - \{\hat{\mathbf{W}}_i^B\}_{1,1} \end{bmatrix} \boldsymbol{\vartheta}_i^B\,, \end{aligned} \tag{C.34}$$

$$
\begin{aligned}
\boldsymbol{\vartheta}^B = \tfrac{1}{2} \sum_{i=1}^{n} &\begin{bmatrix} \left\{\hat{\mathbf{W}}_i^B\right\}_{2,3} - \left\{\hat{\mathbf{W}}_i^B\right\}_{3,2} \\ \left\{\hat{\mathbf{W}}_i^B\right\}_{1,3} - \left\{\hat{\mathbf{W}}_i^B\right\}_{3,1} \\ \left\{\hat{\mathbf{W}}_i^B\right\}_{1,2} - \left\{\hat{\mathbf{W}}_i^B\right\}_{2,1} \end{bmatrix} \mathrm{s}_i \\
+ \tfrac{1}{2} \sum_{i=1}^{n} &\begin{bmatrix} -\left\{\hat{\mathbf{W}}_i^B\right\}_{3,3} - \left\{\hat{\mathbf{W}}_i^B\right\}_{2,2} & \left\{\hat{\mathbf{W}}_i^B\right\}_{2,1} & \left\{\hat{\mathbf{W}}_i^B\right\}_{3,1} \\ -\left\{\hat{\mathbf{W}}_i^B\right\}_{1,2} & \left\{\hat{\mathbf{W}}_i^B\right\}_{1,1} + \left\{\hat{\mathbf{W}}_i^B\right\}_{3,3} & -\left\{\hat{\mathbf{W}}_i^B\right\}_{3,2} \\ \left\{\hat{\mathbf{W}}_i^B\right\}_{1,3} & \left\{\hat{\mathbf{W}}_i^B\right\}_{2,3} & -\left\{\hat{\mathbf{W}}_i^B\right\}_{2,2} - \left\{\hat{\mathbf{W}}_i^B\right\}_{1,1} \end{bmatrix} \boldsymbol{\vartheta}_i^B \,,
\end{aligned}
\tag{C.35}
$$

where $\{\bullet\}_{r,c}$ is the matrix element in row r and column c, and $\hat{\mathbf{W}}_i^B$ is a constant matrix given by

$$
\hat{\mathbf{W}}_i^B = \hat{\mathbf{G}}^B \mathbf{e}_i^B {\mathbf{e}_i^B}^{\mathsf{T}} \,. \tag{C.36}
$$

Note that, for a device composed of a triad of single-axis sensors oriented along the elemental directions of B, the overall scale-factor $\mathbf{s}^B$, non-orthogonality $\boldsymbol{\eta}^B$ and misalignment $\boldsymbol{\vartheta}^B$ have much simpler mappings to the individual scale-factors s_i and misalignments $\boldsymbol{\vartheta}_i^B$, being

$$
\mathbf{s}^B = \begin{bmatrix} \mathrm{s}_1 \\ \mathrm{s}_2 \\ \mathrm{s}_3 \end{bmatrix} \,, \quad \boldsymbol{\eta}^B = \begin{bmatrix} -\left\{\boldsymbol{\vartheta}_2^B\right\}_1 + \left\{\boldsymbol{\vartheta}_3^B\right\}_1 \\ \left\{\boldsymbol{\vartheta}_1^B\right\}_2 - \left\{\boldsymbol{\vartheta}_3^B\right\}_2 \\ -\left\{\boldsymbol{\vartheta}_1^B\right\}_3 + \left\{\boldsymbol{\vartheta}_2^B\right\}_3 \end{bmatrix} \,, \quad \text{and} \quad \boldsymbol{\vartheta}^B = \begin{bmatrix} -\left\{\boldsymbol{\vartheta}_2^B\right\}_1 - \left\{\boldsymbol{\vartheta}_3^B\right\}_1 \\ \left\{\boldsymbol{\vartheta}_1^B\right\}_2 + \left\{\boldsymbol{\vartheta}_3^B\right\}_2 \\ -\left\{\boldsymbol{\vartheta}_1^B\right\}_3 - \left\{\boldsymbol{\vartheta}_2^B\right\}_3 \end{bmatrix} \,. \tag{C.37}
$$

The mappings (C.33)–(C.35) apply to both gyro and accelerometer models. Subscripts g and a were omitted for simplicity.

The covariance models of the overall scale-factor $\mathbf{s}^B$, non-orthogonality $\boldsymbol{\eta}^B$ and misalignment $\boldsymbol{\vartheta}^B$ can be easily obtained from (C.33)–(C.35). That of the gyro scale-factor is

$$
\begin{aligned}
\mathbf{P}_{\mathrm{s,g}} &= E\left\{\mathbf{s}_{\mathrm{g}}^B {\mathbf{s}_{\mathrm{g}}^B}^{\mathsf{T}}\right\} \\
&= \begin{bmatrix} \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \mathrm{s}_{\mathrm{g},1}} & \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \mathrm{s}_{\mathrm{g},2}} & \dots & \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \mathrm{s}_{\mathrm{g},n}} \end{bmatrix} E\left\{ \begin{bmatrix} \mathrm{s}_{\mathrm{g},1} \\ \mathrm{s}_{\mathrm{g},2} \\ \vdots \\ \mathrm{s}_{\mathrm{g},n} \end{bmatrix} \begin{bmatrix} \mathrm{s}_{\mathrm{g},1} & \mathrm{s}_{\mathrm{g},2} & \dots & \mathrm{s}_{\mathrm{g},n} \end{bmatrix} \right\} \begin{bmatrix} \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \mathrm{s}_{\mathrm{g},1}}^{\mathsf{T}} \\ \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \mathrm{s}_{\mathrm{g},2}}^{\mathsf{T}} \\ \vdots \\ \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \mathrm{s}_{\mathrm{g},n}}^{\mathsf{T}} \end{bmatrix} \\
&+ \begin{bmatrix} \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \boldsymbol{\vartheta}_{\mathrm{g},1}^B} & \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \boldsymbol{\vartheta}_{\mathrm{g},2}^B} & \dots & \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \boldsymbol{\vartheta}_{\mathrm{g},n}^B} \end{bmatrix} E\left\{ \begin{bmatrix} \boldsymbol{\vartheta}_{\mathrm{g},1}^B \\ \boldsymbol{\vartheta}_{\mathrm{g},2}^B \\ \vdots \\ \boldsymbol{\vartheta}_{\mathrm{g},n}^B \end{bmatrix} \begin{bmatrix} {\boldsymbol{\vartheta}_{\mathrm{g},1}^B}^{\mathsf{T}} & {\boldsymbol{\vartheta}_{\mathrm{g},2}^B}^{\mathsf{T}} & \dots & {\boldsymbol{\vartheta}_{\mathrm{g},n}^B}^{\mathsf{T}} \end{bmatrix} \right\} \begin{bmatrix} \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \boldsymbol{\vartheta}_{\mathrm{g},1}^B}^{\mathsf{T}} \\ \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \boldsymbol{\vartheta}_{\mathrm{g},2}^B}^{\mathsf{T}} \\ \vdots \\ \dfrac{\partial \mathbf{s}_{\mathrm{g}}^B}{\partial \boldsymbol{\vartheta}_{\mathrm{g},n}^B}^{\mathsf{T}} \end{bmatrix} \,,
\end{aligned}
\tag{C.38}
$$

which, if scale-factors and misalignments of different single-axis units are assumed to be independent, is reduced to

$$\mathbf{P}_{\mathrm{s,g}} = \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}} E\left\{\mathrm{s}^2_{\mathrm{g},i}\right\} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}}^{\mathsf{T}} + \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}} E\left\{\boldsymbol{\vartheta}^B_{\mathrm{g},i} \boldsymbol{\vartheta}^B_{\mathrm{g},i}{}^{\mathsf{T}}\right\} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}}^{\mathsf{T}} . \tag{C.39}$$

The covariances $\mathbf{P}_{\vartheta,\mathrm{g}}$ and $\mathbf{P}_{\eta,\mathrm{g}}$, of overall misalignment $\boldsymbol{\vartheta}^B_{\mathrm{g}}$ and non-orthogonality $\boldsymbol{\eta}^B_{\mathrm{g}}$, respectively, have similar forms in the case of mutual independence between axes

$$\mathbf{P}_{\eta,\mathrm{g}} = \sum_{i=1}^{n} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}} E\left\{\mathrm{s}^2_{\mathrm{g},i}\right\} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}}^{\mathsf{T}} + \sum_{i=1}^{n} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}} E\left\{\boldsymbol{\vartheta}^B_{\mathrm{g},i} \boldsymbol{\vartheta}^B_{\mathrm{g},i}{}^{\mathsf{T}}\right\} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}}^{\mathsf{T}} \tag{C.40}$$

$$\mathbf{P}_{\vartheta,\mathrm{g}} = \sum_{i=1}^{n} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}} E\left\{\mathrm{s}^2_{\mathrm{g},i}\right\} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}}^{\mathsf{T}} + \sum_{i=1}^{n} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}} E\left\{\boldsymbol{\vartheta}^B_{\mathrm{g},i} \boldsymbol{\vartheta}^B_{\mathrm{g},i}{}^{\mathsf{T}}\right\} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}}^{\mathsf{T}} . \tag{C.41}$$

In the case of correlation between axes, $\mathbf{P}_{\eta,\mathrm{g}}$ and $\mathbf{P}_{\vartheta,\mathrm{g}}$ assume forms similar to (C.38). The partial derivatives in (C.38)–(C.41) can be obtained from the mappings (C.33)–(C.35).

The cross-covariances between overall scale-factor, non-orthogonality and misalignment are

$$\mathbf{P}_{\mathrm{s}\eta,\mathrm{g}} = E\left\{\mathbf{s}^B_{\mathrm{g}} \boldsymbol{\eta}^B_{\mathrm{g}}{}^{\mathsf{T}}\right\} = \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}} E\left\{\mathrm{s}^2_{\mathrm{g},i}\right\} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},1}}^{\mathsf{T}} + \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}} E\left\{\boldsymbol{\vartheta}^B_{\mathrm{g},i} \boldsymbol{\vartheta}^B_{\mathrm{g},i}{}^{\mathsf{T}}\right\} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}}^{\mathsf{T}} , \tag{C.42}$$

$$\mathbf{P}_{\mathrm{s}\vartheta,\mathrm{g}} = E\left\{\mathbf{s}^B_{\mathrm{g}} \boldsymbol{\vartheta}^B_{\mathrm{g}}{}^{\mathsf{T}}\right\} = \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}} E\left\{\mathrm{s}^2_{\mathrm{g},i}\right\} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},1}}^{\mathsf{T}} + \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}} E\left\{\boldsymbol{\vartheta}^B_{\mathrm{g},i} \boldsymbol{\vartheta}^B_{\mathrm{g},i}{}^{\mathsf{T}}\right\} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}}^{\mathsf{T}} , \tag{C.43}$$

$$\mathbf{P}_{\eta\vartheta,\mathrm{g}} = E\left\{\boldsymbol{\eta}^B_{\mathrm{g}} \boldsymbol{\vartheta}^B_{\mathrm{g}}{}^{\mathsf{T}}\right\} = \sum_{i=1}^{n} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},i}} E\left\{\mathrm{s}^2_{\mathrm{g},i}\right\} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \mathrm{s}_{\mathrm{g},1}}^{\mathsf{T}} + \sum_{i=1}^{n} \frac{\partial \boldsymbol{\eta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}} E\left\{\boldsymbol{\vartheta}^B_{\mathrm{g},i} \boldsymbol{\vartheta}^B_{\mathrm{g},i}{}^{\mathsf{T}}\right\} \frac{\partial \boldsymbol{\vartheta}^B_{\mathrm{g}}}{\partial \boldsymbol{\vartheta}^B_{\mathrm{g},i}}^{\mathsf{T}} , \tag{C.44}$$

where independence between scale-factor and misalignment of different single-axis units was assumed. If this is not valid, then (C.42)–(C.44) need to be given in an array form as (C.38). This is here omitted for brevity.

The covariance matrices of the equivalent multiplicative uncertainties of the accelerometer model (C.60) ($\mathbf{s}^B_{\mathrm{a}}$, $\boldsymbol{\eta}^B_{\mathrm{a}}$ and $\boldsymbol{\vartheta}^B_{\mathrm{a}}$) have a similar form, and are thus omitted.

G-sensitivity

The gyro G-sensitivity of each single-axis sensor, modeled as proportional to the acceleration parallel to the sensing axis (as in [10]), can also be aggregated and projected onto the sensor axes (of B frame). The G-sensitivity term in (C.25) can be

$$\hat{\mathbf{G}}_{\mathrm{g}}^{B}\sum_{i=1}^{n}\hat{\mathbf{e}}_{\mathrm{g},i}^{B}\mathrm{m}_{\mathrm{g},i}\hat{\mathbf{e}}_{\mathrm{g},i}^{B}{}^{\top}\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}=\mathbf{M}_{\mathrm{g}}^{B}\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}\,, \tag{C.45}$$

where $\mathbf{M}_{\mathrm{g}}^{B}$ is the overall (3 × 3) G-sensitivity matrix. The mapping between the nine entries of $\mathbf{M}_{\mathrm{g}}^{B}$ and each of the n individual coefficients $\mathrm{m}_{\mathrm{g},i}$, used later on to obtain the covariance of the collapsed term $\mathbf{M}_{\mathrm{g}}^{B}$, is

$$\mathsf{vec}\left(\mathbf{M}_{\mathrm{g}}^{B}\right)=\left[\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},1}^{B}\right)\quad\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},2}^{B}\right)\quad\dots\quad\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},n}^{B}\right)\right]\begin{bmatrix}\mathrm{m}_{\mathrm{g},1}\\ \mathrm{m}_{\mathrm{g},2}\\ \vdots\\ \mathrm{m}_{\mathrm{g},n}\end{bmatrix}\,, \tag{C.46}$$

with $\mathsf{vec}\,(\bullet)$ being the operator that reshapes a 3×3 matrix into a 9×1 column vector, i.e.,

$$\mathsf{vec}\left(\mathbf{M}_{\mathrm{g}}\right)=\begin{bmatrix}\left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{1,1}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{1,2}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{1,3}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{2,1}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{2,2}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{2,3}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{3,1}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{3,2}\\ \left\{\mathbf{M}_{\mathrm{g}}^{B}\right\}_{3,3}\end{bmatrix}\,. \tag{C.47}$$

Note that the term $\mathbf{M}_{\mathrm{g}}^{B}\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}$ in (C.45) can be rearranged to have the entries of $\mathbf{M}_{\mathrm{g}}^{B}$ as a column array [as $\mathsf{vec}(\mathbf{M}_{\mathrm{g}}^{B})$ in (C.47)], allowing their inclusion in the estimator state vector. That is

$$\mathbf{M}_{\mathrm{g}}^{B}\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}=\begin{bmatrix}\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}{}^{\top} & \mathbf{0}^{\top} & \mathbf{0}^{\top}\\ \mathbf{0}^{\top} & \boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}{}^{\top} & \mathbf{0}^{\top}\\ \mathbf{0}^{\top} & \mathbf{0}^{\top} & \boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}{}^{\top}\end{bmatrix}\mathsf{vec}\left(\mathbf{M}_{\mathrm{g}}^{B}\right)\,, \tag{C.48}$$

where $\mathbf{0}$ is a 3×1 column vector of zeros.

Note also that if the inertial unit is composed of three gyro axes orthogonally arranged and aligned with the unit directions of B frame, then, as previously explained, $\hat{\mathbf{G}}^{B}$ and $\sum_{i=1}^{n}\hat{\mathbf{e}}_{i}^{B}(\hat{\mathbf{e}}_{i}^{B})^{\top}$ are both identity matrices making $\mathbf{M}_{\mathrm{g}}^{B}$ a pure diagonal matrix ($\mathbf{W}_{\mathrm{g},i}^{B}$ would also be an identity matrix) and reducing the overall gyro G-sensitivity to a 3-element array $\mathbf{m}_{\mathrm{g}}^{B}$ [instead of the 9-element $\mathsf{vec}\,(\mathbf{M}_{\mathrm{g}}^{B})$]. In such case, (C.48) reduces to

$$\mathbf{M}_{\mathrm{g}}^{B}\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}=\mathsf{diag}\left(\mathbf{m}_{\mathrm{g}}^{B}\right)\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}=\mathsf{diag}\left(\boldsymbol{\Delta}\mathbf{v}_{\mathrm{sf}}^{B}\right)\mathbf{m}_{\mathrm{g}}^{B}\,. \tag{C.49}$$

In fact, the orthogonal sensor triad of such special case need not be aligned with the axes of B frame for the G-sensitivity coefficients to be reducible to a set of 3, being a simple axes rotation required to diagonalize $\mathbf{M}_\mathrm{g}$ and achieve this reduction. This is a straightforward variation of (C.49) and thus not further derived.

The covariance matrix of the overall gyro G-sensitivity (in array form) can be derived from (C.45) yielding

$$\begin{aligned}\mathbf{P}_{\mathsf{m,g}} &= E\left\{\mathsf{vec}\left(\mathbf{M}_\mathrm{g}^B\right)\mathsf{vec}\left(\mathbf{M}_\mathrm{g}^B\right)^\mathsf{T}\right\}\\ &= \begin{bmatrix}\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},1}^B\right)^\mathsf{T}\\ \mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},2}^B\right)^\mathsf{T}\\ \vdots\\ \mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},n}^B\right)^\mathsf{T}\end{bmatrix}^\mathsf{T} E\left\{\begin{bmatrix}\mathrm{m}_{\mathrm{g},1}\\ \mathrm{m}_{\mathrm{g},2}\\ \vdots\\ \mathrm{m}_{\mathrm{g},n}\end{bmatrix}\begin{bmatrix}\mathrm{m}_{\mathrm{g},1} & \mathrm{m}_{\mathrm{g},2} & \dots & \mathrm{m}_{\mathrm{g},n}\end{bmatrix}\right\}\begin{bmatrix}\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},1}^B\right)^\mathsf{T}\\ \mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},2}^B\right)^\mathsf{T}\\ \vdots\\ \mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},n}^B\right)^\mathsf{T}\end{bmatrix},\end{aligned} \tag{C.50}$$

which, if the different $\mathrm{m}_{\mathrm{g},i}$ coefficients are assumed to be independent, reduces to

$$\mathbf{P}_{\mathsf{m,g}} = \sum_{i=1}^{n}\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},i}^B\right) E\left\{\mathrm{m}_{\mathrm{g},i}^2\right\}\mathsf{vec}\left(\hat{\mathbf{W}}_{\mathrm{g},i}^B\right)^\mathsf{T} . \tag{C.51}$$

with $\hat{\mathbf{W}}_{\mathrm{g},i}^B$ given by (C.36).

If the reduced G-sensitivity coefficient set $\mathbf{m}_\mathrm{g}^B$ is used [i.e., if (C.49) is valid], $\mathbf{P}_{\mathsf{m,g}}$ becomes

$$\mathbf{P}_{\mathsf{m,g}} = E\left\{\mathbf{m}_\mathrm{g}^B\mathbf{m}_\mathrm{g}^{B\,\mathsf{T}}\right\} = E\left\{\begin{bmatrix}\mathrm{m}_{\mathrm{g},1}\\ \mathrm{m}_{\mathrm{g},2}\\ \mathrm{m}_{\mathrm{g},3}\end{bmatrix}\begin{bmatrix}\mathrm{m}_{\mathrm{g},1} & \mathrm{m}_{\mathrm{g},2} & \mathrm{m}_{\mathrm{g},3}\end{bmatrix}\right\} . \tag{C.52}$$

Bias and noise

The n-element bias and noise vectors in (C.26)-(C.25) can also be collapsed to 3-element sets as

$$\mathbf{b}_\mathrm{g}^B = \hat{\mathbf{G}}_\mathrm{g}^B\hat{\mathbf{U}}_\mathrm{g}^B\begin{bmatrix}\mathrm{b}_{\mathrm{g},1}\\ \mathrm{b}_{\mathrm{g},2}\\ \vdots\\ \mathrm{b}_{\mathrm{g},n}\end{bmatrix}, \qquad \mathbf{b}_\mathrm{a}^B = \hat{\mathbf{G}}_\mathrm{a}^B\hat{\mathbf{U}}_\mathrm{a}^B\begin{bmatrix}\mathrm{b}_{\mathrm{a},1}\\ \mathrm{b}_{\mathrm{a},2}\\ \vdots\\ \mathrm{b}_{\mathrm{a},n}\end{bmatrix}, \tag{C.53}$$

and

$$\boldsymbol{\nu}_\mathrm{g}^B = \hat{\mathbf{G}}_\mathrm{g}^B\hat{\mathbf{U}}_\mathrm{g}^B\begin{bmatrix}\nu_{\mathrm{g},1}\\ \nu_{\mathrm{g},2}\\ \vdots\\ \nu_{\mathrm{g},n}\end{bmatrix}, \qquad \boldsymbol{\nu}_\mathrm{a}^B = \hat{\mathbf{G}}_\mathrm{a}^B\hat{\mathbf{U}}_\mathrm{a}^B\begin{bmatrix}\nu_{\mathrm{a},1}\\ \nu_{\mathrm{a},2}\\ \vdots\\ \nu_{\mathrm{a},n}\end{bmatrix}, \tag{C.54}$$

where $\hat{\mathbf{U}}_\mathrm{g}^B$ and $\hat{\mathbf{U}}_\mathrm{a}^B$ are as in (C.6).

The covariance matrix of the overall bias $\mathbf{b}_\mathrm{g}^B$ of the gyro model is

$$\mathbf{P}_{\mathrm{b,g}} = E\left\{\mathbf{b}_\mathrm{g}^B\mathbf{b}_\mathrm{g}^{B\mathsf{T}}\right\} = \hat{\mathbf{G}}_\mathrm{g}^B\hat{\mathbf{U}}_\mathrm{g}^B E\left\{\begin{bmatrix}\mathrm{b}_{\mathrm{g},1}\\ \mathrm{b}_{\mathrm{g},2}\\ \vdots\\ \mathrm{b}_{\mathrm{g},n}\end{bmatrix}\begin{bmatrix}\mathrm{b}_{\mathrm{g},1} & \mathrm{b}_{\mathrm{g},2} & \dots & \mathrm{b}_{\mathrm{g},n}\end{bmatrix}\right\}\left(\hat{\mathbf{G}}_\mathrm{g}^B\hat{\mathbf{U}}_\mathrm{g}^B\right)^\mathsf{T}, \tag{C.55}$$

which, if bias errors of different axes are assumed to be independent, reduces to

$$\mathbf{P}_{\mathrm{b,g}} = \hat{\mathbf{G}}_\mathrm{g}^B\left(\sum_{i=1}^{n}\hat{\mathbf{e}}_{\mathrm{g},i}^B E\left\{\mathrm{b}_{\mathrm{g},i}^2\right\}\hat{\mathbf{e}}_{\mathrm{g},i}^{B\;\mathsf{T}}\right)\hat{\mathbf{G}}_\mathrm{g}^{B\;\mathsf{T}}. \tag{C.56}$$

The covariance matrix of the noise $\boldsymbol{\nu}_\mathrm{g}^B$ has a similar form

$$\mathbf{P}_{\nu,\mathrm{g}} = E\left\{\boldsymbol{\nu}_\mathrm{g}^B\boldsymbol{\nu}_\mathrm{g}^{B\mathsf{T}}\right\} = \hat{\mathbf{G}}_\mathrm{g}^B\hat{\mathbf{U}}_\mathrm{g}^B E\left\{\begin{bmatrix}\nu_{\mathrm{g},1}\\ \nu_{\mathrm{g},2}\\ \vdots\\ \nu_{\mathrm{g},n}\end{bmatrix}\begin{bmatrix}\nu_{\mathrm{g},1} & \nu_{\mathrm{g},2} & \dots & \nu_{\mathrm{g},n}\end{bmatrix}\right\}\left(\hat{\mathbf{G}}_\mathrm{g}^B\hat{\mathbf{U}}_\mathrm{g}^B\right)^\mathsf{T}, \tag{C.57}$$

which, if axes noises are independent, gives

$$\mathbf{P}_{\nu,\mathrm{g}} = \hat{\mathbf{G}}_\mathrm{g}^B\left(\sum_{i=1}^{n}\hat{\mathbf{e}}_{\mathrm{g},i}^B E\left\{\nu_{\mathrm{g},i}^2\right\}\hat{\mathbf{e}}_{\mathrm{g},i}^{B\;\mathsf{T}}\right)\hat{\mathbf{G}}_\mathrm{g}^{B\;\mathsf{T}}. \tag{C.58}$$

The bias and noise covariance matrices of the overall accelerometer model have the same form of (C.55)–(C.56) and (C.57)–(C.58), respectively.

Overall collapsed model

The aboved proposed collapsing of the n-axis unit error sources turns the model (C.25)–(C.26) into a more compact form, similar to that used in [187], given as

$$\Delta\tilde{\boldsymbol{\theta}}^B = \left(\mathbf{I} + \mathrm{diag}\left(\mathbf{s}_\mathrm{g}^B\right) + \left[\boldsymbol{\vartheta}_\mathrm{g}^B\times\right] + \mathbf{S}\left(\boldsymbol{\eta}_\mathrm{g}^B\right)\right)\Delta\boldsymbol{\theta}^B + \mathbf{M}_\mathrm{g}^B\Delta\mathbf{v}_\mathrm{sf}^B + \Delta t\mathbf{b}_\mathrm{g}^B + \boldsymbol{\nu}_\mathrm{g}^B, \tag{C.59}$$

$$\Delta\tilde{\mathbf{v}}_\mathrm{sf}^B = \left(\mathbf{I} + \mathrm{diag}\left(\mathbf{s}_\mathrm{a}^B\right) + \left[\boldsymbol{\vartheta}_\mathrm{a}^B\times\right] + \mathbf{S}\left(\boldsymbol{\eta}_\mathrm{a}^B\right)\right)\left(\Delta\mathbf{v}_\mathrm{sf}^B + \Delta\mathbf{v}_\mathrm{cent}^B + \Delta\mathbf{v}_\mathrm{euler}^B\right) + \Delta t\mathbf{b}_\mathrm{a}^B + \boldsymbol{\nu}_\mathrm{a}^B, \tag{C.60}$$

with overall centrifugal and Euler acceleration terms

$$\Delta\mathbf{v}_\mathrm{cent}^B \approx \hat{\mathbf{G}}_\mathrm{a}^B\sum_{i=1}^{n}\hat{\mathbf{e}}_{\mathrm{a},i}^B\hat{\mathbf{e}}_{\mathrm{a},i}^{B\;\mathsf{T}}\Delta\mathbf{v}_{\mathrm{cent},i}^B \tag{C.61}$$

$$\Delta\mathbf{v}_\mathrm{euler}^B \approx \hat{\mathbf{G}}_\mathrm{a}^B\sum_{i=1}^{n}\hat{\mathbf{e}}_{\mathrm{a},i}^B\hat{\mathbf{e}}_{\mathrm{a},i}^{B\;\mathsf{T}}\Delta\mathbf{v}_{\mathrm{euler},i}^B. \tag{C.62}$$

The potential estimator states in model (C.59)–(C.60) are shown in Table C.1. The dynamics models for these states are derived in the following section.

C.3.3 Uncertainty state dynamics models

The previous point collapsed the n-axes IMU model (C.21)–(C.22) to the three unit directions of B frame [giving (C.59)–(C.60)]. This process created a set of fixed-sized potential states (see Table C.1). The dynamics of these states can be obtained from their mappings to the states of the original single axis models (C.21)–(C.22).

Table C.1: IMU overall model states

State name	State symbol	Size
Gyro scale-factor	$\mathbf{s}_{\mathsf{g}}^{B}$	3×1
Acc. scale-factor	$\mathbf{s}_{\mathsf{a}}^{B}$	3×1
Gyro non-orthogonality	$\boldsymbol{\eta}_{\mathsf{g}}^{B}$	3×1
Acc. non-orthogonality	$\boldsymbol{\eta}_{\mathsf{a}}^{B}$	3×1
Gyro misalignment	$\boldsymbol{\vartheta}_{\mathsf{g}}^{B}$	3×1
Acc. misalignment	$\boldsymbol{\vartheta}_{\mathsf{g}}^{B}$	3×1
Gyro G-sensitivity	$\mathsf{vec}\left(\mathbf{M}_{\mathsf{g}}^{B}\right)$ or $\mathbf{m}_{\mathsf{g}}^{B}$	9×1 or 3×1
Gyro bias	$\mathbf{b}_{\mathsf{g}}^{B}$	3×1
Acc. bias	$\mathbf{b}_{\mathsf{a}}^{B}$	3×1

Scale-factor, non-orthogonality and misalignment

The scale-factor $\mathbf{s}^{B}$, non-orthogonality $\boldsymbol{\eta}^{B}$, and misalignment $\boldsymbol{\vartheta}^{B}$ in (C.59)–(C.60) arise from the individual scale-factors s_i and misalignments $\boldsymbol{\vartheta}_i^{B}$ of each single-axis sensor. Modeling the individual misalignments as random constants, as done in the simulation model in Section C.2.1, as

$$\boldsymbol{\vartheta}_{i,j+1}^{B} = \boldsymbol{\vartheta}_{i,j}^{B} \ , \tag{C.63}$$

with $\boldsymbol{\vartheta}_{i,0}^{B} \sim \mathcal{N}(\mathbf{0},\, \mathbf{P}_{\theta_i,0})$, and the individual scale-factors as a random-walk processes [as in (C.10)]

$$\mathsf{s}_{i,j+1} = \mathsf{s}_{i,j} + \sqrt{\Delta t_j}\,\sigma_{\mathsf{RW},i}\mathsf{w}_{\mathsf{RW},i,j} \ , \tag{C.64}$$

with $\mathsf{w}_{\mathsf{RW},i,j} \sim \mathcal{N}(0,\, 1)$ and $\mathsf{s}_{i,0} \sim \mathcal{N}(0,\, \sigma_{\mathsf{s}_i,0})$, and working these models into (C.33), (C.34) and (C.35) gives the dynamics of $\mathbf{s}^{B}$, $\boldsymbol{\eta}^{B}$, and $\boldsymbol{\vartheta}^{B}$ as

$$\mathbf{s}_{j+1}^{B} = \mathbf{s}_{j}^{B} + \sqrt{\Delta t_j}\,\mathbf{L}_{\mathsf{s},\mathsf{w}}\mathbf{w}_{\mathsf{RW},j} \tag{C.65}$$

$$\boldsymbol{\eta}_{j+1}^{B} = \boldsymbol{\eta}_{j}^{B} + \sqrt{\Delta t_j}\,\mathbf{L}_{\eta,\mathsf{w}}\mathbf{w}_{\mathsf{RW},j} \tag{C.66}$$

$$\boldsymbol{\vartheta}_{j+1}^{B} = \boldsymbol{\vartheta}_{j}^{B} + \sqrt{\Delta t_j}\,\mathbf{L}_{\vartheta,\mathsf{w}}\mathbf{w}_{\mathsf{RW},j} \ , \tag{C.67}$$

where $\mathbf{w}_{\mathsf{RW},j} \sim \mathcal{N}(\mathbf{0},\, \mathbf{I})$ is a 3×1 noise sample and $\mathbf{L}_{\mathrm{s,w}}$, $\mathbf{L}_{\eta,\mathrm{w}}$, and $\mathbf{L}_{\vartheta,\mathrm{w}}$ are constant 3×3 lower triangular matrices obtained through Cholesky decomposition of

$$\mathbf{L}_{\mathrm{s,w}}\mathbf{L}_{\mathrm{s,w}}^{\mathsf{T}} = \sum_{i=1}^{n} \frac{\partial \mathbf{s}^B}{\partial \mathrm{s}_i} \sigma^2_{\mathsf{RW},i} \frac{\partial \mathbf{s}^B}{\partial \mathrm{s}_i}^{\mathsf{T}} \tag{C.68}$$

$$\mathbf{L}_{\eta,\mathrm{w}}\mathbf{L}_{\eta,\mathrm{w}}^{\mathsf{T}} = \sum_{i=1}^{n} \frac{\partial \boldsymbol{\eta}^B}{\partial \mathrm{s}_i} \sigma^2_{\mathsf{RW},i} \frac{\partial \boldsymbol{\eta}^B}{\partial \mathrm{s}_i}^{\mathsf{T}} \tag{C.69}$$

$$\mathbf{L}_{\vartheta,\mathrm{w}}\mathbf{L}_{\vartheta,\mathrm{w}}^{\mathsf{T}} = \sum_{i=1}^{n} \frac{\partial \boldsymbol{\vartheta}^B}{\partial \mathrm{s}_i} \sigma^2_{\mathsf{RW},i} \frac{\partial \boldsymbol{\vartheta}^B}{\partial \mathrm{s}_i}^{\mathsf{T}} \ . \tag{C.70}$$

The partial derivatives in (C.68)–(C.70) can be obtained from (C.33)–(C.35). Note that the same noise sample $\mathbf{w}_j$ affects $\mathbf{s}^B$, $\boldsymbol{\eta}^B$ and $\boldsymbol{\vartheta}^B$. This leads to (perfect) correlations between the process noises of these states as

$$E\left\{\begin{bmatrix}\mathbf{L}_{\mathrm{s,w}}\\ \mathbf{L}_{\eta,\mathrm{w}}\\ \mathbf{L}_{\vartheta,\mathrm{w}}\end{bmatrix} \mathbf{w}_j \mathbf{w}_j^{\mathsf{T}} \begin{bmatrix}\mathbf{L}_{\mathrm{s,w}}^{\mathsf{T}} & \mathbf{L}_{\eta,\mathrm{w}}^{\mathsf{T}} & \mathbf{L}_{\vartheta,\mathrm{w}}^{\mathsf{T}}\end{bmatrix}\right\} = \begin{bmatrix}\mathbf{L}_{\mathrm{s,w}}\mathbf{L}_{\mathrm{s,w}}^{\mathsf{T}} & \mathbf{L}_{\mathrm{s,w}}\mathbf{L}_{\eta,\mathrm{w}}^{\mathsf{T}} & \mathbf{L}_{\mathrm{s,w}}\mathbf{L}_{\vartheta,\mathrm{w}}^{\mathsf{T}}\\ \mathbf{L}_{\eta,\mathrm{w}}\mathbf{L}_{\mathrm{s,w}}^{\mathsf{T}} & \mathbf{L}_{\eta,\mathrm{w}}\mathbf{L}_{\eta,\mathrm{w}}^{\mathsf{T}} & \mathbf{L}_{\eta,\mathrm{w}}\mathbf{L}_{\vartheta,\mathrm{w}}^{\mathsf{T}}\\ \mathbf{L}_{\vartheta,\mathrm{w}}\mathbf{L}_{\mathrm{s,w}}^{\mathsf{T}} & \mathbf{L}_{\vartheta,\mathrm{w}}\mathbf{L}_{\eta,\mathrm{w}}^{\mathsf{T}} & \mathbf{L}_{\vartheta,\mathrm{w}}\mathbf{L}_{\vartheta,\mathrm{w}}^{\mathsf{T}}\end{bmatrix} . \tag{C.71}$$

The initial conditions of $\mathbf{s}^B$, $\boldsymbol{\eta}^B$ and $\boldsymbol{\vartheta}^B$ are also cross-correlated as all three depend on the initial distributions of all s_i and $\boldsymbol{\vartheta}_i^B$. The initial covariance of $\mathbf{s}^B$, $\boldsymbol{\eta}^B$ and $\boldsymbol{\vartheta}^B$ is

$$E\left\{\begin{bmatrix}\mathbf{s}_0^B\\ \boldsymbol{\eta}_0^B\\ \boldsymbol{\vartheta}_0^B\end{bmatrix}\begin{bmatrix}\mathbf{s}_0^{B\,\mathsf{T}} & \boldsymbol{\eta}_0^{B\,\mathsf{T}} & \boldsymbol{\vartheta}_0^{B\,\mathsf{T}}\end{bmatrix}\right\} = \begin{bmatrix}\mathbf{P}_{\mathrm{s},0} & \mathbf{P}_{\mathrm{s}\eta,0} & \mathbf{P}_{\mathrm{s}\vartheta,0}\\ \mathbf{P}_{\mathrm{s}\eta,0}^{\mathsf{T}} & \mathbf{P}_{\eta,0} & \mathbf{P}_{\eta\vartheta,0}\\ \mathbf{P}_{\mathrm{s}\vartheta,0}^{\mathsf{T}} & \mathbf{P}_{\eta\vartheta,0}^{\mathsf{T}} & \mathbf{P}_{\vartheta,0}\end{bmatrix} , \tag{C.72}$$

where $\mathbf{P}_{\mathrm{s},0}$, $\mathbf{P}_{\mathrm{s},0}$, and $\mathbf{P}_{\mathrm{s},0}$ are obtained with (C.38)–(C.41) and the cross-covariances $\mathbf{P}_{\mathrm{s}\eta,0}$, $\mathbf{P}_{\mathrm{s}\vartheta,0}$, and $\mathbf{P}_{\eta\vartheta,0}$ with (C.42)–(C.44). The initial covariances of s_i and $\boldsymbol{\vartheta}_i^B$ are used in these calculations.

Note that the derivation and modeling presented in this point apply to both gyro and accelerometer. Subscripts a and g have thus been omitted for simplicity.

G-sensitivity

As in the simulation model of Section C.2.1, the G-sensitivity of each single-axis sensor can be modeled as a random constant as

$$\mathrm{m}_{\mathrm{g},i,j+1} = \mathrm{m}_{\mathrm{g},i,j} \ , \tag{C.73}$$

which, as $\hat{\mathbf{W}}_{\mathrm{g},i}^B$ is constant, leads to constant states $\mathsf{vec}(\mathbf{M}_{\mathrm{g}}^B)$ (or $\mathbf{m}_{\mathrm{g}}^B$), i.e.,

$$\mathsf{vec}\left(\mathbf{M}_{\mathrm{g},j+1}^B\right) = \mathsf{vec}\left(\mathbf{M}_{\mathrm{g},j}^B\right) \quad \text{or} \quad \mathbf{m}_{\mathrm{g},j+1}^B = \mathbf{m}_{\mathrm{g},j}^B \ , \tag{C.74}$$

with initial conditions distributed as $\mathsf{vec}(\mathbf{M}_{\mathrm{g},0}^B) \sim \mathcal{N}(\mathbf{0},\, \mathbf{P}_{\mathrm{m,g},0})$ or $\mathbf{m}_{\mathrm{g},0}^B \sim \mathcal{N}(\mathbf{0},\, \mathbf{P}_{\mathrm{m,g},0})$ with $\mathbf{P}_{\mathrm{m,g},0}$ given by (C.51) or (C.52) depending if $\mathsf{vec}(\mathbf{M}_{\mathrm{g}}^B)$ or $\mathbf{m}_{\mathrm{g}}^B$ are used, respectively.

Bias

The overall gyro or accelerometer bias can be modeled [as the single-axis units in (C.13)] to include a random walk process (with turn-on error) and exponentially correlated process, i.e.,

$$\mathbf{b}^B = \mathbf{b}^B_{\mathsf{RW}} + \mathbf{b}^B_{\mathsf{EC}} \,. \tag{C.75}$$

Note that the gyro G-sensitivity term in (C.13) has been treated separately in this estimation model under derivation (see previous point).

The random walk component of the bias of each single-axis sensor can be modeled [as in (C.14)] as

$$\mathsf{b}_{\mathsf{RW},i,j} = \mathsf{b}_{\mathsf{RW},i,j-1} + \sqrt{\Delta t_j}\, \sigma_{\mathsf{K},i}\, \mathsf{w}_{\mathsf{RW},i,j} \,, \tag{C.76}$$

which, plugged into (C.53), leads to the dynamics of the random walk component of the overall bias

$$\mathbf{b}^B_{\mathsf{RW},j+1} = \mathbf{b}^B_{\mathsf{RW},j} + \sqrt{\Delta t_j}\mathbf{L}_{\mathsf{b,RW,w}}\mathbf{w}_{\mathsf{RW},j} \,, \tag{C.77}$$

where the 3 × 1 driving noise is $\mathbf{w}_{\mathsf{RW},j} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and $\mathbf{L}_{\mathsf{b,RW,w}}$ is a constant 3 × 3 lower triangular matrix obtained through the Cholesky decomposition of

$$\mathbf{L}_{\mathsf{b,RW,w}}\mathbf{L}^{\mathsf{T}}_{\mathsf{b,RW,w}} = \hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} \sigma^2_{\mathsf{K},1} & & & \\ & \sigma^2_{\mathsf{K},2} & & \\ & & \ddots & \\ & & & \sigma^2_{\mathsf{K},n} \end{bmatrix} \left(\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B\right)^{\mathsf{T}} \,. \tag{C.78}$$

The diagonal covariance matrix of single units in (C.78) assumes independence of random walk processes of different unit sensors. If this is not the case, off-diagonal entries of such matrix shall be non-null.

The initial condition of the process (C.77) is $\mathbf{b}^B_{\mathsf{RW},0} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{\mathsf{b,RW},0})$ with covariance $\mathbf{P}_{\mathsf{b,RW},0}$ computed using (C.55) or (C.56), depending, respectively, if the initial turn-on bias of different single-axis units is assumed correlated or independent.

The exponentially correlated term of the single-axis sensor has been given in (C.15) as

$$\mathsf{b}_{\mathsf{EC},i,j} = e^{-\beta_{\mathsf{EC},i}\Delta t_j}\mathsf{b}_{\mathsf{EC},i,j-1} + \sigma_{\mathsf{EC},i,\infty}\sqrt{1 - e^{-2\beta_{\mathsf{EC},i}\Delta t_j}}\mathsf{w}_{\mathsf{EC},i,j} \,. \tag{C.79}$$

Combining (C.79) with the overall bias mapping (C.53) yields

$$\mathbf{b}_{\mathsf{EC},j+1} = \hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} e^{-\beta_{\mathsf{EC},1}\Delta t_j} & & & \\ & e^{-\beta_{\mathsf{EC},2}\Delta t_j} & & \\ & & \ddots & \\ & & & e^{-\beta_{\mathsf{EC},n}\Delta t_j} \end{bmatrix} \begin{bmatrix} \mathsf{b}_{\mathsf{EC},1,j} \\ \mathsf{b}_{\mathsf{EC},2,j} \\ \vdots \\ \mathsf{b}_{\mathsf{EC},n,j} \end{bmatrix} + \sqrt{\Delta t_j}\mathbf{L}_{\mathsf{b,EC,w}}\mathbf{w}_{\mathsf{EC},j} \,, \tag{C.80}$$

with $\mathbf{w}_{\mathsf{EC},j} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and, again, the constant 3 × 3 lower triangular matrix $\mathbf{L}_{\mathsf{b,EC,w}}$ is obtained by Cholesky decomposition of

$$\begin{aligned}
&\mathbf{L}_{\mathsf{b,EC,w}}\mathbf{L}_{\mathsf{b,EC,w}}^{\mathsf{T}} \\
&\quad = \frac{1}{\Delta t_j}\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} \sigma^2_{\mathsf{EC},1,\infty}\left(1-e^{-2\beta_{\mathsf{EC},1}\Delta t_j}\right) & & \\ & \ddots & \\ & & \sigma^2_{\mathsf{EC},n,\infty}\left(1-e^{-2\beta_{\mathsf{EC},n}\Delta t_j}\right)\end{bmatrix}\left(\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B\right)^{\mathsf{T}} \\
&\quad \approx \hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} 2\beta_{\mathsf{EC},1}\sigma^2_{\mathsf{EC},1,\infty} & & \\ & \ddots & \\ & & 2\beta_{\mathsf{EC},n}\sigma^2_{\mathsf{EC},n,\infty}\end{bmatrix}\left(\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B\right)^{\mathsf{T}} . \qquad \text{(C.81)}
\end{aligned}$$

where the last step uses the approximation $e^{-2\beta_{\mathsf{EC}}\Delta t} \approx 1 - 2\beta_{\mathsf{EC}}\Delta t$ (see, e.g., [137]), which is valid for $\beta_{\mathsf{EC}}\Delta t \ll 1$.

It is directly evident that the dynamics of the exponentially correlated component of the overall bias given in (C.77) is not written in the expected recursive form: $x_{j+1} = ax_j + b$. This is because the first term on the right-hand side cannot (in general) be written as an affine term of $\mathbf{b}^B_{\mathsf{EC},j}$ [which has the form of (C.53)]. One may try to impose this affinity and obtain, for instance,

$$\begin{aligned}
&\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} e^{-\beta_{\mathsf{EC},1}\Delta t_j} & & \\ & \ddots & \\ & & e^{-\beta_{\mathsf{EC},n}\Delta t_j}\end{bmatrix}\begin{bmatrix} \mathsf{b}_{\mathsf{EC},1,j} \\ \mathsf{b}_{\mathsf{EC},2,j} \\ \vdots \\ \mathsf{b}_{\mathsf{EC},n,j}\end{bmatrix} \\
&\quad = \hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} e^{-\beta_{\mathsf{EC},1}\Delta t_j} & & \\ & \ddots & \\ & & e^{-\beta_{\mathsf{EC},n}\Delta t_j}\end{bmatrix}\left((\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B)^{\mathsf{T}}\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B\right)^{-1}\left(\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B\right)^{\mathsf{T}}\mathbf{b}^B_{\mathsf{EC},j} , \qquad \text{(C.82)}
\end{aligned}$$

simply to realize that, for $n > 3$, the matrix $(\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B)^{\mathsf{T}}\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B$ is singular; naturally, since $\hat{\mathbf{U}}^B$, as given by (C.6), is $3 \times n$.

One possible way out is to approximate the inverse of the time-constant of all exponentially correlated processes (i.e., $\beta_{\mathsf{EC},i}$) by a single value. This effectively turns the diagonal matrix of exponentials on the right-hand side of (C.80) into a scalar multiplying an identity matrix (i.e., $e^{-\beta_{\mathsf{EC}}\Delta t_j}\mathbf{I}$). Of course, this approximation is valid (and works best) if the time-constants of the exponentially correlated processes of the different sensor units are close. This is generally the case as IMU devices are composed of (non-defective) sensor units of the same model and specification with only small variation of error characteristics.

If the IMU includes single-axis units of different model/type/specification, an alternative solution is to perform frequency analysis of the projections of all processes $\mathsf{b}_{\mathsf{EC},i}$ onto $\mathbf{b}^B_{\mathsf{EC}}$ [through the mapping (C.53)] and approximate the dynamics of each component of $\mathbf{b}^B_{\mathsf{EC}}$ as a decoupled first-order system (e.g., taking only the dominant pole in the spectrum of each component). This approach is deemed unnecessary in most realistic scenarios and is thus not pursued further.

Following the former suggested approximation, the dynamics of exponentially correlated overall bias (C.80) becomes

$$\mathbf{b}_{\mathsf{EC},j+1} = e^{-\beta_{\mathsf{EC}}\Delta t_j}\mathbf{b}_{\mathsf{EC},j+1} + \sqrt{\Delta t_j}\mathbf{L}_{\mathsf{b,EC,w}}\mathbf{w}_{\mathsf{EC},j} \;, \tag{C.83}$$

with initial condition $\mathbf{b}_{\mathsf{EC},0} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{\mathsf{b,EC},0})$ and initial covariance $\mathbf{P}_{\mathsf{b,EC},0}$ given by (C.55) or (C.56), depending, respectively, if the initial condition of the exponentially correlated bias of different single-axis units is assumed correlated or independent.

Note that, as mentioned in the beginning of this point, the bias modeling here presented applies to both gyro and accelerometer. Therefore, for simplicity the subscripts a and g have been omitted.

Noise

Finally, the overall noise can be modeled as the result of contributions from single-axis white noise components. These have been given in (C.17) as

$$\nu_{\mathsf{WN},i,j} = \sqrt{\Delta t_j}\sigma_{\mathsf{N},i}\mathsf{w}_{\mathsf{WN},i,j} \;, \tag{C.84}$$

which, plugged into (C.54), gives

$$\boldsymbol{\nu}_j^B = \sqrt{\Delta t_j}\mathbf{L}_{\nu,\mathsf{w}}\boldsymbol{w}_{\nu,j} \;, \tag{C.85}$$

where $\mathbf{w}_{\nu,j} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and the constant 3×3 lower triangular matrix $\mathbf{L}_{\nu,\mathsf{w}}$ yields from the Cholesky decomposition of

$$\mathbf{L}_{\nu,\mathsf{w}}\mathbf{L}_{\nu,\mathsf{w}}^{\mathsf{T}} = \hat{\mathbf{G}}^B\hat{\mathbf{U}}^B \begin{bmatrix} \sigma^2_{\mathsf{N},1} & & & \\ & \sigma^2_{\mathsf{N},2} & & \\ & & \ddots & \\ & & & \sigma^2_{\mathsf{N},n} \end{bmatrix} \left(\hat{\mathbf{G}}^B\hat{\mathbf{U}}^B\right)^{\mathsf{T}} . \tag{C.86}$$

As with the bias random walk driving noise, the diagonal covariance matrix in (C.86) makes the assumption of independent unit sensor noises $\mathsf{w}_{\mathsf{WN},i,j}$. Non-null off-diagonal entries appear in such matrix if that is not the case.

The covariance matrix $\mathbf{P}_\nu$ of the noise $\boldsymbol{\nu}^B$ is as given by (C.57) or (C.57), depending, respectively, if the single-axis white noises are assumed correlated or independent.

Again, the modeling in this point applies to both gyro and accelerometer. Subscripts g and a have thus been skipped.

C.4 Allan variance analysis and modeling

Developed as a time-domain analysis method for the study the frequency stability of oscillators [1, 2, 3], the Allan variance has been extensively used for the stochastic characterization and modeling of inertial sensors [96, 168, 94, 77, 151]. In fact, it has become a standard method in the characterization and testing of FOG [81] and RLG [82] gyroscopes, and of nongyroscopic accelerometers [80]. Also known as *2-sample variance*, the Allan variance is a particular case of the more general M-sample

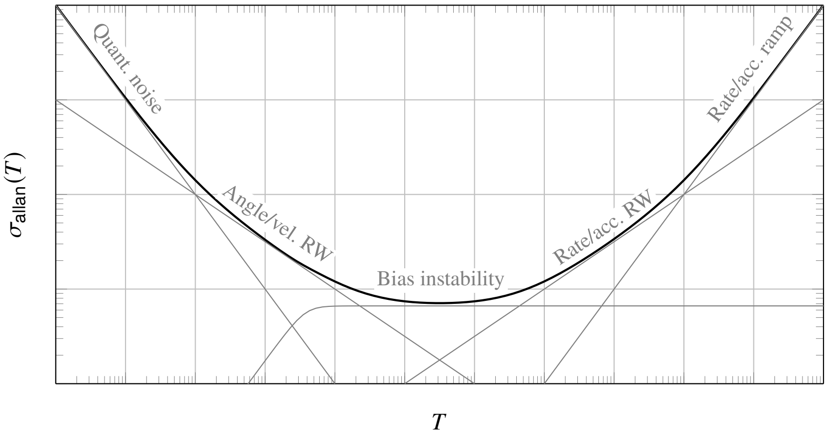


Figure C.5: U-shape Allan std. dev. profile of a sensor as the sum of the typical error sources

variance [2]. For a cluster time $T = n\Delta t$, with sensor sampling time Δt, the Allan variance is [1]

$$\sigma^2_{\mathsf{allan}}(T) = \frac{1}{2T^2}\left\langle\left(\theta_{m+2n} - 2\theta_{m+n} + \theta_m\right)^2\right\rangle \ , \tag{C.87}$$

where $\langle\bullet\rangle$ is the ensemble average. It can be estimated, for instance, as in [3] as

$$\sigma^2_{\mathsf{allan}}(T) = \frac{1}{2T^2\left(N - 2n + 1\right)} \sum_{l=1}^{N-2n+1} \left(\theta_{m+2n} - 2\theta_{m+n} + \theta_m\right)^2 \ , \tag{C.88}$$

where θ is the accumulated sensor measurements, in this case gyro, i.e., $\theta_m = \sum_{l=1}^{m} \Delta\theta_l$. In the study of an accelerometer, angle θ is replaced by velocity, v, and thus $\Delta\theta$ by Δv.

The log-log Allan variance (or standard deviation) plot of a sensor (also known as *Green chart* [96] or *sigma plot* [94]) often describes a U-shape curve as depicted in Figure C.5 (also shown in [81]), with each region of T mostly affected by one dominant noise type.

Figure C.6 shows the Allan standard deviation profiles [$\sigma_{\mathsf{allan}}(T)$] of different noise source types. (See [81] for the exact expressions of these curves.) The discrete process expressions for the angular/velocity RW, rate/acceleration RW and exponentially correlated (1st-order GM) were previously given in (C.17), (C.14), and (C.15), respectively. $\mathsf{A_{SIN}}$ and $\mathsf{f_{SIN}}$ are, respectively, the amplitude and frequency of the sinusoidal noise.

Although not shown, the regular profile in Figure C.5 can also be corrupted by exponentially correlated and sinusoidal sources, especially in the middle to low frequency range (i.e., mid to high cluster times). Analysis and characterization is done by computing and plotting the Allan variance of a (very) long sequence of static sensor data and to it fitting the slopes and shapes of the different error sources (e.g., those of Figure C.6) in order to obtain the parameters of their processes. These may then be used to produce both simulation and filter models of the sensor in question.

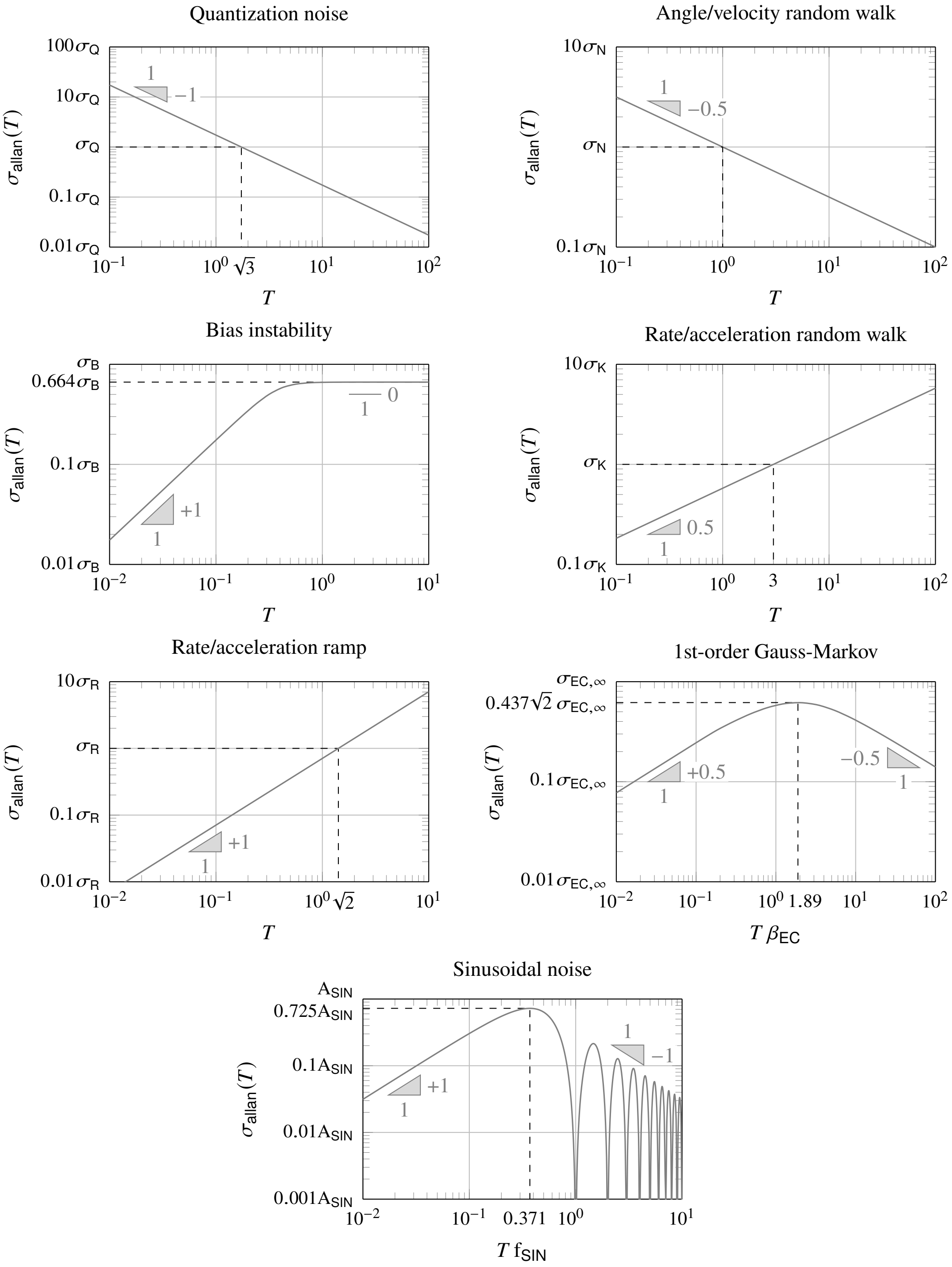


Figure C.6: Allan std. dev. of different noise sources with curve slopes given in decade per decade (adapted from [81])

D

Additional GNSS Filter Models

This appendix lays out the filter update models for GNSS measurement types that, despite described and analyzed in Chapter 2, were not included in the navigation system designed in Chapters 3, 4 and 5. These are GNSS receiver solution [position (POS) and velocity (VEL)], and pseudorange-rate (PRR).

D.1 GNSS receiver solution update

GNSS receiver position and velocity outputs can also be used to update the navigation filter. As described in Section 2.3.1, this renders the architecture loosely coupled.

The measurement models for receiver position and velocity are

$$\begin{aligned} \mathbf{y}_{\mathrm{r}_{\mathsf{gnss}},k} = \tilde{\mathbf{r}}^{E}_{\mathsf{gnss},k} &= \boldsymbol{h}_{\mathrm{r},k}(\mathbf{x}_k) + \boldsymbol{\nu}_{\mathrm{r},k} \\ &= \mathbf{r}^{E}_{k} + \mathbf{C}^{E_k}_{B_k}\mathbf{l}^{B}_{\mathsf{ant}} + \mathbf{b}^{E}_{\mathrm{r},\mathsf{gnss},k} + \boldsymbol{\nu}_{\mathrm{r},k} \end{aligned} \tag{D.1}$$

$$\begin{aligned} \mathbf{y}_{\mathrm{v}_{\mathsf{gnss}},k} = \tilde{\mathbf{v}}^{E}_{\mathsf{gnss},k} &= \boldsymbol{h}_{\mathrm{v},k}(\mathbf{x}_k) + \boldsymbol{\nu}_{\mathrm{v},k} \\ &= \mathbf{v}^{E}_{k} + \left(\mathbf{C}^{E_k}_{B_k}\boldsymbol{\Omega}^{B}_{IB,k} - \boldsymbol{\Omega}^{E}_{IE}\mathbf{C}^{E_k}_{B_k}\right)\mathbf{l}^{B}_{\mathsf{ant}} + \mathbf{b}^{E}_{\mathrm{v},\mathsf{gnss},k} + \boldsymbol{\nu}_{\mathrm{v},k}\ , \end{aligned} \tag{D.2}$$

where $\mathbf{l}_{\mathsf{ant},k}$ is the arm between the center of the IMU and the GNSS antenna phase center, in B coordinates, $\mathbf{b}^{E}_{\mathrm{r},\mathsf{gnss},k}$ and $\mathbf{b}^{E}_{\mathrm{v},\mathsf{gnss},k}$ are, respectively, the GNSS position and velocity bias states, $\boldsymbol{\nu}_{\mathrm{r},k}$ and $\boldsymbol{\nu}_{\mathrm{v},k}$ are Gaussian noise vectors (of position and velocity, respectively) with covariance matrices $\mathbf{R}_{\mathrm{r},k}$ and $\mathbf{R}_{\mathrm{v},k}$.

The error-measurement model is then for the position

$$\begin{aligned} \delta\mathbf{y}_{\mathrm{r}_{\mathsf{gnss}},k} &= \boldsymbol{h}_{\mathrm{r},k}(\hat{\mathbf{x}}_k + \delta\mathbf{x}_k) - \boldsymbol{h}_{\mathrm{r},k}(\hat{\mathbf{x}}_k) + \boldsymbol{\nu}_{\mathrm{r},k} \\ &\approx \hat{\mathbf{r}}^{E}_{k} + \delta\mathbf{r}^{E}_{k} - \hat{\mathbf{r}}^{E}_{k} + \hat{\mathbf{C}}^{E_k}_{B_k}\left(\mathbf{I} - \left[\delta\boldsymbol{\theta}^{B}_{k}\times\right]\right)\mathbf{l}^{B}_{\mathsf{ant}} - \hat{\mathbf{C}}^{E_k}_{B_k}\mathbf{l}^{B}_{\mathsf{ant}} + \hat{\mathbf{b}}^{E}_{\mathsf{gnss},\mathrm{r},k} + \delta\mathbf{b}^{E}_{\mathsf{gnss},\mathrm{r},k} \\ &\quad - \hat{\mathbf{b}}^{E}_{\mathsf{gnss},\mathrm{r},k} + \boldsymbol{\nu}_{\mathrm{r},k} \\ &\approx \delta\mathbf{r}^{E}_{k} + \hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\delta\boldsymbol{\theta}^{B}_{k} + \delta\mathbf{b}^{E}_{\mathsf{gnss},\mathrm{r},k} + \boldsymbol{\nu}_{\mathrm{r},k}\ , \end{aligned} \tag{D.3}$$

and the velocity measurement

$$\begin{aligned}
\boldsymbol{\delta}\mathbf{y}_{\mathrm{v}_{\mathsf{gnss}},k} &= \boldsymbol{h}_{\mathrm{v},k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - \boldsymbol{h}_{\mathrm{v},k}(\hat{\mathbf{x}}_k) + \boldsymbol{\nu}_{\mathrm{v},k} \\
&\approx \hat{\mathbf{v}}_k^E + \boldsymbol{\delta}\mathbf{v}_k^E - \hat{\mathbf{v}}_k^E + \hat{\mathbf{C}}_{B_k}^{E_k}\left(\mathbf{I} - \left[\boldsymbol{\delta\theta}_k^B\times\right]\right)\left(\hat{\boldsymbol{\Omega}}_{IB,k}^B + \boldsymbol{\delta\Omega}_{IB,k}^B\right)\mathbf{l}_{\mathsf{ant}}^B \\
&\quad - \boldsymbol{\Omega}_{IE}^E\hat{\mathbf{C}}_{B_k}^{E_k}\left(\mathbf{I} - \left[\boldsymbol{\delta\theta}_k^B\times\right]\right)\mathbf{l}_{\mathsf{ant}}^B - \left(\hat{\mathbf{C}}_{B_k}^{E_k}\hat{\boldsymbol{\Omega}}_{IB,k}^B - \boldsymbol{\Omega}_{IE}^E\hat{\mathbf{C}}_{B_k}^{E_k}\right)\mathbf{l}_{\mathsf{ant}}^B + \hat{\mathbf{b}}_{\mathsf{gnss,r},k}^E \\
&\quad + \boldsymbol{\delta}\mathbf{b}_{\mathsf{gnss,r},k}^E - \hat{\mathbf{b}}_{\mathsf{gnss,r},k}^E + \boldsymbol{\nu}_{\mathrm{v},k} \\
&\approx \boldsymbol{\delta}\mathbf{v}_k^E + \left(\frac{1}{\Delta t}\hat{\mathbf{C}}_{B_k}^{E_k}\left[\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}_k^B\times\mathbf{l}_{\mathsf{ant}}^B\times\right] - \boldsymbol{\Omega}_{IE}^E\hat{\mathbf{C}}_{B_k}^{E_k}\left[\mathbf{l}_{\mathsf{ant}}^B\times\right]\right)\boldsymbol{\delta\theta}_k^B - \frac{1}{\Delta t}\hat{\mathbf{C}}_{B_k}^{E_k}\left[\mathbf{l}_{\mathsf{ant}}^B\times\right]\boldsymbol{\delta\Delta\theta}_k^B \\
&\quad + \boldsymbol{\delta}\mathbf{b}_{\mathsf{gnss,r},k}^E + \boldsymbol{\nu}_{\mathrm{v},k}\ , \qquad \text{(D.4)}
\end{aligned}$$

where second order error terms have been neglected.

Finally, the non-null Jacobian submatrices are for the position measurement model

$$\mathbf{H}_{\mathrm{r}_{\mathsf{gnss}},k}^{\mathrm{r}} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{r},k}}{\partial\boldsymbol{\delta}\mathbf{r}_k^E} = \mathbf{I} \qquad \text{(D.5)}$$

$$\mathbf{H}_{\mathrm{r}_{\mathsf{gnss}},k}^{\theta} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{r},k}}{\partial\boldsymbol{\delta\theta}_k^B} \approx \hat{\mathbf{C}}_{B_k}^{E_k}\left[\mathbf{l}_{\mathsf{ant}}^B\times\right] \qquad \text{(D.6)}$$

$$\mathbf{H}_{\mathrm{r}_{\mathsf{gnss}},k}^{\mathrm{b_r}} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{r},k}}{\partial\boldsymbol{\delta}\mathbf{b}_{\mathsf{gnss,r},k}^E} = \mathbf{I}\ , \qquad \text{(D.7)}$$

and for the velocity

$$\mathbf{H}_{\mathrm{v}_{\mathsf{gnss}},k}^{\mathrm{v}} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{v},k}}{\partial\boldsymbol{\delta}\mathbf{v}_k^E} = \mathbf{I} \qquad \text{(D.8)}$$

$$\mathbf{H}_{\mathrm{v}_{\mathsf{gnss}},k}^{\theta} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{v},k}}{\partial\boldsymbol{\delta\theta}_k^B} \approx \frac{1}{\Delta t}\hat{\mathbf{C}}_{B_k}^{E_k}\left[\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}_k^B\times\mathbf{l}_{\mathsf{ant}}^B\times\right] - \boldsymbol{\Omega}_{IE}^E\hat{\mathbf{C}}_{B_k}^{E_k}\left[\mathbf{l}_{\mathsf{ant}}^B\times\right] \qquad \text{(D.9)}$$

$$\mathbf{H}_{\mathrm{v}_{\mathsf{gnss}},k}^{\mathrm{x}_{\mathsf{imu}}} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{v},k}}{\partial\boldsymbol{\delta}\mathbf{x}_{\mathsf{imu},k}} \approx -\frac{1}{\Delta t}\hat{\mathbf{C}}_{B_k}^{E_k}\left[\mathbf{l}_{\mathsf{ant}}^B\times\right]\boldsymbol{\Phi}_{\Delta\theta,k}^{\mathsf{imu}} \qquad \text{(D.10)}$$

$$\mathbf{H}_{\mathrm{v}_{\mathsf{gnss}},k}^{\mathrm{b_r}} = \frac{\partial\boldsymbol{\delta}\mathbf{y}_{\mathrm{v},k}}{\partial\boldsymbol{\delta}\mathbf{b}_{\mathrm{v},\mathsf{gnss},k}^E} = \mathbf{I}\ . \qquad \text{(D.11)}$$

D.2 Pseudorange-rate update

As described in Section 2.5.1, GNSS pseudorange-rate measurements can be used to support velocity estimation in the navigation filter. This measurement is tested against other GNSS raw outputs in a covariance analysis in Section 2.5.3. The update model used for this analysis is here presented in a similar level of detail to the remaining updates used in this work. Note, however, that the pseudorange-rate measurement type is not actually used by the navigation system designed in Chapters 3 and 4.

As the pseudoranges in Section 3.9.5, the set of psudorange-rate measurements can be modelled in a similar form as

$$\mathbf{y}_{\dot{\rho},k} = \boldsymbol{h}_{\dot{\rho},k}(\mathbf{x}_k) + \boldsymbol{\nu}_{\dot{\rho},k}\ , \qquad \text{(D.12)}$$

where $\boldsymbol{\nu}_{\dot{\rho},k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_{\dot{\rho},k}\right)$ is the pseudorange-rate channel noise, and the vector function of measurements $\boldsymbol{h}_{\dot{\rho}}(\mathbf{x}_k)$ is for channel i

$$h_{\dot{\rho},i,k}(\mathbf{x}_k) = \dot{\rho}_{i,k}\left(\mathbf{x}_k\right) + c\left(\dot{\tau}_{\mathrm{r},k} + \dot{\tau}_{\mathrm{T},i,k}\left(\mathbf{x}_k\right) + \dot{\tau}_{\mathrm{I},i,k}\left(\mathbf{x}_k\right)\right) + \mathrm{b}_{\dot{\rho},i,k}\ . \tag{D.13}$$

$\dot{\tau}_{\mathrm{r},k}$ is the receiver clock drift, $\dot{\tau}_{\mathrm{T},k}$ and $\dot{\tau}_{\mathrm{I},k}$ are the tropospheric and ionospheric delay drift corrections, and $\mathrm{b}_{\dot{\rho},i,k}$ is the pseudorange-rate residual bias. The receiver clock drift $\dot{\tau}_{\mathrm{r},k}$ is modeled as a random walk state while the pseudorange-rate residual bias $\mathrm{b}_{\dot{\rho},i,k}$ is modeled as a constant state. Note that, similarly to the pseudorange measurement model in Section 3.9.5, the atmospheric delay effects are corrected using derivative versions of the models described in Sections 3.9.3 and 3.9.4 (initially introduced in Section 2.5.2), i.e.,

$$\dot{\tau}_{\mathrm{T},i,k}\left(\mathbf{x}_k\right) = \frac{\mathrm{d}\,(1+\mathrm{s}_{\mathrm{T},i})h_{\tau_{\mathrm{T}},i,k}(\mathbf{x}_k)}{\mathrm{d}\,t} \tag{D.14}$$

$$\dot{\tau}_{\mathrm{I},i,k}\left(\mathbf{x}_k\right) = \frac{\mathrm{d}\,h_{\tau_{\mathrm{I}},i,k}(\mathbf{x}_k)}{\mathrm{d}\,t}\ , \tag{D.15}$$

where the functions $h_{\tau_{\mathrm{T}},i,k}$ and $h_{\tau_{\mathrm{I}},i,k}$ are as in (2.30) and (2.31), respectively, and $\mathrm{s}_{\mathrm{T},i}$ is a scale-factor introduced (in Section 3.9.3) for robustness against model residuals.

The range-rate $\dot{\rho}_{i,k}$ can be derived by differentiation of the range expression (2.15) in Inertial coordinates, as

$$\begin{aligned}\dot{\rho}_{i,k}\left(\mathbf{x}_k\right) = \frac{\mathrm{d}\,\rho_{i,k}\left(\mathbf{x}_k\right)}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left\|\mathbf{r}^I_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) - \mathbf{r}^I_{\mathsf{ant},k}\right\| \\ &= {\mathbf{e}^I_{\rho,i,k}}^{\mathsf{T}}\left(\mathbf{v}^I_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right)\frac{\mathrm{d}\,t_{\mathsf{s},i,k}}{\mathrm{d}t} - \mathbf{v}^I_{\mathsf{ant},k}\right)\ ,\end{aligned} \tag{D.16}$$

where, as in Section (2.16), $t_{\mathsf{s},i,k}$ is the time of signal transmission and t_k is the time of signal reception (here implied by the index k on the terms $\mathbf{r}^I_{\mathsf{ant},k}$, $\mathbf{v}^I_{\mathsf{ant},k}$ and $\mathbf{e}^I_{\rho,i,k}$); $\mathbf{r}^I_{\mathsf{s},i}$ and $\mathbf{v}^I_{\mathsf{s},i}$ are the inertial position and velocity of the satellite tracked by channel i, respectively, while $\mathbf{r}^I_{\mathsf{ant},k}$ and $\mathbf{v}^I_{\mathsf{ant},k}$ are the receiver antenna inertial position and velocity, respectively; $\mathbf{e}^I_{\rho,i,k}$ is the receiver-to-satellite unit direction vector in Inertial coordinates.

Note that, as stated in (2.16),

$$t_{\mathsf{s},i,k} = t_k - \Delta t_{i,k} \qquad \text{and} \qquad \Delta t_{i,k} = \frac{\rho_{i,k}}{c}\ , \tag{D.17}$$

and therefore

$$\frac{\mathrm{d}\,t_{\mathsf{s},i,k}}{\mathrm{d}t} = 1 - \frac{\dot{\rho}_{i,k}}{c}\ , \tag{D.18}$$

which, replaced in (D.16) and rearranged, leads to

$$\dot{\rho}_{i,k} = \frac{{\mathbf{e}^I_{\rho,i,k}}^{\mathsf{T}}\left(\mathbf{v}^I_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right) - \mathbf{v}^I_{\mathsf{ant},k}\right)}{1 + \frac{1}{c}\,{\mathbf{e}^I_{\rho,i,k}}^{\mathsf{T}}\mathbf{v}^I_{\mathsf{s},i}\left(t_{\mathsf{s},i,k}\right)}\ . \tag{D.19}$$

Writing all kinematics quantities in ECEF coordinates yields

$$\dot{\rho}_{i,k} = \frac{{\mathbf{e}^{E}_{\rho,i,k}}^{\top}\left(\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\left(\mathbf{v}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k}) + \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k})\right) - \mathbf{v}^{E}_{\mathsf{ant},k} - \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathsf{ant},k}\right)}{1 + \frac{1}{c}\,{\mathbf{e}^{E}_{\rho,i,k}}^{\top}\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\left(\mathbf{v}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k}) + \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k})\right)}\,, \tag{D.20}$$

which, realizing that ${\mathbf{e}^{E}_{\rho}}^{\top}\boldsymbol{\Omega}^{E}_{IE}(\mathbf{r}^{E}_{i} - \mathbf{r}^{E}_{\mathsf{ant}}) = \mathbf{0}$, gives

$$\dot{\rho}_{i,k} = \left(1 + \beta_{i,k}\right)\mathbf{e}^{E}_{\rho,i,k}{}^{\top}\left(\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\mathbf{v}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k}) - \mathbf{v}^{E}_{\mathsf{ant},k}\right)\,, \tag{D.21}$$

where the receiver antenna velocity $\mathbf{v}^{E}_{\mathsf{ant},k}$ depends on the kinematics whole-states as

$$\mathbf{v}^{E}_{\mathsf{ant},k} = \mathbf{v}^{E}_{k} + \left(\mathbf{C}^{E_k}_{B_k}\boldsymbol{\Omega}^{B}_{IB,k} - \boldsymbol{\Omega}^{E}_{IE}\mathbf{C}^{E_k}_{B_k}\right)\mathbf{l}^{B}_{\mathsf{ant}}\,, \tag{D.22}$$

and $\beta_{i,k}$ is as in (3.130), i.e.,

$$\beta_{i,k} = -\frac{\frac{1}{c}\,\mathbf{e}^{E}_{\rho,i,k}{}^{\top}\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\left(\mathbf{v}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k}) + \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k})\right)}{1 + \frac{1}{c}\,\mathbf{e}^{E}_{\rho,i,k}{}^{\top}\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\left(\mathbf{v}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k}) + \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k})\right)}\,. \tag{D.23}$$

As discussed in Section 3.9.5, GNSS satellites travel with inertial velocities about five orders of magnitude lower than the speed of light in vacuum c [as in (3.131)], yielding $\beta_{i,k} \sim 10^{-5}$ and thus $1 + \beta_{i,k} \approx 1$. For this reason the term $\beta_{i,k}$ can be neglected in the application at hand, being thus

$$\dot{\rho}_{i,k} = \mathbf{e}^{E}_{\rho,i,k}{}^{\top}\left(\mathbf{C}^{E_k}_{E(t_{\mathsf{s},i,k})}\mathbf{v}^{E}_{\mathsf{s},i}(t_{\mathsf{s},i,k}) - \mathbf{v}^{E}_{\mathsf{ant},k}\right)\,. \tag{D.24}$$

The pseudorange-rate error-measurement model is defined similarly to that of the pseudorange [in (3.121)] as

$$\boldsymbol{\delta}\mathbf{y}_{\dot{\rho},k} \;=\; \boldsymbol{h}_{\dot{\rho},k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - \boldsymbol{h}_{\dot{\rho},k}(\hat{\mathbf{x}}_k) + \boldsymbol{\nu}_{\dot{\rho},k}\,, \tag{D.25}$$

where for channel i

$$h_{\dot{\rho},i,k}(\hat{\mathbf{x}}_k + \boldsymbol{\delta}\mathbf{x}_k) - h_{\dot{\rho},i,k}(\hat{\mathbf{x}}_k) = \delta\dot{\rho}_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) + c\left(\delta\dot{\tau}_{\mathsf{r},k} + \delta\dot{\tau}_{\mathrm{T},i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right)\right) + \delta\mathrm{b}_{\dot{\rho},i,k}\,, \tag{D.26}$$

where the error introduced by the ionospheric correction is assumed to be negligible (as discussed in Section 3.9.4).

The error-range-rate $\delta\dot{\rho}_{i,k}$ is given by

$$\begin{aligned}\delta\dot{\rho}_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) &= \dot{\rho}_{i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) - \hat{\dot{\rho}}_{i,k}\\ &\approx \left(\hat{\mathbf{e}}^{E}_{\rho,i,k} + \boldsymbol{\delta}\mathbf{e}^{E}_{\rho,i,k}\right)^{\top}\left(\mathbf{C}^{E_k}_{E(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k})}\mathbf{v}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k} - \delta\Delta t_{i,k}\right) - \hat{\mathbf{v}}^{E}_{\mathsf{ant},k} - \boldsymbol{\delta}\mathbf{v}^{E}_{\mathsf{ant},k}\right) - \hat{\dot{\rho}}_{i,k}\,.\end{aligned} \tag{D.27}$$

The satellite velocity term in (D.27) can be obtained through a similar derivation to that done in (3.126) for its position. It goes

$$\begin{aligned}
\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k})}\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k}\right) &= \mathbf{C}^{E_k}_{I}\mathbf{v}^{I}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k}\right) - \mathbf{C}^{E_k}_{I}\boldsymbol{\Omega}^{I}_{IE}\mathbf{r}^{I}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k}\right) \\
&\approx \mathbf{C}^{E_k}_{I}\mathbf{v}^{I}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) - \mathbf{C}^{E_k}_{I}\boldsymbol{\Omega}^{I}_{IE}\left(\mathbf{r}^{I}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) - \delta\Delta t_{i,k}\mathbf{v}^{I}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right)\right) \\
&\approx \mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) + \delta\Delta t_{i,k}\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\boldsymbol{\Omega}^{E}_{IE}\left(\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) + \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right)\right) \ .
\end{aligned} \tag{D.28}$$

Note that, as in the satellite position perturbation derivation in (3.125), the satellite velocity was assumed constant within the time scale of the time error $\delta\Delta t_{i,k}$. Replacing (3.127) and then (3.132) in (D.28), and further using inequality (3.131), gives

$$\begin{aligned}
\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k})}\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k}\right) &= \mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) \\
&\quad -\frac{1}{c}\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\boldsymbol{\Omega}^{E}_{IE}\left(\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) + \boldsymbol{\Omega}^{E}_{IE}\mathbf{r}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right)\right)\hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k} \\
&\approx \mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) \ .
\end{aligned} \tag{D.29}$$

The error introduced by the estimated range unit vector can be derived as

$$\begin{aligned}
\boldsymbol{\delta}\mathbf{e}^{E}_{\rho,i,k} &= \mathbf{e}^{E}_{\rho,i,k} - \hat{\mathbf{e}}^{E}_{\rho,i,k} \\
&= \frac{\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k})}\mathbf{r}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k}\right) - \hat{\mathbf{r}}^{E}_{\mathrm{ant},k} - \boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}}{\left\|\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k})}\mathbf{r}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}-\delta\Delta t_{i,k}\right) - \hat{\mathbf{r}}^{E}_{\mathrm{ant},k} - \boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}\right\|} - \hat{\mathbf{e}}^{E}_{\rho,i,k} \\
&\approx \frac{\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\mathbf{r}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) - \hat{\mathbf{r}}^{E}_{\mathrm{ant},k} - \boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}}{\hat{\rho}_{i,k} + \hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}} - \hat{\mathbf{e}}^{E}_{\rho,i,k} \qquad \text{(D.30)} \\
&\approx \frac{\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\mathbf{r}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) - \hat{\mathbf{r}}^{E}_{\mathrm{ant},k} - \boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}}{\hat{\rho}_{i,k}}\left(1 + \frac{\hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}}{\hat{\rho}_{i,k}}\right) - \hat{\mathbf{e}}^{E}_{\rho,i,k} \\
&\approx \hat{\mathbf{e}}^{E}_{\rho,i,k} - \frac{\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}}{\hat{\rho}_{i,k}} + \hat{\mathbf{e}}^{E}_{\rho,i,k}\frac{\hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k}}{\hat{\rho}_{i,k}} - \hat{\mathbf{e}}^{E}_{\rho,i,k} \\
&\approx \frac{1}{\hat{\rho}_{i,k}}\left[\hat{\mathbf{e}}^{E}_{\rho,i,k}\times\right]^{2}\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k} \ , \qquad \text{(D.31)}
\end{aligned}$$

where (D.29) is used in step (D.30).

Plugging (D.29) and (D.31) into (D.27) yields

$$\delta\dot{\rho}_{i,k} \approx \frac{1}{\hat{\rho}_{i,k}}\left(\mathbf{C}^{E_k}_{E(\hat{t}_{\mathrm{s},i,k})}\mathbf{v}^{E}_{\mathrm{s},i}\left(\hat{t}_{\mathrm{s},i,k}\right) - \hat{\mathbf{v}}^{E}_{\mathrm{ant},k}\right)^{\mathsf{T}}\left[\hat{\mathbf{e}}^{E}_{\rho,i,k}\times\right]^{2}\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k} - \hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\boldsymbol{\delta}\mathbf{v}^{E}_{\mathrm{ant},k} \ , \tag{D.32}$$

where the antenna position error is mapped to the error-states as

$$\boldsymbol{\delta}\mathbf{r}^{E}_{\mathrm{ant},k} \approx \boldsymbol{\delta}\mathbf{r}^{E}_{k} + \hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathrm{ant}}\times\right]\boldsymbol{\delta\theta}^{B}_{k} \ , \tag{D.33}$$

and the velocity error [as derived in (D.4)] is approximately

$$\begin{aligned}\boldsymbol{\delta}\mathbf{v}^{E}_{\mathsf{ant},k} &\approx \boldsymbol{\delta}\mathbf{v}^{E}_{k}\\ &+\left(\frac{1}{\Delta t_k}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^{B}_{k}\times\mathbf{l}^{B}_{\mathsf{ant}}\times\right]-\boldsymbol{\Omega}^{E}_{IE}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\right)\boldsymbol{\delta\theta}^{B}_{k}-\frac{1}{\Delta t_k}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\boldsymbol{\delta\Delta\theta}^{B}_{k}\,.\end{aligned} \tag{D.34}$$

Considering, as proposed in Section 3.9.3, that the real tropospheric delay differs from the correction model by a constant scale factor $\mathrm{s_T}$, then the tropospheric error-delay-rate $\delta\dot{\tau}_{\mathrm{T},i,k}$ in (D.26) can be given as

$$\delta\dot{\tau}_{\mathrm{T},i,k}\left(\boldsymbol{\delta}\mathbf{x}_k\right) \approx \mathrm{s}_{\mathrm{T},i}\dot{\tau}_{\mathrm{T},i,k}\left(\hat{\mathbf{x}}_k\right)+\mathbf{h}_{\dot{\tau}_{\mathrm{T}},i,k}\boldsymbol{\delta}\mathbf{x}_k\ . \tag{D.35}$$

where $\mathbf{h}_{\dot{\tau}_{\mathrm{T}},i,k}$ is the Jacobian row vector of the time derivative of the tropospheric delay model function $\tau_{\mathrm{T},i,k}$ with respect to the error-state vector.

The pseudorange-rate measurement innovation covariance [as defined in (3.30)] is

$$E\left\{\mathbf{z}_{\dot{\rho},k}\mathbf{z}^{\mathsf{T}}_{\dot{\rho},k}\right\}=\mathbf{H}_{\dot{\rho},k}\mathbf{P}_k\mathbf{H}^{\mathsf{T}}_{\dot{\rho},k}+c^2\sigma^2_{\mathrm{s_T}}\boldsymbol{h}_{\dot{\tau}_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)\boldsymbol{h}_{\dot{\tau}_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)^{\mathsf{T}}+\mathbf{R}_{\dot{\rho},k}\ , \tag{D.36}$$

where $\boldsymbol{h}_{\dot{\tau}_{\mathrm{T}},k}(\hat{\mathbf{x}}_k)$ is the array of estimated tropospheric delay-rates. The Jacobian $\mathbf{H}_{\dot{\rho},k}$ is shaped as

$$\mathbf{H}_{\dot{\rho},k}=\left[\mathbf{h}^{\mathsf{T}}_{\dot{\rho},1,k}\quad\cdots\quad\mathbf{h}^{\mathsf{T}}_{\dot{\rho},i,k}\quad\cdots\quad\mathbf{h}^{\mathsf{T}}_{\dot{\rho},n,k}\right]^{\mathsf{T}}\ , \tag{D.37}$$

with non-null partial derivatives

$$\mathbf{h}^{\mathsf{v}}_{\dot{\rho},i,k}=-\hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}+c\,\mathbf{h}^{\mathsf{v}}_{\dot{\tau}_{\mathrm{T}},i,k} \tag{D.38}$$

$$\mathbf{h}^{\mathsf{r}}_{\dot{\rho},i,k}=\frac{1}{\hat{\rho}_{i,k}}\left(\mathbf{C}^{E_k}_{E(\hat{t}_{\mathsf{s},i,k})}\mathbf{v}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right)-\hat{\mathbf{v}}^{E}_{\mathsf{ant},k}\right)^{\mathsf{T}}\left[\hat{\mathbf{e}}^{E}_{\rho,i,k}\times\right]^2+c\,\mathbf{h}^{\mathsf{r}}_{\dot{\tau}_{\mathrm{T}},i,k} \tag{D.39}$$

$$\begin{aligned}\mathbf{h}^{\theta}_{\dot{\rho},i,k}&=\frac{1}{\hat{\rho}_{i,k}}\left(\mathbf{C}^{E_k}_{E(\hat{t}_{\mathsf{s},i,k})}\mathbf{v}^{E}_{\mathsf{s},i}\left(\hat{t}_{\mathsf{s},i,k}\right)-\hat{\mathbf{v}}^{E}_{\mathsf{ant},k}\right)^{\mathsf{T}}\left[\hat{\mathbf{e}}^{E}_{\rho,i,k}\times\right]^2\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\\&\qquad-\hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\left(\frac{1}{\Delta t_k}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\boldsymbol{\Delta}\hat{\boldsymbol{\theta}}^{B}_{k}\times\mathbf{l}^{B}_{\mathsf{ant}}\times\right]-\boldsymbol{\Omega}^{E}_{IE}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\right)+c\,\mathbf{h}^{\theta}_{\dot{\tau}_{\mathrm{T}},i,k}\end{aligned} \tag{D.40}$$

$$\mathbf{h}^{\mathsf{x_{imu}}}_{\dot{\rho},i,k}=\frac{1}{\Delta t_k}\hat{\mathbf{e}}^{E}_{\rho,i,k}{}^{\mathsf{T}}\hat{\mathbf{C}}^{E_k}_{B_k}\left[\mathbf{l}^{B}_{\mathsf{ant}}\times\right]\boldsymbol{\Phi}^{\mathsf{imu}}_{\Delta\theta,k}+c\,\mathbf{h}^{\mathsf{x_{imu}}}_{\dot{\tau}_{\mathrm{T}},i,k} \tag{D.41}$$

$$\mathrm{h}^{\dot{\tau}_{\mathrm{r}}}_{\dot{\rho},i,k}=c \tag{D.42}$$

$$\mathrm{h}^{\mathrm{b}_{\dot{\rho},i}}_{\dot{\rho},i,k}=1\ , \tag{D.43}$$

where $\mathbf{h}^{\mathsf{v}}_{\dot{\tau}_{\mathrm{T}},i,k}$, $\mathbf{h}^{\mathsf{v}}_{\dot{\tau}_{\mathrm{T}},i,k}$, $\mathbf{h}^{\theta}_{\dot{\tau}_{\mathrm{T}},i,k}$ and $\mathbf{h}^{\mathsf{x_{imu}}}_{\dot{\tau}_{\mathrm{T}},i,k}$ are the sensitivities of the tropospheric delay-rate model function $h_{\dot{\tau}_{\mathrm{T}},i,k}\left(\mathbf{x}_k\right)$ with respect to the error-states $\boldsymbol{\delta}\mathbf{v}^{E}_{k}$, $\boldsymbol{\delta}\mathbf{r}^{E}_{k}$, $\boldsymbol{\delta\theta}^{B}_{k}$ and $\delta\mathbf{x}_{\mathsf{imu}}$ [the error state set of $\mathbf{x}_{\mathsf{imu}}$ as defined in (3.45)], respectively.

E

Additional Plots

Figures E.1– E.12 complement the result plots presented in Section 4.5.2.

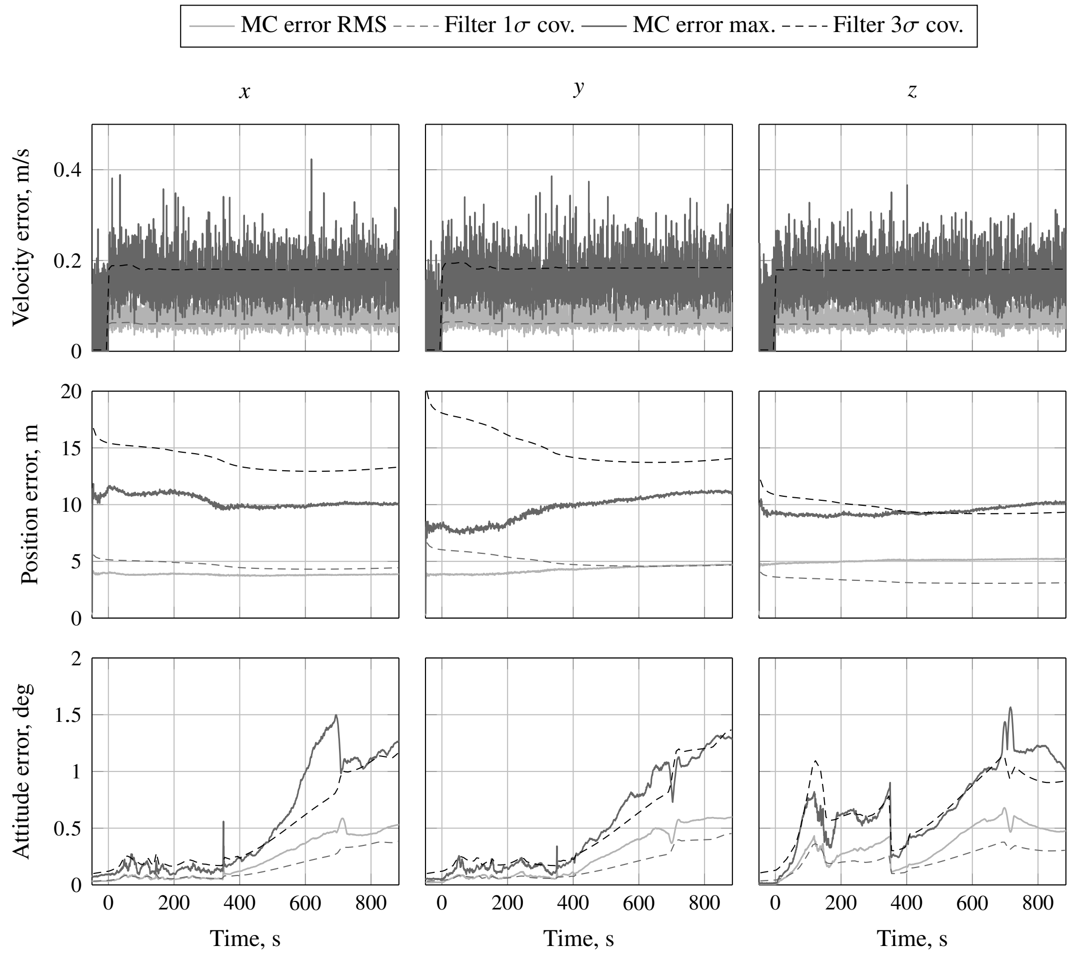


Figure E.1: Kinematics states in the **full-order** filter with **Low-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Position and velocity expressed in ECEF axes, attitude in Body. Nominal conditions along a Vega VV02 trajectory.

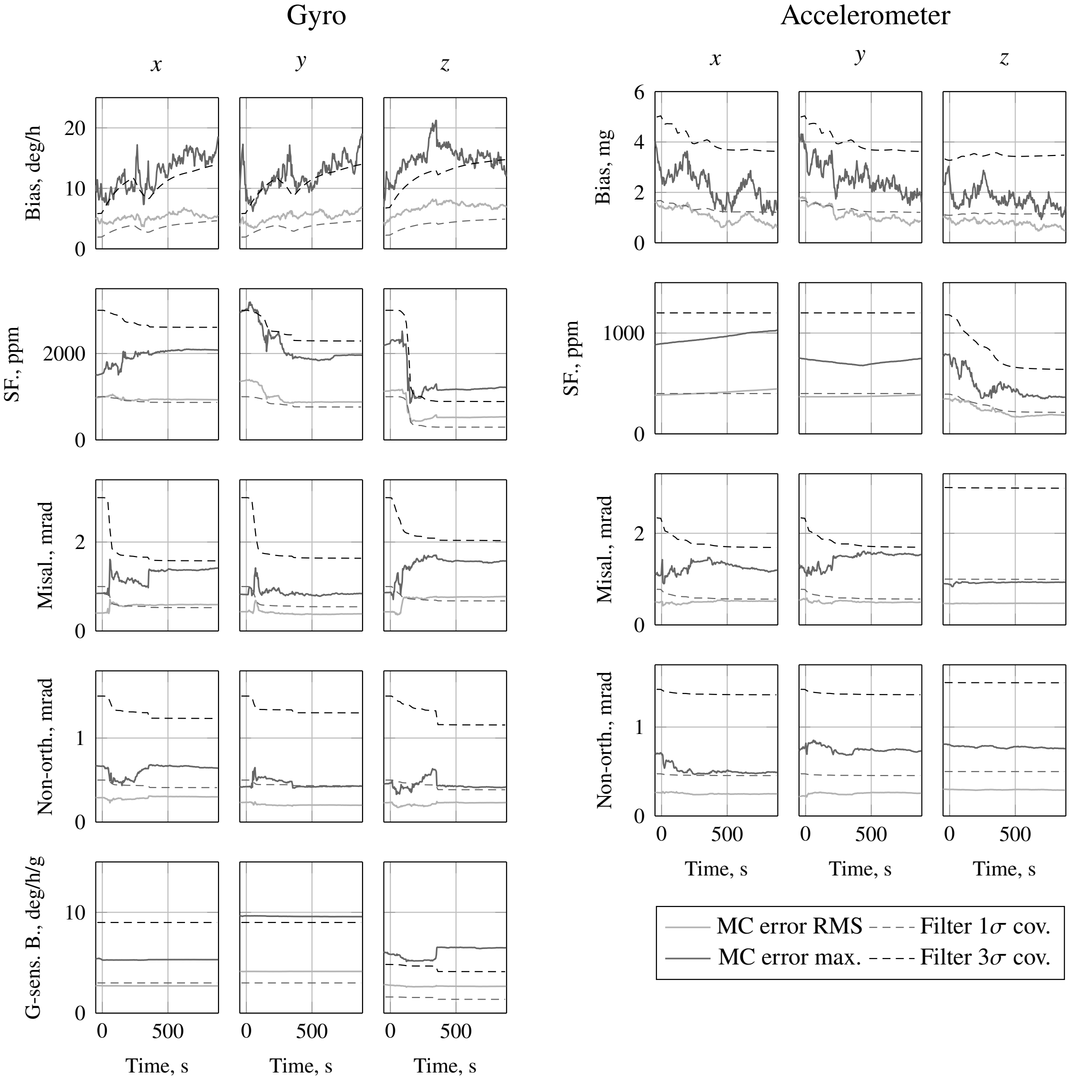


Figure E.2: Gyro and accelerometer states in the **full-order** filter with **Low-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Body frame axes. Nominal conditions along a Vega VV02 trajectory.

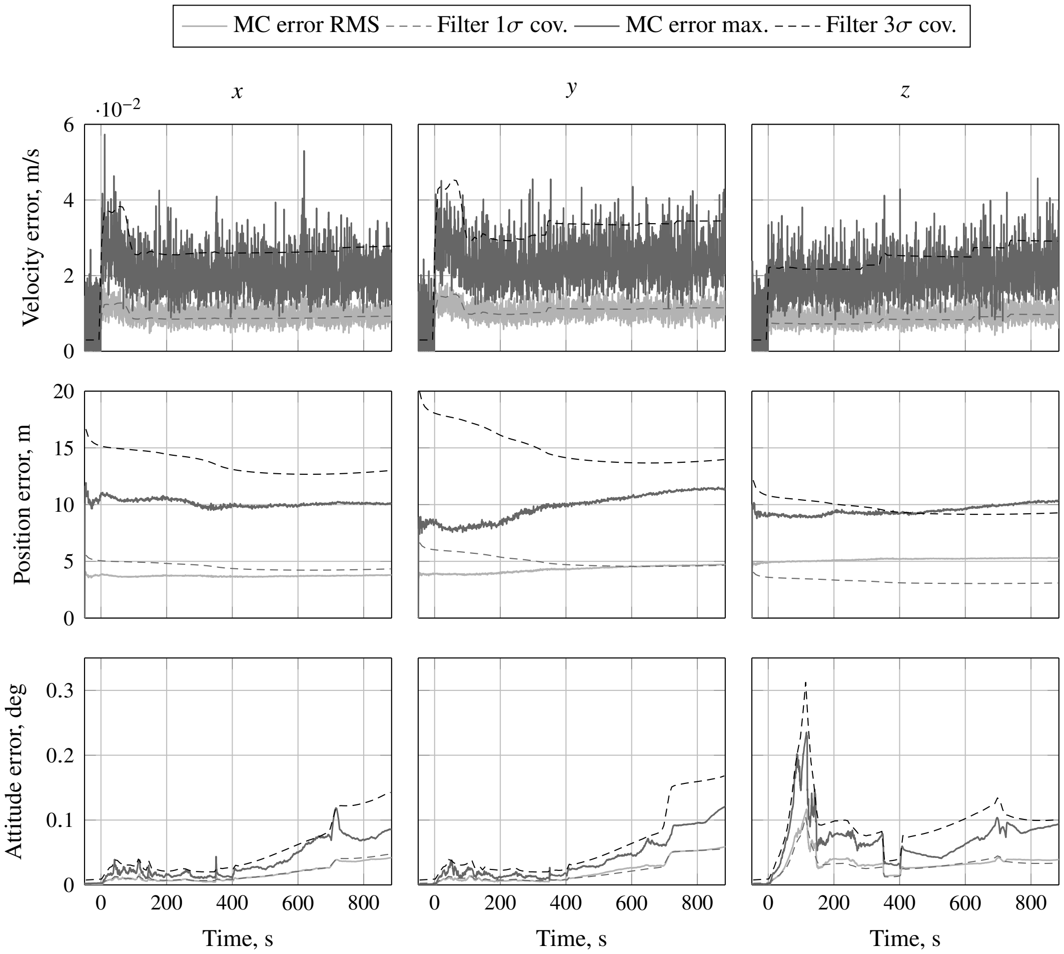


Figure E.3: Kinematics states in the **full-order** filter with **Medium-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Position and velocity expressed in ECEF axes, attitude in Body. Nominal conditions along a Vega VV02 trajectory.

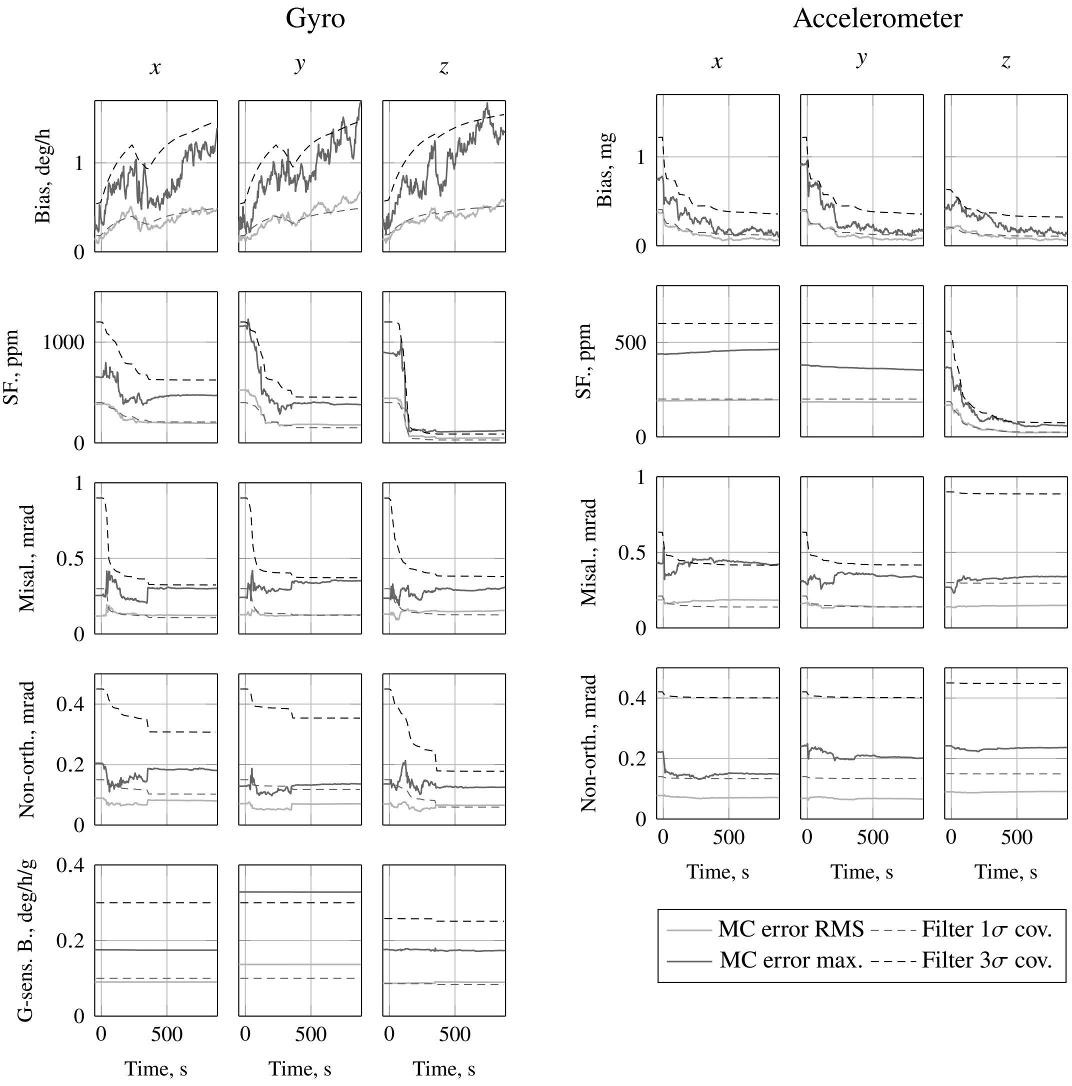


Figure E.4: Gyro and accelerometer states in the **full-order** filter with **Medium-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Body frame axes. Nominal conditions along a Vega VV02 trajectory.

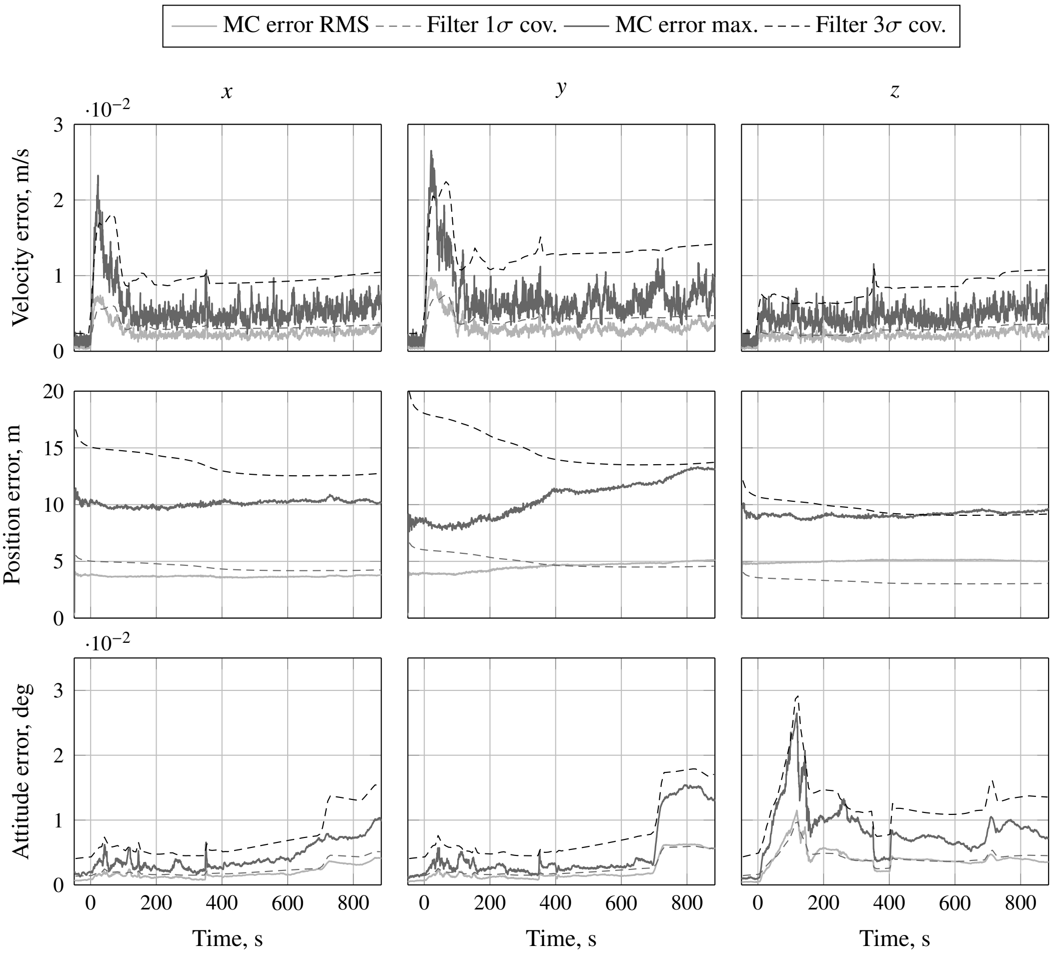


Figure E.5: Kinematics states in the **full-order** filter with **High-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Position and velocity expressed in ECEF axes, attitude in Body. Nominal conditions along a Vega VV02 trajectory.

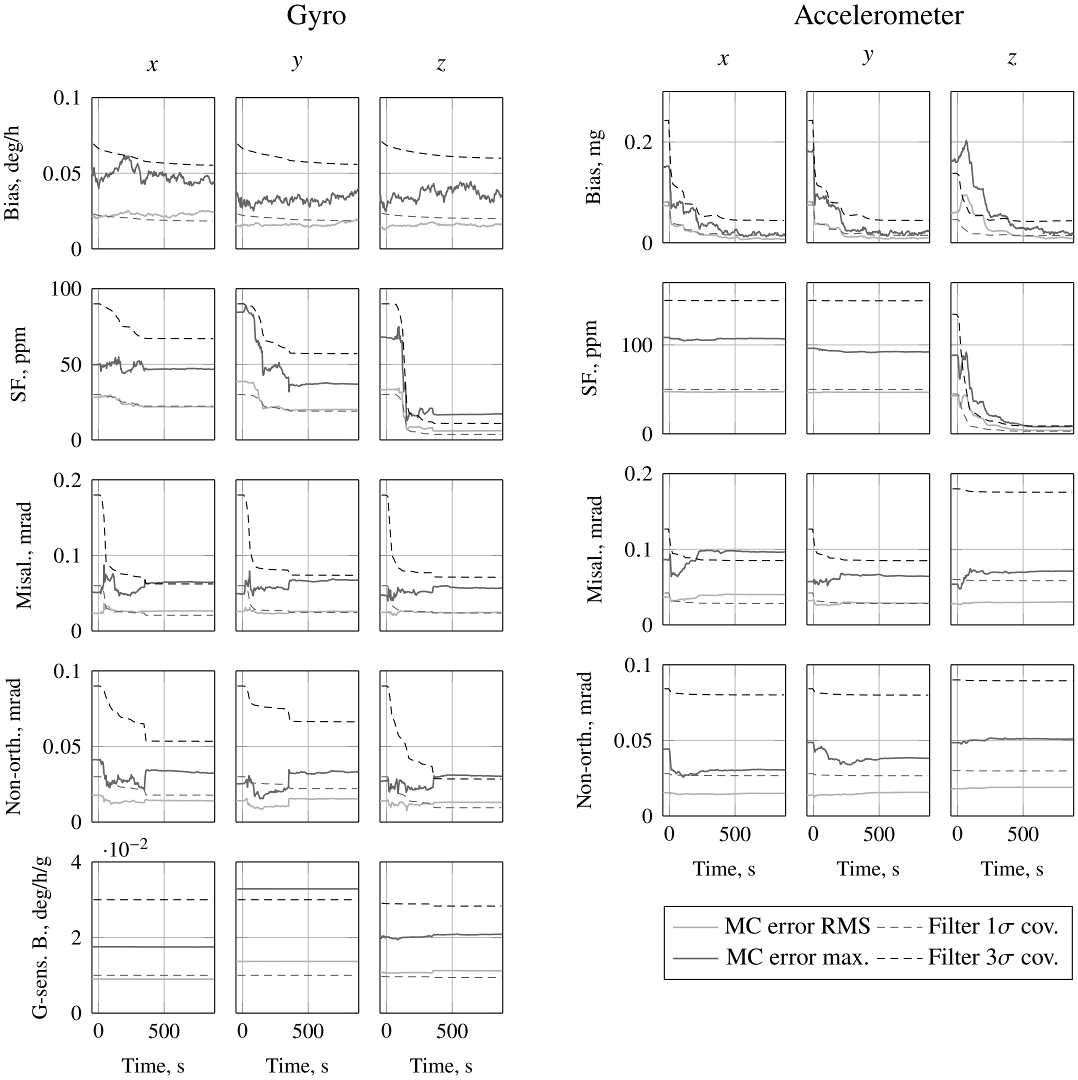


Figure E.6: Gyro and accelerometer states in the **full-order** filter with **High-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Body frame axes. Nominal conditions along a Vega VV02 trajectory.

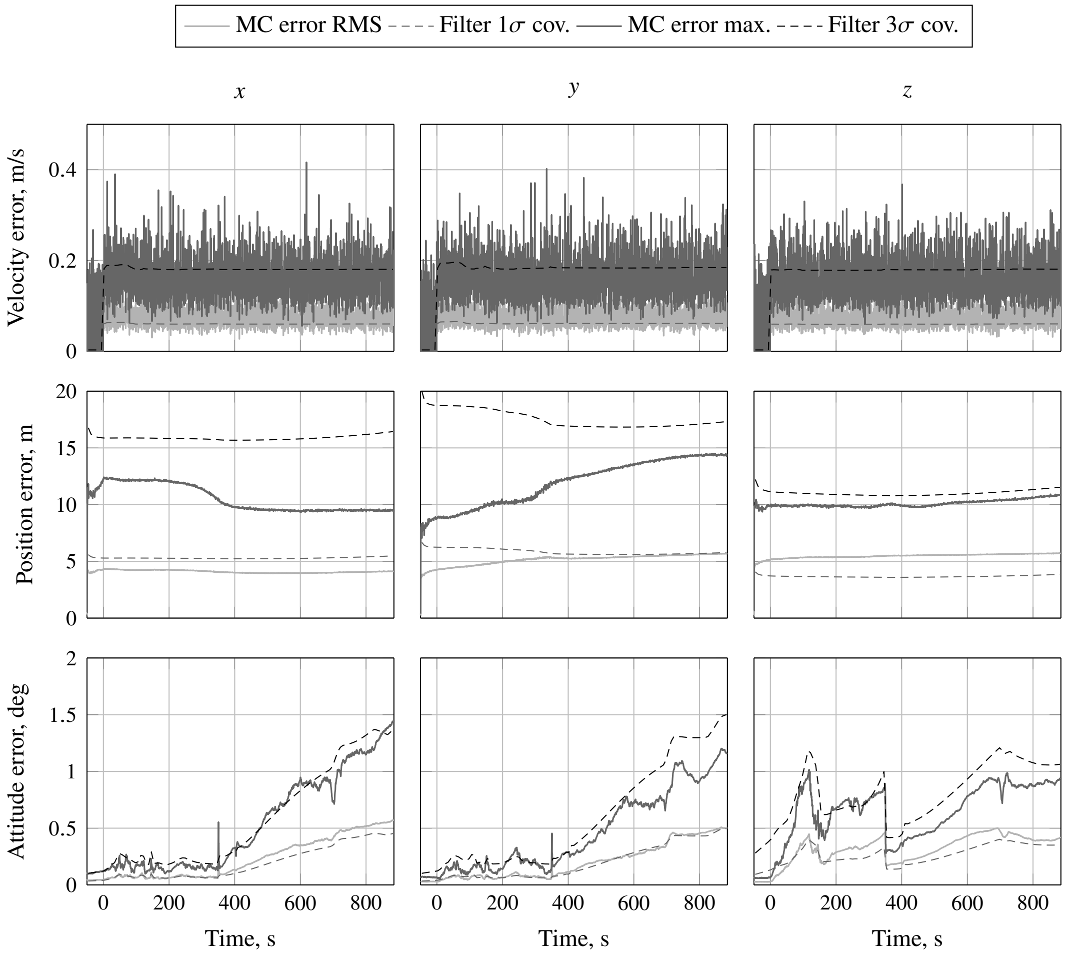


Figure E.7: Kinematics states in the **reduced-order** filter with **Low-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Position and velocity expressed in ECEF axes, attitude in Body. Nominal conditions along a Vega VV02 trajectory.

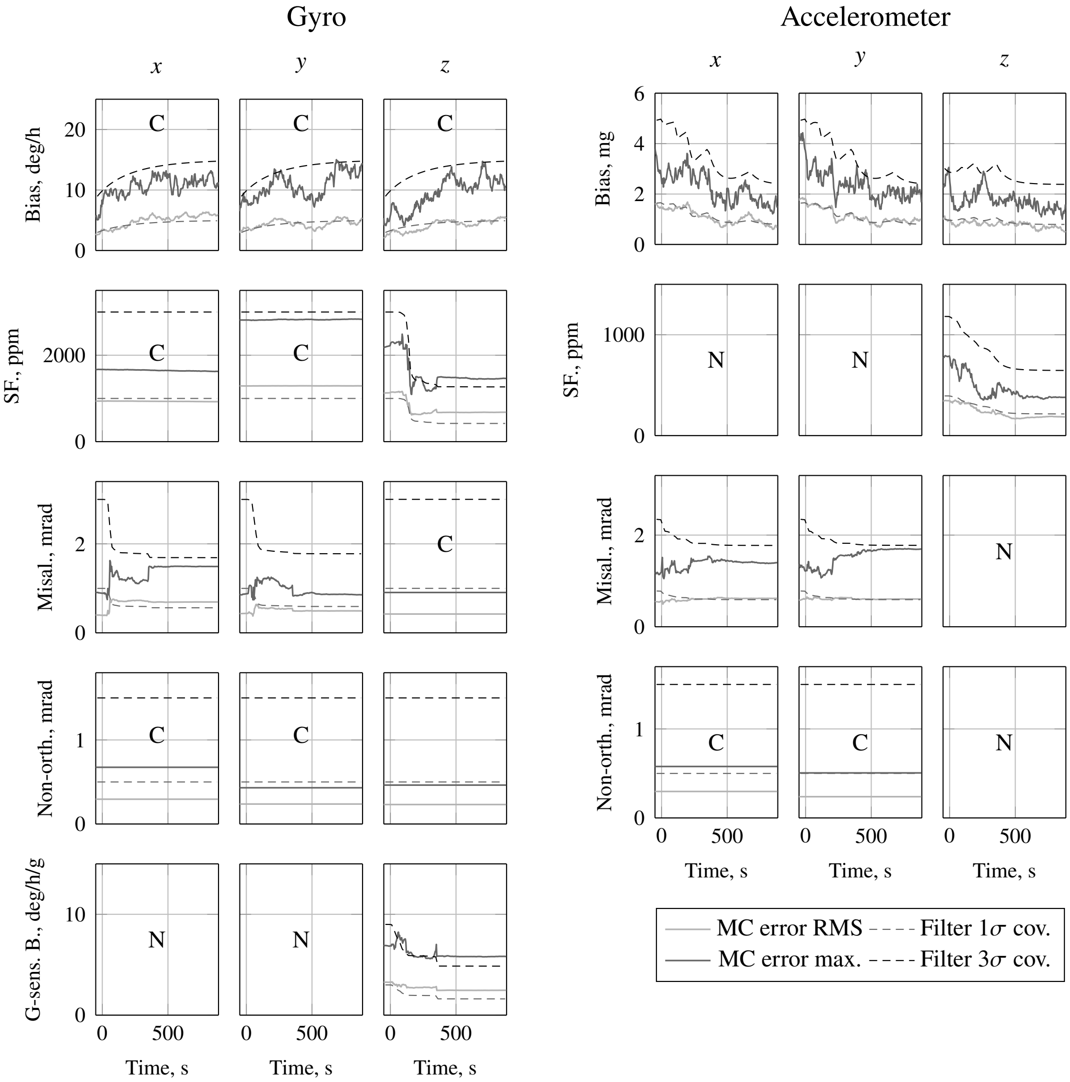


Figure E.8: Gyro and accelerometer states in the **reduced-order** filter with **Low-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. States in Body axes directions with considered and neglected parameters marked as C and N, respectively. Nominal conditions along a Vega VV02 trajectory.

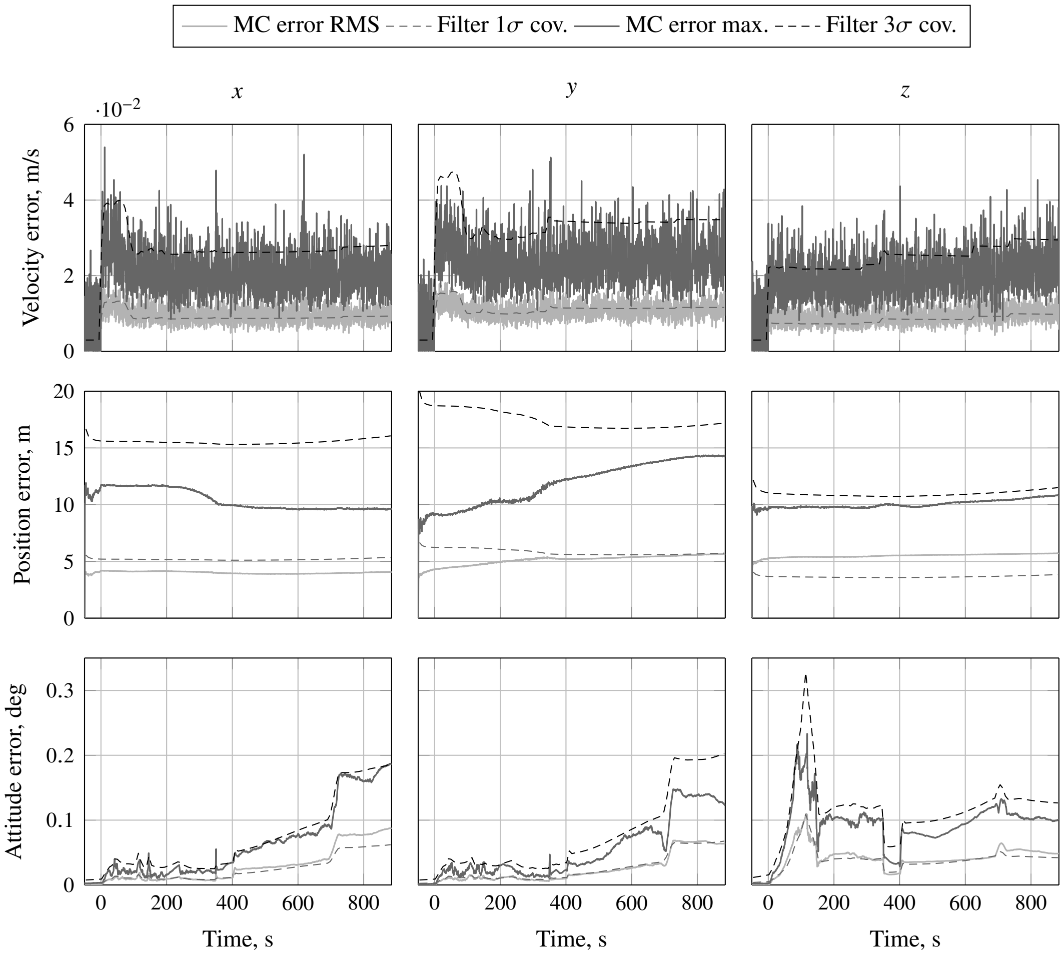


Figure E.9: Kinematics states in the **reduced-order** filter with **Medium-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Position and velocity expressed in ECEF axes, attitude in Body. Nominal conditions along a Vega VV02 trajectory.

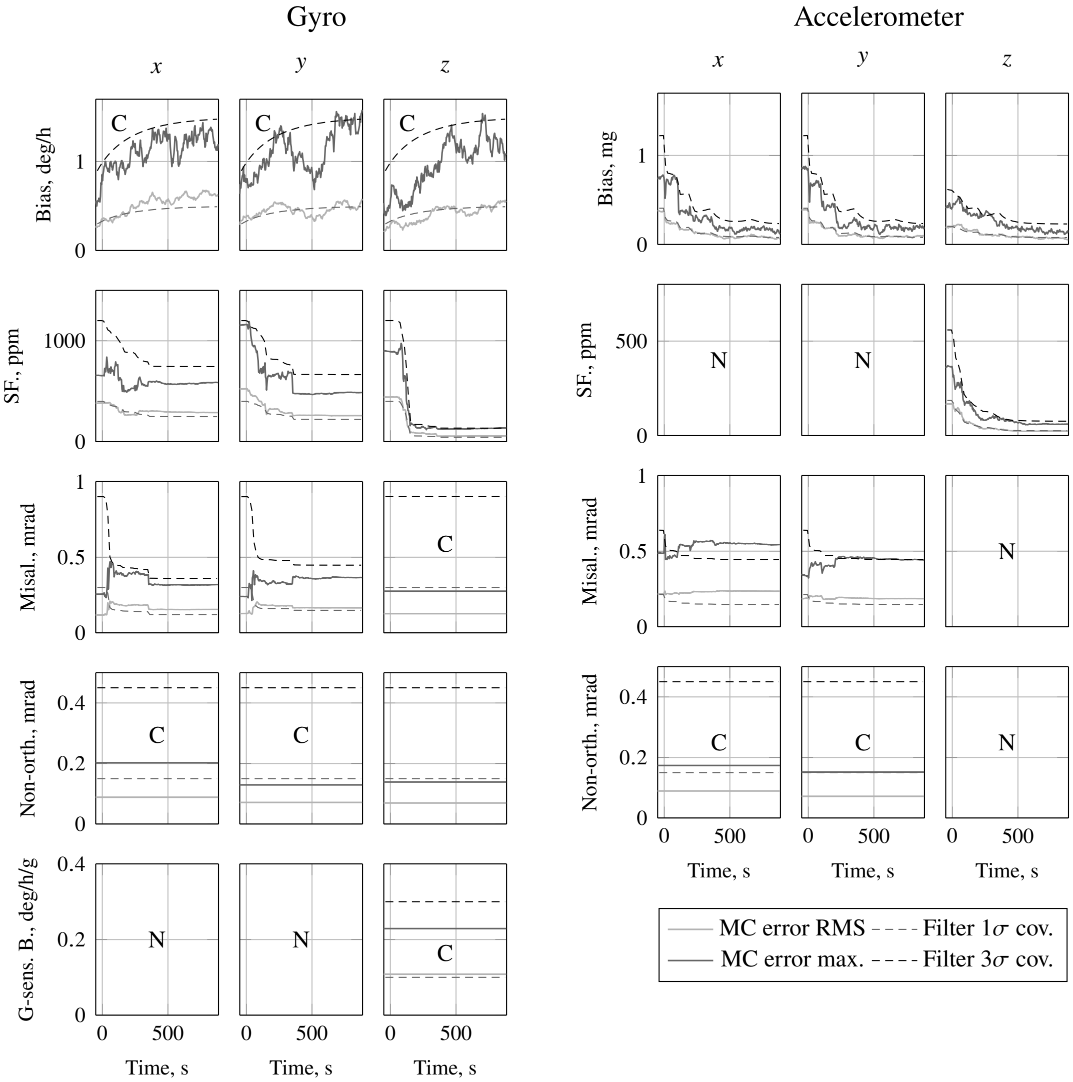


Figure E.10: Gyro and accelerometer states in the **reduced-order** filter with **Medium-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. States in Body axes directions with considered and neglected parameters marked as C and N, respectively. Nominal conditions along a Vega VV02 trajectory.

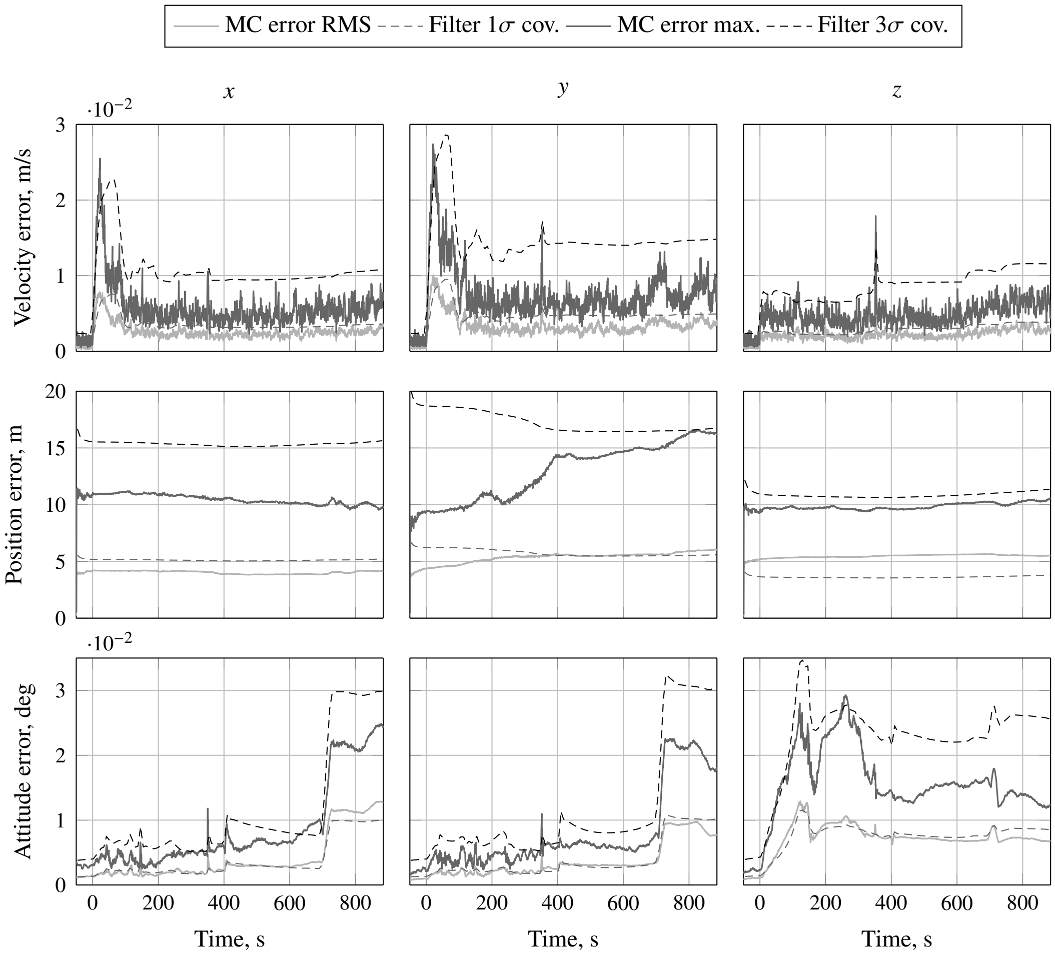


Figure E.11: Kinematics states in the **reduced-order** filter with **High-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. Position and velocity expressed in ECEF axes, attitude in Body. Nominal conditions along a Vega VV02 trajectory.

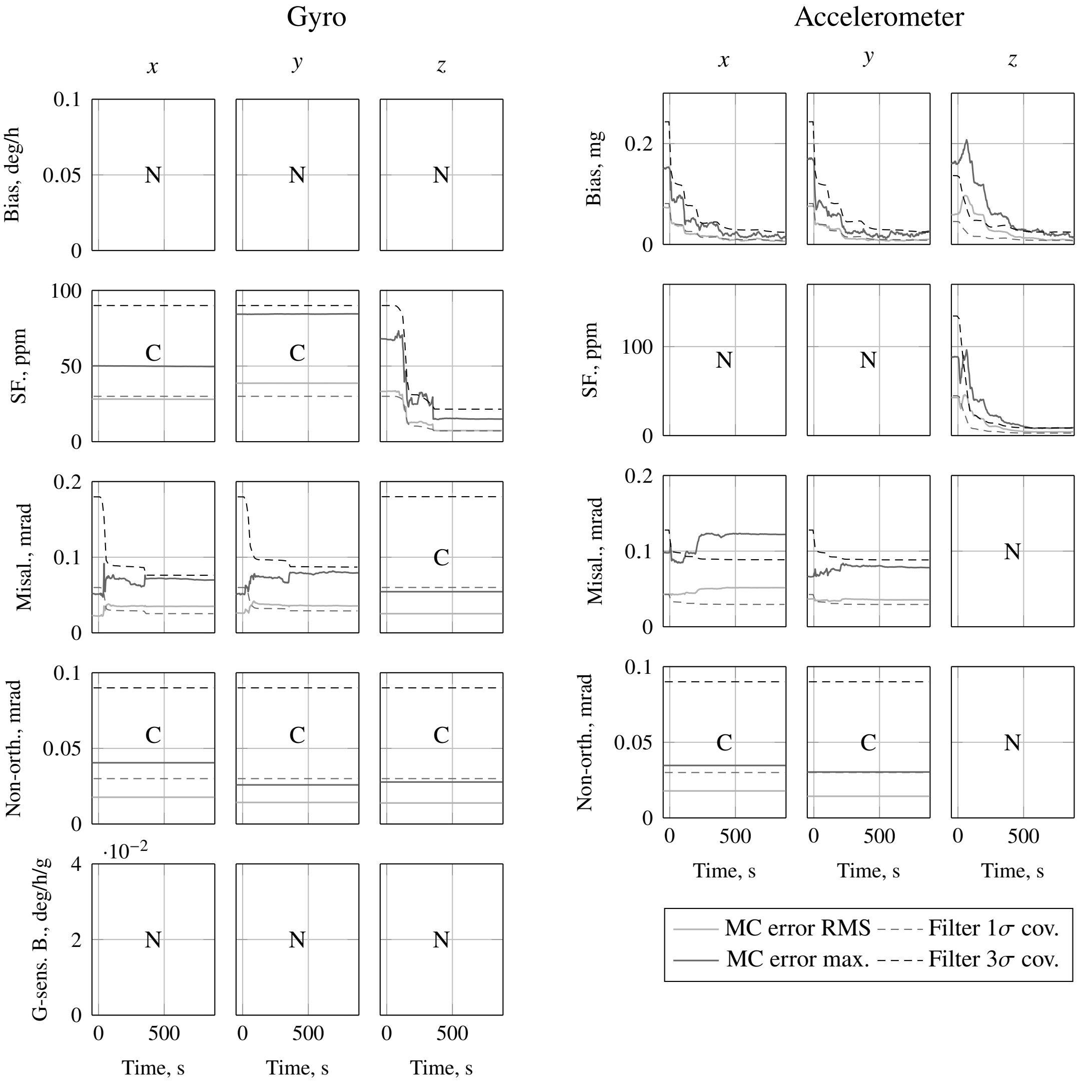


Figure E.12: Gyro and accelerometer states in the **reduced-order** filter with **High-grade** inertial sensors. Monte Carlo RMS and maximum bounds *vs* predicted 1σ and 3σ filter covariance. States in Body axes directions with considered and neglected parameters marked as C and N, respectively. Nominal conditions along a Vega VV02 trajectory.

References

[1] D. W. Allan. “A Mdofied "Allan Variance" With Increased Oscillator Characterization Ability”. In: *35th Annual Frequency Control Symposium*. 1981. DOI: 10.1109/FREQ.1981.200514 (cit. on pp. 186, 187).

[2] D. W. Allan. “Statistics of Atomic Frequency Standards”. In: *Proceedings of the IEEE* 54.2 (Feb. 1966), pp. 221–230. DOI: 10.1109/PROC.1966.4634 (cit. on pp. 186, 187).

[3] D. W. Allan. “Time and Frequency (Time-Domain) Characterization, Estimation, and Prediction of Precision Clocks and Oscillators”. In: *IEEE Transportations on Ultrasonics, Ferroelectrics, and Frequency Control* UFFC-34.6 (1987), pp. 647–654 (cit. on pp. 186, 187).

[4] G. L. Andersen, A. C. Christensen, and O. Ravn. “Augmented models for improving vision control of a mobile robot”. In: *IEEE 3rd Conference on Control Applications*. Vol. 1. 1994, pp. 53–58. DOI: 10.1109/CCA.1994.381270 (cit. on p. 132).

[5] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis*. Wiley & Sons, 2003 (cit. on p. 68).

[6] *Arianespace Lancement VV02 Vega: Proba-V, VNREDSat-1 et ESTCube-1 Succès de la mission*. In French. Last accessed: 2019-07-15. Arianespace. URL: http://www.arianespace.com/wp-content/uploads/2015/10/5-7-2013_FR.pdf (cit. on p. 165).

[7] *Arianespace Lancement VV03: Succès de la mission Le satellite KazEOSat-1 (DZZ-HR) est en orbite*. In French. Last accessed: 2019-07-15. Arianespace. URL: http://preproduction.arianespace.com/wp-content/uploads/2015/09/4-29-2014-VV03-launch-success_FR.pdf (cit. on p. 165).

[8] *Atlas V Launch Services User’s Guide*. United Launch Alliance. U.S., Mar. 2010 (cit. on p. 4).

[9] M. Bak, T. D. Larsen, and P. M. and Nørgård. “Location Estimation using Delayed Measurements”. In: *IEEE 5th International Workshop on Advanced Motion Control*. Coimbra, Portugal, 1998, pp. 180–185. DOI: 10.1109/AMC.1998.743533 (cit. on p. 131).

[10] J. B. Bancroft and G. Lachapelle. “Estimating MEMS Gyroscope G-Sensitivity Errors in Foot Mounted Navigation”. In: *2nd International Conference on Ubiquitous Positioning, Indoor Navigation and Location-Based Service*. Helsinki, Finland, 2012 (cit. on pp. 21, 172, 178).

[11] I. Y. Bar-Itzhack and N. Berman. “Control Theoretic Approach to Inertial Navigation System”. In: *Journal of Guidance* 11.3 (May 1988), pp. 237–245. DOI: 10.2514/3.20299 (cit. on p. 81).

[12] Y. Bar-Shalom. “Update with Out-of-Sequence Measurements in Tracking: Exact Solution”. In: *IEEE Transactions on Aerospace and Electronic Systems* 38.3 (July 2002), pp. 769–778 (cit. on p. 131).

[13] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. Wiley & Sons, 2001 (cit. on pp. 45, 68, 69, 100).

[14] N. M. Barbour. “Inertial Navigation Sensors”. In: *Advances in Navigation Sensors and Integration Technology*. NATO RTO Lecture Series-232. NATO, 2003 (cit. on p. 18).

[15] R. H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics*. AIAA Education Series. New York: AIAA, 1987 (cit. on p. 95).

[16] D. S. Bayard and P. B. Brugarolas. "An Estimation Algorithm for Vision-Based Exploration of Small Bodies in Space". In: *American Control Conference*. 2005, pp. 4589–4595. DOI: 10.1109/ACC.2005.1470719 (cit. on p. 132).

[17] S. Belin, S. Villers, and A. Conde Reis. "Requirements toward GNSS chain for Ariane 5 Mid-Life Evolution". In: *5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*. 2010. DOI: 10.1109/NAVITEC.2010.5708078 (cit. on pp. 1, 2, 5, 11, 13, 39, 93).

[18] N. Bergman. "Recursive Bayesian Estimation: Navigation and Tracking Applications". PhD thesis. Sweden: Linköping University, 1999 (cit. on p. 81).

[19] S. M. Bezick, A. J. Pue, and C. M. Patzelt. "Inertial Navigation for Guided Missile Systems". In: *Johns Hopkins APL Technical Digest* 1.4 (2010), pp. 331–342 (cit. on p. 18).

[20] U. I. Bhatti and W. Y. Ochieng. "Failure Modes and Models for Integrated GPS/INS Systems". In: *The Journal of Navigation* (2007). DOI: 10.1017/S0373463307004237 (cit. on pp. 97, 98).

[21] G. J. Bierman. *Factorization Methods for Discrete Sequential Estimation*. Ed. by R. Bellman. Academic Press, 1977 (cit. on p. 30).

[22] S. Blair. "V For Vegetation - The mission of Proba-V". In: *ESA Bulletin* 153 (Feb. 2013), pp. 10–21 (cit. on p. 165).

[23] S. Boulade, B. Frapard, and A. Biard. "GPS/INS Navigation System for Launchers and Re-Entry Vehicles - Developments and Adaptation to Ariane 5". In: *5th International ESA Conference on Guidance, Navigation and Control Systems*. Frascati, Italy, 2002 (cit. on pp. 5, 24, 52, 93).

[24] B. Braun, A. Grillenberger, and M. Markgraf. "Performance analysis of an IMU-augmented GNSS tracking system on board the MAIUS-1 sounding rocket". In: *CEAS Space Journal* 10.3 (Sept. 2018), pp. 407–425. DOI: 10.1007/s12567-018-0206-8 (cit. on p. 150).

[25] B. Braun, M. Markgraf, and O. Montenbruck. "Performance analysis of IMU-augmented GNSS tracking systems for space launch vehicles". In: *CEAS Space Journal* 8.2 (2016), pp. 117–133. DOI: 10.1007/s12567-016-0113-9 (cit. on pp. 1–3, 5, 11, 14, 18, 26, 37, 54, 95).

[26] L. Breger and J. P. How. "Gauss's Variational Equation-Based Dynamics and Control for Formation Flying Spacecraft". In: *Journal of Guidance, Control and Dynamics* 30.2 (Mar. 2007), pp. 437–448. DOI: 10.2514/1.22649 (cit. on p. 95).

[27] K. R. Britting. *Inertial Navigation Systems Analysis*. Wiley & Sons, 1971 (cit. on pp. 1, 14, 16, 17, 24, 25).

[28] K. R. Britting and T. Palsson. "Self Alignment Techniques for Strapdown Inertial Navigation Systems with Aircraft Application". In: *Journal of Aircraft* 7.4 (1970), pp. 302–307. DOI: 10.2514/3.44168 (cit. on p. 24).

[29] R. Broquet et al. "HiNAV Inertial / GNSS Hybrid Navigation System for Launchers and Re-entry Vehicles". In: *5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*. ESA. 2010. DOI: 10.1109/NAVITEC.2010.5707998 (cit. on pp. 2, 5).

[30] R. G. Brown. "Global Positioning System: Theory and Applications". In: ed. by B. W. Parkinson and J. J. Spilker Jr. Vol. 2. AIAA, 1996. Chap. Receiver Autonomous Integrity Monitoring (cit. on pp. 3, 37, 102, 128).

[31] R. G. Brown and P. Y. C. Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. Wiley & Sons, 1997 (cit. on p. 172).

[32] P. Buist. "The Baseline Constrained LAMBDA Method for Single Epoch, Single Frequency Attitude Determination Applications". In: *20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007)*. 2007, pp. 2962–2973 (cit. on p. 150).

[33] E. Burke, E. Rutkowski, and E. Rutkowski. “Vehicle Based Independent Tracking System (VBITS): A Small, Modular, Avionics Suite for Responsive Launch Vehicle and Satellite Applications”. In: *6th Responsive Space Conference*. 2008 (cit. on p. 4).

[34] A. Buysse, W. H. Steyn, and A. Schutte. “Flight control system for a reusable rocket booster on the return flight through the atmosphere”. In: *11th IAA Symposium on Small Satellites*. Berlin, Germany, 2017 (cit. on p. 150).

[35] A. Chatfield. *Fundamentals of High Accuracy Inertial Navigation*. Ed. by P. Zarchan. Vol. 174. Progress in Astronautics and Aeronautics. AIAA, 1997 (cit. on pp. 2, 16, 24, 25).

[36] S. Y. Cho, H. K. Lee, and H. K. Lee. “Observability and Estimation Error Analysis of the Initial Fine Alignment Filter for Nonleveling Strapdown Inertial Navigation System”. In: *Journal of Dynamic Systems, Measurement, and Control* 135 (Mar. 2013), pp. 1–9. DOI: 10.1115/1.4007552 (cit. on p. 82).

[37] J. P. Collins. “Assessment and Development of a Tropospheric Delay Model for Aircraft Users of the Global Positioning System”. MA thesis. Department of Geodesy and Geomatics Engineering Technical Report No. 203, University of New Brunswick, 1999 (cit. on pp. 31, 32).

[38] J. L. Crassidis and J. L. Junkins. *Optimal Estimation of Dynamic Systems*. Chapman & Hall/CRC, 2012 (cit. on pp. 37, 68, 69, 71, 131).

[39] J. L. Crassidis, F. L. Markley, and Y. Cheng. “A Survey of Nonlinear Attitude Estimation Methods”. In: *Journal of Guidance, Control and Dynamics* 30.1 (Jan. 2007), pp. 12–28. DOI: 10.2514/1.22452 (cit. on p. 46).

[40] *Delta IV Launch Services User’s Guide*. United Launch Alliance. U.S., June 2013 (cit. on p. 4).

[41] J. Demmel, L. Grigori, M. F. Hoemmen, and J. Langou. *Communication-optimal parallel and sequential QR and LU factorizations*. Tech. rep. UCB/EECS-2008-89. Electrical Engineering and Computer Sciences, University of California at Berkeley, 2008 (cit. on p. 88).

[42] *Department of Defense World Geodetic System, 1984 (WGS-84), Its Definition and Relationships With Local Geodetic Systems*. TR8350.2, 3rd Ed., Amendment 1. National Imagery and Mapping Agency (NIMA), Jan. 2000 (cit. on p. 14).

[43] J. L. Devore. *Probability and Statistics for Engineering and the Sciences*. 8th Ed. Brooks/Cole, 2012 (cit. on p. 69).

[44] A. J. van Dierendonck, J. B. McGraw, and R. G. Brown. “Relationship Between Allan Variances and Kalman Filter Parameters”. In: *16th Annual PTTI Meeting*. 1984 (cit. on p. 29).

[45] V. D. Dishel and E. L. Mezhiritskiy. “Principals Of Integrated INS/GLONASS+GPS GNC Systems For Space Launchers. Results Of Realized Missions And Future Prospects”. In: *8th International ESA Conference on Guidance, Navigation & Control Systems*. June 2011 (cit. on pp. 2, 4).

[46] S. Dussy, D. Durrant, T. Moy, N. Perriault, and B. Celerier. “MEMS gyro for space applications: Overview of European activities”. In: *AIAA Guidance, Navigation and Control Conference and Exhibit*. 2005. DOI: 10.2514/6.2005-6466 (cit. on p. 19).

[47] S. Dutta and R. D. Braun. “Cramér-Rao Lower-Bound Optimization of Flush Atmospheric Data System Sensor Placement”. In: *Journal of Spacecraft and Rockets* 51.6 (Nov. 2014), pp. 1773–1788. DOI: 10.2514/1.A32824 (cit. on p. 81).

[48] R. E. Ebner and J. G. Mark. “Redundant Integrated Flight-Control/Navigation Inertial Sensor Complex”. In: *Journal of Guidance and Control* 1.2 (Mar. 1978), pp. 143–149. DOI: 10.2514/3.55757 (cit. on p. 169).

[49] J. A. Estefan and O. J. Sovers. *A Comparative Survey of Current and Proposed Tropospheric Refraction-Delay Models for DSN Radio Metric Data Calibration*. Tech. rep. JPL, NASA, 1994 (cit. on p. 32).

[50] F. A. Evans and J. C. Wilcox. “Experimental strapdown redundant sensor inertial navigation system”. In: *Journal of Spacecraft and Rockets* 7.9 (Sept. 1970), pp. 1070–1074. DOI: 10.2514/3.30104 (cit. on p. 169).

[51] J. A. Farrell. *Aided Navigation - GPS with High Rate Sensors*. McGraw-Hill, 2008 (cit. on pp. 16, 20, 21, 25, 29, 32, 51).

[52] P. C. Fenton, W. Kunysz, and G. Garbe. “Using GPS for Position and Attitude Determination of the Canadian Space Agency’s Active Rocket Mission”. In: *11th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1998)*. 1998, pp. 1791–1800 (cit. on p. 150).

[53] B. Ferrell, J. Simpson, R. Zoerner, J. B. Bull, and R. J. Lanzi. “Autonomous Flight Safety System”. In: *41st Space Congress*. 2004 (cit. on p. 4).

[54] J.-J. Floch, F. Soualle, P. D. Resta, S. Belin, and S. Reynaud. “Galileo Signal Tracking and Acquisition Performance Predictions for a Dynamic Launcher”. In: *Proceedings of the 2008 National Technical Meeting of the Institute of Navigation*. Institute of Navigation. 2008, pp. 636–647 (cit. on p. 5).

[55] D. J. Flynn. *A discussion of coning errors exhibited by inertial navigation systems*. Tech. rep. AD-A147-846. Royal Aircraft Establishment, 1984 (cit. on p. 17).

[56] M. Giannini, M. Melara, and C. Roux. “ALTS Localization System Of Vega Launcher: VV02 Post-Flight Analysis & GPS Issues For Future Hybrid Navigation”. In: *9th International ESA Conference on Guidance, Navigation & Control Systems*. June 2014 (cit. on p. 5).

[57] B. Gibbs. *Advanced Kalman Filtering, Least-Squares and Modeling*. Wiley & Sons, 2011 (cit. on pp. 102, 122).

[58] J. P. Gilmore and R. A. McKern. “A Redundant Strapdown Inertial Reference Unit (SIRU)”. In: *Journal of Spacecraft* 9.1 (Jan. 1972), pp. 39–47. DOI: 10.2514/3.61628 (cit. on p. 169).

[59] *Global Positioning and Inertial Measurements Range Safety Tracking Systems’ Commonality Standard*. RCC 324-01. Range Commanders Council, 2001 (cit. on p. 11).

[60] S. Gomez. *Three Years of Global Positioning System Experience on International Space Station*. NASA/TP-2006-213168. NASA. Houston, Texas, U.S., Aug. 2006 (cit. on p. 4).

[61] S. Gomez and M. Lammer. “Lessons Learned from Two Years of On-Orbit Global Positioning System Experience on International Space Station”. In: *ION-GNSS Meeting*. Long Beach, U.S., 2004 (cit. on p. 4).

[62] S. Gonseth, F. Rudolf, C. Eichenberger, D. Durrant, and P. Airey. “Miniaturized high-performance MEMS accelerometer detector”. In: *CEAS Space Journal* 7.2 (2015), pp. 263–270. DOI: 10.1007/s12567-015-0093-1 (cit. on p. 19).

[63] R. Goodstein, B. K. Tse, and D. J. WinkeI. “Flight Test Results of the Inertial Upper Stage Redundant Inertial Measurement Unit Redundancy Management Technique”. In: *AlAA 22nd Aerospace Sciences Meeting*. 1984. DOI: 10.2514/6.1984-568 (cit. on p. 169).

[64] F. van Graas and A. Soloviev. “Precise Velocity Estimation Using a Stand-Alone GPS Receiver”. In: *Journal of the Institute of Navigation* 51.4 (2004), pp. 283–292. DOI: 10.1002/j.2161-4296.2004.tb00359.x (cit. on p. 55).

[65] M. S. Grewal, L. R. Weill, and A. P. Andrews. *Global Positioning Systems, Inertial Navigation, and Integration*. Wiley & Sons, 2001 (cit. on p. 12).

[66] B. Griffiths, H. Li, and T. Gray. “Launch Vehicle Tracking Enhancement Through Global Positioning System Metric Tracking”. In: *IEEE Aerospace Conference*. June 2016. DOI: 10.1109/AERO.2016.7500533 (cit. on p. 4).

[67] J. Gross, Y. Gu, S. Gururajan, B. Seanor, and M. R. Napolitano. “A Comparison of Extended Kalman Filter, Sigma-Point Kalman Filter, and Particle Filter in GPS/INS Sensor Fusion”. In: *AIAA Guidance, Navigation, and Control Conference*. 2010. DOI: 10.2514/6.2010-8332 (cit. on p. 36).

[68] P. D. Groves. “Navigation using Inertial Sensors”. In: *IEEE Aerospace and Electronic Systems Magazine* 30.2 (2015), pp. 42–69. DOI: 10.1109/MAES.2014.130191 (cit. on pp. 2, 17, 20).

[69] P. D. Groves. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Artech House, 2008 (cit. on pp. 3, 12–14, 16, 24, 25, 27, 29, 55, 58, 95, 97, 101, 139).

[70] X. Guochang. *GPS Theory, Algorithms and Applications*. 2nd Ed. Springer, 2007 (cit. on pp. 27, 32).

[71] Hacettepe University. *TEC for Single Station, IONOLAB-TEC Online (v1.29)*. http://www.ionolab.org/webtec/single.html. Last accessed: 2019.07.12 (cit. on p. 157).

[72] F. M. Ham and R. G. Brown. “Observability, Eigenvalues, and Kalman Filtering”. In: *IEEE Transactions on Aerospace and Electronic Systems* AES-19.2 (Mar. 1983), pp. 269–273. DOI: 10.1109/TAES.1983.309446 (cit. on pp. 81, 82).

[73] A. Hauschild et al. “Results of the GNSS Receiver Experiment OCAM-G on Ariane-5 flight VA 219”. In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 231.6 (June 2015), pp. 1100–1114. DOI: 10.1177/0954410016648351 (cit. on pp. 4, 5, 54, 66, 95, 150).

[74] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle. *GNSS - Global Navigation Satellite Systems*. Springer, 2008 (cit. on p. 32).

[75] S. Hong, H. H. Chun, S. H. Kwon, and M. H. Lee. “Observability Measures and Their Application to GPS/INS”. In: *IEEE Transactions on Vehicular Technology* 57.1 (Jan. 2008), pp. 97–106. DOI: 10.1109/TVT.2007.905610 (cit. on p. 81).

[76] H. Hou. “Modeling Inertial Sensors Errors Using Allan Variance”. PhD thesis. University of Calgary, 2004 (cit. on p. 20).

[77] H. Hou and N. El-Sheimy. “Inertial Sensors Errors Modeling Using Allan Variance”. In: *16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*. 2003, pp. 2860–2867 (cit. on p. 186).

[78] R. Hunger. *Floating Point Operations in Matrix-Vector Calculus*. Tech. rep. Tenchical University of Munich, 2007 (cit. on pp. 87, 88).

[79] *IEEE Standard for Inertial Sensor Terminology*. Std 528-2001. IEEE Aerospace and Electronic Systems Society, Nov. 2001. DOI: 10.1109/IEEESTD.2001.93360 (cit. on p. 19).

[80] *IEEE Standard Specification Format Guide and Test Procedure for Linear Single-Axis, Nongyroscopic Accelerometers*. IEEE Std 1293™-1998. IEEE-SA Standards Board. DOI: 10.1109/IEEESTD.2011.5960745 (cit. on p. 186).

[81] *IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fiber Optic Gyros*. IEEE Std 952-1997. IEEE-SA Standards Board. DOI: 10.1109/IEEESTD.1998.86153 (cit. on pp. 186–188).

[82] *IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Laser Gyros*. IEEE Std 647-1995. IEEE-SA Standards Board. DOI: 10.1109/IEEESTD.2006.246241 (cit. on p. 186).

[83] M.B. Ignagni. “Efficient Class of Optimized Coning Compensation Algorithms”. In: *Journal of Guidance, Control and Dynamics* 19.2 (1996), pp. 424–429. DOI: 10.2514/3.21635 (cit. on pp. 17, 18).

[84] *iMAR iIMU-FCAI datasheet*. Last accessed: August 2016. iMAR. URL: http://www.cbil.co.uk/index.php/download_file/view/796/214/ (cit. on pp. 5, 154).

[85] *Independent Assessment of X-37 Safety & Mission Assurance Processes and Design Features*. NASA Headquarters Office of Safety & Mission Assurance. U.S., June 2001 (cit. on p. 4).

[86] H. W. Janes, R. B. Langley, and S. P. Newby. "Analysis of tropospheric delay prediction models: comparisons with ray-tracing and implications for GPS relative positioning". In: *Bulletin Géodésique* 65 (1991), pp. 151–161 (cit. on p. 32).

[87] R. Jategaonkar, R. Behr, W. Gockel, and C. Zorn. "Data Analysis of Phoenix Reusable Launch Vehicle Demonstrator Flight Test". In: *Journal of Aircraft* 42.2006-12 (2006), pp. 1732–1737. DOI: 10.2514/1.19602 (cit. on p. 150).

[88] A. H. Jazwinski. *Stochastic Processes And Filtering Theory*. Academic Press, 1970 (cit. on p. 81).

[89] C. Jekeli. *Inertial Navigation Systems with Geodetic Applications*. Walter de Gruyter, 2000 (cit. on pp. 16, 17, 22).

[90] G. M. Jenkins and D. G. Watts. *Spectral Analysis and its Applications*. Holden-Day, 1968 (cit. on p. 69).

[91] John A. Volpe National Transportation Systems Center. *Vulnerability Assessment of the Transportation Infrastructure Relying on the Global Positioning System*. Tech. rep. U. S. Department of Transportation, 2001 (cit. on p. 3).

[92] E. D. Kaplan and C. J. Hegarty. *Understanding GPS: Principles and Applications*. 2nd Ed. Artech House, 2006 (cit. on pp. 3, 12, 13, 26, 27, 30, 156).

[93] V. Kempe. *Inertial MEMS: Principles and Practice*. 1st Ed. Cambridge University Press, 2011 (cit. on pp. 19, 21).

[94] A. D. King. "Characterization of Gyro In-run Drift". In: *DGON Symposium Gyro Technology*. 1984 (cit. on pp. 186, 187).

[95] J.A. Klobuchar. "Global Positioning System: Theory and Applications". In: ed. by B. W. Parkinson and J. J. Spilker Jr. AIAA, 1996. Chap. Ionospheric effects on GPS (cit. on p. 34).

[96] C. R. Kochakian. "Time-Domain Uncertainty Charts (Green Charts): A Tool for Validating the Design of IMU/Instrument Interfaces". In: *AIAA Guidance and Control Conference*. 1980 (cit. on pp. 186, 187).

[97] T. D. Larsen, N. A. Andersen, O. Rovn, and N. K. Poulsen. "Incorporation of Time Delayed Measurements in a Discrete-time Kalman Filter". In: *37th IEEE Conference on Decision & Control*. Tampa, Florida, Dec. 1998 (cit. on p. 131).

[98] A. Lawrence. *Modern Inertial Technology - Navigation, Guidance, and Control*. 2nd Ed. Springer, 1998 (cit. on p. 19).

[99] R. Leandro, M. Santos, and R. Langley. "UNB Neutral Atmosphere Models: Development and Performance". In: *2006 National Technical Meeting of The Institute of Navigation*. 2006, pp. 564–573 (cit. on p. 157).

[100] W.M. Lear. *GPS navigation for low-earth orbiting vehicles*. NASA 87-FM-2, JSC-32031. Rev. 1. NASA, Lyndon B. Johnson Space Center, 1987 (cit. on p. 34).

[101] J. S. Leung, G. L. Fay II, T. A. Patrick, and R. W. Seibold. *Space-Based Navigation for RLVs and ELVs*. Tech. rep. U. S. Department of Transportation, 2006 (cit. on p. 37).

[102] S. C. Maki. "A low-cost GPS inertial guidance (GPSIG) for space boosters". In: *IEEE Position Location and Navigation Symposium*. 1990 (cit. on pp. 3, 95).

[103] M. Markgraf. "Radiation Effects on Integrated Circuits and Systems for Space Applications". In: ed. by R. Velazco, D. McMorrow, and J. Estela. Springer, 2019. Chap. The Phoenix GPS Receiver for Rocket and Satellite Applications: An Example for the Successful Utilization of COTS Technology in Space Projects. DOI: 10.1007/978-3-030-04660-6_14Âă (cit. on p. 5).

[104] M. Markgraf and O. Montenbruck. "Phoenix-HD - A Miniature GPS Tracking System for Scientific and Commercial Rocket Launches." In: *6th International Symposium on Launcher Technologies*. 2005 (cit. on pp. 5, 54, 154).

[105] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Ed. by J. R. Wertz. Springer, 2014 (cit. on p. xv).

[106] P. Matisko and V. Havlena. "Optimality tests and adaptive Kalman filter". In: *IFAC Proceedings Volumes* 45.16 (2012), pp. 1523–1528. DOI: 10.3182/20120711-3-BE-2027.00011 (cit. on p. 69).

[107] P. S. Maybeck. *Stochastic Models, EstControl, and Control*. Vol. I. Academic Press, 1982 (cit. on p. 68).

[108] P. S. Maybeck. *Stochastic Models, EstControl, and Control*. Vol. II. Academic Press, 1982 (cit. on pp. 36, 37).

[109] R. A. McKern. "A Study of Transformation Algorithms for Use in a Digital Computer". MA thesis. University of Illinois, 1968 (cit. on pp. 17, 41).

[110] R. K. Mehra and J. Peschon. "An Innovations Approach to Fault Detection and Diagnosis in Dynamic Systems". In: *Automatica* 7.5 (1971), pp. 637–640. DOI: 10.1016/0005-1098(71)90028-8 (cit. on p. 69).

[111] V. B. Mendes. "Modeling the neutral-atmosphere propagation delay in radiometric space techniques". PhD thesis. Department of Geodesy and Geomatics Engineering Technical Report No. 199, University of New Brunswick, Fredericton, Canada, 1999 (cit. on p. 32).

[112] O. Montenbruck, T. Ebinuma, E. G. Lightsey, and S. Leung. "A real-time kinematic GPS sensor for spacecraft relative navigation". In: *Aerosol Science and Technology* 6.6 (Oct. 2002), pp. 435–449. DOI: 10.1016/S1270-9638(02)01185-9 (cit. on pp. 30, 157, 159, 161).

[113] O. Montenbruck and E. Gill. "Ionospheric Correction for GPS Tracking of LEO Satellites". In: *The Journal of Navigation* 55.2 (2002), pp. 293–304. DOI: 10.1017/S0373463302001789 (cit. on pp. 33, 34).

[114] O. Montenbruck, E. Gill, and M. Markgraf. "Phoenix-XNS: A Miniature Real-Time Navigation System for LEO Satellites". In: *NAVITEC 2006*. The Netherlands, Dec. 2006 (cit. on p. 5).

[115] O. Montenbruck and M. Markgraf. "Global Positioning System Sensor with Instantaneous-Impact-Point Prediction for Sounding Rockets". In: *Journal of Spacecraft and Rockets* 41.4 (2004), pp. 644–650. DOI: 10.2514/1.1962 (cit. on p. 5).

[116] O. Montenbruck and M. Markgraf. *User's Manual for the GPS Orion-S/-HD Receiver*. GTN-MAN-0110. Deutsches Zentrum für Luft- und Raumfahrt (DLR). Oberpfaffenhofen, Germany, 2003. URL: https://www.dlr.de/rb/en/Portaldata/38/Resources/dokumente/GSOC_dokumente/RB-RFT/GTN-MAN-0110_prot.pdf (cit. on pp. 32, 157, 161, 163).

[117] O. Montenbruck and M. Markgraf. *User's Manual for the Phoenix GPS Receiver*. GTN-MAN-0120. Deutsches Zentrum für Luft- und Raumfahrt (DLR). Oberpfaffenhofen, Germany, 2008 (cit. on pp. 32, 157, 163).

[118] O.and Gill E. Montenbruck. *Satellite Orbits: Models, Methods, and Applications*. 2nd Ed. Springer, 2001 (cit. on pp. 27, 58, 139).

[119] T. C. Moore, M. A. Carr, and H. D. Friesen. "Launch vehicle mission capability enhancement through Global Positioning System Metric Tracking". In: *IEEE Aerospace Conference*. Apr. 2011. DOI: 10.1109/AERO.2011.5747486 (cit. on p. 4).

[120] T. A. Musabayev et al. "First in-orbit results from KazEOSat-1: the high-resolution Earth observation mission of the Republic of Kazakhstan". In: *Proceedings of the 65th International Astronautical Congress (IAC 2014)*. Paper: IAC-14-B1.2.2. Toronto, Canada, Sept. 2014 (cit. on p. 165).

[121] Narmada, S. Reynaud, P. Delaux, and A. Biard. “Use of GNSS for Next European Launcher Generation”. In: *6th International ESA Conference on Guidance, Navigation & Control Systems*. Loutraki, Greece, Oct. 2005 (cit. on pp. 1–3, 5).

[122] S. Nassar. “Improving the Inertial Navigation System (INS) Error Model for INS and INS/DGPS Applications”. PhD thesis. University of Calgary, 2003 (cit. on p. 20).

[123] *NATO Standardization Agreement (STANAG)*. Doc. 4294, Edition 1. NATO, 1993 (cit. on pp. 31, 32).

[124] *Navstar GPS Space Segment/Navigation User Interfaces*. IS-GPS-200H. Global Positioning Systems Directorate, 2014 (cit. on pp. 27, 29, 55, 58, 139, 157).

[125] N. Neumann, M. A. Samaan, M. Conradt, and S. Theil. “Attitude Determination for the SHEFEX 2 Mission Using a Low Cost Star Tracker”. In: *AIAA Guidance, Navigation and Control Conference and Exhibit 2009*. AIAA 2009-6279. Aug. 2009 (cit. on p. 154).

[126] *Orion: America’s Next Generation Spacecraft*. NP-2010-10-025-JSC. NASA, Johnson Space Center. Houston, Texas, U.S., 2010 (cit. on p. 4).

[127] B. W. Parkinson and J. J. Spilker Jr. *Global Positioning System: Theory and Applications*. Vol. 1. AIAA, 1996 (cit. on pp. 26, 27).

[128] B. W. Parkinson and J. J. Spilker Jr. *Global Positioning System: Theory and Applications*. Vol. 2. AIAA, 1996 (cit. on p. 12).

[129] N. K. Pavlis, S. A. Holmes, S. C. Kenyon, and J. K. Factor. “The development and evaluation of the Earth Gravitational Model 2008 (EGM2008)”. In: *Journal of Geophysical Research* 117.B4 (2012), pp. 1–38. DOI: 10.1029/2011JB008916 (cit. on p. 43).

[130] M. G. Petovello. “Real-Time Integration of a Tactical-Grade IMU and GPS for High-Accuracy Positioning and Navigation”. PhD thesis. University of Calgary, 2003 (cit. on pp. 3, 37, 101, 116).

[131] M. St-Pierre and D. Gingras. “Comparison between the unscented Kalman filter and the extended Kalman filter for the position estimation module of an integrated navigation information system”. In: *IEEE Intelligent Vehicle Symposium*. 2004. DOI: 10.1109/IVS.2004.1336492 (cit. on p. 36).

[132] M. E. Pittelkau. “Rotation Vector in Attitude Estimation”. In: *Journal of Guidance, Control and Dynamics* 26.6 (Nov. 2003), pp. 855–860. DOI: 10.2514/2.6929 (cit. on pp. 43, 45).

[133] B. Polle et al. “Robust INS/GPS Hybrid Navigator Demonstrator Design for Launch, Re-entry and Orbital Vehicles”. In: *7th International ESA Conference on Guidance, Navigation & Control Systems*. Ireland, 2008 (cit. on p. 5).

[134] M. Ragab, F. M. Cheatwood, S. J. Hughes, and A. Lowry. “Launch Vehicle Recovery and Reuse”. In: *AIAA SPACE 2015 Conference and Exposition*. 2015 (cit. on p. 150).

[135] A. Ramanandan, A. Chen, and J. A. Farrell. “Inertial Navigation Aiding by Stationary Updates”. In: *IEEE Transactions on Intelligent Transportation Systems* 13.1 (2012), pp. 235–248. DOI: 10.1109/TITS.2011.2168818 (cit. on p. 25).

[136] J. Reis. “Real-time GPS Heading Determination”. MA thesis. Lisbon, Portugal: Instituto Superior Técnico, Technical University of Lisbon, 2009 (cit. on p. 150).

[137] R. M. Rogers. *Applied Mathematics in Integrated Navigation Systems*. 2nd Ed. AIAA Education Series. AIAA, 2003 (cit. on pp. 24, 25, 185).

[138] I. Rongier et al. “IMU/GPS Hybrid Navigation Application to Ariane 5”. In: *5th International Conference on Launcher Technology (CNES)*. Centre National d’Études Spatiales. 2003 (cit. on p. 4).

[139] K. M. Roscoe. “Equivalency Between Strapdown Inertial Navigation Coning and Sculling Integrals/Algorithms”. In: *Journal of Guidance, Control and Dynamics* 24.2 (2001), pp. 201–205. DOI: 10.2514/2.4718 (cit. on p. 18).

[140] M. Sagliano, S. R. Steffes, and S. Theil. “Mars Rendezvous Relative Navigation Implementation for SINPLEX”. In: *AIAA/AAS Astrodynamics Specialist Conference*. 2014. DOI: 10.2514/6.2014-4104 (cit. on p. 51).

[141] M. Sagliano, G. F. Trigo, and R. Schwarz. “Preliminary Guidance and Navigation Design for the Upcoming DLR Reusability Flight Experiment (ReFEx)”. In: *69th International Astronautical Congress*. Bremen, Germany, 2018 (cit. on pp. 7, 71, 81, 150).

[142] M. A. Samaan and S. Theil. “Development of a low cost star tracker for the SHEFEX mission”. In: *Aerospace Science and Technology* 23.1 (2011), pp. 469–478. DOI: 10.1016/j.ast.2011.09.013 (cit. on pp. 5, 154).

[143] S. Särkkä. “Unscented Rauch-Tung-Striebel Smoother”. In: *IEEE Transactions on Automatic Control* 53.3 (2008), pp. 845–849. DOI: 10.1109/TAC.2008.919531 (cit. on p. 154).

[144] P. G. Savage. “Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms”. In: *Journal of Guidance, Control and Dynamics* 21.1 (1998), pp. 19–28. DOI: 10.2514/2.4228 (cit. on pp. 17, 18).

[145] P. G. Savage. “Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms”. In: *Journal of Guidance, Control and Dynamics* 21.2 (1998), pp. 208–221. DOI: 10.2514/2.4242 (cit. on pp. 17, 18, 43).

[146] M. Schlotterer. “Navigation System for Reusable Launch Vehicle”. In: *31st Annual AAS Guidance and Control Conference*. 2008 (cit. on pp. 2, 13, 150).

[147] M. Schlotterer. “Robuste Schätzung und Sensorfusion zur Navigation von wieder verwendbaren Raumtransportern”. In German. PhD thesis. Bremen University, 2008 (cit. on pp. 2, 98).

[148] S. Schmidt. “Application of State-Space Methods to Navigation Problems”. In: *Advances in Control Systems* 3 (1966), pp. 293–340. DOI: 10.1016/B978-1-4831-6716-9.50011-4 (cit. on p. 37).

[149] R. Schwarz et al. “Overview of Flight Guidance, Navigation, and Control for the DLR Reusability Flight Experiment (ReFEx)”. In: *8th EUCASS, Conference on Reusable Systems for Space Access*. Madrid, Spain, 2019 (cit. on pp. 7, 71, 81, 150).

[150] U. Sezen, F. Arikan, O. Arikan, O. Ugurlu, and A. Sadeghimorad. “Online, automatic, near-real time estimation of GPS-TEC: IONOLAB-TEC”. In: *Space Weather* 11.5 (2013), pp. 297–305 (cit. on p. 157).

[151] N. El-Sheimy, H. Hou, and X. Niu. “Analysis and Modeling of Inertial Sensors Using Allan Variance”. In: *IEEE Transactions on Instrumentation and Measurement* 57.1 (Jan. 2008), pp. 140–149. DOI: 10.1109/TIM.2007.908635 (cit. on p. 186).

[152] D. Simon. “Kalman filtering with state constraints: a survey of linear and nonlinear algorithms”. In: *IET Control Theory and Applications* 4.8 (2009), pp. 1303–1318. DOI: 10.1049/iet-cta.2009.0032 (cit. on p. 25).

[153] D. Simon. *Optimal State Estimation*. Wiley & Sons, 2006 (cit. on pp. 3, 37, 81, 131).

[154] J. Simpson et al. “Testing of the International Space Station and X-38 Crew Return Vehicle GPS Receiver”. In: *12th International Technical Meeting of the Satellite Division of The ION*. 1999 (cit. on p. 4).

[155] M. Sippel et al. “Progress in the Design of a Reusable Launch Vehicle Stage”. In: *11th AIAA/AAAF Space Planes and Hypersonic Systems*. Orleans, France, 2002 (cit. on p. 150).

[156] S. Slivinsky, C. Nesbit, C. Bartone, R. Phillips, and R. Rexrode. “Development and Demonstration of a Ballistic Missile Range Safety Technology System”. In: *Journal of the Institute of Navigation* 49.2 (2002), pp. 91–102. DOI: 10.1002/j.2161-4296.2002.tb00258.x (cit. on p. 4).

[157] P. D. Solomon, J. Wang, and C. Rizos. "Latency Determination and Compensation in Real-Time GNSS/INS Integrated Navigation Systems". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science* 38.1 (2011) (cit. on pp. 14, 131).

[158] J. J. Spilker Jr. "Global Positioning System: Theory and Applications". In: ed. by B. W. Parkinson and J. J. Spilker Jr. Vol. 1. AIAA, 1996. Chap. Tropospheric Effects on GPS (cit. on p. 32).

[159] S. R. Steffes. "Computationally Distributed Real-Time Dual Rate Kalman Filter". In: *Journal of Guidance, Control and Dynamics* 37.4 (2014), pp. 1064–1068. DOI: 10.2514/1.G000179 (cit. on pp. 14, 36, 132, 133).

[160] S. R. Steffes. "Development and Analysis of SHEFEX-2 Hybrid Navigation System Experiment". PhD thesis. University of Bremen, Apr. 2013 (cit. on pp. 5, 18, 25, 36, 39, 62, 133–135, 137, 138, 154, 155).

[161] S. R. Steffes. "Real-Time Navigation Algorithm for the SHEFEX2 Hybrid Navigation System Experiment". In: *AIAA Guidance, Navigation, and Control Conference*. 2012. DOI: 10.2514/6.2012-4990 (cit. on pp. 14, 20, 25, 26, 30, 37, 43, 62, 64, 98, 131, 133, 139, 154, 155).

[162] S. R. Steffes, M. A. Samaan, M. Conradt, and S. Theil. "Reconfigurable Hardware-in-the-Loop Test Bench for the SHEFEX2 Hybrid Navigation System Experiment". In: *AIAA Modeling and Simulation Technologies Conference*. Portland, Oregon, Aug. 2011 (cit. on pp. 141, 154, 155).

[163] S. R. Steffes, S. Theil, and M. A. Samaan. "Post-Mission Analysis Of Flight Results From The SHEFEX2 Hybrid Navigation System". In: *9th International ESA Conference on Guidance, Navigation & Control Systems*. Porto, Portugal, June 2014 (cit. on pp. 154, 155, 157).

[164] S. R. Steffes, S. Theil, M. A. Samaan, and M. Conradt. "Flight Results from the SHEFEX2 Hybrid Navigation System Experiment". In: *AIAA Guidance, Navigation, and Control Conference*. 2012. DOI: 10.2514/6.2012-4991 (cit. on pp. 54, 131, 153–155).

[165] P. Stoica. "A Test for Whiteness". In: *IEEE Transactions on Automatic Control* 22.6 (Dec. 1977), pp. 992–993. DOI: 10.1109/TAC.1977.1101659 (cit. on p. 69).

[166] Y. Tang, W. Zhong, J. Shou, and W. Hu. "Exploration of BD2/SINS Deeply Integrated Navigation in CZ-7 Launch Vehicle Guidance System". In: *China Satellite Navigation Conference (CSNC)*. Vol. 3. 2014 (cit. on p. 4).

[167] J. H. Taylor. "The Cramér-Rao estimation lower bound computation for deterministic nonlinear systems". In: *IEEE Transactions on Automatic Control* 24.2 (Apr. 1979), pp. 343–344. DOI: 10.1109/CDC.1978.268121 (cit. on p. 81).

[168] M. M. Tehrani. "Ring laser gyro data analysis with cluster sampling technique". In: *SPIE, Fiber Optic and Laser Sensors I*. Vol. 412. 1983. DOI: 10.1117/12.935818 (cit. on p. 186).

[169] P. J. G. Teunissen. "An Integrity and Quality Control Procedure for use in Multi Sensor Integration". In: *Proceedings of the 3rd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS 1990)*. 1990, pp. 513–522 (cit. on p. 104).

[170] P. J. G. Teunissen and A. Kleusberg. "GPS for Geodesy". In: ed. by A. Kleusberg and P. J. G. Teunissen. Springer, 1996. Chap. GPS Observation Equations and Positioning Concepts, pp. 175–218 (cit. on p. 29).

[171] P. J. G. Teunissen and M. A. Salzmann. "A recursive slippage test for use in state-space filtering". In: *Manuscripta Geodaetica* 14 (1989), pp. 383–390 (cit. on p. 116).

[172] S. Theil, M. Schlotterer, M. Conradt, and M. Hallmann. "Integrated Navigation System for the second SHarp Edge Flight EXperiment (SHEFEX-2)". In: *31th Annual AAS Guidance and Control Conference*. AAS 08-012. Breckenridge, Colorado, 2008 (cit. on pp. 151, 154).

[173] S. Theil, S. R. Steffes, M. A. Samaan, and M. Conradt. "Hybrid Navigation System for Spaceplanes, Launch and Re-Entry Vehicles". In: *16th AIAA/DLR/DGLR International Space Planes and Hypersonic Systems and Technologies Conference*. AIAA 2009-7381. 2009 (cit. on pp. 11, 154).

[174] S. Theil et al. “Hybrid Navigation System for the SHEFEX-2 Mission”. In: *AIAA Guidance, Navigation and Control Conference and Exhibit*. AIAA 2008-6991. Aug. 2008. DOI: 10.2514/6.2008-6991 (cit. on pp. 5, 151, 154).

[175] E. H. Thompson, J. L. Farrell, and J. W. Knight. “Alignment Methods for Strapdown Inertial Systems”. In: *AIAA Journal of Spacecraft and Rockets* 3.9 (1966), pp. 1432–1434. DOI: 10.2514/3.28671 (cit. on p. 24).

[176] D. Titterton and J. Weston. *Strapdown Inertial Navigation Technology*. 2nd Ed. The Institution of Electrical Engineers, 2004 (cit. on pp. 2, 14, 16–18, 21, 22, 24, 25).

[177] G. F. Trigo, D. Donas-Boto, C. Silva, and J. E. Sanguino. “Vehicle Heading Estimation Using a Two Low-Cost GPS Receiver Configuration”. In: *IEEE 73rd VTC*. Budapest, Hungary, 2011. DOI: 10.1109/VETECS.2011.5956715 (cit. on p. 150).

[178] G. F. Trigo, B. Maass, H. Krüger, and S. Theil. “Hybrid optical navigation by crater detection for lunar pin-point landing: trajectories from helicopter flight tests”. In: *10th International ESA Conference on Guidance, Navigation & Control Systems*. Salzburg, Austria, 2017 (cit. on pp. 7, 39, 41).

[179] G. F. Trigo, B. Maass, H. Krüger, and S. Theil. “Hybrid optical navigation by crater detection for lunar pin-point landing: trajectories from helicopter flight tests”. In: *CEAS Space Journal* 10.4 (2018), pp. 567–581. DOI: 10.1007/s12567-017-0188-y (cit. on pp. 7, 39, 41).

[180] G. F. Trigo and S. Theil. “Advances in Aerospace Guidance, Navigation and Control”. In: ed. by B. Dolega, R. Glebocki, D. Kordos, and M. Zugaj. Springer, 2018. Chap. Improved Hybrid Navigation for Space Transportation, pp. 295–315. DOI: 10.1007/978-3-319-65283-2_16 (cit. on pp. 7, 30, 39, 62).

[181] G. F. Trigo and S. Theil. “Architectural Elements of Hybrid Navigation Systems for Future Space Transportation”. In: *10th International ESA Conference on Guidance, Navigation & Control Systems*. Salzburg, Austria, 2017 (cit. on pp. 7, 9, 97).

[182] G. F. Trigo and S. Theil. “Architectural Elements of Hybrid Navigation Systems for Future Space Transportation”. In: *CEAS Space Journal* 10.2 (2018), pp. 231–250. DOI: 10.1007/s12567-017-0187-z (cit. on pp. 7, 9, 97).

[183] G. F. Trigo and S. Theil. “Improved Hybrid Navigation for Space Transportation”. In: *4th CEAS Specialist Conference on Guidance, Navigation & Control*. Warsaw, Poland, 2017 (cit. on pp. 7, 17, 28, 36, 39, 62).

[184] G. F. Trigo, S. Theil, J. Vandersteen, S. Bennani, and C. Roux. “Robust Tightly-coupled Hybrid Navigation for Space Transportation”. In: *AIAA Journal of Spacecraft and Rockets* 56.2 (2018), pp. 596–609. DOI: 10.2514/1.A34232 (cit. on pp. 7, 71, 87).

[185] G. Tumino and Y. Gerard. “IXV: the Intermediate eXperimental Vehicle”. In: *ESA Bulletin* 128 (Nov. 2006), pp. 62–67 (cit. on p. 165).

[186] T. Upadhyay, I. Progri, J. Lomas, and J. Blucker. “Precision Relative Navigation for Automated Rendezvous and Docking”. In: *22nd Annual AAS Guidance and Control Conference*. 1999 (cit. on p. 59).

[187] J. Vandersteen, S. Bennani, and C. Roux. “Robust Rocket Navigation with Sensor Uncertainties - Vega Launcher Application”. In: *AIAA Journal of Spacecraft and Rockets* (2017), pp. 1–14. DOI: 10.2514/1.A33884 (cit. on pp. 3, 5, 14, 19, 20, 48, 77, 90, 172, 181).

[188] *Vega Flight VV02 Launch Kit - Second Vega launch from the Guiana Space Center*. Last accessed: 2019-07-15. Arianespace. URL: http://www.arianespace.com/wp-content/uploads/2013/05/VV02-PROBA-V-VNREDSat-1-GB-1.pdf (cit. on p. 165).

[189] *Vega Flight VV03 Launch Kit - Third Vega launch from the Guiana Space Center*. Last accessed: 2019-07-15. Arianespace. URL: http://www.arianespace.com/wp-content/uploads/2015/09/VV03-launchkit-EN2.pdf (cit. on p. 165).

[190] *Vega Flight VV04 Launch Kit - VV04 IXV, Fourth Vega launch from the Guiana Space Center*. Last accessed: 2019-07-15. Arianespace. URL: http://www.arianespace.com/wp-content/uploads/2015/09/VV04-launchkit-GB.pdf (cit. on p. 165).

[191] *Vega User's Manual*. Issue 4, Revision 0. Arianespace. Apr. 2014 (cit. on pp. 5, 165).

[192] *Vega VV04: lancement du démonstrateur de rentrée atmosphérique IXV*. In French. Last accessed: 2019-07-15. Arianespace. URL: http://www.arianespace.com/wp-content/uploads/2015/09/2-11-2015-Lancement-IXV-FR.pdf (cit. on p. 165).

[193] H. Weihs. "Sounding Rockets for Entry Research: SHEFEX Flight Test Program". In: *21st ESA Symposium on Rocket and Balloon Programmes*. Thun, Switzerland, 2013 (cit. on pp. 5, 151, 152).

[194] H. Weihs, J. Longo, and J. Turner. "The Sharp Edge Flight Experiment SHEFEX II, a Mission Overview and Status". In: *15th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*. 2008. DOI: 10.2514/6.2008-2542 (cit. on p. 151).

[195] H. Weihs, J. Turner, and M. Hörschgen. "SHEFEX II - The Next Step within Flight Testing of Re-entry Technology". In: *57th International Astronautical Congress*. Valencia, Spain, 2006. DOI: 10.2514/6.IAC-06-D2.5.03 (cit. on p. 151).

[196] H. Weinberg. *Gyro Mechanical Performance: the Most Important Parameter*. Tech. rep. MS-2158. Analog Devices, 2011 (cit. on p. 21).

[197] J. Wendel, J. Metzger, R. Moenikes, A. Maier, and G. F. Trommer. "A Performance Comparison of Tightly Coupled GPS/ INS Navigation Systems based on Extended and Sigma Point Kalman Filters". In: *Journal of the Institute of Navigation* 53.1 (2006), pp. 21–32. DOI: 10.1002/j.2161-4296.2006.tb00368.x (cit. on p. 36).

[198] J. Wendel, C. Schaile, and G. F. Trommer. "Direct Kalman Filtering of GPS/INS for Aerospace Applications". In: *International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation*. 2001 (cit. on p. 14).

[199] J. Wendel and G. F. Trommer. "Tightly coupled GPS/INS integration for missile applications". In: *Aerospace Science and Technology* 8.7 (2004), pp. 627–634. DOI: 10.1016/j.ast.2004.07.003 (cit. on p. 28).

[200] A. Williams, M. Villa, and J. Puig-Suari. "Platform Independent Launch Vehicle Avionics". In: *28th Annual AIAA/USU Conference on Small Satellites*. 2014 (cit. on p. 4).

[201] B. Willms. "Space Integrated GPS/INS (SIGI) Navigation System for the Space Shuttle". In: *18th Digital Avionics Systems Conference*. 1999 (cit. on p. 4).

[202] D. P. Woodbury. "Accounting for Parameter Uncertainty in Reduced-Order Static and Dynamic Systems". PhD thesis. Texas A&M University, 2011 (cit. on p. 37).

[203] D. P. Woodbury and J. L. Junkins. "On the Consider Kalman Filter". In: *AIAA Guidance, Navigation and Control Conference*. Toronto, Canada, 2010. DOI: 10.2514/6.2010-7752 (cit. on pp. 3, 37).

[204] L. Xing, Z. Xiong, J. Liu, W. Luo, and Y. Yue. "Offline Calibration for MEMS Gyroscope G-sensitivity Error Coefficients Based on the Newton Iteration and Least Square Methods". In: *The Journal of Navigation* 71.2 (2018), pp. 352–370. DOI: 10.1017/S0373463317000625 (cit. on p. 21).

[205] R. Zanetti, K. J. DeMars, and R. H. Bishop. "Underweighting Nonlinear Measurements". In: *Journal of Guidance, Control and Dynamics* 33.5 (2010). DOI: 10.2514/1.50596 (cit. on pp. 36, 62).

[206] J. Zhou, S. Knedlik, and O. Loffeld. "Sequential Processing of Integrated Measurements in Tightly-coupled INS/GPS Integrated Navigation System". In: *AIAA Guidance, Navigation, and Control Conference*. 2010. DOI: 10.2514/6.2010-8190 (cit. on p. 28).

Listings

List of Figures

List of Tables

Acknowledgements

The project behind this dissertation was first proposed by Samir Bennani at the Guidance, Navigation and Control Section of the European Space Agency; he, along with Stephan Theil at the Institute of Space System of the German Aerospace Center, and Christof Büskens at the Optimization and Optimal Control Working Group of the University of Bremen, set up the cooperation, through ESA's Network Partnership Initiative (NPI), that funded and supported this work. I am deeply grateful to them for entrusting me with this topic and for paving a smooth road for my research path.

The majority of the work herein was carried out at DLR, in Bremen, under supervision of Stephan Theil and within the Navigation Systems group led by Hans Krüger. Their vast experience and constant support have made this research far better. Like them, many others at DLR left their mark in this project: René Schwarz, Michael Dumke and Piyapat Saranrittichai gave invaluable IT and hardware support; Stephen Steffes, Bolko Maass, David Heise and Ansgar Heidecker (among others) offered, on countless occasions, enlightening discussions on inertial sensors, filtering and estimation; Oliver Montenbruck, Markus Markgraf and Benjamin Braun provided crucial support on the Phoenix GPS receiver and general advice on GNSS topics; and, finally, Federico Fumenti, Marco Sagliano, Emre Arslantas and Svenja Woicke kept my morale high throughout our shared PhD journeys. A very special thank-you to all of them.

Another word of gratitude is owed to Jeroen Vandersteen, who, together with Samir Bennani has helped immensely in guiding this research, especially during my visiting period with the Guidance, Navigation and Control Section of ESA at ESTEC in Noordwijk, The Netherlands. Christophe Roux, of ELV-AVIO, has also provided important, and much appreciated, insight and data during this period.

Near the end, Matthias Knauer, Kai Wah Chan and Kira Lara Löper at the Optimization and Optimal Control Working Group, University of Bremen, and Andreas Rademacher at the Modelling and Scientific Computing Working Group, University of Bremen, helped organize and/or served in the committee of this dissertation's defense. I am thankful to them.

Along the way, many others have eased the trail I trod. Friends and family were constant sources of encouragement. The examples of perseverance set by my parents and grandparents inspired me and kept me grounded and focused. Their support propelled me. And then there was Alice; her love and understanding carried me through. Our *piccola* Elisa gave me the very final push and made it all worth it. I am profoundly indebted to all of you.

A todos o meu mais sincero obrigado.