



Navigation Algorithm-Agnostic Integrity Monitoring based on Solution Separation with Constrained Computation Time and Sensor Noise Overbounding

Vasileios Bosdelekidis¹ · Torleiv H. Bryne¹ · Nadezda Sokolova^{2,1} · Tor A. Johansen¹

Received: 20 January 2022 / Accepted: 8 July 2022
© The Author(s) 2022

Abstract

Integrity monitoring (IM) in autonomous navigation has been extensively researched, but currently available solutions are mainly applicable to specific algorithms and sensors, or limited by linearity or 'Gaussianity' assumptions. This study investigates a Solution Separation (SS) based framework for universal IM, scalable to multi-sensor fusion as each hypothesis assumes a whole sensor measurement set as faulty. Architecturally we consider that: 1) multi sensor systems must account for various sensor noise models which lead to inconsistent estimates of uncertainties, 2) a module must be able to detect sensor failure or sensor noise mismodeling and suggest better bounds for the error, without being constantly conservative, 3) some algorithms are computationally heavy to monitor in the SS setting or the provided covariances cannot be interpreted in IM. A hybrid SS architecture can be practical, where some solutions are evaluated with a navigation algorithm with known characteristics, although the all-sensor-in solution is evaluated with the monitored algorithm. Experiments are run on filter and smoothing-based navigation algorithms. In addition, we experiment with hybrid SS monitoring and time-correlated noise to evaluate the appropriability of our framework in the context of the above-mentioned requirements. This is a novel framework in the IM domain, directly integrable in existing navigation solutions and, in our opinion, it will facilitate the quantification of the effect of different sensors in navigation safety.

Keywords Universal integrity monitoring · Time-correlated noise · Error State Kalman filter · Log likelihood based fault detection · GTSAM

1 Introduction

We currently observe a tremendous interest in the development of autonomous systems tasked to carry out safety-critical operations in varying and challenging conditions. Specific examples with major civilian interest are unmanned cars, surface vessels (e.g. autonomous passenger ferries) and aerial vehicles (e.g. air taxis and ambulances). Their operation may take place in dense urban-environments and the safe interaction with other users is of paramount importance.

At the same time, their navigation should be as much uninterrupted as possible.

The Integrity Risk of a positioning system refers to the probability of an unavoidable state estimation error to result in a situation where the estimate deviates from the true value by more than a certain Protection Level (PL). A report from the European Global Navigation Satellite Systems Agency [1] emphasized that a well-established and trustable framework to set integrity risks and compute PLs is mandatory for any safety critical application that uses position estimates as input. Integrity monitors can contribute to exceptionally safe autonomous system operation and this quantification can result in overcoming the trust burden from humans and achieve better compliance with local regulations. At the same time, they can promote a better understanding of the trade-off between safety and system availability [2, 3] and eventually motivate the development of more efficient navigation solutions.

✉ Vasileios Bosdelekidis
vasileios.bosdelekidis@ntnu.no

¹ Department of Engineering Cybernetics, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway

² SINTEF, 7034 Trondheim, Norway

IM is a well-studied topic in safety-critical applications, e.g. in aviation. The concept of IM has been specially developed and extensively applied to Global Navigation Satellite Systems (GNSSs). On the user side, the most widely used algorithm is called Receiver Autonomous IM (RAIM). RAIM consists of an independent Fault Detection module that evaluates the consistency of the positioning solution by using redundant satellite ranging measurements. A second function of RAIM is the determination of PLs which are upper bounds on the positioning errors that the Fault Detector is not expected to detect. There are also numerous studies that attempted to transfer the knowledge gained in the aviation industry to land applications. Many of them are inspired by the RAIM method. There are two popular types of methodologies: SS and Residual-Based (RB) RAIM. In the former case several faulty measurement hypotheses are constructed, and PLs are computed using the same number of test statistics as hypotheses, while in the latter method a single RB test statistic is used [4]. An interesting example of SS-based IM is in feature-based navigation, e.g. landmark-based with camera or laser sensors, where multiple association hypotheses are used to evaluate the impact of errors to Feature Extraction or Data Association [5]. One of the interesting variations of RAIM that has also shown to be applicable to multi-sensor navigation is called Particle RAIM which uses a particle filter to form a multi-modal probability distribution over position [6].

Most of the research on the development of an IM exclusively targets a specific sensor type. However, modern autonomous systems utilize advanced sensor fusion algorithms, exploiting auxiliary non-GNSS sensors like cameras or lasers together with the standard inertial or GNSS sensors. Most of the works that investigate the integrity of multi-sensor systems are constrained by linearity assumptions (e.g. the research of Meng and Hsu [7]), or they attempt to develop an algorithm that performs both the estimation and IM collaboratively (e.g. the research of Mohanty et al. [6]). It is desirable for a universal IM framework to work with non-linear systems, that are commonly used today, and be independent from the navigation algorithm. To the best of our knowledge, research around a universal IM solution is still lacking. In this study we extend a SS method for IM to remove the linearity and Gaussian noise assumption. We evaluate our method with a filtering-based, as well as a smoothing-based, navigation algorithm.

The development of a universal solution must account for the particular error models of various sensors. Fault detection is a subprocess of an IM tasked to identify and isolate a faulty sensor, allowing the navigation system to continue its operation. Nevertheless, no widely adopted approach exists to account for time-correlated errors with unknown parameters. Our developed IM adopts a method by Crespillo et al. [8] to account for time-correlations by introducing

overbounding hypotheses in a SS context. These hypotheses are used conditionally, and the decision is made automatically by a log-likelihood based algorithm. This leads to more reliable IM with tighter Protection Levels in comparison to methods with continuous overbounding.

We introduce an IM that is independent from the navigation algorithm, which enables easier integration with existing solutions. On a high-level, we utilize a SS methodology, where a sensor is completely excluded in each hypothesis. It is worth mentioning that some navigation algorithms are computationally heavy to execute repeatedly in a SS setting, or do not provide absolute covariances, or do not provide covariances at all. We experimented with a smoothing-based navigation algorithm. In this algorithm, we optionally use an Error State Kalman Filter (ESKF) implementation to evaluate faulty hypotheses, independently of the smoothing-based navigation algorithm used in the all-source-in hypothesis.

For our experiments we selected to use one simulated UAV trajectory, and one real trajectory from a car driven in an urban environment. First, we evaluated the proposed IM on a ESKF implementation. We are confident that ESKF is a proper choice of a modern non-linear estimator, in comparison to the standard Extended Kalman Filter (EKF). We also evaluated the IM on an algorithm based on Factor Graph Optimization, which is a very popular algorithm in the domain of smoothing-based navigation.

The main contribution of our work is a novel IM architecture, which is navigation algorithm-agnostic, in terms of internal architecture as well as time complexity, and is scalable to various sensors and sensor noise models. A limitation is that the sensors should be capable to provide independent position solutions. In addition, the presence of a fault or, in the general case, the exclusion of a sensor can lead to challenges with sensor initialization or the calibration process, e.g. of a visual sensor. The sensor initialization or calibration of a sensor on the run is not addressed in this paper. More specifically, the contributions are the following:

- We demonstrate that a SS IM algorithm, working independently of any navigation algorithm, can employ statistics on Log Likelihoods of various measurement subsets: such metrics can identify presence of measurement faults as well as improve the understanding of underestimated uncertainties provided by the navigation algorithm. We propose a simple statistic based on the rolling standard deviation of Log Likelihood (LLRSD) of each subset that can accurately detect inconsistencies in time, instead of the already studied but more complex Log Likelihood Ratio (LLR) [9]
- Multi-sensor fusion must account for sensor noise models which lead to underestimated uncertainties and invalid bounds. We propose overbounding the common case of time-correlated noise. We observe that overbound-

ing techniques, previously derived for Kalman Filter (KF)-like methods, provide promising results with our SS-based architecture with ESKF backend, even when the knowledge of noise model parameters is approximate. This is an empirical result and we do not provide theoretical guarantees of overbounding. Nevertheless, the important contribution of our framework is that accounting for a different sensor noise model is as easy as constructing hypotheses to apply appropriate overbounding techniques. This is useful, for example, when a sensor's noise distribution has been learned offline under nominal or non-nominal conditions.

- There is wide variety of sensor fusion navigation algorithms and in some cases position covariances are unavailable or cannot be interpreted in the absolute coordinate system. In these cases the IM might fail. In addition, it can be inefficient to execute a computationally heavy navigation algorithm multiple times in the SS setting. It seems beneficial to incorporate a standardized algorithm for the evaluation of some hypotheses. We include results for a GTSAM algorithm for IMU and GNSS fusion, whose monitoring was done with and without the ESKF for evaluation of faulty hypotheses.

The remaining part of the paper is organized as follows. Section 2 presents related works. Section 3 introduces our proposed IM architecture (Section 3.1) and the methods for PL computation (Section 3.2), correlated noise overbounding (Section 3.3) and the Fault Detection statistics (Section 3.4). The data used for evaluation of our IM and the experimental results are discussed in Section 4. Section 5 concludes the paper.

2 Related Work

2.1 On SS and the Benefits of SS vs RB RAIM

Joerger et al. [4] designed a fault detector and a non-least-squares estimator in RAIM. In the first part of their work they focused on multi-measurement faults and that is why they derived analytical expressions of the worst-case fault magnitude and direction, along with an expression for the PL. By a comprehensive comparison they proved the superiority of SS in comparison to RB in terms of the detection statistic, something that is justified by the fact that SS detection is directly related to the faulty measurement hypotheses. In addition, the non-least-squares estimator, that they developed in the second part of their work, achieved the optimization of the integrity risk, at the cost of lower accuracy. A formulation of both the integrity and continuity risk was made in the subsequent work of Joerger et al. [2], and from there they defined simple formulas for both the single and

multiple measurement fault case. Their theoretical proofs are very important in the IM domain and their error bounding formulas are used also by us. In SS the corresponding formulations are derived in the positioning domain. This makes SS more appropriate for a universal IM as it facilitates scalability to multiple sensors and independence from the navigation algorithm.

2.2 On Multi-Sensor IM

A large number of available navigation platforms and algorithms are multi-sensor based. However, multiple modalities of the various measurements, as well as the necessity for recursive estimation place many challenges to the classical IM methods. Multi-sensor IM has gained significant interest but, to the best of our knowledge, proposed solutions in literature maintain some dependence on a specific navigation algorithm.

Gupta and Gao (2019)[10] introduced the Particle RAIM framework. At each time the position is weighted by incorporating its likelihood of generating the subset of measurements. In a subsequent work, Mohanty et al. [6] explored a GNSS and Camera integration with the Particle RAIM framework. Due to the use of the visual sensor, they accounted also for ambiguities during feature and image association, and for errors originating from sensor fusion. The Particle RAIM framework leverages collaborative estimation and IM, and uses the full probability distribution over position, instead of position point estimates. This means that this framework captures very well the uncertainty in the state estimation algorithm.

Meng and Hsu [7] considered the system state propagation as an additional measurement in a KF-based navigation algorithm. This allowed them to calculate the integrity of a recursive algorithm like KF which was impossible with other classic snapshot-based methods. The dependence, however, on KF-based algorithms is quite limiting. Otherwise, for scalability and due to the different nature of measurements, at each hypothesis a whole sensor is considered as faulty and their method works directly on output position solutions. A method to maintain navigation integrity despite the multiple-sensor measurements was developed also by Appleget et al. [11]. They emphasized that traditional SS is not that effective in handling sensor faults that result in insignificant differences in state estimation. For this reason, their algorithm employs the expected value of the covariance of the measurement residuals in a moving average χ^2 -test. Multiple filters are used, and residual covariance distances are calculated for each sensor/ subfilter pair. The residual measurement is based on the EKF's predicted states.

Bhamidipati and Gao [12] utilized a linearized Graph-SLAM framework for worst-case failure mode slope analysis and, subsequently, IM when there are multiple faults in

both GNSS and vision. The visual observations are used to distinguish Line Of Sight (LOS) and Non Line Of Sight (NLOS) satellites by recognizing the sky, formulating the GNSS measurement covariances. The state vector updates are done before performing Fault Detection & Isolation, which depends on the comparison with an empirical distribution of measurement residuals. Their method is capable to monitor multiple faults in both GNSS and vision. Nevertheless, their method was developed for specific sensors, applies measurement preprocessing, and performs estimation and IM collaboratively. Thus, it is not directly extendable to other sensor types or navigation algorithms.

2.3 On Sequential Estimation Navigation Integrity

Recursive IM has been a topic of interest for many researchers, beyond those who investigated multi-sensor fusion. For example, Arana et al. [13] presented a multi-hypothesis method that can monitor multiple landmark association faults at different times. They avoid assumptions in the nature of faults, as integrity is evaluated under the worst combination of sensor faults. The sequential method of Tanil et al. [14] used the innovation sequence from a single KF for fault detection before IM. For fault isolation, they proposed that the sub-set solutions can still be extracted from the full-set solution without the need to run the sub-filters. However, neither of the methods are navigation algorithm-independent, as they are based on the KF innovation vectors sequence. Nonetheless, both methods accounted for the fact that the estimation error and the fault detection are affected by faults back in time, and this is an important consideration also in our method.

2.4 On Fault Detection and Threshold Optimization

Fault detection and isolation is an important module in the IM process which improves the navigation solution's reliability. In the presence of redundant measurements, it is typical to identify measurement faults by comparing positioning solutions when using the full measurement set and different measurement subsets. Faults that can be isolated will be excluded and will not be considered in the integrity risk evaluation.

He et al. [9] highlighted that a detection statistic should account for both optimal availability and integrity, and it should be independent of underlying statistical assumptions. Taking this into consideration, they tested a nonlinear optimization algorithm for the fault detection bound, using the cumulative LLR statistic. LLR and most other Log-Likelihood based statistics do not rely on the internal procedure of a navigation method, but only on the measurement set, the position state estimations and the uncertainties. We selected to incorporate a simpler statistic that relies on the rolling

standard deviation of maximum Log Likelihoods of each measurement subset for our Fault Detection module, and we include comparison results with the LLR statistic.

Jurado et al. [15] were interested in presenting a Solution Separation-based FD method without the assumptions of simultaneously redundant, synchronous sensors with valid measurement models and without constraining the statistical distribution of the faults. In the fault detection procedure, they refrained to rely on competing distributions (e.g. under all-source and faulty hypotheses) but rather they utilized a residual-based statistic under each hypothesis and each sensor. Finally, the authors extended their method to multiple fault detection. However, the FD statistic depends on a state covariance matrix from the navigation algorithm which is a limitation in comparison to our SS-based method with standardized ESKF in backend.

2.5 On Integrity Monitoring of Smoothing-Based Navigation

IM of Visual localization algorithms is quite challenging as there is a vast amount of measurements (e.g. landmarks and features), and it is common that more than one measurement are faulty at each time [16]. Visual localization is a typical example of applications that are better solved with optimization instead of filter-based methods. It is therefore essential to test IM methods with an optimization-based navigation approach too. In the existing literature, an IM method targeted to a nonlinear pose optimization problem was developed by Li and Waslander [16]. They emphasized that their method cannot guarantee that a proposed bound will always be valid, as is true for any nonlinear system with outliers. They employed a variation of the Parity Space Approach as statistical tests to remove multiple outliers for a batch of measurements, which eventually leads to PL calculation. An assumption in this test is that the noise of the measurement model follows a Gaussian distribution. One central contribution of our method is that the sensor measurement noise is not necessarily Gaussian, and we execute experiments where at least one sensor has time-correlated noise.

3 Proposed Method

In this section we describe the proposed IM method, where Fig. 1 outlines the high-level architecture. Previous literature has shown that a method that does collaborative estimation and IM is more efficient (position errors bounded reliably and tightly). However, such highly integrated methods seem inappropriate when we are interested in universality and easy adaptation in existing navigation solutions. We have therefore used a method that is independent of the navigation algorithm. An additional prerequisite for the universality of

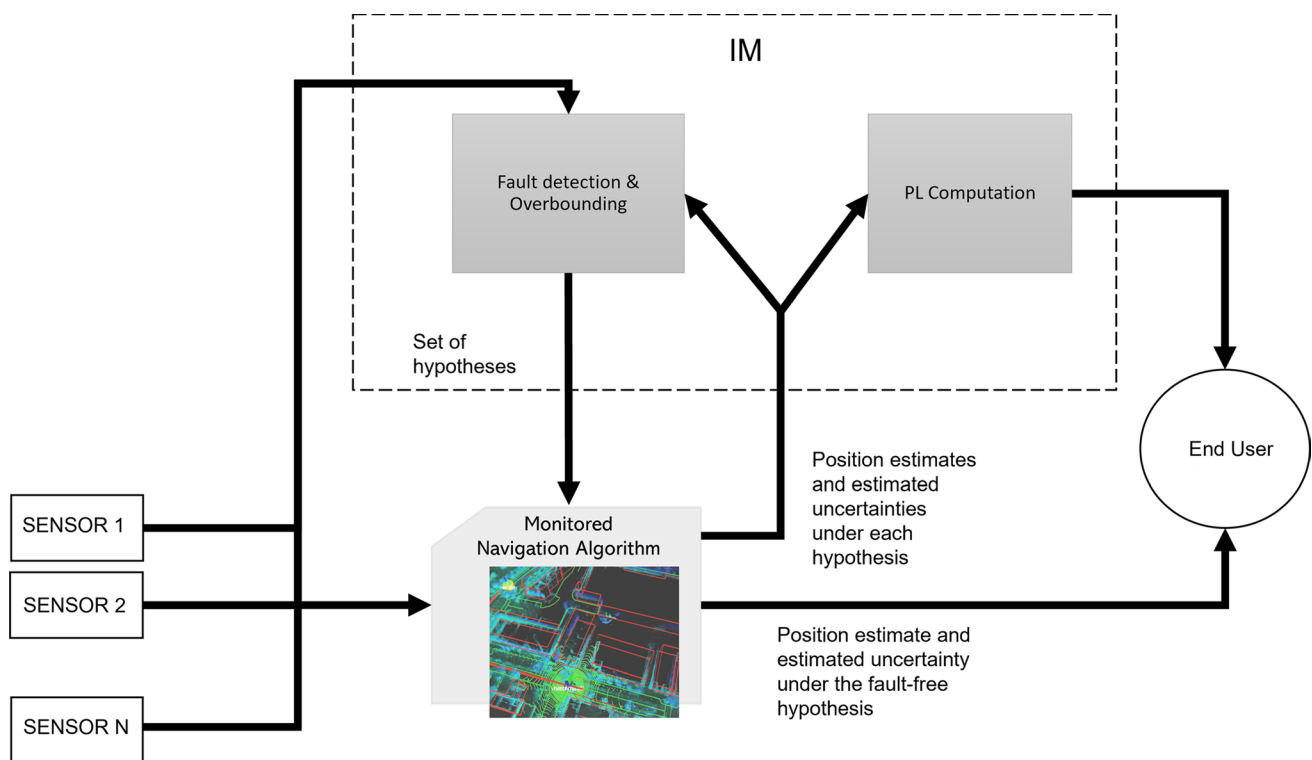


Fig. 1 Architecture of the proposed IM framework

our method is to facilitate the computation of reliable PLs in the presence of different noise models. A commonly used technique is to inflate the process noise covariances in a way to guarantee that the estimated state uncertainty overbounds the true error. We introduce the concept of overbounding hypotheses, where the estimator is run with the inflated noise covariances. These hypotheses can be employed conditionally if a sensor appears to be inconsistent (see Section 3.4). We experimented with overbounding hypotheses of time-correlated noise, which is common for many sensors (see Section 3.3). In our architecture it is straightforward to extend the system with techniques dedicated to other noise models by evaluating and computing bounds when the sensors have different noise parametrization, that is, with additional hypotheses.

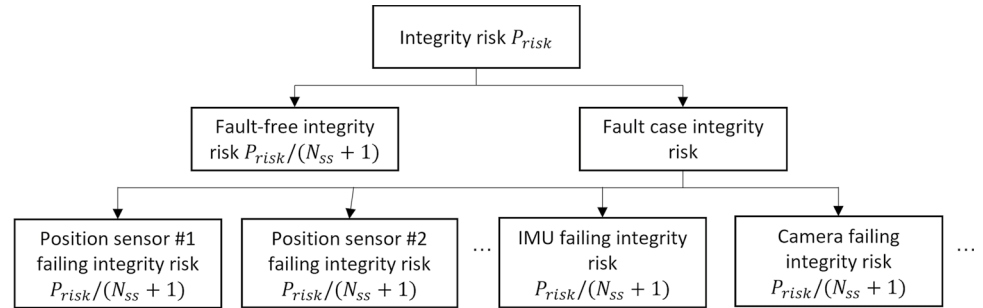
The method belongs to the class of SS methods operating in the position domain. In SS methods the estimation algorithm runs each time for a subset of the input measurements. The different position estimates are compared together and with the all-source estimate (estimate after using the whole set of measurements). In this way we evaluate if some measurements are outliers and are correctly hypothesized as faulty. It helps also to quantify the uncertainty that is introduced by various measurements, resulting in more accurate PL calculation. An important assumption that is often made in SS-based IM is that at each time there is no more

than one fault in sensor measurements. However, it has been emphasized in previous literature that the maximum number of simultaneous faults related to different sensors might vary a lot. A typical integrity requirement considered in previous works is in the order of $10^{-7} - 10^{-8}$ failures / hour of operation and agrees with the gold standard for automotive functional safety [17]. In a GNSS-only solution Fu et al. [18] determined that the probability of more than one satellites failing simultaneously is very low and insignificant for such integrity requirements. However, according to them, errors such as those in feature detection or in landmark locations will likely affect more than one vision measurements (e.g. pseudorange and bearings to landmarks).

The current work develops an IM method that works universally and is scalable to a range of sensors. Therefore, the integrity is evaluated at a high level considering loosely integrated sensor measurements. This means that each time we hypothesize a sensor as faulty instead of its individual measurements. Nonetheless, it is not yet well studied how probable is the event of two or more sensors failing simultaneously and we tested the IM method on a system where only up to one sensor is failing at each time.

The next subsections first present this SS strategy which is scalable to multiple sensors. The formulation of the PL calculation follows. Section 3.3 describes how we deal with time correlated noise, a characteristic that is common for

Fig. 2 Example integrity risk allocation on sensor-level



many sensors. Section 3.4 is dedicated to the log-likelihood based fault detection method, with application to overbounding the mismodeled time-correlated noise.

3.1 SS on Sensor-Level

The concept of SS is closely related to the allocation of an integrity risk budget, as the integrity risk tree demonstrates in Fig. 2. Instead of depicting a generic allocation among all the various errors that can contribute, this figure demonstrates the proposed, on sensor-level, allocation. It is worth mentioning that the sensor set in the diagram serves only as an example, and we experimented only with a subset of the mentioned sensors. In the diagram, N_{ss} denotes the number of fault hypotheses. P_{risk} is the predetermined integrity risk requirement. The integrity risk is allocated equally among all hypotheses. This is an arbitrary decision and may result in looser bounds.

SS is a procedure of evaluating the disagreement of the fault-free estimation (the nominal estimation of the navigation algorithm) to the solution corresponding to fault hypotheses. Eventually, in our case, we seek the quantification of this disagreement to identify a faulty sensor, or to estimate a PL that sufficiently bounds the error of a hypothesis. This quantification depends on the output position states and estimated position uncertainty after the execution of the navigation algorithm with the hypothesized fault-free measurement subset as input, therefore, it can be easily integrated with most existing navigation algorithms. Nonetheless, the experiments show that a universal IM should include its own estimation backend, due to two reasons: an algorithm might be computationally heavy to execute multiple times, or the position uncertainty of the algorithm might be incorrect, hard to use or unavailable. The developed multiple-hypotheses IM allows the selection between the native algorithm under monitoring or a standard implementation of another algorithm for the evaluation of some hypotheses, where for the experiments we selected the ESKF.

The computational load that is introduced by a safety monitor is important to consider. Joerger et al. [2] proved that, in contrast to Residual Based (RB) methods, in SS there is no necessity to find the worst case fault vector magnitude

and direction. However, it is required to run the filter multiple times. Nonetheless, parallelization seems feasible in this case.

Another challenge with a SS method is that it strongly relies on the redundancy of measurements among sensors. In our case, this means that at least two redundant measurements are necessary to evaluate the boundaries of a fault-tolerant (to one of these measurements) hypothesis. This indicates that the utilization of the low-level measurements of a sensor might be necessary in some cases. For example, a solution to the redundancy issue, without sacrificing scalability, could be the utilization of the limited number of pseudorange measurements to satellites, along with the single measurements from the non-GNSS sensors. It was mentioned before that the probability of more than one faulty satellite is lower than a typical integrity requirement, satisfying the redundancy requirement.

3.2 PL Computation

PL denotes a position error bound that is determined based on the integrity risk requirement. Mathematically, we will refer to \mathbf{PL} , which is a vector of error bounds, as we work with multi-dimensional navigation. For each direction, it should be guaranteed that the probability of the error exceeding the corresponding bound is smaller than the integrity requirement. That is, under a fault hypothesis i , the \mathbf{PL} should satisfy the following inequality:

$$P(HMI|H_i)P(H_i) < I_{REQ_i} \quad (1)$$

HMI stands for Hazardous Misleading Information and is the event $|\hat{\mathbf{x}}_i - \mathbf{x}| > \mathbf{AL} \ \& \ \mathbf{PL} < \mathbf{AL}$, where $\hat{\mathbf{x}}_i$ is the estimated position state under hypothesis i , \mathbf{x} is the true state, and \mathbf{AL} is the vector of Alert Limits for each dimension. I_{REQ_i} is the preset integrity risk requirement, allocated to the i th hypothesis. Here, as in many works, we select an equal allocation to each hypothesis, something that can result in a looser bound. $P(H_i)$ is the probability of a fault hypothesis and can be computed offline during extensive

simulation runs, but in our work and in most previous works on IM an arbitrary constant value was selected for all fault hypotheses. Then for the fault-free hypothesis we have $P(H_0) = 1 - \sum_{i=1}^{N_{ss}} P(H_i)$.

By adding the contributions of each failure towards the integrity risk it is guaranteed that the overall PLs will bound the error and, therefore, the safety of the method. In the fault-free case, measurements follow a nominal distribution and we assume that the error follows a Gaussian distribution. Joerger et al. [4] derived a simple formula for the **PL** calculation for a fault hypothesis i , obtained for the worst-case fault magnitude, and, in our case, for one of the dimensions it is given by:

$$p_{L_i} = Q^{-1} \left\{ \frac{I_{REQ}}{P(H_i)(N_{ss} + 1)} \right\} \sigma_i + T_{\Delta_i} \tag{2}$$

where Q^{-1} the inverse tail probability of the standard normal distribution, σ_i the standard deviation of the fault-free solution under the i th hypothesis and $T_{\Delta_i} = Q^{-1} \left(\frac{P_{CONT}}{N_{ss}} \right) \sigma_{\Delta_i}$ where P_{CONT} continuity risk allocated to the fault hypotheses and $\sigma_{\Delta_i}^2 = \sigma_i^2 - \sigma_0^2$, with σ_0^2 the variance for the all-source solution.

The intuitive interpretation of Eq. 2 is that if we assume that the position error follows a standard normal distribution then the factors $Q \left\{ \frac{I_{REQ}}{P(H_i)(N_{ss} + 1)} \right\}$ will bound the probability of Hazardous Misleading Information. The second term, T_{Δ_i} , can be considered a fault isolation threshold which has to satisfy the continuity risk requirement.

For the fault-free case we will have $p_{L_0} = Q^{-1} \left\{ \frac{I_{REQ}}{2(N_{ss} + 1)} \right\} \sigma_i$ where we multiply by 2 because both tails of the error distribution need to be accounted for.

3.3 Overbounding Time-Correlated Noise

It is a fact that the noise of some sensors is time-correlated and not white. The estimated covariance from some navigation algorithms will be unreliable unless the correlated noise is accounted for. Subsequently, in many cases the computed PLs will be unreliable.

A first order Gauss-Markov process (GMP) is the random process used in this work to model correlated noise. These processes are fully defined by a time constant τ , steady state variance σ^2 and initial variance σ_0^2 . The discrete first-order Gauss-Markov process is defined with the following equation for the discrete time step ζ [19]:

$$y(\zeta) = \alpha y(\zeta - 1) + u(\zeta) \tag{3}$$

y denotes a random variable (in this case the correlated measurement noise), $u(\zeta)$ a random variable that follows

zero-mean Gaussian distribution with variance σ_u^2 and α is a coefficient in the range $[0, 1]$.

The correlation time T is the lag time corresponding to an autocorrelation value of $1/e$, or:

$$Autocorrelation(y(\zeta), y(\zeta + T)) = \frac{1}{e} \tag{4}$$

For two samples m, n with $m > n$ the autocorrelation of $y(n)$ is

$$\begin{aligned} Autocorrelation(y(m), y(n)) &= a^{m-n} \\ \Rightarrow Autocorrelation(y(\zeta), y(\zeta + T)) &= a^T \end{aligned} \tag{5}$$

By combining it with Eq. 4 we find that: $a = e^{-1/T}$.

Crespillo et al. [8] designed a GNSS + Inertial Navigation System (INS) integration scheme where noise processes are correlated and the parameter values are known only to reside in a (wide) range of values. Langel et al. [20] formulated a way to guarantee that the estimated covariance from a KF overbounds the actual error distribution of the estimate. They based their proof on the propagation of the error matrix $\mathbf{E} = \mathbf{\Sigma} - \mathbf{P}$, where \mathbf{P} is the estimated covariance matrix of the KF and $\mathbf{\Sigma}$ is the covariance matrix to define, so that $\mathbf{\Sigma} \geq \mathbf{P}$. To achieve that, they concluded that the process noise power spectral density matrix \mathbf{Q} should be populated with the upper bound values of its individual elements. We should use the maximum time constant of the range and the upper bound for the uncertain steady-state variance parameter of the correlated noise, inflated by the ratio of the maximum and minimum values of the time constant:

$$\tau = \tau_{max} \tag{6}$$

$$\sigma^2 = \sigma_{max}^2 (\tau_{max} / \tau_{min}) \tag{7}$$

Finally, the initial variance of the GMP is determined in a way to satisfy the condition $\mathbf{E}(0) \geq 0$ which is another prerequisite to have that $\mathbf{\Sigma} \geq \mathbf{P}$ at all $t > 0$:

$$\sigma_0^2 \geq \frac{2\sigma_{max}^2}{1 + (\tau_{min} / \tau_{max})} \tag{8}$$

We can set σ_0^2 to this lower bound. Formal proofs are provided by Langel et al. [20].

This paper provides some empirical results after utilization of this method with the IM with ESKF back-end. Specifically, in the simulation experiments run, a position sensor, that behaves similarly to a GNSS, is used with noise that follows a GMP (see a next section for more details). As the paper by Langel et al. [20] demonstrated, the GMP noise in each sensor can be sufficiently accounted for by augmenting the state vector with additional bias states and a subsequent inflation of the process noise covariance

Table 1 A comparison of the RBT values (see section 4.4) for the two trajectories and for various algorithms or IM setups, with or without the Fault Detection and Overbounding (FDO) module. Here, the measured RBT value for the ESKF navigation algorithm is used as reference for the comparisons at each row. In each case, only the maximum RBT value appearing in any of the dimensions is compared. The results for the two datasets are not directly comparable, as the accuracy of their GT might be different

Trajectory	ESKF		GTSAM		GTSAM + ESKF hypotheses	
	w/o FDO	with FDO	w/o FDO	with FDO	w/o FDO	with FDO
UAV	Maximum RBT for $\tau = 12^2$					
	11.4	+27.8%	+349%	+349%	-10%	+37.3%
	Maximum RBT for $\tau = 60^2$					
	46.96	-37.6%	+445%	+445%	-46.2%	-50%
	Full estimation + IM Avg execution time per step (s)					
0.1183	0.1483	0.2264	0.2718	0.1551	0.1482	
KITTI	Maximum RBT for $\tau = 12^2$					
	15.73	+12.7%	+71%	+71%	+22%	+31.3%
	Maximum RBT for $\tau = 60^2$					
	15.73	+12.7%	+755%	+755%	+22%	+31.3%
	Full estimation + IM Avg execution time per step (s)					
0.07	0.075	0.089	0.117	0.074	0.087	

matrix \mathbf{Q} . Time-correlated noise is just an example of the challenges that can deteriorate the quality of IM, and our proposed architecture is compatible with additional overbounding techniques to handle other challenging cases.

Appendix 1 serves as an introduction to the ESKF implementation and the strategy to use the GMP parameter values determined in Eq. 6, 7, 8 to inflate \mathbf{Q} .

3.4 LLR and LLRSD for Fault Detection

He et al. [9] introduced a method to optimize the fault detection bound for improved integrity and availability. Their method uses the cumulative LLR to identify sensors that are likely faulty. An intuitive understanding of how LLR can be used in fault detection is the following: In the absence of the faulty measurement, someone expects a better agreement of the remaining measurements in comparison to the fault-free hypothesis, i.e. the LLR will be higher in this case. In our method we will see empirically that a simpler metric (only the faulty hypothesis is used in computations), which we call LLRSD, provides a better indication of the times when a measurement is faulty. For completeness the reader may consult Appendix 2 which includes an introduction to the LLR metric. The formulation of the LLRSD metric follows.

For a time interval a to b , let us accumulate the Log-Likelihoods of a measurement subset y^j containing all measurements except the j th. This is the nominator of the LLR metric too (Eq. 27):

$$LL_a^{b,j} = \sum_{k=a}^b \ln p(y_k^j | x_k) \quad (9)$$

Then, similarly to Appendix 2, at each time k get the maximum, $\psi_k^j = \max_{k-x+1 \leq a \leq k} \{LL_a^{k,j}\}$, $0 \leq j \leq m$, with x the

accumulation time window size and m the total number of measurements.

Using a window of configurable size N it is attempted to find how much the maximum log-likelihood of subset j at current time (ψ_k^j) deviates from log-likelihoods at previous times, inside the window.

$$LLRSD_k^j = \sqrt{\frac{\sum_{z=k-N}^k (\psi_z^j - \mu_k^j)^2}{N}} \quad (10)$$

where $\mu_k^j = \sum_{z=k-N}^k \frac{\psi_z^j}{N}$ is the mean of the maximum log-likelihoods in the window.

The experimental results, presented in Sections 4.2.1 and 4.2.2, showed that the risk of fault misdetection is lower when using this metric than when using the LLR metric.

4 Experiments

This section introduces the trajectories, sensor combinations and integration strategies as well as the navigation algorithms that were evaluated with the developed IM method. Table 1 will summarize the IM results for each trajectory and monitored navigation algorithm.

4.1 Data

We verify our method for two trajectories. The first trajectory corresponds to a 3D UAV scenario with a GNSS and an IMU sensor with perfectly aligned reference frames. The noise processes of the sensors are Gaussian and the trajectory was created in MATLAB. This trajectory is given in Fig. 3. The second trajectory includes data collected from a car driving in an urban environment with, among others, measurements from a GNSS receiver and an inertial sensor

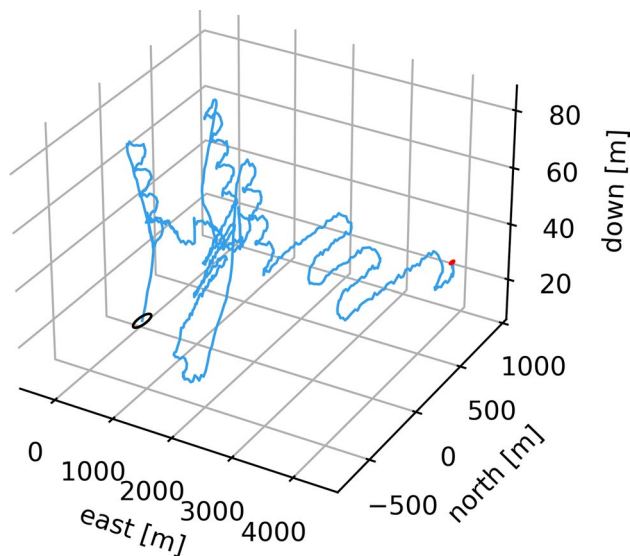


Fig. 3 3D simulated trajectory of a UAV in NED coordinates. The starting and ending positions are indicated with a back circle and a red dot respectively

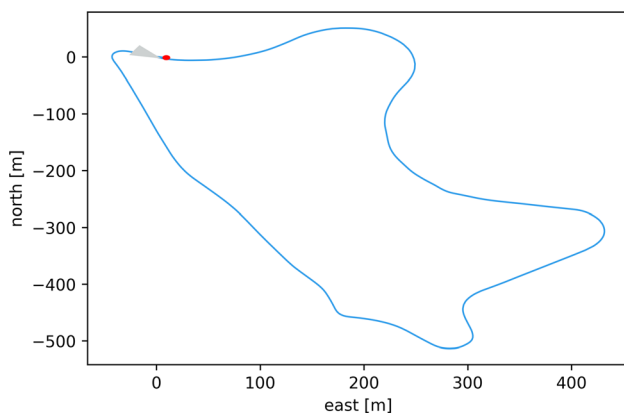


Fig. 4 2D collected data in NED coordinates from a real car drive. Trajectory from the KITTI raw dataset. An arrow indicates the starting position and direction, and the red dot indicates the ending position. Thus, the starting and ending position of the car are almost the same

module. This trajectory is included in the KITTI raw dataset [21]. It is shown in Fig. 4. We limit the experiments to a loose integration of position fix and IMU sensors. The name POS_SENSOR1 is used for the purpose of the text and refers to the GNSS sensor. The next subsection describes the navigation algorithms under evaluation. For the testing of the fault detection method and the overbounding of correlated noise we synthesize the measurements of an additional position sensor called POS_SENSOR2. Specifically, the time correlation between successive position measurement errors is modeled employing a Gauss-Markov process, as described earlier, according to the Eq. 3, and this noise is added around

the measurements of POS_SENSOR1. The parameters of the noise are as follows:

$$T = 300s$$

$$\sigma_u = 0.3m$$

The frequencies of each sensor measurements are:

$$\text{IMU: } 10 \text{ Hz}$$

$$\text{POS_SENSOR1: } 1 \text{ Hz}$$

$$\text{POS_SENSOR2: } 2 \text{ Hz}$$

The difference in frequency between the two position sensors was accounted for in the measurement error matrices by setting the measurement noise covariance matrices as $R_{pos2} = R_{pos1}/0.5$ [22]. The elements in R_{pos1} , that correspond to the measurements of POS_SENSOR1, have been tuned for the simple POS_SENSOR1 + INS integration and based on estimated uncertainty consistency tests.

Figure 5 depicts the measurements of POS_SENSOR1 and POS_SENSOR2.

Both trajectories contain also ground truth (GT) measurements, with respect to the world North-East-Down (NED) frame, which we use for the computation of the estimation error.

4.2 Navigation Algorithms Evaluated with the IM

We evaluated the developed IM on a loosely coupled sensor integration and on two types of navigation algorithms. The first was an ESKF, described in Section 4.2.1, and the second one was GTSAM, which is based on smoothing and is described in Section 4.2.2.

A fault hypothesis is constructed after considering the full measurement set of a sensor as faulty, thus, excluding it. Our SS-based IM executes the navigation algorithm for the original measurement set (all sensors are fault-free) and for the subsets under the fault hypotheses. The IMU is not hypothesized as faulty to ensure redundancy of measurements at each time. Therefore, there are totally three hypotheses in our experiments:

- All-source: all sensors are assumed non-faulty and used in the navigation algorithm
- Out-POS_SENSOR1: exclusion of the POS_SENSOR1
- Out-POS_SENSOR2: exclusion of the POS_SENSOR2

4.2.1 ESKF

The ESKF is a successor of the original KF and the EKF for the cases when either the process or the measurement model or both are nonlinear [23, 24]. In this approach, the

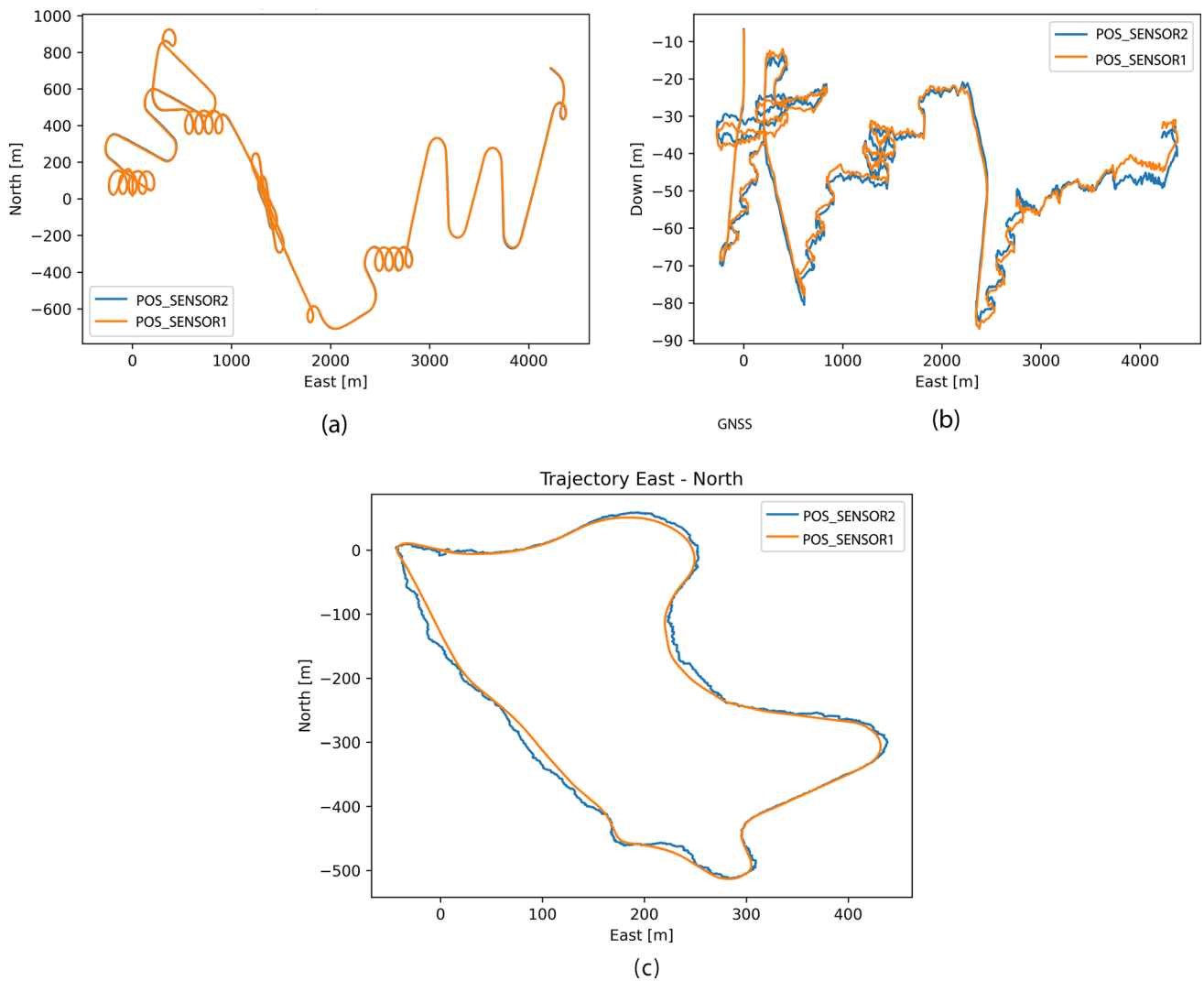


Fig. 5 The measurements of POS_SENSOR1 and POS_SENSOR2. The coordinates were converted to the NED system. (a) Measurements in the 3D UAV trajectory North vs East direction (b) Measure-

ments in the 3D UAV trajectory East vs Down direction (c) Measurements in the 2D KITTI trajectory North vs East direction

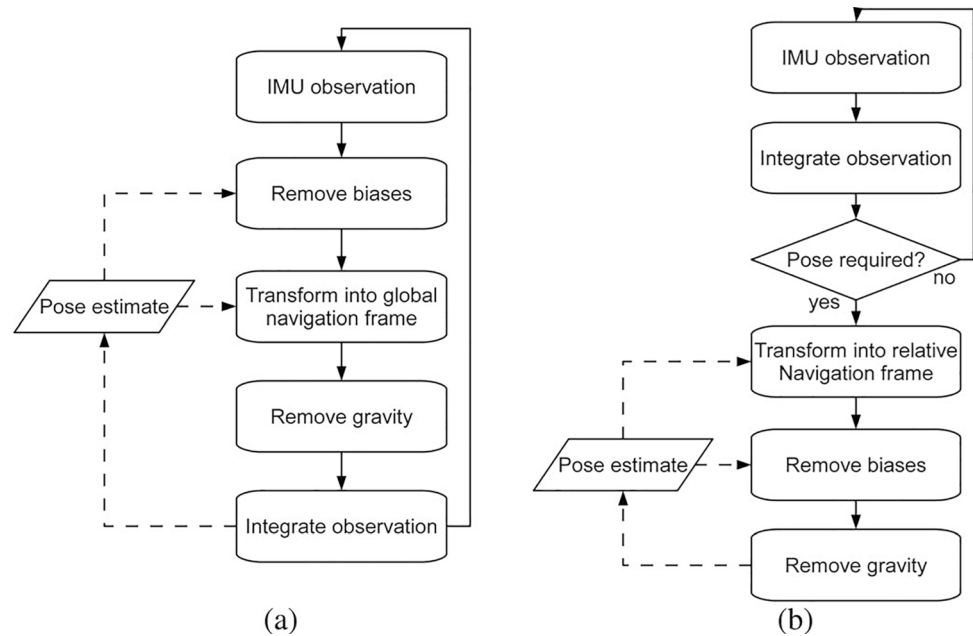
error in the states is estimated using a KF, instead of the state itself. The linearity in the error state dynamics means that the application of KF is feasible. Thus, ESKF takes advantage of KF's optimal estimations [23]. This filter is typically used for the fusion of IMU data with any other kinds of sensors. We use this filter to loosely integrate one or two position sensors with the IMU sensor and validate the proposed IM. Appendix 1 is a brief introduction to the ESKF implementation.

4.2.2 GTSAM

GTSAM is a C++ library that implements smoothing and mapping (SAM) in robotics and vision, using Factor Graphs and Bayes Networks as the underlying computing paradigm rather than sparse matrices [25].

We evaluate our IM with GTSAM with the IMU + position sensor integration. Although this integration is far from being state-of-the-art, it should be a demonstrative example of the straight-forward applicability of our method with more complex integrations used in smoothing-based navigation. In GTSAM we use the implementation of an IMU factor as proposed by Lupton and Sukkarieh [26]. Figure 6 compares the steps followed in standard inertial integration with those implemented in the IMU factor. This architecture allows the reparameterization of the navigation frame and the pre-integration of the inertial observations which facilitates the initialization of the system (all initial conditions become linearly dependent on the estimated states). In addition, the gravity vector must be considered after the observations are already integrated instead of accounting it during integration of the inertial observations in the velocity equation.

Fig. 6 Flowcharts comparing the steps in standard inertial integration (a) with those implemented in GTSAM's IMUFactor (b). Figure from Lupton and Sukkarieh [26]



Like most nonlinear optimization libraries, GTSAM optimizes for a change with respect to a linearization point. This is very important to note, because in the case of GTSAM the covariance matrices are given in relative, and not absolute, coordinates [25].

4.2.3 IM of the GTSAM Navigation Algorithm using the ESKF Backend

Finally, we examine how the IM with ESKF backend behaves in the case of the GTSAM navigation algorithm. The idea here is to use the covariances estimated by ESKF to compute the bounding PLs of the estimation error of GTSAM. Specifically, the optimization procedure in GTSAM is run for the all-sensor input. The ESKF estimates the covariances when leave-one-out sensor input is used (fault hypotheses). This is an alternative implementation of the IM and, as mentioned before, can result in higher efficiency, depending on the output format and time complexity of the monitored navigation algorithm.

4.3 Results and Discussion

4.3.1 ESKF Experiments

Figures 7 and 8 present, for each of the trajectories (2 or 3 dimensions), the computed PL and the position error of the estimations relative to the GT positions and for different integrations of the POS_SENSOR1 and / or POS_SENSOR2 with the IMU. It is reminded that we cannot use multiple-hypothesis monitoring when solely the POS_SENSOR1 or POS_SENSOR2 is used with the IMU due to lack of

redundancy in the measurements. In that case we evaluate only the fault-free hypothesis. The filter has quite good knowledge of the noise of the POS_SENSOR1 and the IMU, although the parametrization is not optimal. In the case of the simulated trajectory, it is observed that the IM fails to calculate PLs that actually bound the error, when the POS_SENSOR2 is used. This is especially visible in Fig. 7c, where the POS_SENSOR1 has been completely excluded. In the case of the KITTI trajectory, we observed that the IM achieved sufficient bounds when SS was employed in the presence of both POS_SENSOR1 and POS_SENSOR2.

Figure 9a shows for the UAV scenario the Pearson correlation coefficient of the LLRSD or the LLR metric with the absolute position error, in the case of the integration of the POS_SENSOR1 + POS_SENSOR2 + IMU and for each hypothesis (in each hypothesis either the POS_SENSOR1 or the POS_SENSOR2 is assumed fault free.). Figure 9b depicts the mean and std values of the LLRSD and LLR metric for each hypothesis, at the times when the PLs do not bound the error. In both cases, there is a weak to medium positive correlation among the metric in hypothesis H1 and the error in hypothesis H0 only on the East axis. There is also a weak negative correlation among the metric in hypothesis H2 and the error in hypothesis H1. The advantage of using the LLRSD metric over the LLR metric is apparent in Figs. 9b where it is straight-forward to identify that H1 is a valid hypothesis. In addition, the standard deviations (shown with the black vertical lines on the bars) are in general lower for the LLRSD metric and the H1 or H2 hypotheses, reducing the probability of fault misdetection. Figure 10 presents the results after applying the overbounding technique exclusively for the uncertainty of the measurements of the position sensors (see Section 3.3

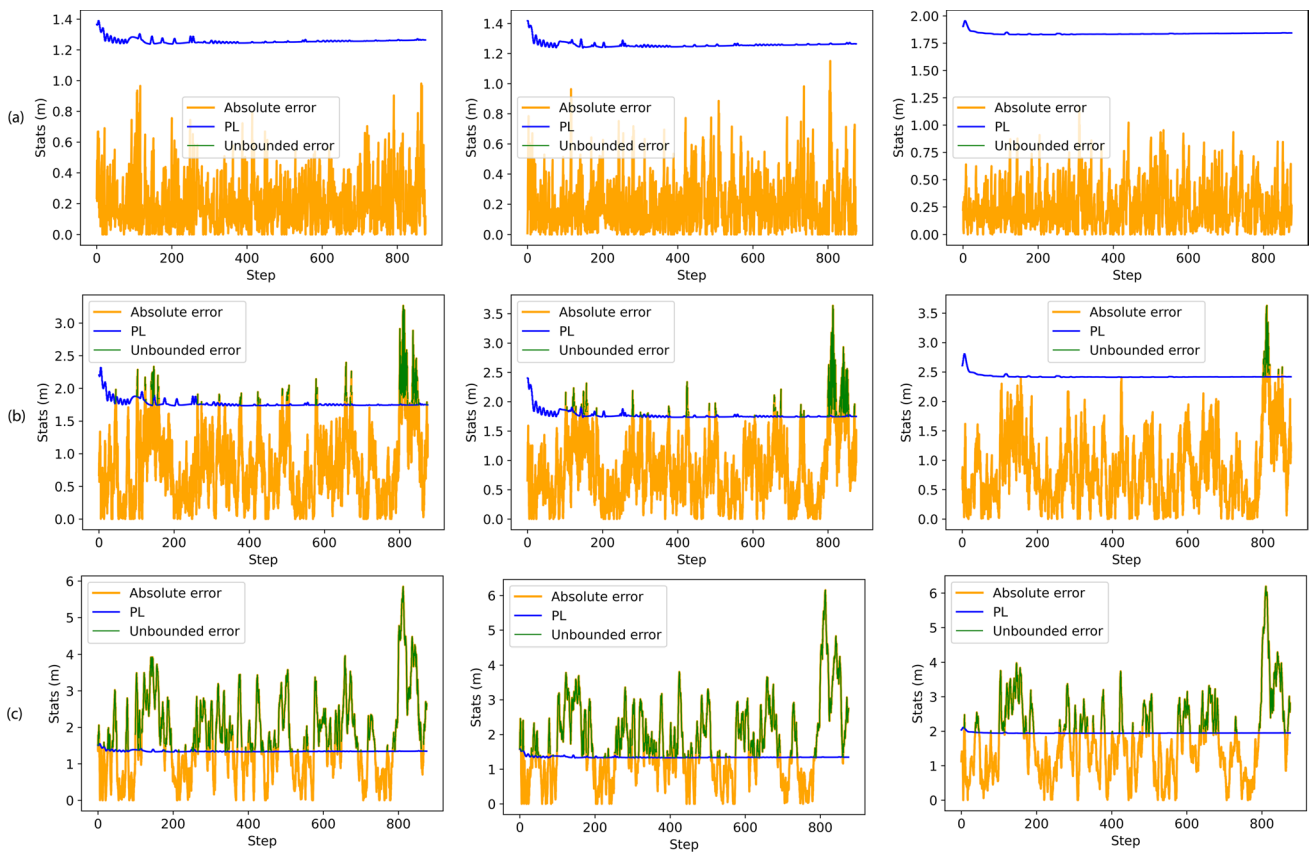


Fig. 7 PL and absolute errors for various settings of POS_SENSOR1+POS_SENSOR2+IMU loose integration with ESKF for the UAV simulated trajectory. We marked with green the parts of the error plot that exceed the corresponding PL. Each column corresponds to one direction, North (left), East (middle), Down (right). (a)

Utilization of measurements only from the POS_SENSOR1 and the IMU. (b) Utilization of measurements from both the POS_SENSOR1 and POS_SENSOR2, along with the IMU. (c) Utilization of measurements only from the POS_SENSOR2

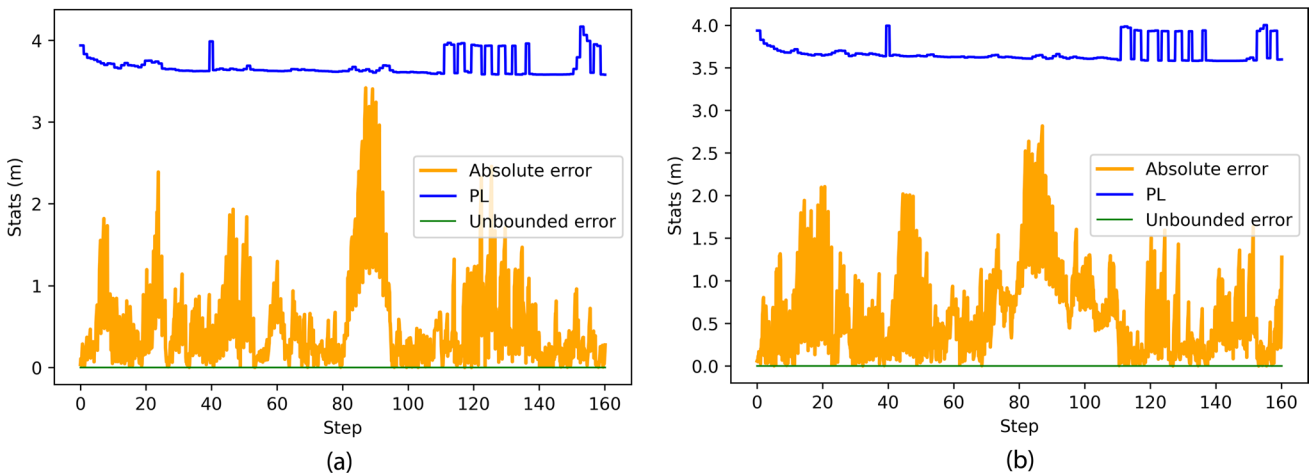


Fig. 8 PL and errors for the POS_SENSOR1+POS_SENSOR2+IMU loose integration with ESKF for the KITTI trajectory. (a) North axis (b) East axis

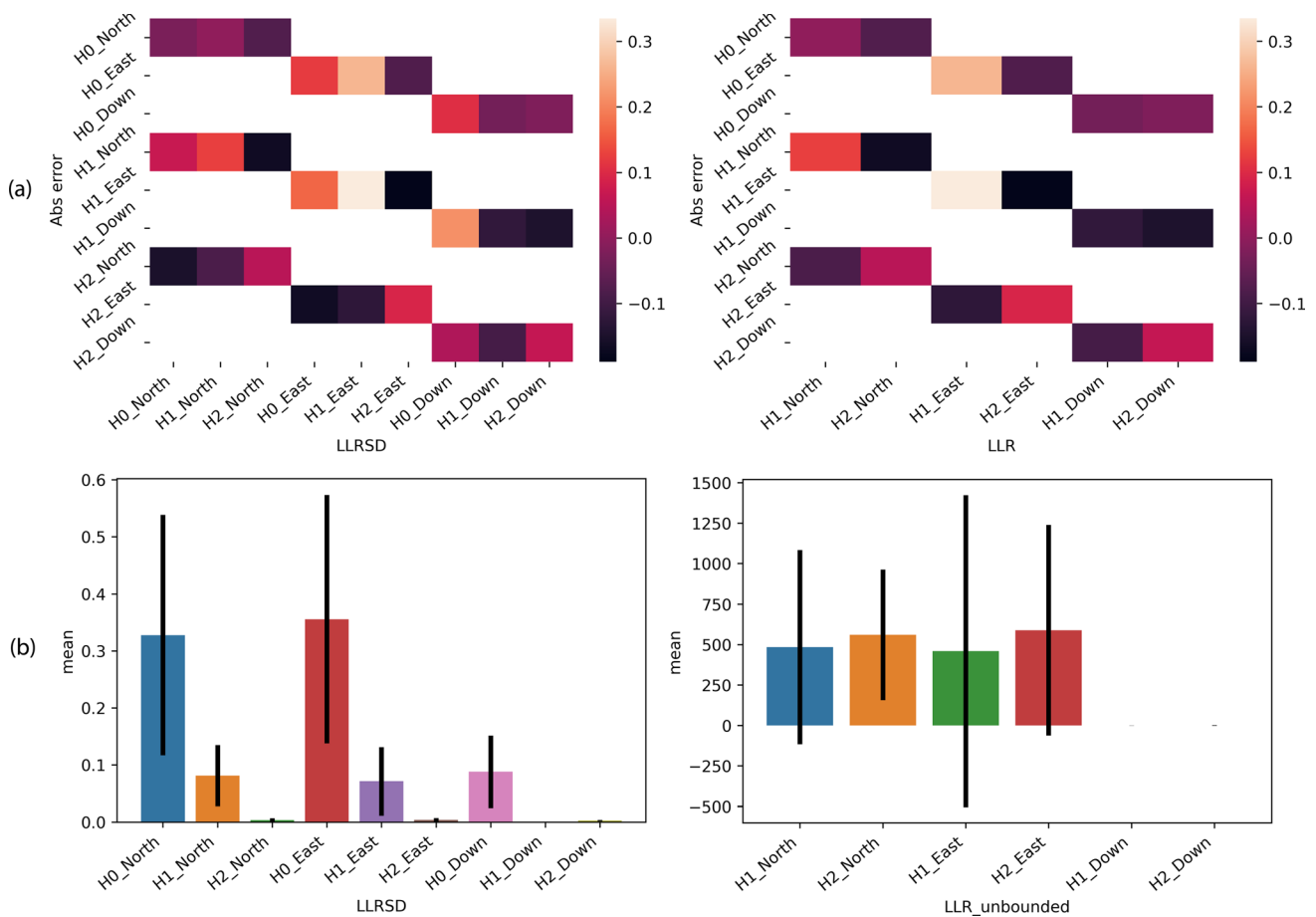


Fig. 9 Comparison between the LLRS and LLR metric for the POS_SENSOR1+POS_SENSOR2+IMU integration in ESKF for the UAV trajectory. The analysis was done for each hypothesis (denoted with H* in the labels) and each axis (North, East, Down) (a) Pearson correlation coefficient among the LLRS or LLR metric and

the absolute position error. (b) Mean and standard deviation of the LLRS or the LLR metric in the regions where bounding failed (see Fig. 7). Each standard deviation is half the length of the black vertical line depicted on top of the corresponding bar

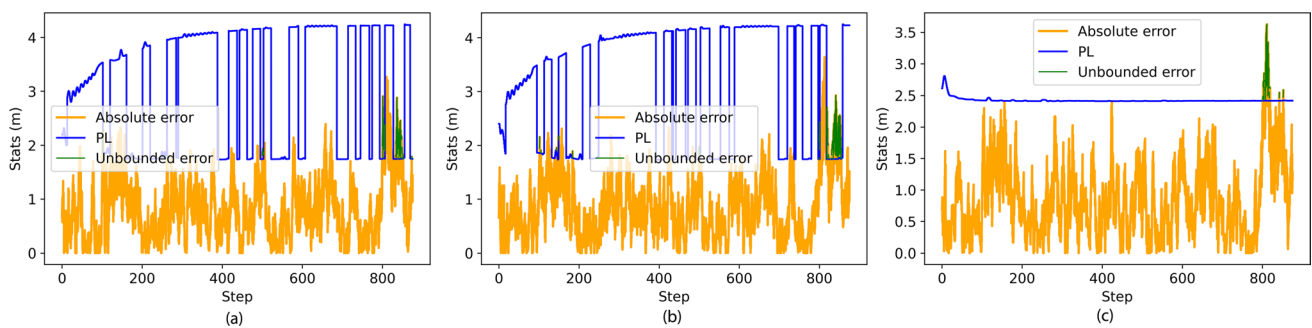


Fig. 10 IM of the ESKF with conditional overbounding for the UAV trajectory. The PLs and the true errors are shown for the (a) North axis (b) East axis (c) Down axis

and Appendix 1), and for the times when the LLRS value exceeds a preset threshold. The threshold used is 0.075 and was selected empirically based on the results for both trajectories. Overbounding means that a higher bound is computed for the hypothesis that includes the faulty sensor. The LLRS

statistic can identify a faulty sensor, with the assumption that at most one sensor is faulty at each time. We confirm that the error is bounded almost all the time, avoiding very loose bounds when not needed.

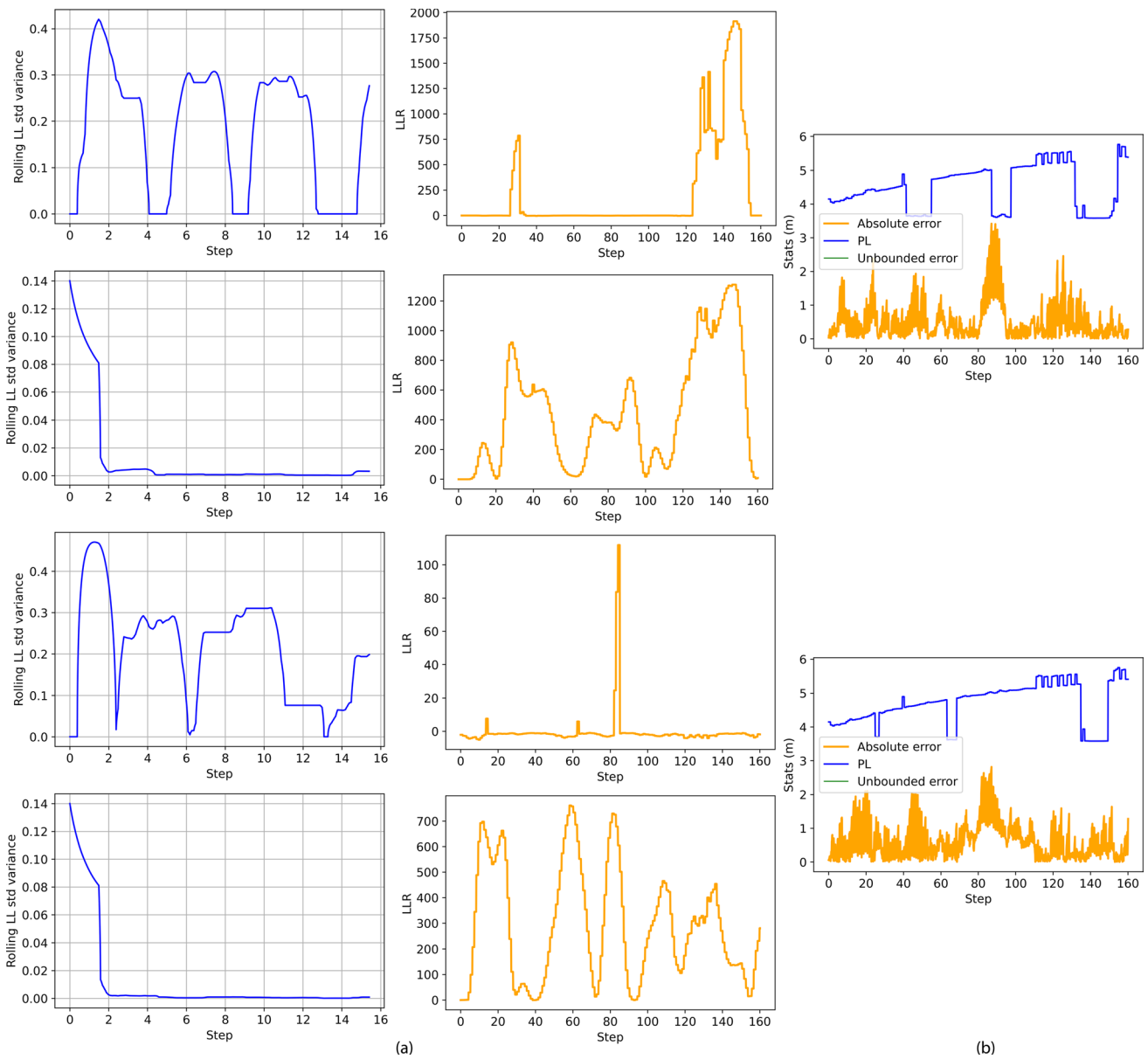


Fig. 11 Evaluation of IM of the ESKF with overbounding for the KITTI trajectory **(a)** LLRSD (left column) and LLR (right column). Each row from top to bottom depicts the results for the following hypotheses and axes: Out-POS_SENSOR1 & North axis, Out-POS_SENSOR2 & North axis, Out-POS_SENSOR1 & East axis, Out-POS_SENSOR2 & East axis **(b)** PL and error for the POS_SENSOR1+POS_SENSOR2+IMU integration after conditional overbounding for the North (top row) and East (bottom row) axes. The decision for overbounding was only based on the LLRSD metric

Figure 11 presents the evaluation of LLR and LLRSD for the KITTI trajectory. The figure and Table 1 show that overbounding did result in unnecessarily looser bounds. This was expected as the monitored algorithm was already configured with large enough measurement uncertainties for the IMU and POS_SENSOR1 sensors. This is an important result that shows that overbounding might deteriorate the IM performance depending on how the tuning of the parameters was done in the first place and how uncertain are the correlated noise parameters.

Out-POS_SENSOR2 & East axis **(b)** PL and error for the POS_SENSOR1+POS_SENSOR2+IMU integration after conditional overbounding for the North (top row) and East (bottom row) axes. The decision for overbounding was only based on the LLRSD metric

4.3.2 GTSAM Experiments

Figure 12 depicts the PLs and errors for the UAV trajectory for the IMU + POS_SENSOR1 + POS_SENSOR2 integration. It is apparent that it is overconfident to use the covariance returned from the GTSAM software. However, the LLRSD is still a simple way to detect that the noise of POS_SENSOR2 is mismodeled. This is confirmed by Fig. 13 which shows the mean and standard deviation of the LLRSD and LLR metrics.

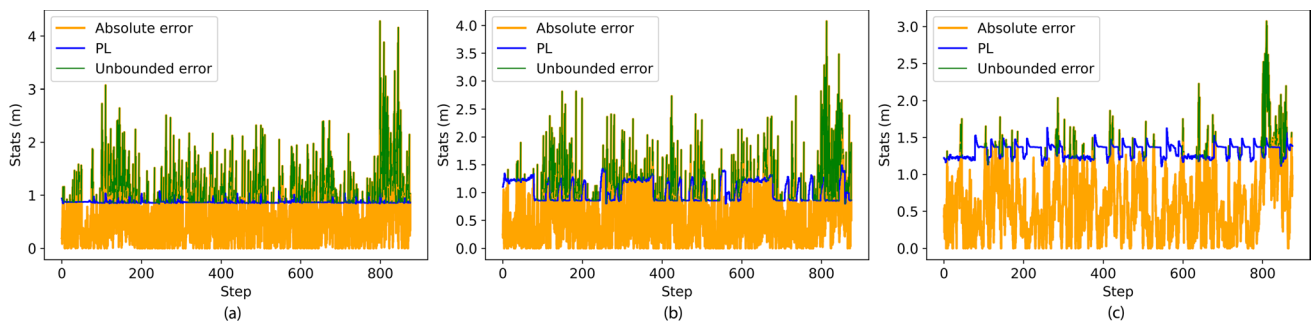


Fig. 12 PL and errors for the POS_SENSOR1 + POS_SENSOR2 + IMU loose integration with GTSAM for the UAV trajectory (a) North axis (b) East axis (c) Down axis

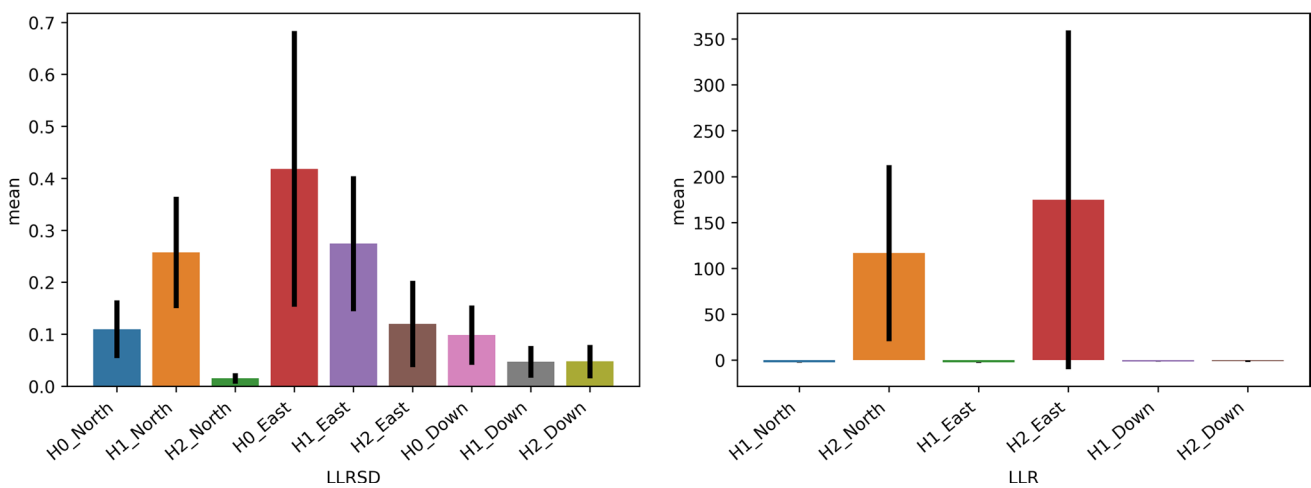


Fig. 13 LLRS and LLR mean and standard deviation (black lines on the bars) for the POS_SENSOR1 + POS_SENSOR2 + IMU loose integration with GTSAM for the UAV trajectory

4.3.3 Experiments on GTSAM using the ESKF Backend for IM

The reduction of the execution time was apparent when this implementation was used instead of the GTSAM-based evaluation for all hypotheses, as Table 1 demonstrates. Figure 14 shows that PLs bound quite well the error in contrast to the PLs computed when purely GTSAM-based IM is used (Fig. 12). However, the uncertainty is underestimated around the end of the trajectory. We remind that the overbounding technique was proven for KF-like algorithms.

4.4 Quantitative Evaluation of our IM Method

Li and Waslander [16] proposed a relaxed tightness metric to quantitatively evaluate the performance of PLs. They highlighted that in the nonlinear system case with outlier measurements it is not possible to guarantee that the error will be sufficiently bounded by a given PL at all times. They

proposed a novel Relaxed Bound Tightness (RBT) metric that quantifies both how much of the time the error is sufficiently bounded and the tightness of the bound. This metric is calculated as follows:

$$Z = \sqrt{\frac{\sum_{i=1}^N \rho \left(\frac{p_{L_i} - |e_i|}{\sigma_i} \right)^2}{N}}, \tag{11}$$

where p_{L_i} and e_i are respectively the PL and the error for a sample time i in one direction, N is number of samples, σ_i is the error covariance for the sample and ρ is a weight function that should penalize bounding failures more than loose bounds. It is given as:

$$\rho = \begin{cases} 1, & p_{L_i} \geq |e_i| \\ \tau, & p_{L_i} < |e_i| \end{cases} \tag{12}$$

To compute meaningful absolute RBT values the constant τ should be selected in a way to minimize

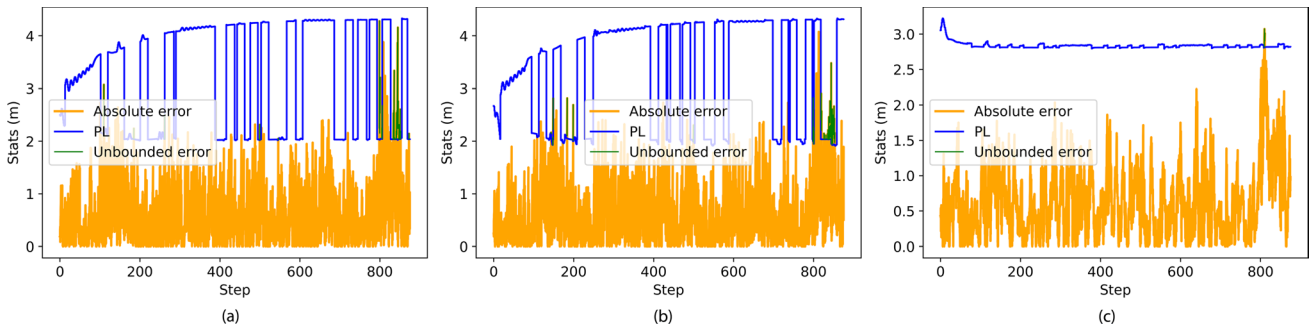


Fig. 14 PL and errors for the integration of the POS_SENSOR1, POS_SENSOR2 and IMU, in GTSAM for the UAV trajectory. Here we use the ESKF for the computation of the bounds in the fault

hypotheses' case. The filter underestimates the noise of the POS_SENSOR2. We also use the conditional overbounding technique

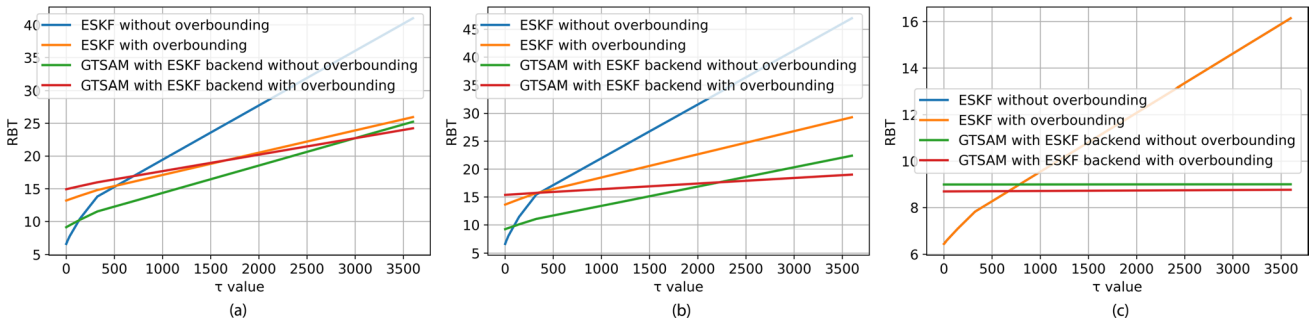


Fig. 15 RBT values of examined IM algorithms, for various values of the unbounded error penalty constant τ , for the UAV simulation trajectory. (a) North axis (b) East axis (c) Down axis

Z given an ideal bound v^* . After assuming that the error follows a Gaussian distribution without outliers, the latter is calculated as the quantile function for a Gaussian distribution, depending on the predefined minimum probability of error bounding. However, here we arbitrarily selected a set of values between 1^2 and 60^2 for τ to run the evaluation and compare the IM performance for different sensor integration schemes and navigation algorithms. Table 1 presents the results along with execution times. Overbounding is justifiable in safety-critical applications where bounding the error over long periods is a strict requirement, that is, for higher τ . In contrast, overbounding methods are disadvantageous when tighter PLs are preferred (lower τ). In addition, one should consider that overbounding leads to longer execution times when IM is run sequentially, mainly due to the evaluation of more hypotheses. In the absence of bounding failures, the value of τ does not affect the RBT value, as is confirmed from the results for the KITTI trajectory.

Finally, Fig. 15 plots the RBT values for the simulated UAV trajectory for a few selected τ values in the range $[1, 60^2]$. The figure shows that additional criterium for the decision to utilize overbounding or not should be

the algorithm that is monitored. Overbounding becomes justifiable already for a value of $\tau = 500$ for the ESKF case, whereas, for the GTSAM with ESKF backend, it is not justifiable until a value of $\tau = 2000$. This is because the position error was generally lower in the GTSAM case. To generalize, the results highlight that overbounding methods can be avoided in IM if the tuning of the noise parameters in the navigation algorithm has been done optimally (i.e. the position error is low) or if the noise parameters have been set conservatively (i.e. PLs will be already loose enough).

5 Conclusions

This paper is dedicated on the design of a universal IM solution for multi-sensor navigation that operates in the position domain. It is a SS-based monitor where an entire sensor's measurement set is assumed faulty in each hypothesis. The monitor works directly with the output position states and uncertainties of the underlying navigation algorithm. However, we suggest that the utilization of the monitored algorithm to evaluate all hypotheses can result in degraded performance when

computationally heavy algorithms are evaluated or due to non-standardized format of provided covariances. Instead, we propose the utilization of a standardized filter for the evaluation of all but the fault-free hypotheses, and, therefore, our method removes the direct dependence on output covariances from the navigation algorithm. We selected the ESKF in our experiments as it usually has good performance with non-linear systems.

The challenge with an independent solution is the unreliable uncertainties provided by the navigation algorithm not caught by consistency checks. Despite the more efficient IM of solutions that run collaboratively with the navigation algorithm, we propose that measures based on measurements' log likelihood (LL) can provide a good understanding of sensor measurement uncertainties. This is especially helpful when the navigation algorithm only has very approximate knowledge of the measurement noise models. We include an overbounding technique which is applied conditionally, based on the rolling standard deviation of LL. We experimented by synthesizing a measurement that includes time correlated noise, something that is unknown in the navigation algorithm. Instead, the algorithm has some knowledge of broad ranges of values where the noise parameters lie. The results demonstrate the effectiveness of LL based measurements in providing sufficient indications of underestimated uncertainties. Previous theoretical results of overbounding the error of KF estimates in presence of correlated sensor noise were applied with the ESKF. Applying this method showed promising results: Protection Levels that are bounding the error and that are not too loose when not needed.

As the experiments confirmed, the proposed method's architecture achieves the goal of universality in terms of navigation algorithm's internal architecture as well as time complexity, while it is extendable to a variety of sensors with challenging noise models. A limitation of the method is that the included sensors should provide independent position solutions which is not true in many occasions, and, therefore, this remains a topic for future research.

The method and the experimental results are expected to be adoptable in future developments in multi-sensor navigation platform IM. It is currently unclear how camera or LiDAR sensors affect the integrity and under which conditions. Current single-sensor methods work with individual keyframes or landmarks and, due to the huge number of hypotheses, they are not directly scalable for multi-sensor fusion. Our architecture works with hypotheses on sensor-level and avoids the scalability issues.

Appendix 1: ESKF and Gauss-Markov Process Noise Overbounding

IMUs often suffer stochastic errors that cannot be estimated. It is common to model these errors as the sum of random constant turn-on-bias, a time-correlated process and a white Gaussian noise term [8]. Therefore, the angular rate and specific force measurements responsible for the linear acceleration of the sensor can be expressed as:

$$\tilde{\mathbf{w}}^b = \mathbf{w}^b + \mathbf{b}_{w,0} + \mathbf{b}_w + \mathbf{n}_w, \tag{13}$$

$$\tilde{\mathbf{f}}^b = \mathbf{f}^b + \mathbf{b}_{f,0} + \mathbf{b}_f + \mathbf{n}_f, \tag{14}$$

where $\tilde{\mathbf{w}}^b$ and $\tilde{\mathbf{f}}^b$ are respectively the measured turn rates and specific forces in 3 axes with respect to the body frame b , \mathbf{w}^b and \mathbf{f}^b their true values, $\mathbf{b}_{w,0}$ and $\mathbf{b}_{f,0}$ are respectively the turn-on random constant bias of the angular rates and the specific forces, \mathbf{b}_w and \mathbf{b}_f are the time-correlated biases, and \mathbf{n}_w and \mathbf{n}_f are white Gaussian noise vectors. The time-correlated biases are usually modeled with a Gauss-Markov process.

The ESKF estimated error state $\delta\mathbf{x}$ consists of the errors in position $\delta\rho$, velocity $\delta\mathbf{v}$, angles vector $\delta\boldsymbol{\theta}$, linear acceleration bias $\delta\mathbf{b}_a$ and angular velocity bias $\delta\mathbf{b}_w$. In our experiments the second position sensor is affected by time-correlated noise and, therefore, we have to account for it as well. Generally, the state vector can be augmented with additional states to account for additional time correlated errors, e.g. in position sensor's measurements. Therefore, in the general case, the state vector becomes:

$$\delta\mathbf{x} = (\delta\rho, \delta\mathbf{v}, \delta\boldsymbol{\theta}, \delta\mathbf{b}_a, \delta\mathbf{b}_w, \delta\mathbf{b}_{pos_1}, \dots, \delta\mathbf{b}_{pos_m})^T \in \mathbb{R}^{15+3m}, \tag{15}$$

where m is the number of position sensors. In our experiments $m = 2$ and, therefore, the total number of states will be 21.

Prediction

The continuous error state kinematics will be:

$$\delta\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x})\delta\mathbf{x} + \mathbf{G}(\mathbf{x})\mathbf{n}, \tag{16}$$

where \mathbf{n} is the process noise with spectral density $\mathbf{Q} \in \mathbb{R}^{(12+3m) \times (12+3m)}$ where the velocity, orientation and bias estimates are modeled by white Gaussian processes.

The error state dynamics matrix $\mathbf{A}(\mathbf{x})$ and error state noise matrix $\mathbf{G}(\mathbf{x})$ are formulated by means of first order-approximations:

$$A = \left[\begin{array}{cccc|ccc} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & \\ \mathbf{0} & \mathbf{0} & -R(\mathbf{q})S(\tilde{\mathbf{f}} - \hat{\mathbf{b}}_a) & -R(\mathbf{q}) & \mathbf{0} & & \\ \mathbf{0} & \mathbf{0} & -S(\tilde{\mathbf{w}} - \hat{\mathbf{b}}_w) & \mathbf{0} & -\mathbf{I} & & \mathbf{0}_{15 \times 3m} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -p_{b_a} \mathbf{I} & \mathbf{0} & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -p_{b_w} \mathbf{I} & & \\ \hline & & \mathbf{0}_{3m \times 15} & & -p_{b_{pos_1}} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ & & & & \mathbf{0} & \ddots & \vdots \\ & & & & \mathbf{0} & \dots & -p_{b_{pos_m}} \mathbf{I} \end{array} \right] \quad (17)$$

$$G = \left[\begin{array}{cccc|ccc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & & \\ -R(\mathbf{q}) & \mathbf{0} & \mathbf{0} & \mathbf{0} & & & \mathbf{0}_{15 \times 3m} \\ \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & & & \\ \mathbf{0} & \mathbf{0} & g_{b_a} \mathbf{I} & \mathbf{0} & & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & g_{b_w} \mathbf{I} & & & \\ \hline & & \mathbf{0}_{3m \times 12} & & g_{b_{pos_1}} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ & & & & \mathbf{0} & \ddots & \vdots \\ & & & & \mathbf{0} & \dots & g_{b_{pos_m}} \mathbf{I} \end{array} \right], \quad (18)$$

with $\mathbf{0}$ and \mathbf{I} without subscript indicating a 3×3 matrix of zeros and the 3×3 identity matrix respectively, the symbol S denoting the cross-product operation, \mathbf{q} the attitude quaternion, $R(\mathbf{q})$ the rotation matrix, p_{b_*} are the inverse time constants of each sensor and g_{b_*} a component that will inflate the corresponding covariance of a bias state in \mathbf{Q} . Then:

$$Q = \left[\begin{array}{cccc|ccc} \sigma_a^2 \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & & \\ \mathbf{0} & \sigma_w^2 \mathbf{I} & \mathbf{0} & \mathbf{0} & & & \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & & & \mathbf{0}_{12 \times 3m} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & & & \\ \hline & & & & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ & & \mathbf{0}_{3m \times 12} & & \mathbf{0} & \ddots & \vdots \\ & & & & \mathbf{0} & \dots & \mathbf{I} \end{array} \right], \quad (19)$$

where σ_a^2 and σ_w^2 the noise variance of the accelerometer and the gyroscope respectively.

The FDO module conditionally selects to inflate the uncertainty that corresponds to a sensor by setting $p_{b_*} = 1/T_{b_*}$ and $g_{b_*} = \sqrt{\frac{2}{T_{b_*}} \sigma_{b_*}^2}$ with T_{b_*} and σ_{b_*} given in Eqs. 6 and 7 as this design choice has been proven to lead in an estimated covariance $\hat{\mathbf{P}}$ that is larger than the true covariance \mathbf{P} [8].

The propagation matrix A_k and covariance matrix Q_d in the discrete time model can be computed with the Van Loan's method and the predicted covariance in discrete-time, similarly to a standard KF becomes:

$$\hat{\mathbf{P}}_{k|k-1} = A_{k|k-1} \hat{\mathbf{P}}_{k-1|k-1} A_{k|k-1}^T + Q_{d(k|k-1)} \quad (20)$$

Update

The measurement model relates to the true state as:

$$z = \mathbf{h}(\mathbf{x}_t) + \mathbf{w}, \quad (21)$$

denoting as \mathbf{x}_t the true state vector and \mathbf{w} the measurement noise modeled as a zero mean Gaussian with covariance matrix \mathbf{R}_k .

However, as we aim for the estimation of the error state instead of the true state we require the relationship between the true state and the error state. This is formulated as:

$$\mathbf{x}_t = \left[\begin{array}{c} \rho + \delta\rho \\ \mathbf{v} + \delta\mathbf{v} \\ \mathbf{q} \otimes \left[\begin{array}{c} 1 \\ \frac{1}{2} \delta\boldsymbol{\theta} \end{array} \right] \\ \mathbf{b}_a + \delta\mathbf{b}_a \\ \mathbf{b}_w + \delta\mathbf{b}_w \\ \mathbf{b}_{pos_1} + \delta\mathbf{b}_{pos_1} \\ \vdots \\ \mathbf{b}_{pos_m} + \delta\mathbf{b}_{pos_m} \end{array} \right] \quad (22)$$

Substituting in Eq. 16 and by linearizing around $\delta\mathbf{x} = \mathbf{0}$ (error state is small):

$$z \approx \mathbf{h}(\mathbf{x} \oplus \mathbf{0}) + \left. \frac{\partial \mathbf{h}(\mathbf{x} \oplus \delta\mathbf{x})}{\partial \delta\mathbf{x}} \right|_{\delta\mathbf{x}=\mathbf{0}} \delta\mathbf{x} + \mathbf{w} = \mathbf{h}(\mathbf{x}) + \mathbf{H} \delta\mathbf{x} + \mathbf{w}, \quad (23)$$

with \mathbf{H} denoting the Jacobian. Having this model, we can directly retrieve the updated error state and covariance as in the EKF framework.

In the experiments presented in this paper the fixes are directly the measurements from the position sensors and the matrix \mathbf{H} for the position update from POS_SENSOR i will be:

$$\mathbf{H}_{pos_i} = \left[\mathbf{I} \ \mathbf{0} \ -R(\mathbf{q})S(\mathbf{r}_{pos_i}) \ \mathbf{0}_{3 \times 6} \ \vdots \ \mathbf{H}'_{pos_i} \right] \in \mathbb{R}^{3 \times (15+3m)}, \quad (24)$$

where \mathbf{r}_{pos_i} the lever arm from body to POS_SENSOR i and \mathbf{H}'_{pos_i} is a $3 \times 3m$ matrix which is non-zero only for the elements that correspond to \mathbf{b}_{pos_i} :

$$H'_{pos_i} = [0_{3 \times 3(i-1)} \quad I \quad 0_{3 \times 3(m-i)}] \tag{25}$$

This completes an introduction to our ESKF implementation, where we focused on showing the application of the overbounding method.

Appendix 2: LLR for Faulty Sensor Detection

Let:

y_k^0 the subset that contains all measurements at time k
 $y_k^j = [\rho_k^1, \rho_k^2, \dots, \rho_k^{j-1}, \rho_k^{j+1}, \dots, \rho_k^m]$ the subset that contains all measurements except the j -th (m the total number of measurements at time k).

We define the likelihood:

$$p(y_k^j | x_k) = \prod_{w=1 \& w \neq j}^m \frac{1}{\sqrt{2\pi}c_k} e^{-\frac{\rho_k^{(w)} - \tilde{\rho}_k^{(w)}}{2c_k}}, \tag{26}$$

ρ_k the observation from the sensor, $\tilde{\rho}_k$ the algorithm's estimation for the same measurement.

For a time range a to b , the cumulative LLR between two observation subsets is:

$$S_a^{b,j} = \sum_{k=a}^b \ln \left(\frac{p(y_k^j | x_k)}{p(y_k^0 | x_k)} \right), 1 \leq j \leq m \tag{27}$$

Define $\beta_k^j = \max_{k-x+1 \leq a \leq k} \{S_a^{k,j}\}$, x is the accumulation time window size, and $\beta_k = [\beta_k^1, \beta_k^2, \dots, \beta_k^m]$ is the test statistic.

So:

- If we have sensors that provide redundant measurements
- If we assume only up to one faulty sensor at each time

Then having hypotheses with one sensor excluded at each, a hypothesis that correctly assumes a sensor faulty will have much higher LLR than other hypotheses. Intuitively, it can be seen as a measurement of better agreement of all sensors.

Authors' Contributions V. Bosdelekidis: methodology, software implementation, experimental tests, paper editing. T. H. Bryne: methodology, experimental tests, paper editing, work supervision. N. Sokolova: methodology, experimental tests, paper editing, work supervision. T. A. Johansen: methodology, experimental tests, paper editing, work supervision.

Funding This work was supported by the Research Council of Norway and Centre for Autonomous Marine Operations and Systems for the corresponding PhD project at the Norwegian University of Science and Technology (RCN grant number: 305051).

Data Availability The KITTI trajectory is publicly available at <http://www.cvlibs.net/datasets/kitti/>. The UAV simulated trajectory can be provided to anyone upon request to the corresponding author of this paper.

Code Availability Not applicable.

Declarations

Ethics Approval Not applicable.

Consent to Participate Not applicable.

Consent for Publication Not applicable.

Conflicts of Interest/Competing Interests The authors report no conflict of interest or competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. European GNSS Agency: Report on The Performance and Level of Integrity For Safety and Liability Critical Multi-Applications. (2015). https://www.gsa.europa.eu/sites/default/files/calls_for_proposals/Annex%202.pdf. Accessed 11 Jan 2022
2. Joerger, M., Chan, F.-C., Pervan, B.: Solution Separation Versus Residual-Based RAIM. *Navigation*. **61** (2014)
3. Calhoun, S.-M., Raquet, J.: Integrity determination for a vision based precision relative navigation system. 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), Savannah, GA, pp. 294–304 (2016)
4. Joerger, M., Chan, C., Langel, S., Pervan, B.: RAIM detector and estimator design to minimize the integrity risk. 2012, ION GNSS (2012)
5. Joerger, M., Jamoom, M., Spenko, M., Pervan, B.: Integrity of laser-based feature extraction and data association. 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), pp. 557–571 (2016)
6. Mohanty, A., Gupta, S., Gao, G.-X.: A particle filtering framework for integrity risk of GNSS-camera sensor fusion. Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2020) (2021)
7. Meng, Q., Hsu, L.-T.: Integrity monitoring for all source navigation enhanced by Kalman Filter based Solution Separation. 2020. *IEEE Sensors J.* (2020)
8. Crespillo, O. G., Joerger, M., Langel, S.: Overbounding GNSS/INS Integration with Uncertain GNSS Gauss-Markov Error Parameters. 2020 IEEE/ION Position, Location and Navigation Symposium (PLANS), pp. 481–489 (2020)

9. He, P., Liu G., Tan, C., Lu Y.-E: Nonlinear fault detection threshold optimization method for RAIM algorithm using a heuristic approach. *GPS Solut* **20**, 863–875 (2016)
10. Gupta, S., Gao, G.-X.: Particle RAIM for integrity monitoring. *ION GNSS+* (2019)
11. Appleget, A., Leishman R.-C., Gipson, J.: Evaluation of sensor-agnostic all-source residual monitoring for navigation. Proceedings of the 2021 International Technical Meeting of The Institute of Navigation (2021)
12. Bhamidipati, S., Gao, G.-X.: SLAM-based Integrity Monitoring Using GPS and Fish-eye Camera. Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation, *ION GNSS+* 2019, pp. 4116–4129 (2019)
13. Arana, G.-D., Hafez, O.-A., Joerger, M., Spenko, M.: Recursive Integrity Monitoring for Mobile Robot Localization Safety. 2019 International Conference on Robotics and Automation (ICRA), pp. 305–311 (2019)
14. Tanil, C., Khanafseh, S., Joerger, M., Pervan, B.: Sequential Integrity Monitoring for Kalman Filter Innovations-Based Detectors. Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (*ION GNSS+* 2018), Miami, Florida, September 2018, pp. 2440–2455 (2018)
15. Jurado, J., Raquet, J., Schubert Kabban, C.M., Gipson, J.: Residual-based multi-filter methodology for all-source fault detection, exclusion, and performance monitoring. *Navigation* **67**, 493–509 (2020)
16. Li, C., Waslander, S.-L.: Visual Measurement Integrity Monitoring for UAV Localization. 2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR) (2019)
17. Reid, T.-G.-R., Houts, S.-E., Cammarata, R., Mills, G., Agarwal, S., Vora, A., Pandey, G.: Localization Requirements for Autonomous Vehicles. *SAE Int. J. Connect. Autom. Veh.* **2**(3), 173–190 (2019)
18. Fu, L., Zhang, J., Li, R., Cao, X., Wang, J.: Vision-Aided RAIM: A New Method for GPS Integrity Monitoring in Approach and Landing Phase. *Sensors* **15**, 22854–22873 (2015)
19. Mohleji, S.-C., Wang, G.: Modeling ADS-B position and velocity errors for airborne merging and spacing in interval management application. The MITRE Corporation (2010)
20. Langel S., Crespillo, O.-G., Joerger, M.: Bounding sequential estimation errors due to gauss-markov noise with uncertain parameters. Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (*ION GNSS+* 2019), Miami, Florida, pp. 3079–3098 (2019)
21. Geiger, A., Lenz, P., Stiller, C., Urtasun, R. Vision meets robotics: The KITTI Dataset. *Int. J. Robot. Res. (IJRR)* (2013)
22. Brown, R. G., & Hwang P.-Y.-C.: Transition from the discrete to continuous filter equations. In: Brown, R. G., & Hwang P. Y. C. *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises*, pp. 371–375. Wiley. ISBN: 978-0-470-60969-9 (2012)
23. Madyastha, V., Ravindra, V., Mallikarjunan, S., Goyal, A.: Extended Kalman Filter vs. Error State Kalman Filter for Aircraft Attitude Estimation. *AIAA Guidance, Navigation, and Control Conference* 2011 (2011)
24. Roumeliotis, S.-I., Sukhatme, G.-S., & Bekey, G.-A.: Circumventing dynamic modeling: evaluation of the error-state Kalman filter applied to mobile robot localization. Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C), pp. 1656–1663 vol.2 (1999)
25. Dellaert, F.: Factor Graphs and GTSAM: A Hands-on Introduction. Georgia Tech Technical Report, *GT-RIM-CP&R-2012-002*, Tech. Rep. (2012)
26. Lupton T., Sukkarieh, S.: Visual-Inertial-Aided Navigation for High-Dynamic Motion in Built Environments Without Initial Conditions. *IEEE Trans. Robot.* **28**(1) (2012)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Vasileios Bosdelekidis received the integrated M.Sc. degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2016. Later, he was employed for five years at HERE Technologies, with a focus on precise car navigation and highly automated driving, first as Software Engineer in Product Operations team and, later, as Data Scientist in a research team. He is currently pursuing the Ph.D. degree with the Department of Engineering Cybernetics at Norwegian University of Science and Technology (NTNU), Trondheim, Norway. He is affiliated with the Centre for Autonomous Marine Operations and Systems (AMOS). His research is focused on navigation integrity in safety-critical autonomous operations with sensor- and navigation algorithm-agnostic framework.

Torleiv H. Bryne received his MSc and PhD in Engineering Cybernetics in 2013 and 2017, respectively, both from the Norwegian University of Science and Technology (NTNU). He has previously been a Research Scientist at SINTEF, Trondheim, Norway, and is currently an Associate Professor at the Department of Engineering Cybernetics, NTNU. His research interests are in the field of estimation applied to navigation and autonomous systems. Unmanned aerial vehicles and marine applications are the main focus areas of his research. He recently co-founded the spin-off company SentiSystems.

Nadezda Sokolova received her PhD degree in Telecommunications from Norwegian University of Science and Technology (NTNU). She also holds a MSc degree in Geomatics Engineering from University of Calgary, Canada, and a MSc degree in Space and Aeronautical Engineering from Narvik University, Norway. Currently, she works as a senior research scientist at SINTEF Digital and adjunct associate professor at the Engineering Cybernetics Department, NTNU, focusing on high integrity resilient multi-sensor navigation.

Tor Arne Johansen received the MSc degree in 1989 and the PhD degree in 1994, both in electrical and computer engineering, from the Norwegian University of Science and Technology, Trondheim, Norway. From 1995 to 1997, he worked at SINTEF as a researcher before he was appointed Associated Professor at the Norwegian University of Science and Technology in Trondheim in 1997 and Professor in 2001. He has published several hundred articles in the areas of control, estimation and optimization with applications in the marine, aerospace, automotive, biomedical and process industries. In 2002 Johansen co-founded the company Marine Cybernetics AS where he was Vice President until 2008. Prof. Johansen received the 2006 Arch T. Colwell Merit Award of the SAE, and is currently a principal researcher within the Center of Excellence on Autonomous Marine Operations and Systems (NTNU-AMOS) and director of the Unmanned Aerial Vehicle Laboratory at NTNU and the SmallSat Laboratory at NTNU. He recently co-founded the spin-off companies Scout Drone Inspection, UBIQ Aerospace, Zeabuz and SentiSystems.