

RESEARCH

Open Access



Reliable urban vehicle localization under faulty satellite navigation signals

Shubh Gupta¹ and Grace X. Gao^{1*}

*Correspondence:
gracegao@stanford.edu

¹ Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, USA

Abstract

Reliable urban navigation using global navigation satellite system requires accurately estimating vehicle position despite measurement faults and monitoring the trustworthiness (or integrity) of the estimated location. However, reflected signals in urban areas introduce biases (or faults) in multiple measurements, while blocked signals reduce the number of available measurements, hindering robust localization and integrity monitoring. This paper presents a novel particle filter framework to address these challenges. First, a Bayesian fault-robust optimization task, formulated through a Gaussian mixture model (GMM) measurement likelihood, is integrated into the particle filter to mitigate faults in multiple measurement for enhanced positioning accuracy. Building on this, a novel test statistic leveraging the particle filter distribution and the GMM likelihood is devised to monitor the integrity of the localization by detecting errors exceeding a safe threshold. The performance of the proposed framework is demonstrated on real-world and simulated urban driving data. Our localization algorithm consistently achieves smaller positioning errors compared to existing filters under multiple faults. Furthermore, the proposed integrity monitor exhibits fewer missed and false alarms in detecting unsafe large localization errors.

Keywords: Particle filter, Integrity monitoring, Gaussian mixture model

1 Introduction

Global navigation satellite system (GNSS) in urban environments is often affected by limited satellite visibility, signal attenuation, non-line-of-sight signals, and multipath effects [55]. Such impairments to GNSS signals result in fewer measurements as compared to open-sky conditions as well as time-varying biases, or *faults*, simultaneously in multiple measurements. Improper handling of these faults during localization may unknowingly result in large positioning errors, posing a significant risk to the safety of the navigating vehicle. Therefore, safe navigation in urban environments requires addressing two challenges: accurately estimating the position of the navigating vehicle from the faulty measurements and determining the estimated position's trustworthiness, or *integrity* [55].

1.1 Position estimation under faulty GNSS measurements

For estimating the position of a ground-based vehicle, many existing approaches use GNSS measurements in tandem with odometry measurements from an inertial navigation system. Many of these approaches rely on state estimation by means of filters, such as Kalman filter and particle filter [36]. These filters track the probability distribution of the navigating vehicle's position across time, approximated by a Gaussian distribution in a Kalman filter and by a multimodal distribution represented as a set of samples, or particles, in a particle filter. However, the traditional filters are based on the assumption of Gaussian noise (or overbounds) in GNSS measurements, which is often violated in urban environments due to faults caused by multipath and non-line-of-sight errors [55].

Several methods to address non-Gaussian measurement errors in filtering have been proposed in the area of robust state estimation. Karlgaard [27] integrated the Huber estimation technique with Kalman filter for outlier-robust state estimation. Pesonen [40], Medina et al. [34], and Crespillo et al. [8] developed robust estimation schemes for localization using fault-contaminated GNSS measurements. Lesouple et al. [30] incorporated sparse estimation theory in mitigating GNSS measurement faults for localization. However, these techniques are primarily designed for scenarios where a large fraction of non-faulty measurements are present at each time instant, which is not necessarily the case for urban environment GNSS measurements.

In the context of achieving robustness against GNSS measurement faults, several techniques have been developed under the collective name of Receiver Autonomous Integrity Monitoring (RAIM) [25]. RAIM algorithms mitigate measurement faults either by comparing state estimates obtained using different groups of measurements (solution-separation RAIM) [6, 25, 28] or by iteratively excluding faulty measurements based on measurement residuals (residual-based RAIM) [19, 23, 38, 46]. Furthermore, several works have combined RAIM algorithms with filtering techniques for robust positioning. Grosch et al. [15], Hewitson and Wang [18], Leppakoski et al. [29], and Li et al. [32] utilized residual-based RAIM algorithms to remove faulty GNSS measurements before updating the state estimate using KF. Boucher et al. [4], Ahn et al. [1], and Wang et al. [52, 53] constructed multiple filters associated with different groups of GNSS measurements and used the logarithmic likelihood ratio between the distributions tracked by the filters to detect and remove faulty measurements. Pesonen [41] proposed a Bayesian filtering framework that tracks indicators of multipath bias in each GNSS measurement along with the state. A limitation of these approaches is that the computation required for comparing state estimates from several groups of GNSS measurements increases combinatorially both with the number of measurements and the number of faults considered [25]. Furthermore, recent research has shown that poorly distributed GNSS measurement values can cause significant positioning errors in RAIM, exposing serious vulnerabilities in these approaches [47].

In a recent work [16], we developed an algorithm for robustly localizing a ground vehicle in challenging urban scenarios with several faulty GNSS measurements. To handle situations where a single position cannot be robustly estimated from the GNSS measurements, the algorithm employed a particle filter for tracking a multimodal distribution of the vehicle position. Furthermore, the algorithm equipped the particle filter with a Gaussian mixture likelihood model of GNSS measurements to ensure that various

position hypotheses supported by different groups of measurements are assigned large probability in the tracked distribution.

1.2 Integrity monitoring in GNSS-based localization

For assessing the trustworthiness of an estimated position of the vehicle, the concept of GNSS integrity has been developed in the previous literature [42, 55]. Integrity is defined through a collection of metrics that together express the ability of a navigation system to provide timely warnings when the system output is unsafe to use [36, 39]. Some metrics of interest for monitoring integrity are defined as follows

- *Positioning Failure* is defined as the event that the position error exceeds a specified maximum value, known as the alarm limit (AL).
- *Accuracy* represents the statistical uncertainty in the position output of a navigation system, expressed as the radius of a circle that contains a specified percentage of the positioning solutions.
- *Availability* of a navigation system indicates whether the system output is safe to use at a given time instant. A system is considered available if it meets the required accuracy threshold and has not detected a positioning failure.
- *Integrity Risk* is the probability of the event where the position error exceeds AL, but the system is declared available by the integrity monitoring algorithm.

Different approaches for monitoring integrity have been proposed in the literature. Solution-separation RAIM algorithms monitor integrity based on the agreement between the different state estimates obtained for different groups of measurements [6, 25, 28]. Residual-based RAIM algorithms determine the integrity metrics from statistics derived from the measurement residuals for the obtained state estimate [19, 23, 38, 46]. Tanil et al. [48] utilized the innovation sequence of the Kalman filter to derive integrity metrics and determine the availability of the system. However, the approach is designed and evaluated for single satellite faults and does not necessarily generalize to scenarios where several measurements are affected by faults. Furthermore, these techniques have been designed for the Kalman filter and cannot directly be applied to a particle filter-based framework. In Bayesian RAIM [41], the availability of the system is determined by comparing statistics computed from the tracked probability distribution with a vector-valued threshold. The proposed technique is general and can be applied to a variety of filters, including the particle filter. However, the tracked filter probability distribution is concentrated about the mean and has poor modeling fidelity in the tail regions of the probability distribution. Hence, the derived statistics have limited performance in integrity monitoring, which focuses on the tail distributions [37].

1.3 Our approach

In this paper, we build upon our prior work on a particle filter-based framework [16] to present a novel algorithm for urban GNSS-based localization and integrity monitoring. Our algorithm tackles the challenges of limited visibility and multiple faults by integrating an optimization objective for fault mitigation through a Gaussian mixture model (GMM) [13] measurement likelihood that handles faults present in multiple

measurements. The GMM likelihood enables the particle filter to effectively capture the multimodal uncertainty in localization that arises due to potential faulty measurements, by considering particles that support different subsets of measurements, as depicted in Fig. 1. By integrating vehicular dynamics with these particles and measurements, our algorithm assimilates measurement information over time within the particle filter framework. This allows mitigating the impact of erroneous measurements while accurately tracking the multimodal uncertainty in the vehicle's location. Furthermore, it continuously monitors the integrity of the estimated location and raise an alarm when the error exceeds a specified limit.

The contributions of this paper are threefold:

1. We present a novel particle filter-based localization algorithm that integrates an optimization objective for fault mitigation and estimates the location using a Gaussian mixture model measurement likelihood. This algorithm offers enhanced robustness and accuracy in urban environments by effectively handling multiple faults in GNSS measurements.
2. We propose a statistical testing technique that leverages measures derived from the particle filter probability distribution and the available GNSS measurements to evaluate the integrity of the estimated location. This technique enables the system to detect situations where the error exceeds a predefined threshold and trigger an alarm accordingly.
3. We evaluated the performance of our algorithm by testing it in real-world and simulated driving scenarios. To assess its effectiveness, we compare the performance of our framework against existing localization baselines and integrity monitoring techniques. The results demonstrate the superior performance of our framework, par-

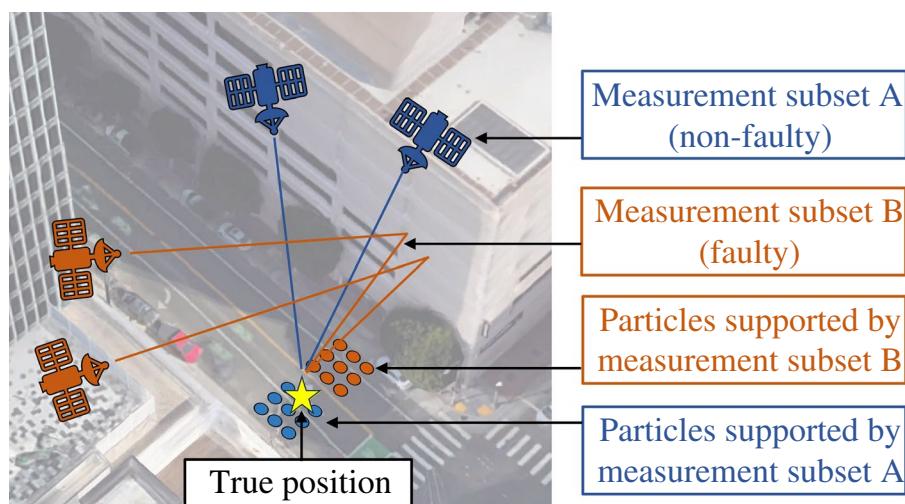


Fig. 1 Our proposed particle filter-based algorithm for fault-robust GNSS localization. The algorithm incorporates GNSS pseudorange measurements, divided into unknown subsets of non-faulty (**A**, blue) and faulty (**B**, orange) measurements. By considering particles that support either subset, our algorithm effectively captures the multimodal uncertainty in localization, accommodating the potential presence of faulty measurements

ticularly in challenging urban environments, where it outperforms the existing methods.

Our approach provides several advantages over prior approaches for fault-robust localization and integrity monitoring. This is due to our weighting scheme that simultaneously assigns weights to GNSS measurements and particles instead of individually considering several subsets of the measurements. Since our approach tracks several position hypotheses for mitigating measurement faults, it can track the position probability distribution in scenarios with a large fraction of faulty GNSS measurements, unlike the existing robust state estimation approaches. Unlike existing particle filter-based approaches that rely on a known map of the environment [20, 21, 35] or a known transition model of faults [41], our approach requires no assumptions about existing environment information for mitigating faults. Furthermore, our integrity monitor accounts for the multimodal probability distribution of errors present in the measurements, unlike the purely filter-based approach presented in [41].

The rest of the paper is organized as follows: Section 2 presents related work in the field of urban GNSS-based localization. Section 3 provides the problem setup and background information. In Sect. 4, we describe our proposed framework in detail. Section 5 presents the experimental results and performance evaluation. Finally, Sect. 6 concludes the paper, summarizing the contributions and discussing future research directions.

2 Problem formulation

Consider a vehicle navigating in an urban environment, equipped with an onboard GNSS receiver and an IMU. At each time step, denoted by t , the vehicle acquires a set of K_t GNSS pseudorange measurements. We denote this set of measurements as $m_t = \{\rho_t^k\}_{k=1}^{K_t}$, where each $\rho_t^k \in \mathbb{R}$ is the pseudorange measurement from the k -th satellite at time t .

In addition to pseudorange measurements, the vehicle also obtains an odometry measurement at each time step, denoted as $u_t \in \mathbb{R}^3$, representing the vehicle's velocity at the time step.

The objective of our estimation problem is twofold. Firstly, we aim to estimate the vehicle's state (or position) at time t , denoted as $x_t \in \mathbb{R}^3$. This estimate is based on the sequence of GNSS pseudorange measurements $\mathcal{M}_{1:t} = \{m_1, \dots, m_t\}$ and odometry measurements $\mathcal{U}_{1:t} = \{u_1, \dots, u_t\}$, where $\mathcal{M}_{t_0:t}$ and $\mathcal{U}_{t_0:t}$ denote all pseudorange and odometry measurements from time t_0 to time t , respectively.

The second objective is to raise an alarm whenever the positioning error Δx_t exceeds a user-specified alarm limit α_t^{AL} . In this paper, we refer to these events as positioning failures. Here, the positioning error Δx_t represents the difference between the estimated state, x_t , and the true state, x_t^* . The alarm limit, α_t^{AL} , is a user-defined threshold that specifies the maximum allowable positioning error in the system at time t .

3 Background on GNSS pseudorange-based localization

After defining the problem, we give an overview of the localization methods based on GNSS pseudorange measurements. We begin by discussing the snapshot least-squares estimation (LSE)-based positioning algorithm, which leverages all the available

measurements at a given time instance for positioning. We then explore fault mitigation strategies based on residuals, aimed at improving positioning accuracy in the presence of a single fault. Subsequently, we delve into the extension of these strategies to multiple time steps within a filtering framework, enabling robust algorithms that utilize information from multiple time steps while considering a single fault at each time step. Finally, we formalize the objective of detecting positioning failures for integrity monitoring.

3.1 Snapshot LSE GNSS-based positioning

GNSS positioning requires the solution of a set of nonlinear equations, which arise from the satellite positioning geometry and the signal's time-of-flight. Each equation links the receiver's unknown position and clock bias to the known positions of several satellites and their respective transmission times.

Snapshot LSE positioning can be expressed as the following optimization problem:

$$\min_{x_t, \beta_t} \sum_{k=1}^{K_t} (r_t^k)^2 \quad (1)$$

$$\text{where } r_t^k = \frac{\rho_t^k - h^k(x_t, \beta_t)}{\sigma_t^k}, \quad (2)$$

Here, x_t and β_t denote the estimated receiver's position and the estimated clock bias at time t , respectively. The objective function represents the sum of squared residuals r_t^k , which represent the differences between the observed and the estimated pseudoranges for all satellites scaled by a standard deviation σ_t^k . The standard deviation σ_t^k is typically obtained empirically, through the satellite geometry, or using received signal characteristics.

The term $h^k(x_t, \beta_t)$ is the estimated pseudorange for the k -th satellite given by a model of satellite signal propagation:

$$h^k(x_t, \beta_t) = \|x_t - X_t^k\| + c(\beta_t - \beta_t^k) + \epsilon^{\text{iono}} + \epsilon^{\text{tropo}} + \epsilon^{\text{mp}}. \quad (3)$$

In the above equation, X_t^k denotes the position of the satellite corresponding to the k -th measurement at time t , and β_t^k denotes the estimated clock bias error of the k -th satellite. The terms ϵ^{iono} , ϵ^{tropo} , and ϵ^{mp} account for the estimated errors from ionospheric and tropospheric, and multipath effects, respectively. The term c denotes the speed of light. The satellite position and clock bias are determined from the navigation message embedded in the signal, while the ionospheric and tropospheric corrections are typically obtained from physical models or augmentation systems [14]. As the multipath error is dependent on the receiver's surroundings, it is assumed to be zero within this model unless additional environmental data are available.

The nonlinear optimization problem is usually solved using the Newton–Raphson algorithm, an iterative method designed to successively improve approximations of the solution [26]. Beginning with an initial estimate for the receiver position and clock bias, the algorithm iteratively enhances this estimate by taking steps in the direction that maximally reduces of the objective function, guided by its gradient. The primary

advantage of the Newton–Raphson method in snapshot GNSS positioning is its ability to quickly converge on a precise solution, despite the nonlinear nature of the problem. However, when multipath or other sources of error result in significant discrepancies between the modeled and observed pseudoranges, termed as faults, the application of more advanced algorithms becomes necessary for reliable positioning [9].

3.2 Snapshot LSE positioning with residual-based fault mitigation

In the context of reliable GNSS-based positioning, it is common practice to address scenarios where one of the available measurements exhibits faults. Initially, an all-measurement solution for the position and clock bias is estimated by solving the previously discussed optimization problem (Eq. 1). Subsequently, the residuals for each measurement from the all-measurement solution are analyzed to determine faults. By identifying the measurement with the largest residual, potential faults can be identified and mitigated. This process of identifying and removing fault-containing measurements is also referred to as fault exclusion in literature.

For improved positioning performance, several algorithms such as Receiver Autonomous Integrity Monitoring (RAIM) incorporate both a fault detection test and a fault mitigation algorithm that relies on residuals [5]. This test utilizes statistical tools, such as a Chi-squared-based detector, to detect the presence of faults. If a fault is not detected, the all-measurement solution is used for positioning.

Conversely, upon successful fault detection, an improved solution can be obtained by re-solving the optimization problem, excluding the measurement identified as faulty. The revised optimization problem can be formulated as follows:

$$\min_{x_t, \beta_t} \sum_{k=1, k \neq k'}^{K_t} (r_t^k)^2. \quad (4)$$

Here, k' is the index of the identified faulty measurement. The rest of the variables retain their previous definitions.

While the residual-based fault mitigation strategy improves localization performance in some cases, the solution of the aforementioned optimization problem does not guarantee an accurate solution. Certain combinations of fault magnitudes and satellite geometry may mislead the algorithm into incorrectly flagging a non-faulty measurement as faulty, resulting in erroneous positioning [33]. Nevertheless, in practical applications with occasional measurement faults, such as aviation, this modified approach often improves the accuracy of the positioning solution and enhances the robustness of the snapshot LSE positioning process.

This robustness, however, is contingent on a few important assumptions—the given set of GNSS measurements contains only one faulty measurement and that there are an adequate number of non-faulty measurements to derive a reliable initial solution. These assumptions hold true in aviation, where several satellites are within the line-of-sight of the receiver and measurement faults are infrequent. However, urban environments often present challenges like signal obstruction and reflections, which frequently violate this assumption.

Furthermore, while the aforementioned strategy can be repeated after each fault exclusion to identify and remove multiple faults, the likelihood of missing faulty measurements increases with more than one fault, resulting in significantly diminished effectiveness of the approach in urban environments. In these scenarios, the initial solution, derived from all the measurements, often deviates significantly from the actual solution. Consequently, distinguishing between faulty and non-faulty measurements based on residuals becomes challenging.

3.3 Sequential positioning with residual-based fault mitigation

Addressing the limitations of snapshot LSE algorithms, several methods have been developed that leverage information across multiple time steps to improve the localization accuracy and robustness to measurement errors [23, 32]. These methods aim to mitigate the impact of insufficient fault-free measurements at individual time steps by combining measurements and incorporating knowledge of the vehicle dynamics over time.

In the context of sequential localization, the optimization problem can be formulated as follows:

$$\min_{x_{1:T}, \beta_{1:T}} \sum_{t=1}^T \sum_{k=1, k \neq k'_t}^{K_t} (r_t^k)^2, \quad (5)$$

$$\text{s.t. } x_t, \beta_t = f_t(x_{t-1}, \beta_{t-1}). \quad (6)$$

where T represents the total number of time steps, $x_{1:T}$ and $\beta_{1:T}$ correspond to the sequences of receiver positions and clock biases, respectively, spanning the entire time frame. The function $f_t(\cdot)$ represents the state transition at time t , derived from the vehicle dynamics. The objective of this optimization problem (Eq. 5) is to minimize the sum of squared residuals over all time steps and satellites, with the exclusion of satellite k'_t at each time step t , which is considered faulty. By considering information from multiple time steps, this sequential localization approach offers improved robustness compared to snapshot LSE methods.

To efficiently solve this optimization problem, it can be tackled incrementally at each time step t within a filtering framework, using the following probabilistic optimization formulation:

$$\min_{p(x_t, \beta_t)} \mathbb{E} \left[\sum_{k=1, k \neq k'_t}^{K_t} (r_t^k)^2 \right], \quad (7)$$

$$x_t, \beta_t = f_t(x_{t-1}, \beta_{t-1}), \quad (8)$$

$$x_{t-1}, \beta_{t-1} \sim p(x_{t-1}, \beta_{t-1}). \quad (9)$$

The term $p(x_t, \beta_t)$ denotes the probability distribution of the joint state space comprising of position x_t and clock bias β_t . Filtering algorithms, such as Kalman filters (employing multivariate Gaussian distributions) or particle filters (utilizing collections of weighted

samples), can effectively estimate the location and clock bias by modeling the probability distribution.

To mitigate faults using residuals in the sequential setting, similar strategies as in the snapshot LSE case are often employed. These strategies involve estimating a solution using all available measurements and subsequently removing the measurement with the largest residual from the solution. While this approach can be more robust in the sequential setting due to the initial proximity of the all-measurement solution to the true solution, it is limited to handling single or few faults. In scenarios with multiple faults, the initial solution can exhibit significant bias, leading to challenges in fault detection and accurate positioning [3].

4 Formalism for detecting positioning failures

GNSS-based localization has an essential requirement of integrity—despite the adoption of fault-tolerant strategies, the localization error is not guaranteed to be insignificant. This is often the product of various influencing factors, such as sub-optimal satellite geometry, inadequate visibility, and the existence of unmodeled errors, all of which can substantially compromise the positioning performance [2]. Consequently, to counteract this limitation, a majority of integrity monitoring algorithms are equipped to detect positioning failures and trigger alarms when the solution is unreliable.

A practical approach to detecting positioning failures at time t involves the formulation of the following optimization problem:

$$\min_{\text{Alarm}_t} p(\Delta x_t \leq \alpha_t^{\text{AL}} \mid \text{Alarm}_t) \quad (10)$$

$$\text{where } p(\Delta x_t > \alpha_t^{\text{AL}} \mid \overline{\text{Alarm}}_t) \leq p_{\text{MD}}, \quad (11)$$

where Alarm_t and $\overline{\text{Alarm}}_t$ denote the complementary binary events that represent whether an alarm is triggered (or not) by the system at time t to signal the presence of positioning failures. The term p_{MD} represents an upper limit on the probability of missed detection, which is a non-detection of a positioning failures scenario. Therefore, the primary objective underpinning the design of a good detection algorithm is one that minimizes the likelihood of false alarms—situations where an alarm is triggered erroneously under nominal system conditions—while meeting the missed identification performance requirement.

In view of the above outlined optimization problem, the criteria for triggering an alarm can be informed either by a single time step's measurement, similar to snapshot LSE algorithms, or by multiple time steps' measurements, similar to sequential algorithms.

4.1 Snapshot integrity monitoring

In the case of snapshot algorithms, the event Alarm is typically determined by the following equation:

$$\text{Alarm}_t = \{\tau(m_t, x_t, \beta_t) > \alpha_t\}, \quad (12)$$

In this equation, $\tau(\cdot)$ is a test statistic derived from the current measurements and solutions, and α_t is a threshold for the statistic at time t . For snapshot integrity monitoring, both the statistic and the threshold are structured around Gaussian statistics or

the Chi-squared test, with degrees of freedom equivalent to the difference between the number of available measurements and the dimensionality of the solution space [28].

4.2 Sequential integrity monitoring

The previous statistical testing framework for detecting positioning failures in snapshot settings can be generalized to sequential settings. In this case, the test statistic $\tau(\cdot)$ is computed using measurements $\mathcal{M}_{1:t}$ spanning all previous time steps, in conjunction with an estimated probability distribution of the joint sequence of positions $x_{1:t}$ and clock biases $\beta_{1:t}$. The resulting detection criterion can be formulated as follows:

$$\text{Alarm}_t = \{\tau(\mathcal{M}_{1:t}, p(x_{1:t}, \beta_{1:t})) > \alpha_t\}. \quad (13)$$

This general formulation encompasses a variety of detection criteria that have been developed within existing sequential integrity monitoring algorithms. For instance, in the context of localization systems utilizing Kalman filters, a typical test statistic used is the sum of squared innovations [23]. This statistic depends on the measurements as well as the mean and covariance of the probability distribution tracked by the Kalman filter at each time step.

In systems employing a bank of filters for localization, a common test statistic incorporates the likelihood ratios between different components of the probability distribution tracked by the bank [44]. Similarly, in particle filter-based methods, vector-valued test statistics that depend on the tracked probability distribution have been employed [41].

While a majority of the test statistics have been designed for aviation contexts, where stringent assumptions about faults and measurement noise apply, they exhibit a broader applicability to integrity monitoring in complex urban environments. By conservatively tailoring the threshold to meet the missed identification requirement while simultaneously minimizing false alarms, these algorithms can be employed to detect positioning failures in a variety of challenging settings [12]. However, it is crucial to note that current algorithms have yet to demonstrate the capacity to concurrently maintain rates of false alarms and missed detections in urban environments that are comparable to aviation. This highlights the necessity to focus on designing novel methods aimed at improving the integrity monitoring performance in urban environments.

Having defined the problem setting and the background on GNSS-based fault-robust localization and integrity monitoring, we are now ready to present our approach for sequential fault-robust localization and integrity monitoring in the presence of multiple faults and limited measurements.

5 Methodology

In this section, we present our approach to address the challenges of fault-robust localization and integrity monitoring in scenarios involving multiple faults and limited measurements. We begin by generalizing the residual-based fault-robust localization optimization problem, enabling simultaneous mitigation of the faulty measurement while determining the position and clock bias solution. This generalized optimization problem is further extended to handle sequential settings and situations with multiple faults.

Subsequently, we derive our Gaussian mixture model (GMM) likelihood of measurements, which is constructed based on the aforementioned optimization problem. The GMM likelihood captures the multimodal nature of the measurement errors and of the fault-robust localization optimization problem.

Next, we introduce our particle filter-based algorithm, which estimates a multimodal probability distribution of the position and clock bias using the GMM likelihood. By leveraging the particle filter framework, our algorithm effectively captures the multimodal nature of the solution-space, enabling accurate and robust localization and representation of uncertainty.

Furthermore, we present our method that detects positioning failures by utilizing the available measurements and the estimated probability distribution. This allows us to identify potential risks and trigger alarm to ensure the safety of the vehicle.

Finally, we provide a computational analysis of our algorithm, assessing its performance and discussing the computational complexity of our approach.

5.1 Generalized fault-robust localization using residuals

5.1.1 Optimization problem

We extend the optimization problem for fault-robust localization presented in Sect. 3.3 extends the generalized setting of identifying and removing faulty measurements. The generalized problem includes a measurement selection vector, $s_t \in \{0, 1\}^{K_t}$ with k th element denoted by s_t^k , within the optimization variables. The problem is thus formulated as:

$$\min_{x_t, \beta_t, s_t} \sum_{k=1}^{K_t} s_t^k (r_t^k)^2. \quad (14)$$

Here, s_t assumes the value 1 for all except one measurement, for which it is zero. The optimization objective is minimized when s_t allocates zero to the largest residual-bearing measurement, given optimal position x_t and clock bias β_t estimates. Expanding the approach to sequential settings, we can formulate the probabilistic optimization objective in a manner similar to Eq. 7:

$$\min_{p(x_t, \beta_t), p(s_t)} \mathbb{E} \left[\sum_{k=1}^{K_t} s_t^k (r_t^k)^2 \right], \quad (15)$$

$$x_t, \beta_t = f_t(x_{t-1}, \beta_{t-1}), \quad (16)$$

$$x_{t-1}, \beta_{t-1} \sim p(x_{t-1}, \beta_{t-1}). \quad (17)$$

By formulating the optimization problem in this manner, we can simultaneously mitigate a faulty measurement while estimating the position and clock bias. Moreover, we can further extend this formulation to handle the mitigation of multiple faults by including k different measurement selection vectors. This is equivalent to setting k entries of the selection vector s to zero while keeping the rest as one. In this context, minimizing the objective function involves removing the k largest residuals from the optimal solution. It

is important to note, however, that the number of faulty measurements must be known in advance for this method of optimization. Furthermore, the potential values that the selection vector can assume grow combinatorially with the number of faults, making the optimization computationally demanding for more than a single fault.

In addressing the challenges of localization in complex urban environments, it is important to account for uncertainty and multiple potential faults. Therefore, we consider an extreme case where the measurement selection vector s_t assumes the value 1 for only a single measurement while assigning 0 to the remaining measurements. The optimization problem with this selection vector can be formulated similarly to Eq. 14. The extreme scenario provides a safety-oriented perspective, particularly when the true measurement quality or fault conditions are unknown.

The efficient optimization of this extreme scenario is facilitated by the fact that the number of potential values that the selection vector can assume is equal to the number of measurements. This enables optimization by effectively exploring different configurations of the selection vector, and allows identifying the optimal assignment that minimizes the residuals and achieves fault-robust localization.

Building upon this formulation of the optimization problem, we next construct a Gaussian mixture model (GMM) likelihood for the measurements. This likelihood serves as a fundamental component of our algorithm, which is based on a particle filter framework, designed for fault-robust localization and integrity monitoring in the presence of multiple faults.

5.1.2 Gaussian mixture model likelihood of measurements

Traditional GNSS localization approaches that rely on filter-based techniques design the measurement likelihood model under the assumption that all incorporated measurements are both independent and devoid of any faults. The probability $p(\rho_t^k | x_t)$ associated with the k th measurement at time t for x_t is modeled as a Gaussian distribution [7, 24, 41, 54].

However, this model evaluates each position based on its ability to concurrently satisfy all measurements, thereby generating a unimodal probability distribution around a single solution point x_t . This methodology, although effective under conditions of independent and fault-free measurements, fails to maintain its efficacy when multiple GNSS measurements are faulty. In such instances, the unimodal model tends to oversimplify the measurement errors, as it is incapable of handling the complex, multimodal errors arising from multiple faults. Hence, in scenarios with multiple faults, an alternative modeling approach capable of representing a multimodal probability distribution is required to model the measurement likelihood.

In response to this, we derive a novel measurement likelihood grounded in our optimization problem for sequential fault-robust localization (Eq. 14). In line with the probabilistic interpretation of optimization problems, we construct the objective function as the expected value of the negative logarithm of the measurement. To efficiently solve this optimization problem, it can be tackled incrementally at each time step t within a filtering framework, using the following probabilistic optimization formulation:

$$\min_{p(x_t, \beta_t), p(s_t)} \mathbb{E} \left[\sum_{k=1}^{K_t} s_t^k (r_t^k)^2 \right] \equiv \min_{p(x_t, \beta_t), p(s_t)} \mathbb{E}[-\log \ell_t(x_t, \beta_t, s_t)], \quad (18)$$

where $\ell_t(x_t, \beta_t, s_t)$ denotes the measurement probability $p(m_t | x_t, \beta_t, s_t)$. Using this interpretation, we can determine the form of the likelihood as follows:

$$-\log \ell_t(x_t, \beta_t, s_t) \propto \sum_{k=1}^{K_t} s_t^k (r_t^k)^2, \quad (19)$$

$$\ell_t(x_t, \beta_t, s_t) \propto \prod_{k=1}^{K_t} e^{-C_1 s_t^k (r_t^k)^2}, \quad (\text{For some constant } C_1 > 0) \quad (20)$$

$$\propto \sum_{k=1}^{K_t} \mathbf{1}[s_t^k = 1] e^{-C_1 (r_t^k)^2}. \quad (21)$$

Applying the law of total probability and eliminating s_t , we arrive at the following expressions:

$$p(m_t | x_t, \beta_t) = \sum_{k=1}^{K_t} \ell_t(x_t, \beta_t, s_t) p(s_t^k = 1), \quad (22)$$

$$= \sum_{k=1}^{K_t} C_2 p(s_t^k = 1) e^{-C_1 (r_t^k)^2}, \quad (\text{For some constant } C_2 > 0) \quad (23)$$

$$\ell_t(x_t, \beta_t) = \sum_{k=1}^{K_t} \gamma_t^k p_{\mathcal{N}(0,1)}(r_t^k) \text{ s.t. } \sum_{k=1}^{K_t} \gamma_k = 1, \quad (24)$$

where $p_{\mathcal{N}(\mu, \sigma^2)}(\cdot)$ denotes the probability under normal distribution with mean μ and variance σ^2 .

These equations lead us to model the likelihood $\ell_t(x_t, \beta_t) = p(m_t | x_t, \beta_t)$ as a GMM. This model leverages additive contributions of each measurement to the likelihood, ensuring that faults in any small subset of measurements do not dominate the overall likelihood.

In the GMM likelihood, each measurement component is associated with a weight coefficient $\gamma_t^k = p(s_t^k = 1)$ which modulates the impact of that component on the overall likelihood. Consequently, the likelihood maps the space of potential x_t and β_t values to the multimodal GMM probability distribution via a nonlinear residual computation function. The resulting GMM implies a multimodal probability distribution over the solution space, marked by peaks at positions supported by different subsets of component measurements. The measurement weights γ_t^k determine the relative

confidence among these modes, implying that with identical standard deviation values, the mode with a higher measurement weight has a higher probability value [13].

In the subsequent sections, we will employ this interpretation of the GMM likelihood in relation to our optimization problem. We incorporate the likelihood into our particle-filter-based algorithm for fault-robust localization and integrity monitoring, with the goal of improving the localization accuracy and integrity in situations with multiple faults.

5.2 Particle filter-based fault-robust localization and integrity monitoring

In this section, we describe our algorithm for fault-robust localization and integrity monitoring. Our overall algorithm can be subdivided into seven components:

1. The *Propagation step* receives as input the previous time particle distribution of receiver state and the odometry measurement and generates as output propagated particles in the *extended state-space*, which are associated with different GNSS measurements. The extended state-space consists of the receiver state along with an integer-valued selection of a GNSS measurement associated with the particle.
2. The *Iterative Weighting step* receives as input the extended state-space particles and the GNSS measurements and infers the particle weights and GMM weight coefficients in three steps: First, the *Measurement Voting step* computes a probabilistic confidence, referred to as vote, in the association between the extended state-space particle and the chosen GNSS measurement. Next, the *Vote Pooling step* combines the votes to obtain an overall confidence in each GNSS measurement, referred to as measurement weights. Finally, the *GMM Weighting step* updates the particle weights using the GMM constructed from the measurement weights as the measurement likelihood function. The process is then repeated for a fixed number of iterations, or till convergence.
3. The *Reduced Resampling step* receives as input the extended state-space particles and their computed weights and generates a new set of particles in the receiver state space.
4. For detecting positioning failure, we compute a vector-valued statistic comprising of *precision* and *Probability of Positioning Failures* using the particles and the GMM likelihood. Then, we trigger alarm by comparing the obtained precision and probability of positioning failure against user-specified thresholds.

Figure 2 illustrates the architecture of our algorithm. We first describe our model for the particle-based probability distribution, followed by details about the individual components of our method. Finally, we end the section with the run-time analysis of our approach.

5.2.1 Particle-based multimodal state probability distribution

The vehicle's state probability distribution, denoted as π_t , is approximated via a particle filter in the following manner:

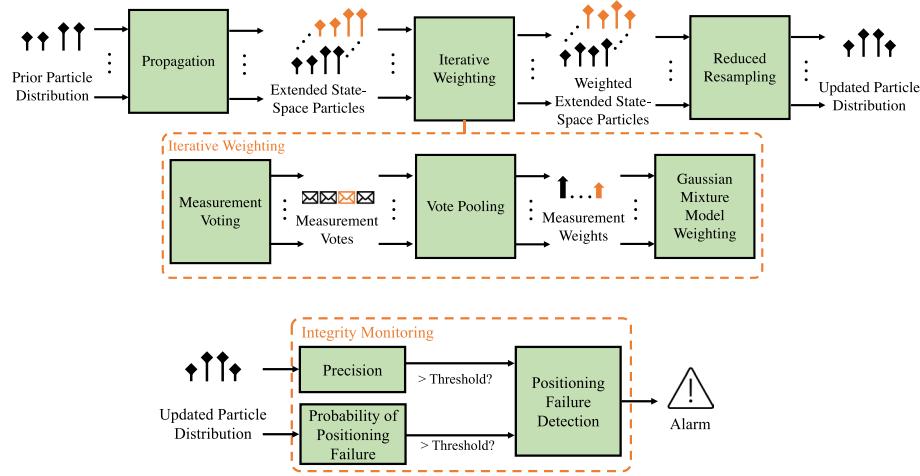


Fig. 2 Architecture of the proposed algorithm for fault-robust localization and integrity monitoring. The algorithm incorporates GNSS and odometry measurements to infer the probability distribution of the vehicle state as a set of weighted particles in a particle filter. Robustness to faults in GNSS measurements is achieved using the GMM weighting, measurement voting and vote pooling steps. By analyzing the estimated probability distribution and GNSS measurements, the framework derives metrics of precision and probability of positioning failure, enabling continuous detection of positioning failure to trigger alarm across all time steps

$$\pi_t = p(x_t | x_{t-1}, m_t, u_t) \approx \hat{\pi}_t = \sum_{i=1}^N w_t^{(i)} \delta(x = x_t^{(i)}). \quad (25)$$

In this expression, $\hat{\pi}_t$ denotes the approximate state probability distribution represented using particles; $x_t^{(i)}$ denotes the state of the i th particle, and includes the position, clock bias, and additional tracked terms (e.g., heading); $w_t^{(i)}$ denotes the weight of the i th particle satisfying $\sum_{i=1}^N w_t^{(i)} = 1$; $\delta(x = x_t)$ denotes the Dirac delta function that assumes the value of zero everywhere except x_t [10]; and N is the total number of particles.

To design our particle filter-based localization algorithm, we assume that the state of the vehicle follows the Markov property, i.e., given the present state of the vehicle, the future states are independent of the past states. By the Markov property among (x_1, \dots, x_t) , the conditional probabilities $p(x_t | x_{t-1}, m_t, u_t)$ and $p(x_t | x_1, \dots, x_{t-1}, \mathcal{M}_{1:t}, u_{1:t})$ are equivalent. Applying Bayes' theorem with independence assumptions on \mathcal{M}_t, u_t , the probability distribution is factored as:

$$p(x_t | x_{t-1}, m_t, u_t) \propto p(m_t | x_t, x_{t-1}, u_t) \cdot p(x_t | x_{t-1}, u_t) \quad (26)$$

$$\propto \underbrace{p(m_t | x_t)}_{\ell_t(x_t)} \cdot \underbrace{p(x_t | x_{t-1}, u_t)}_{\tilde{\pi}_t}, \quad (27)$$

where $\ell_t(x_t)$ can be interpreted as the likelihood of receiving m_t at time t from x_t ; and $\tilde{\pi}_t$ denotes the probability distribution at time t predicted using $\hat{\pi}_{t-1}$ and u_t .

In our approach, we concurrently perform state estimation and fault mitigation via an extended state-space (x, s) of particles, with $s \in \{1, \dots, K_t\}$ indicating an integer corresponding to one of the GNSS measurements in m_t . This is analogous to the selection vector s in the optimization problem Eq. 14. Here, the integer value of s signifies the index corresponding to the value 1 in the vector. Each particle in the extended state-space is assigned a value of s at the time of creation during the propagation step, which remains constant until the resampling step. These extended state-space particles are used subsequently to determine the weight coefficients in the GMM likelihood as well as the weights of the particles.

5.2.2 Propagation step

Firstly, we generate K_t uniform-weighted copies of each existing particle $x_{t-1}^{(i)}$, each corresponding to a unique s value in the set $\{1, \dots, K_t\}$. This overall set of particles captures the joint state-space that includes s . Following this, the particles in this extended state-space, denoted as $x_{t-1}^{(i,s)}$, are propagated in accordance with state dynamics as:

$$x_t^{(i,s)} = f(x_{t-1}^{(i,s)}, u_t) + v^{(i,s)} \quad \forall s = \{1, \dots, K\}, \quad (28)$$

$$\tilde{\pi}_t^{(i)} = \sum_{k=1}^K \gamma_t^k \delta(x = \tilde{x}_t^{(i,s=k)}), \quad (29)$$

$$\tilde{\pi}_t \approx \sum_{i=1}^N w_{t-1}^{(i)} \tilde{\pi}_t^{(i)}, \quad (30)$$

where f is the function to propagate state x using odometry u based on system dynamics. The term γ_t is initialized uniformly across measurements, and $v^{(i,s)} \sim \mathcal{N}(0, (\sigma^v)^2)$ denotes the stochastic term involved in the propagation of each particle with standard deviation σ^v . This is equivalent to drawing samples from the predicted state probability distribution at time t based on the history of measurements [11].

Following this, we compute the measurement likelihood model and update the weights associated with each particle in the weighting step.

5.2.3 Iterative weighting step

To determine the values for $\{\gamma_k\}_{k=1}^{K_t}$ and $\{w_t^{(i)}\}_{i=1}^N$ using the GMM likelihood, we employ an iterative approach inspired by the expectation–maximization (EM) algorithm for GMM [50]. Our approach involves three steps, namely measurement voting, vote pooling and GMM weighting. The steps are described in detail below.

5.2.3.1 Measurement Voting Assuming no prior knowledge about measurement faults, we initialize the value of γ_t^k uniformly with K_t^{-1} for all $k \in \{1, \dots, K_t\}$. For the weights $\{w_t^{(i,s=k)}\}_{i=1}^N$, we set their initial values based on the previous time step weights as

$$w_t^{(i,s=k)} = \frac{1}{K_t} \cdot w_{t-1}^{(i)}, \quad i \in \{1, \dots, N\}. \quad (31)$$

This initialization scheme ensures equal weight distribution across the measurement components for each particle at the start of the iterative process.

In order to compute the updates for both $\{w_t^{(i)}\}_{i=1}^N$ and $\{\gamma_k\}_{k=1}^{K_t}$, we introduce a set of random variables $\{V^k\}_{k=1}^{K_t}$ referred to as *votes*. These votes allow us to indirectly express the confidence associated with each measurement k based on a given value of the state x_t . Unlike $\{\gamma_k\}_{k=1}^{K_t}$, the votes $\{V^k\}_{k=1}^{K_t}$ depend on x_t and depend on the normalized residuals r_t^k for each particle $x_t^{(i,s=k)}$ in π_t . The residuals r_t^k are defined as:

$$r_t^k = (\sigma_t^k)^{-1}(\rho_t^{(k)} - h(x_t^{(i,s=k)})). \quad (32)$$

Using the initial values of $\{w_t^{(i)}\}_{i=1}^N$ and $\{\gamma_k\}_{k=1}^{K_t}$ as well as the computed residuals $\{r_t^{(i,k)}\}_{i=1,k=1}^{N,K_t}$, we proceed to infer the probability distribution of the random variables $\{V^k\}_{k=1}^{K_t}$ in the expectation-step of the EM algorithm [50]. We model this probability distribution using samples of V^k , which we refer to as weighted votes $\{v_t^{(i,k)}\}_{i=1}^N \sim V^k$:

$$v_t^{(i,k)} = p_{\mathcal{N}^2(0,1)}((r_t^{(i,k)})^2), \quad (33)$$

where $p_{\mathcal{N}^2(0,1)}(\cdot)$ denotes the probability density function of the square of a standard normal distribution [45]. The use of the squared standard Gaussian distribution assigns smaller probability values to larger residuals, aligning with RAIM algorithms that use the Chi-squared distribution for detecting GNSS faults [38].

5.2.3.2 Vote pooling In the vote pooling step, we normalize and combine the votes cast by each particle to determine the updated values of γ_k for each measurement. The total empirical probability π_t^{tot} of measurements m_t at time t , incorporating the votes $\{v_t^{(i,k)}\}_{i=1,k=1}^{N,K_t}$ and measurement weights $\{\gamma_k\}_{k=1}^{K_t}$, is expressed as

$$\pi_t^{tot} = \sum_{i=1}^N \sum_{k=1}^{K_t} \gamma_k w_t^{(i,k)} v_t^{(i,k)}. \quad (34)$$

To maximize π_t^{tot} while satisfying the constraint $\sum_{k=1}^{K_t} \gamma_k = 1$, we derive an update rule for vote pooling as

$$\gamma_k = \frac{\sum_{i=1}^N w_t^{(i,k)} v_t^{(i,k)}}{\sum_{i=1}^N \sum_{k'=1}^{K_t} w_t^{(i,k')} v_t^{(i,k')}}. \quad (35)$$

This update rule assigns an empirical probability to each γ_k based on the weighted votes, enabling us to combine the confidences from multiple particles and measurements.

5.2.3.3 GMM weighting Using the computed measurement weights $\{\gamma_k\}_{k=1}^{K_t}$, we update the weights of each extended state-space particle. To ensure numerical stability, we perform the weight update in the logarithmic scale. The GMM log-likelihood, however, contains additive terms inside the logarithm, making direct distribution

infeasible. To address this, we perform the weighting using the extended state-space that includes the measurement association variable s .

The measurement likelihood conditioned on s can now be expressed as a categorical distribution likelihood as per Sect. 5.1.2:

$$\ell_t(x_t) = p(m_t | x_t, s) = \prod_{k=1}^{K_t} \left(\gamma_k p_{\mathcal{N}(0,1)}(r_t^{(k)}) \right)^{\mathbb{I}[s=k]}, \quad (36)$$

where $\mathbb{I}[\cdot]$ denotes an indicator function that equals 1 if the condition in its argument is true and 0 otherwise. Here, $\sum_{k=1}^{K_t} \gamma_k = 1$.

Taking the logarithm of the likelihood, we obtain:

$$\log \ell_t(x_t) = \sum_{k=1}^{K_t} \mathbf{1}[s = k] \left(\log \gamma_k + \log p_{\mathcal{N}(0,1)}(r_t^{(k)}) \right). \quad (37)$$

To compute the new weights $\{w_t^{(i,k)}\}_{i=1, k=1}^{N, K_t}$ for each particle in a stable manner, we use the log-likelihood expression as follows:

$$l_k^{(i)} = \log \ell_t(x_t^{(i,s=k)}) - \max_{i,k} \left(\log \ell_t(x_t^{(i,s=k)}) \right), \quad (38)$$

$$w_t^{(i,k)} = \frac{\exp(l_k^{(i)})}{\sum_{i=1}^N \sum_{k=1}^{K_t} \exp(l_k^{(i)})}. \quad (39)$$

The iterative weighting algorithm is summarized in Algorithm 1. In successive iterations of the iterative weighting algorithm, we can additionally repeat the propagation step to compute new votes instead of reusing the same propagated particles, thus avoiding overfitting.

5.2.4 Reduced resampling step

The reduced resampling step is used to obtain the updated particle distribution $\{x_t^{(i)}\}_{i=1}^N = \hat{x}_t$ by redistributing the weighted extended state-space particles using the sequential importance resampling (SIR) procedure [17]. This step also reduces the number of particles from $N \times K_t$ to the original number of N particles, assigning equal weight to each particle. The algorithm is described in Algorithm 2.

The algorithm redistributes the particles based on their weights across the extended state-space and ensures that the number of particles remains constant. The resulting mean state estimate x_t is calculated as $x_t = \sum_{i=1}^N w_t^{(i)} x_t^{(i)}$.

Algorithm 1 Iterative weighting procedure

Require: Weighted extended state-space particles: $\{(x_t^{(i,s)}, w_t^{(i,s)})\}$ for $i = 1$ to N , $s = 1$ to K_t

Ensure: New particle weights: $\{w_t^{(i,s)}\}$ for $i = 1$ to N , $s = 1$ to K_t

Ensure: Updated measurement weights: $\{\gamma_k\}$ for $k = 1$ to K_t

- 1: Initialize $\gamma_k \leftarrow \frac{1}{K_t}$ for $k = 1$ to K_t
- 2: Initialize $w_t^{(i,s)} \leftarrow \frac{1}{K_t} \cdot w_{t-1}^{(i)}$ for $i = 1$ to N , $s = 1$ to K_t
- 3: **for** iteration = 1 to M **do**
- 4: **Measurement Voting:**
- 5: Compute residuals r_i^k using Eq. 32
- 6: **Vote Pooling:**
- 7: Compute votes $v_t^{i,k}$ using Eq. 33
- 8: Update γ_k using Eq. 35
- 9: **GMM Weighting:**
- 10: Compute log-likelihoods $l_k^{(i)}$ using Eq. 38
- 11: Compute new particle weights $w_t^{(i,s=k)}$ using Eq. 39
- 12: **end for**
- 13: **return** New particle weights: $\{w_t^{(i,s)}\}$ for $i = 1$ to N , $s = 1$ to K_t and updated measurement weights: $\{\gamma_k\}$ for $k = 1$ to K_t

Algorithm 2 Reduced resampling procedure

Require: Weighted extended state-space particles: $\{(x_t^{(i,s)}, w_t^{(i,s)})\}$ for $i = 1$ to N , $s = 1$ to K_t

Ensure: Updated particle distribution: $\{x_t^{(i)}\}$ for $i = 1$ to N

- 1: Set $W_t^{(i)} = \sum_{s=1}^{K_t} w_t^{(i,s)}$ for $i = 1$ to N \triangleright Cumulative sum of particle weights
- 2: Generate N random numbers u_i independently and uniformly distributed in the range $[0, 1]$, where $i = 1, 2, \dots, N$
- 3: Initialize $i = 1$
- 4: **for** $j = 1$ to N **do**
- 5: **while** $u_j > W_t^{(i)}$ **do**
- 6: $i \leftarrow i + 1$
- 7: **end while**
- 8: Set $x_t^{(j)} = x_t^{(i,s=1)}$ \triangleright Copy the i -th particle's state
- 9: Set $w_t^{(j)} = 1/N$ \triangleright Assign equal weights
- 10: **end for**
- 11: **return** Updated particle distribution: $\{x_t^{(i)}\}$ for $i = 1$ to N

5.2.5 Integrity monitoring

One of the key features of our algorithm is the integrity monitoring component, which aims to detect positioning failure conditions. Our integrity monitor incorporates a non-Gaussian probability distribution represented by particles to capture the uncertainty in the vehicle state estimation. It operates based on three fundamental assumptions: (1) The existence of a fault-free measurement at each timestep, (2) redundant positioning information across multiple satellites (over time), odometry, and dynamics, and (3) Gaussian probability distribution of individual fault-free GNSS measurements.

At each time instant, the integrity monitor calculates a vector-valued test statistic comprising of the probability of positioning failure τ^{PF} and precision τ^{P} . These measures are derived and computed based on the probability distribution tracked by the filter along with the available GNSS measurements. Subsequently, the algorithm compares these measures against user-specified thresholds to detect positioning failures. If a positioning failure is detected, an alarm is triggered, and the system is declared as unavailable. Our integrity monitoring algorithm is inspired by the Bayesian framework proposed in [41].

We compute the probability of positioning failure τ^{PF} based on the probability distribution of positions that lie outside the specified alarm limit. In previous particle filter-based methods [41], τ^{PF} is computed by integrating the filter probability distribution over all positions that lie outside the specified alarm limit. However, this approach often leads to inaccurate estimations of τ^{PF} since filtering approaches are conventionally designed to estimate the probability distribution in the vicinity of the most probable states and not the tail-ends [49].

To address this limitation, we derive an alternative approach to estimate τ^{PF} as an approximate upper bound of the positioning failures risk directly from our GMM likelihood. The estimation is based on the filter probability distribution along with the available GNSS measurements. We estimate τ^{PF} as follows:

$$p(\text{PF}) = p(x_t \notin \Omega_I | M_t, u_t, \hat{\pi}_{t-1}) \quad (40)$$

$$= 1 - p(x_t \in \Omega_I | M_t, u_t, \hat{\pi}_{t-1}) \quad (41)$$

$$= 1 - \frac{p(x_t \in \Omega_I | u_t, \hat{\pi}_{t-1})}{p(M_t | u_t, \hat{\pi}_{t-1})} p(M_t | x_t \in \Omega_I, u_t, \hat{\pi}_{t-1}) \quad (42)$$

$$\leq 1 - p(x_t \in \Omega_I | u_t, \hat{\pi}_{t-1}) p(M_t | x_t \in \Omega_I, u_t, \hat{\pi}_{t-1}) \quad (43)$$

$$\approx 1 - p(x_t \in \Omega_I | u_t, \hat{\pi}_{t-1}) \int_{x_t \in \Omega_I} |\Omega_I|^{-1} \sum_{k=1}^{K_t} \gamma_k p_{\mathcal{N}(\hat{\rho}_{x_t}^k, (\sigma_{x_t}^k)^2)} \left(\rho_t^{(k)} \right) dx_t \quad (44)$$

$$= \tau^{\text{PF}} \quad (45)$$

Here, Ω_I denotes the set of positions that lie within the alarm limit around the mean state estimate x_t and $|\Omega_I|$ denotes its total area. To simplify the computation, we approximate

the conditional distribution $p(x_t | \{x_t \in \Omega_I\}, u_t, \hat{\pi}_{t-1}) \approx p(x_t | \{x_t \in \Omega_I\})$ as a uniform distribution $|\Omega_I|^{-1} \forall x_t \in \Omega_I$. While this approximation does not necessarily correspond to the worst-case computation of $p(M_t | x_t \in \Omega_I, u_t, \hat{\pi}_{t-1})$, it admits fast computation using numerical techniques and therefore we adopt it in this work. Future research will explore advanced techniques, such as importance sampling, to improve the approximation while accounting for the worst case. Nevertheless, τ^{PF} is subject to approximation errors from filter, the measurement likelihood, uniform prior distribution, and numerical integration, and can therefore trigger false alarms or missed detections for different choices of threshold.

To efficiently compute the integral, we employ cubature techniques for two-dimensional disks, as described in [31]. The term $p(\{x_t \in \Omega_I\} | u_t, \hat{\pi}_{t-1})$ is computed by adding the weights of particles that lie inside the alarm limit based on the propagated distribution $\tilde{\pi}_t$. This approximation enhances computational efficiency over exact calculation while providing a reasonable estimate of τ^{PF} for determining positioning failure.

We define precision τ^P as the radius of the smallest disk around the estimated position x_t that contains the true vehicle position with at least 0.5 probability. Unlike the probability of positioning failure τ^{PF} , precision is determined primarily by the probability mass near the state estimate rather than the tail-ends. This is in line with the way precision (called “accuracy” in the paper) is computed in [41]. Unlike classical accuracy, which measures closeness to ground truth, τ^P captures the precision (or resolution) of positions that the filter can distinguish with the particles at a timestep. While the term is limited in capturing integrity information on its own, it helps identify situations where the filter distribution itself has a high variance and therefore can lead to incorrect estimation of τ^{PF} , which relies on the filter distribution.

To compute precision, we approximate the particle distribution by a Gaussian distribution with the mean parameter x_t and the covariance matrix Σ_t . The covariance matrix Σ_t is computed as a weighted estimate from the particle samples, where the weights are denoted as $w_t^{(i)}$ and the particle samples $x_t^{(i)}$. The computation of Σ_t is given by:

$$\Sigma_t = \frac{1}{1 - \sum_{i=1}^N (w_t^{(i)})^2} \sum_{i=1}^N w_t^{(i)} (x_t^{(i)} - x_t)(x_t^{(i)} - x_t)^\top. \quad (46)$$

Once we have the covariance matrix Σ_t , we estimate the precision τ^P using the inverse cumulative probability distribution of the Gaussian distribution. Specifically, we compute τ^P as follows:

$$\tau^P = \max_{i=1,2} \sqrt{\Sigma_{t,i,i}} \Phi^{-1}(p_P), \quad (47)$$

where $\Phi^{-1}(\cdot)$ denotes the inverse cumulative probability function for standard Gaussian distribution, and p_P denotes the desired precision level, which we set to 0.5. A smaller value of τ^P implies higher precision in positioning.

To detect positioning failures, we compare the vector-valued statistic $[\tau^{\text{PF}}, \tau^P]$ against a user-specified threshold $[\alpha^{\text{PF}}, \alpha^A]$. The integrity monitor checks the following condition:

$$(\tau^{PF} \geq \alpha^{PF}) \quad \text{or} \quad (\tau^P \geq \alpha^A). \quad (48)$$

Hence, if either the estimated probability of positioning failure or the estimated precision exceeds their respective thresholds, the integrity monitor detects a positioning failure and triggers an alarm.

5.3 Computation requirement

The computational requirement of our algorithm is proportional to the number of measurements k , just like existing least-squares and robust state estimation approaches. For residual-based RAIM algorithms [19, 23, 38, 46], the maximum required computation grows proportionally to mk , where m is the maximum number of iterations. For existing particle filter-based methods that run a separate filter for each fault hypothesis [1, 4, 52, 53], the computation depends on the number of filters (fault hypotheses) and on the number of particles n , which scales proportionally to nk^m , where m is the maximum number of simultaneous faults considered within the set of measurements.

In contrast to existing particle filter-based methods, our approach grows linearly in computation with the number of available measurements. For n particles and k available measurements, each component in our algorithm has a maximum computational requirement proportional to nk irrespective of the number of faults present. Hence, our approach exhibits smaller computational requirements than existing particle filter-based methods, and similar requirements to residual-based RAIM, least-squares and robust state estimation algorithms with respect to the number of GNSS measurements.

Furthermore, each of these steps can be parallelized across particles and measurements, which can further reduce the computational requirement. For example, all votes from particles to measurements can be computed in parallel, and pooling of votes can be performed at the same time across different measurements. Additionally, weights for all the particles can be computed simultaneously from the likelihood model. This can lead to significant speedups, especially for large state spaces.

However, the number of particles required to adequately approximate the probability distribution increases exponentially with the size of the state space [11]. This means that the computational requirement of our algorithm can also increase exponentially with the size of the state space. In order to maintain a small computation cost with large state-spaces, more complex strategies, such as Rao–Blackwellization or using factored probability distributions, need to be employed.

6 Experimentation results

This section presents results evaluating the performance of the algorithm for localization and integrity monitoring. First, localization performance is examined in simulated and real-world driving scenarios. Next, integrity monitoring results are shown for a simulated scenario with multiple faults. Finally, the results are discussed.

6.1 Localization performance

In order to assess the performance of our fault-robust localization algorithm, we conducted evaluations using both simulated and real-world scenarios. On each scenario, we compared the results against existing baselines for fault-robust localization.

The rest of this section is organized as follows: first, we defined the setup for the simulated and real-world experiments. Then, we introduced the baselines against which we compare our approach. Next, we outlined the metrics used to evaluate the performance. Subsequently, we presented the localization results obtained from the experiments. Finally, we conducted a sensitivity analysis to investigate the impact of the noise parameters on the performance of our approach.

6.1.1 Simulated experiment scenario

Our choice of simulations is motivated by two factors: the availability of accurate ground-truth information and the ability to obtain multiple measurement sequences with distinct noise values. To evaluate the effectiveness in localization by our approach and the baselines, we introduced varying noise profiles in GNSS and odometry measurements.

To create a realistic simulation environment, we designed a scenario in which a vehicle traversed a horizontal surface along 50 randomly generated trajectories, each spanning approximately 4000 m in length. During the simulation, the vehicle obtained GNSS ranging measurements from simulated satellites, which were subject to noise, as well as noisy odometry measurements. The satellites were simulated to move at a constant velocity of 1000 m/s along separate paths, randomly selected, while maintaining a fixed height of 2×10^7 m from the horizontal surface.

The odometry measurements included vehicle speed readings and the heading direction of the vehicle. Each driving scenario lasted for a duration of 400 s, with measurements acquired at frequency of 1 Hz. In our state estimation algorithms for the simulated scenario, we focused on tracking the 2D state of the vehicle, specifically the coordinates (x, y) . This restriction as motivated by the computational constraints associated with handling larger state-spaces. The simulation and filter parameters used in our experiments are given in Table 1.

To emulate real-world scenarios, we introduced two distinct noise profiles in our pseudorange measurements during the simulation process. Firstly, we incorporated zero mean random noise in all measurements. Secondly, we introduced random bias

Table 1 Experimental parameters for simulated scenario

Parameter	Value
# of satellites	5–10
GNSS ranging noise σ_{GNSS}	5–10 m
GNSS ranging bias error magnitude	50–200 m
Maximum # of faulty measurements	1–6
GNSS fault change probability	0.2
Vehicle speed	10 m/s
Odometry noise σ^v	5 m/s
# of particles	500
Filter initialization σ_{init}	5 m
Filter measurement model noise $\{\sigma^k\}_{k=1}^K$	5 m
Filter propagation model noise σ_f	5 m
Alarm limit AL	15–20 m

noise in a subset of measurements. The subset of measurements subject to bias noise was randomly initialized and had a small probability of transitioning to a different random subset at each time instant. The number of affected measurements was stochastically determined, ranging from zero up to a predetermined maximum number of faults during each transition. Additionally, the Gaussian noise applied to measurements was adjusted to have double the variance for biased measurements.

By employing our noise model, we effectively simulated the impact of non-line-of-sight signals and multipath propagation on pseudorange measurements within the context of urban GNSS navigation scenarios.

6.1.2 Real-world experiment scenario

In the real-world case, we utilized GNSS pseudorange and odometry measurements from the publicly available urban GNSS dataset collected by TU Chemnitz for the SmartLoc project [43]. Specifically, we focused on the Frankfurt Westend-Tower trajectory from the dataset for evaluating our approach. This trajectory had a length of approximately 2300 m and encompassed both dense and sparse urban regions, with approximately 32% of the total measurements being NLOS signals. Figure 3a visualizes the trajectory with the start and end points.

The dataset was collected using a low-cost u-blox EVK-M8T GNSS receiver operating at a frequency of 5 Hz, along with CAN (control area network) data for odometry obtained at a frequency of 50 Hz. The odometry data provided velocity and yaw-rate measurements from the vehicle's CAN bus. To establish a reliable ground truth, a high-cost NovAtel GNSS receiver operating at 20 Hz was employed to provide reference positions. Additionally, corrections to satellite ephemeris data were provided in the form of an SP3 file.

The receiver collected approximately 12 pseudorange measurements at each time instant from the GPS, GLONASS, and Galileo constellations. We assume that the starting position of the vehicle and an estimate of the clock drift rate are known and use this information to correct for clock biases at the start. Specifically, we removed

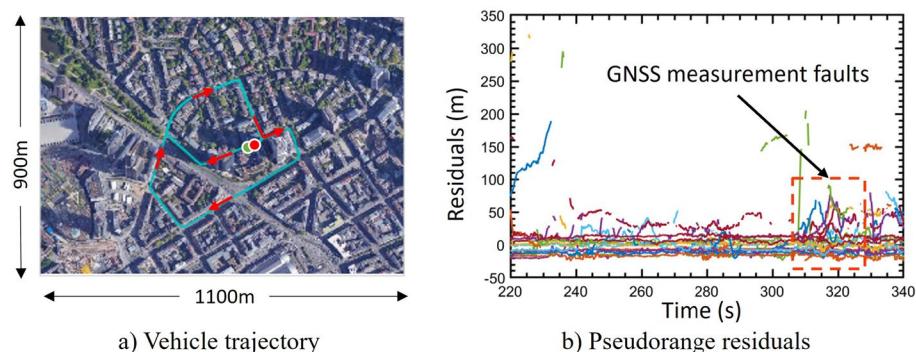


Fig. 3 Real-world dataset for validation of our approach. We use Frankfurt Westend-Tower sequence from smartLoc project dataset collected by TU Chemnitz [43]. The trajectory followed by the vehicle is shown in (a). The vehicle moves from the start point (green) to end point (red) along the trajectory (cyan) in the direction specified by red arrows. (b) The residuals of the pseudorange measurements with respect to ground-truth position spread across 2 min (receiver clock drift errors removed via filtering). Multiple measurements demonstrate bias errors due to multipath and non-line-of-sight effects in the urban environment

inter-constellation clock error differences by subtracting the initial measurement residuals from all measurements. The adjustment ensured that the measurements had zero residuals at the initial position, enhancing the accuracy of subsequent calculations. Figure 3b illustrates the measurement residuals with respect to the ground-truth position in a dense urban region over a duration of 2 minutes.

In our state-space model, we considered the vehicle position (x, y), heading θ , and clock bias error β . To mitigate the impact of clock drift, β represents the difference between the true receiver clock bias and a simple linear drift model with precomputed drift rate based on initial measurement samples. We estimated this difference across time instants without explicitly tracking the clock drift, thereby maintaining a manageable state size for computational tractability. It should be noted that in practical applications, the clock parameters can be determined separately using least-squares techniques, while our approach can focus on tracking the positioning parameters for efficient computation. In our approach, we employed 1000 particles to accommodate the larger state space compared to the simulated scenario. Additionally, we utilized 5 iterations of weighting in our approach, which was determined empirically to yield small positioning errors in the given scenario.

6.1.3 Baselines

For evaluating the localization performance, we consider two baselines:

1. *Joint state-space particle filter (J-PF)*: J-PF is based on the particle filter algorithm proposed in [41]. In J-PF, both the vehicle state and the measurement fault vector are considered in the particle filter state-space. This is in contrast to traditional particle filter, which only consider the vehicle state. We refer to the combined state-space comprising of all the parameters as the joint state-space.

Since J-PF independently considers both the measurement faults and the state, it can also be interpreted as a bank of particle filters [51, 53]. This means that J-PF is able to track multiple hypotheses of faults simultaneously, each using a separate particle filter with its own probability distribution of the vehicle state. However, the computational requirement of J-PF is combinatorial in the number of considered measurement faults, increasing exponentially as the number of faults increase.

In our experiments, we limit the algorithm to consider at most two faults. Furthermore, we assume a uniformly distributed fault transition probability in the particle filter.

2. *Kalman filter RAIM (KF-RAIM)*: KF-RAIM is a residual-based RAIM algorithm [38] that uses a Kalman filter to track the state of the vehicle. The algorithm is initialized with the ground-truth position and velocity of the vehicle with a specified standard deviation σ_{init} . The algorithm then iteratively detects and removes measurement faults using a global and a local test, similar to existing approaches [22]. The local test is repeated until the global test succeeds and declares the estimated position safe to use. The algorithm is terminated if the global test fails or if the number of iterations exceeds a specified maximum value. We set the maximum number of iterations

to 5 in our experiments. Furthermore, the detection thresholds have been tuned to achieve the best localization performance on the dataset.

6.1.4 Metrics

For evaluating the positioning performance of our approach, we compute the metrics of horizontal root-mean-square error (RMSE) as well as the percentage of estimates that deviate more than 15 m from the ground truth (%>15 m). We compute these metrics as

$$\text{RMSE} = \sqrt{\frac{1}{T} \sum_{t=1}^T \|x_t - x_t^*\|_{\text{pos}}^2} \quad (49)$$

$$\%>15 \text{ m} = \frac{N(\|x_t - x_t^*\| > 15)}{T} \quad (50)$$

where T denotes the total number of time steps for which the algorithm is run; x_t^* denotes the ground-truth state of the receiver; $\|\cdot\|_{\text{pos}}$ denotes the Euclidean norm of the horizontal position coordinates in state; and $N(I)$ denotes the number of occurrences of event I . The metric %>15 m is analogous to probability of positioning failure, defined in Section 1.2 with an alarm limit of 15 m.

6.1.5 Localization performance

In the simulated case, we evaluated the localization performance on scenarios with varying number of available and faulty GNSS measurements at each time instant. The total number of measurements and the maximum number of faults considered in our analysis were as follows: (total # of measurements, max # of faults): (5, 1), (5, 2), (7, 4), (10, 6). To induce noise degradation, we introduced a bias error of 100 m with a standard deviation of 5 m to the faulty GNSS measurements. The localization metrics were computed based on 50 full runs, each with a duration 400 s and utilizing different randomly generated trajectories.

The trajectories for two selected scenarios, (5, 1) and (10, 6), are presented in Fig. 4, illustrating the comparison between our approach, J-PF, and KF-RAIM. The corresponding localization metrics are recorded in Table 2. The qualitative and quantitative results

Table 2 Comparison of localization performance on the simulated dataset

		Few faults		Many faults	
		(5, 1)	(5, 2)	(7, 4)	(10, 6)
KF-RAIM	RMSE(m)	7.6	21.5	34.3	34.4
	%>15 m	5.9	68.7	87.4	91.1
J-PF	RMSE(m)	4.8	5.8	33.5	22.9
	%>15 m	1.2	3.1	94.2	85.4
Ours	RMSE(m)	11.0	12.4	13.2	12.4
	%>15 m	23.4	26.6	33.1	28.7

The dataset consists of varying total number of measurements and maximum number of measurement faults (total # of measurements, max # of faults). Our approach demonstrates better localization performance in scenarios with large number of faults in comparison with J-PF and KF-RAIM. Bold values indicate the best performance in each category

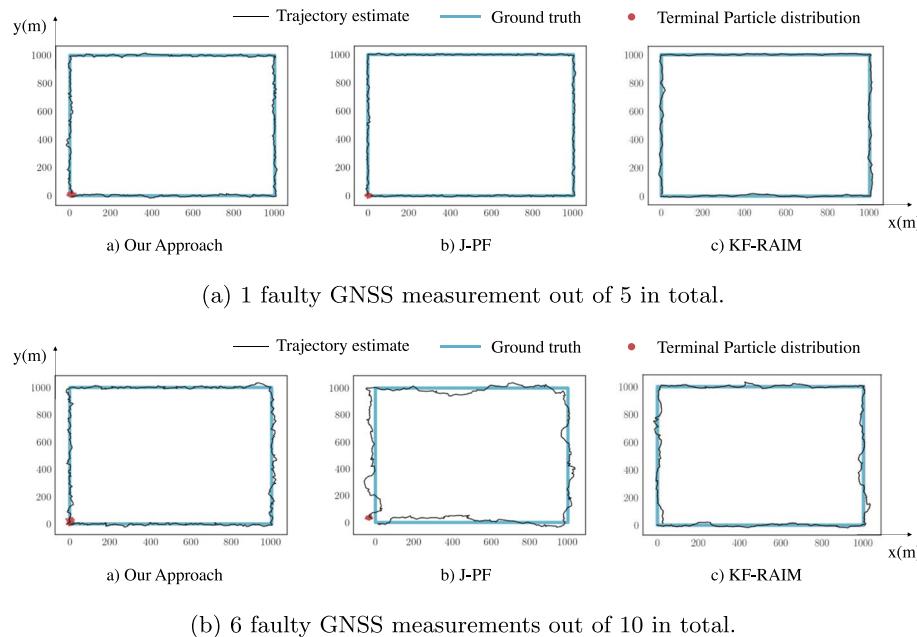


Fig. 4 Localization accuracy comparison between our approach, J-PF, and KF-RAIM approaches for (i) few-fault scenario and (ii) many-fault scenario. We fix the underlying ground-truth trajectory to be a square of side 1000 m with (0, 0) as both the start and end positions. In single fault scenario, all approaches exhibit similarly small positioning errors. Our approach estimates the vehicle state with better accuracy than J-PF and KF-RAIM in scenarios with many faulty measurements

clearly indicate that our approach outperforms the compared approaches in scenarios with high levels of noise degradation while maintaining a low rate of large positioning errors. Specifically, J-PF demonstrates better performance than its counterparts in scenarios with a single fault, as it considers all possibilities of faults separately. However, its performance deteriorates in scenarios with more than two measurement faults. Similarly, KF-RAIM excels in identifying and removing faults in scenarios with few faulty measurements, resulting in lower positioning errors, but exhibits poor performance in many-fault scenarios.

Figure 5 presents a visualization of the estimated vehicle trajectories and positioning mean-squared error over time based on real-world data. Our approach produces trajectories that closely align with the ground-truth trajectory compared to J-PF and KF-RAIM. The horizontal RMSE values, averaged over 20 runs with different randomness in initialization and propagation, indicate that our approach achieves better localization accuracy. Notably, in regions where multiple measurements are affected by bias errors, our approach outperforms the baselines by assigning high measurement weights to the measurements consistent with the particle filter's probability distribution of the vehicle state.

In summary, our approach demonstrates superior localization performance compared to J-PF and KF-RAIM in scenarios with a significant number of faulty measurements. The qualitative and quantitative assessments consistently highlight the efficacy of our approach in mitigating the adverse effects of noise degradation while maintaining accurate and reliable positioning estimates.

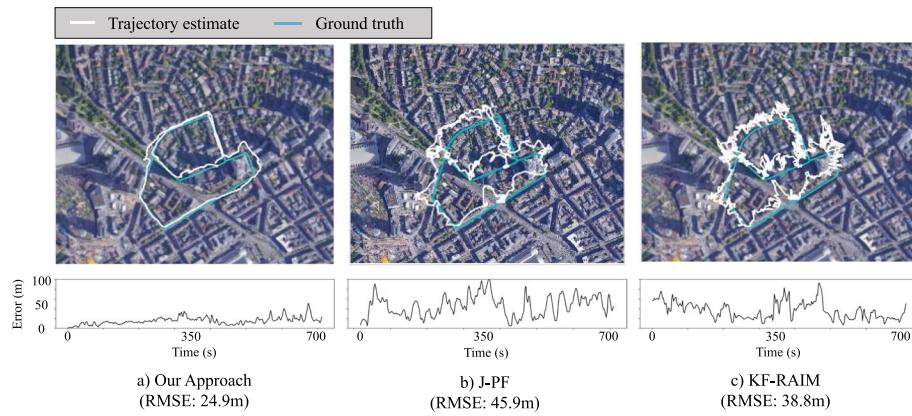


Fig. 5 Estimated vehicle trajectories and mean-squared errors on real-world data. Our approach (a) produces trajectories that consistently have smaller positioning errors as compared to trajectory from J-PF (b) and KF-RAIM (c). The horizontal RMSE values are computed by averaging over 20 runs with different randomness values in initialization and propagation. In regions where multiple measurements contaminated with bias errors, our approach is able to localize better than the baselines by assigning high measurement weights to measurements that are consistent with the particle filter probability distribution of vehicle state

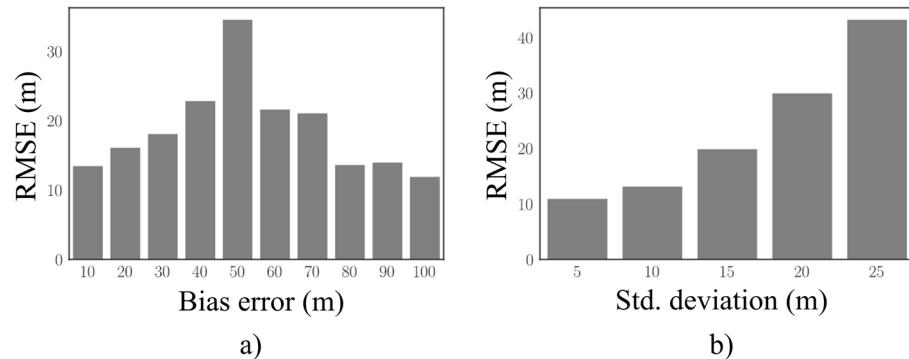


Fig. 6 Sensitivity analysis of the localization performance of our approach with varying values of GNSS measurements: **a** bias and **b** standard deviation. For low values of bias, the faults do not significantly impact the navigation solution, resulting in low RMSE values. Beyond the value of 50 m, our approach is able to remove the impact of faulty measurements resulting in low RMSE values. The performance of the algorithm increasingly deteriorates with higher standard deviation in GNSS measurement noise since both the capability to remove faulty measurements and localization are hampered by an increased noise

6.1.6 Sensitivity analysis on simulated scenario

In this experiment, we conducted a sensitivity analysis to examine the impact of GNSS measurement noise bias and standard deviation on the performance of our algorithm. We conducted two separate studies: one varying bias while keeping the standard deviation fixed, and the other varying the standard deviation while keeping the bias fixed.

In the first study, we fixed the GNSS measurement standard deviation at 5 m and varied the bias between 10 and 100 m in increments of 10 m. For each bias value, we performed 20 runs and computed RMSE as the performance metric. The results of this study are presented in the first plot of Fig. 6.

The analysis reveals that the localization performance of our approach initially deteriorates and then improves as the bias value increases. For low bias error values, the presence of faulty GNSS measurements has a minimal impact on the localization solution, even if the algorithm fails to remove them. However, beyond a bias error value of 50 m, our approach successfully assigns lower measurement weights to faulty measurements, effectively mitigating their influence on the localization solution. Consequently the RMSE decreases for higher bias error values, indicating improved performance.

In the second study, we kept the bias fixed at 100 m and varied the standard deviation between 5 and 25 m in increments of 5 m. Similarly, we conducted 20 runs for each standard deviation value and computed the RMSE. The results are presented in the second plot of Fig. 6.

The findings from this study demonstrate the impact of high measurement noise standard deviation on our approach. As the standard deviation increases, the RMSE exhibits a super-linear increase. The performance deteriorates with higher noise standard deviation since both the fault mitigation capability and the localization accuracy are adversely affected by increased measurement noise. Building upon this work, future research could integrate the proposed method with more robust state estimation methods, such as factor graphs, to address high-noise situations more effectively.

Overall, the sensitivity analysis highlights the importance of managing GNSS measurement noise bias and standard deviation in our approach. The results indicate that our algorithm is capable of effectively mitigating the impact of faulty measurements and achieving accurate localization when the bias error is relatively high. Moreover, the analysis emphasizes the sensitivity of our approach to increased measurement noise standard deviation, as it negatively affects both fault mitigation and localization performance. These insights provide valuable guidance for optimizing the performance of our algorithm in practical GNSS-based localization applications.

6.2 Integrity monitoring performance

6.2.1 Simulated experiment scenario

Due to the difficulty and scope limitations in obtaining a large amount of real-world data for statistical validation, we focused our primary analysis of integrity monitoring performance on simulated data. For the simulated scenario, we generated 400-s-long trajectories with GNSS pseudorange measurements acquired at a rate of 1 Hz (Fig. 7a). The GNSS pseudorange measurements were simulated from 10 satellites. To introduce faults, we added bias errors to up to 60% of the available measurements between 125 and 175 s, resulting in faulty measurements that deviate from the ground-truth position. The position offset for each fault was randomly selected between 50 and 150 m across different runs. For the simulations, we only utilized GNSS pseudorange measurements and did not simulate any odometry measurements. Therefore, the localization was performed solely using the GNSS pseudorange measurements for all the filters. This leads to a more challenging setting where the filters observe frequent positioning failures, and hence the ability of the integrity monitoring algorithm can be properly evaluated. In the particle filter algorithm, we set the propagation noise standard deviation to 20 m to ensure the tracking of the position even in the absence of odometry measurements.

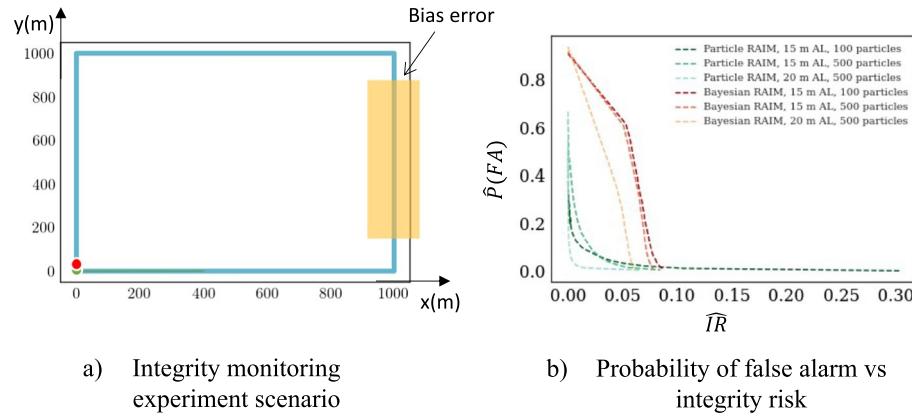


Fig. 7 **a** Simulation experiment for analyzing integrity monitoring performance. The start and end of the 400-s-long trajectory are marked in green and red, respectively. Bias errors are induced in upto 60% of the total 10 simulated GNSS measurements between 125 and 175 s (yellow region). **b** Minimum attained probability of false alarm and integrity risk across different threshold values for our approach and Bayesian RAIM [41] for different number of particles (100, 500) and alarm limit (15 m, 20 m). The different methods are denoted by the triplet (approach, alarm limit, particles). Probability of false alarm and integrity risk are estimated from more than 10^4 samples across multiple runs. Our approach exhibits lower false alarms and smaller integrity risk than Bayesian RAIM across all thresholds for each configuration of the number of particles and alarm limit

6.2.2 Baseline

To evaluate the integrity monitoring performance of our particle filter-based method, we compared our approach against the particle filter-based Bayesian RAIM [41], which detects positioning failures based on statistics computed by integrating the J-PF probability distribution. We note that other integrity monitoring methods have been explored in research, such as methods based on Kalman filter innovations [24], however they cannot be readily applied to our particle filter setting and therefore have not been compared against.

6.2.3 Metrics

The integrity monitoring performance metrics we employed evaluate the capability of our integrity monitor in triggering the alarm when it is under positioning failures. The performance metrics include the estimated probability of false alarm $\hat{P}(\text{FA})$ and empirical integrity risk \hat{IR} (analogous to probability of missed detection) which are computed as:

$$\hat{P}(\text{FA}) = \frac{N(I_{\text{al}} \cap \overline{I}_{\text{ho}})}{T}, \quad (51)$$

$$\hat{IR} = \frac{N(\overline{I}_{\text{al}} \cap I_{\text{ho}})}{T}, \quad (52)$$

where $I_{\text{al}}(\overline{I}_{\text{al}})$ denotes the event when the navigation system triggers the alarm (does not trigger alarm) and $I_{\text{ho}}(\overline{I}_{\text{ho}})$ denotes the event of positioning failures (nominal operations). These metrics are sensitive to the choice of threshold used in the integrity monitoring algorithm. For instance, if a large threshold is used, fewer detections would occur and correspondingly the likelihood of missing a correct detection would increase,

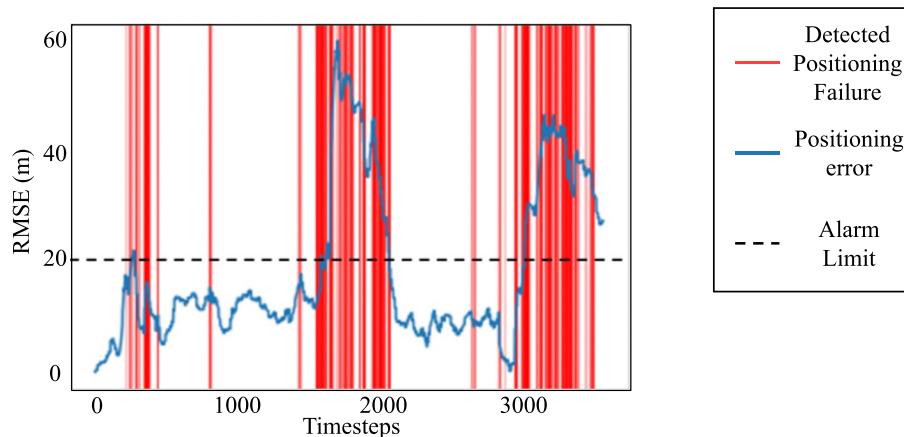


Fig. 8 Detection of positioning failures using our method, visualized against positioning errors on real-world data. Our method is able to detect positioning failures at times where position error exceeds the alarm limit (shown as the dotted line), highlighting its potential for real-world integrity monitoring. However, periods of missed detections exist between correct ones, underlining the need for future research to enhance the algorithm's ability to consistently detect positioning failures

Table 3 Comparative analysis of $\hat{P}(\text{FA})$ and $\hat{I}\hat{R}$ between Bayesian RAIM and our approach on real-world data

	$\hat{P}(\text{FA})$	$\hat{I}\hat{R}$
Bayesian RAIM	0.07	0.80
Our approach	0.06	0.58

Our method maintains lower integrity risk without compromising on the rate of false alarms, highlighting its potential for reliable integrity monitoring in urban environments

driving up $\hat{I}\hat{R}$. Conversely, if a small threshold is used, several detections will occur that might reduce $\hat{I}\hat{R}$, but would increase $\hat{P}(\text{FA})$.

Both Bayesian RAIM [41] and our approach rely on different thresholds to determine positioning failures. Optimal threshold values for minimum false alarms and missed detections depend on the scenario and algorithm. Varying these threshold values results in a trade-off between $\hat{P}(\text{FA})$ and $\hat{I}\hat{R}$. To compare the performance of the integrity monitor, we computed $\hat{P}(\text{FA})$ and $\hat{I}\hat{R}$ across different values for two settings: the number of particles (100, 500) and the alarm limit (10 m, 15 m) (Fig. 7b). All the metrics were calculated using more than 10^4 samples across multiple simulation runs for each algorithm. The results b) demonstrate that our approach consistently generates lower false alarms and missed-identifications compared to Bayesian RAIM across different threshold values for each considered parameter setting.

6.2.4 Real-world experiment

To complement our simulation-based experiments and address the limitations of simulation fidelity, we additionally tested our integrity monitoring algorithm on real-world data utilized in the localization experiments. Given the small size of the dataset which limits the independent samples available for evaluating the integrity

monitor, we executed the filtering algorithms multiple times, aggregating 10^4 samples for each algorithm while keeping the measurement data constant. The alarm limit for the experiments was set to 20 m, and the parameters were tuned to achieve similar false alarms $\hat{P}(\text{FA})$ for ease of comparison. The results, detailed in Table 3, present the estimated probabilities of false alarms $\hat{P}(\text{FA})$ and integrity risk \hat{IR} . Notably, our method achieved a lower integrity risk at a comparable rate of false alarms, demonstrating its efficacy.

However, at this false alarm configuration ($< 10\%$), the integrity risk remains high for both algorithms ($> 50\%$). This is because the algorithm struggles to consistently detect positioning failures with the changing filter distribution, even though it detects the events at several timesteps. Figure 8 illustrates this limitation, showing our algorithm's detections (tuned for higher false alarm allowance at 30% for clarity) alongside position errors and alarm limit. While detections align with periods when errors surpass the alarm limit, validating our method's sensitivity to such events, missed detections between correct detections lead to higher \hat{IR} and indicate areas for improvement. Future research can explore temporal consistency in detections to enhance the algorithm's performance in identifying positioning failures.

6.3 Discussion

The analysis and experiments conducted on both simulated and the real-world driving data indicate several key findings regarding the performance of our algorithm.

First, our framework demonstrated lower positioning errors compared to existing approaches in environments with a high fraction of faulty GNSS measurements. This highlights the effectiveness of our fault mitigation technique and the ability to accurately estimate the vehicle's state even in the presence of multiple faults.

Second, our framework exhibits superior performance in detecting hazardous operating conditions compared to existing particle filter-based integrity monitoring algorithms. The integrity monitoring component successfully identifies situations where positioning error exceeds the alarm limit, ensuring that hazardous operating conditions are detected in a timely manner.

However, there are some limitations and areas for improvement in our framework. In scenarios with few faults, the performance may be inferior to existing approaches. This is due to the conservative design choices made in the framework, particularly in the design of the fault-robust optimization problem and the associated GMM measurement likelihood model. While the conservative design ensures robust state estimation in challenging urban scenarios with multiple faults, it may result in poor performance when the number of faults is small. Exploring hybrid approaches that can switch between localization algorithms may be a promising avenue for future research to address this limitation.

Another drawback of our framework is the generation of false alarms when determining system availability. This is primarily caused by the large uncertainty within the GMM likelihood components, which leads to a conservative estimate of the probability of positioning failures. To mitigate this issue, future work can explore methods to reduce uncertainty by incorporating additional measurements and sources of information such

as camera images, carrier-phase data, and road networks. Incorporating these additional sources of information can help improve the accuracy in detecting positioning failure and reduce false alarm rates.

7 Conclusion

In this paper, we presented a novel probabilistic framework for fault-robust localization and integrity monitoring in challenging scenarios with faults in several GNSS measurements. The presented framework leverages GNSS and odometry measurements to compute a fault-robust probability distribution of the position and declares the navigation system unavailable if a reliable position cannot be estimated. We employed a particle filter for state estimation and developed a novel GMM likelihood for computing particle weights from GNSS measurements while mitigating the impact of measurement errors due to multipath and NLOS signals. Our approach for mitigating these errors is based on the expectation-maximization algorithm and determines the GMM weight coefficients and particle weights in an iterative manner. To determine the system availability, we derived the probability of positioning failures and precision that are compared with specified reference values. Through a series of experiments on challenging simulated and real-world urban driving scenarios, we have shown that our approach achieves lower positioning errors in state estimation as well as small probability of false alarm and integrity risk in integrity monitoring when compared to the existing particle filter-based approach. Furthermore, our approach is capable of mitigating multiple measurement faults with lower computation requirements than the existing particle filter-based approaches. We believe that this work offers a promising direction for real-time deployment of algorithms in challenging urban environments.

Acknowledgements

The authors would like to acknowledge Ashwin V. Kanhere and the rest of NAVLab for proofreading and reviewing this work.

Authors' contributions

SG conceived the idea and wrote the manuscript and implemented the experiments; and GG finalized the manuscript write-up, proofread and checked the technical correctness of the manuscript, and provided future perspectives of the research. All authors read and approved the final manuscript.

Funding

This material is based upon work supported by the National Science Foundation under award #2006162.

Availability of data and materials

Not applicable.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no conflict of interest.

Received: 7 November 2023 Accepted: 9 April 2024

Published online: 20 April 2024

References

1. J.S. Ahn, R. Rosihan, D.H. Won et al., GPS integrity monitoring method using auxiliary nonlinear filters with log likelihood ratio test approach. *J. Electr. Eng. Technol.* **6**(4), 563–572 (2011). <https://doi.org/10.5370/JEET.2011.6.4.563>
2. A. Angrisano, S. Gaglione, C. Gioia, Performance assessment of GPS/GLONASS single point positioning in an urban environment. *Acta Geod. Geophys.* **48**(2), 149–161 (2013). <https://doi.org/10.1007/s40328-012-0010-4>
3. S. Bhattacharyya, Performance analyses of a RAIM algorithm for kalman filter with GPS and NavIC constellations. *Sensors* **21**(24), 8441 (2021). <https://doi.org/10.3390/s21248441>
4. C. Boucher, A. Lahrech, J.C. Noyer, Non-linear filtering for land vehicle navigation with GPS outage, in *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, vol. 2 (IEEE, The Hague, Netherlands, 2004), pp. 1321–1325. <https://doi.org/10.1109/ICSMC.2004.1399808>
5. R.G. Brown, A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods. *Navigation* **39**(3), 301–316 (1992). <https://doi.org/10.1002/j.2161-4296.1992.tb02278.x>
6. R.G. Brown, P.W. McBURNIEY, Self-contained GPS integrity checking using maximum solution separation. *Navigation* **35**(1), 41–53 (1988). <https://doi.org/10.1002/j.2161-4296.1988.tb00939x>
7. Z. Chen et al., Bayesian filtering: from Kalman filters to particle filters, and beyond. *Statistics* **182**(1), 1–69 (2003)
8. O.G. Crespillo, D. Medina, J. Skaloud et al., Tightly coupled GNSS/INS integration based on robust M-estimators, in *2018 IEEE/ION position, location and navigation symposium (PLANS)* (IEEE, Monterey, CA, 2018) (2018). <https://doi.org/10.1109/PLANS.2018.8373551>
9. J.A. Del Peral-Rosado, R.E. I Castillo, J. Miguez-Sanchez et al. Performance analysis of hybrid GNSS and LTE localization in urban scenarios, in *2016 8th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)* (IEEE, Noordwijk, 2016), pp. 1–8. <https://doi.org/10.1109/NAVITEC.2016.7849332>, <http://ieeexplore.ieee.org/document/7849332/>
10. P.A.M. Dirac, *The Principles of Quantum Mechanics*, vol. 27 (Oxford University Press, 1981)
11. A. Doucet, A.M. Johansen, A tutorial on particle filtering and smoothing: Fifteen years later, in *Handbook of Nonlinear Filtering* **12**(656–704), 3 (2009)
12. S. Gaglione, A. Innac, S. Pastore Carbone et al. Robust estimation methods applied to GPS in harsh environments, in *2017 European Navigation Conference (ENC)* (IEEE, Lausanne, Switzerland, 2017), pp 14–25. <https://doi.org/10.1109/EURONAV.2017.7954169>, <http://ieeexplore.ieee.org/document/7954169/>
13. D.N. Geary, G.J. McLachlan, K.E. Basford, Mixture models: inference and applications to clustering. *J. R. Stat. Soc. Ser. A (Stat. Soc.)* **152**(1), 126 (1989). <https://doi.org/10.2307/2982840>
14. M.S. Grewal, Space-based augmentation for global navigation satellite systems. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **59**(3), 497–503 (2012). <https://doi.org/10.1109/TUFFC.2012.2220>
15. A. Grosch, O. Garcia Crespillo, I. Martini et al. Snapshot residual and Kalman Filter based fault detection and exclusion schemes for robust railway navigation, *2017 European navigation conference (ENC)* (IEEE, Lausanne, Switzerland, 2017), pp. 36–47. <https://doi.org/10.1109/EURONAV.2017.7954171>
16. S. Gupta, G.X. Gao, Particle RAIM for integrity monitoring, *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation* (Miami, Florida, 2019), pp. 811–826. <https://doi.org/10.33012/2019.16939>, <https://www.ion.org/publications/abstract.cfm?articleID=16939>
17. F. Gustafsson, F. Gunnarsson, N. Bergman et al., Particle filters for positioning, navigation, and tracking. *IEEE Trans. Signal Process.* **50**(2), 425–437 (2002). <https://doi.org/10.1109/78.978396>
18. S. Hewitson, J. Wang, Extended receiver autonomous integrity monitoring (eRAIM) for GNSS/INS integration. *J. Surv. Eng* **136**(1), 13–22 (2010). [https://doi.org/10.1061/\(ASCE\)0733-9453\(2010\)136:1\(13\)](https://doi.org/10.1061/(ASCE)0733-9453(2010)136:1(13))
19. Z. Hongyu, H. Li, L. Jing, An optimal weighted least squares RAIM algorithm, in *2017 Forum on Cooperative Positioning and Service (CPGPS)* (IEEE, Harbin, China, 2017) pp. 122–127. <https://doi.org/10.1109/CPGPS.2017.8075109>
20. L.T. Hsu, F. Chen, S. Kamijo, Evaluation of multi-GNSSs and GPS with 3D map methods for pedestrian positioning in an urban canyon environment. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **E98.A**(1), 284–293 (2015). <https://doi.org/10.1587/transfun.E98.A.284>
21. L.T. Hsu, Y. Gu, S. Kamijo, NL0S correction/exclusion for GNSS measurement using RAIM and city building models. *Sensors* **15**(7), 17329–17349 (2015). <https://doi.org/10.3390/s150717329>
22. Y. Jiang, S. Pan, Q. Meng et al., Multiple faults detection algorithm based on REKF for GNSS/INS integrated navigation, in *International Conference on Guidance* (Springer, Navigation and Control, 2022), pp. 886–894
23. M. Joerger, B. Pervan, Sequential residual-based RAIM, *Proceedings of the 23rd International Technical Meeting of the Satellite Division of the Institute Of Navigation (ION GNSS 2010)*, pp. 3167–3180 (2010)
24. M. Joerger, B. Pervan, Kalman filter residual-based integrity monitoring against measurement faults, in *AIAA guidance, navigation, and control conference. american institute of aeronautics and astronautics, Minneapolis, Minnesota* (2012). <https://doi.org/10.2514/6.2012-4450>
25. M. Joerger, B. Pervan, Solution separation and Chi-Squared ARAIM for fault detection and exclusion, *2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014* (IEEE, 2014), pp. 294–307
26. E. Kaplan, C. Hegarty, *Understanding GPS: Principles and Applications* (Artech House, 2005)
27. C.D. Karlgaard, Nonlinear regression Huber–Kalman filtering and fixed-interval smoothing. *J. Guid. Control Dyn.* **38**(2), 322–330 (2015). <https://doi.org/10.2514/1.G000799>
28. Y.C. Lee, Others Analysis of range and position comparison methods as a means to provide GPS integrity in the user receiver, in *Proceedings of the 42nd annual meeting of the Institute of Navigation* (Citeseer, 1986), pp. 1–4
29. H. Leppakoski, H. Kuusniemi, J. Takala, RAIM and complementary kalman filtering for GNSS reliability enhancement, in *2006 IEEE/ION position, location, and navigation symposium (ION GNSS 2006)* (IEEE, Coronado, CA, 2006), pp. 948–956. <https://doi.org/10.1109/PLANS.2006.1650695>
30. J. Lesouple, T. Robert, M. Sahmoudi et al., Multipath mitigation for GNSS positioning in an urban environment using sparse estimation. *IEEE Trans. Intell. Transp. Syst.* **20**(4), 1316–1328 (2019). <https://doi.org/10.1109/TITS.2018.2848461>
31. F.G. Lether, A generalized product rule for the circle. *SIAM J. Numer. Anal.* **8**(2), 249–253 (1971). <https://doi.org/10.1137/0708025>

32. Z. Li, D. Song, F. Niu et al. RAIM algorithm based on robust extended Kalman particle filter and smoothed residual, in *Lecture Notes in Electrical Engineering* (2017). https://doi.org/10.1007/978-981-10-4591-2_17
33. C. Macabiau, B. Gerfault, I. Nikiforov et al. RAIM performance in presence of multiple range failures, in *Proceedings of the 2005 National Technical Meeting of The Institute of Navigation* (2005), pp. 779–791
34. D.A. Medina, M. Romanovas, I. Herrera-Pinzon et al. Robust position and velocity estimation methods in integrated navigation systems for inland water applications, in *2016 IEEE/ION position, location and navigation symposium (PLANS)* (IEEE, Savannah, GA, 2016), pp. 491–501. <https://doi.org/10.1109/PLANS.2016.7479737>. <http://ieeexplore.ieee.org/document/7479737/>
35. S. Miura, S. Hisaka, S. Kamijo, GPS multipath detection and rectification using 3D maps, in *16th international IEEE conference on intelligent transportation systems (ITSC 2013)* (IEEE, The Hague, Netherlands, 2013), pp. 1528–1534. <https://doi.org/10.1109/ITSC.2013.6728447>
36. Y.J. Morton, F.S.T. Van Diggelen, J.J. Spilker et al. (eds.), *Position, Navigation, and Timing Technologies in the 21st Century: Integrated Satellite Navigation, Sensor Systems, and Civil Applications, First Edition* (Wiley/IEEE Press, Hoboken, 2020)
37. D. Panagiotakopoulos, A. Majumdar, W.Y. Ochieng, Extreme value theory-based integrity monitoring of global navigation satellite systems. *GPS Solut.* **18**(1), 133–145 (2014). <https://doi.org/10.1007/s10291-013-0317-9>
38. B.W. Parkinson, P. Axelrad, Autonomous GPS integrity monitoring using the pseudorange residual. *Navigation* **35**(2), 255–274 (1988). <https://doi.org/10.1002/j.2161-4296.1988.tb00955.x>
39. B.W. Parkinson, P. Enge, P. Axelrad et al. *Global Positioning System: Theory and Applications*, vol. II (American Institute of Aeronautics and Astronautics, 1996)
40. H. Pesonen, Robust estimation techniques for GNSS positioning, in *Proceedings of NAV07-The Navigation Conference and Exhibition*, issue: 1.11 (2007)
41. H. Pesonen, A framework for Bayesian receiver autonomous integrity monitoring in urban navigation. *Navigation* **58**(3), 229–240 (2011). <https://doi.org/10.1002/j.2161-4296.2011.tb02583.x>
42. S. Pullen, T. Walter, P. Enge, SBAS and GBAS integrity for non-aviation users: moving away from “specific risk”, in *Proceedings of the 2011 International Technical Meeting of The Institute of Navigation* (2011), pp. 533–545
43. P. Reisdorf, T. Pfeifer, J. Breßler et al. The problem of comparable GNSS results—an approach for a uniform dataset with low-cost and reference data, in *Proceedings of International Conference on Advances in Vehicular Systems, Technologies and Applications (VEHICULAR)* (2016)
44. G. Siegert, P. Banys, C.S. Martinez et al. EKF based trajectory tracking and integrity monitoring of AIS data, in *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)* (IEEE, Savannah, GA, 2016), pp. 887–897. <https://doi.org/10.1109/PLANS.2016.7479784>. <http://ieeexplore.ieee.org/document/7479784/>
45. M.K. Simon, *Probability Distributions Involving Gaussian Random Variables: A Handbook for Engineers and Scientists* (Springer, 2007)
46. M.A. Sturza, Navigation system integrity monitoring using redundant measurements. *Navigation* **35**(4), 483–501 (1988). <https://doi.org/10.1002/j.2161-4296.1988.tb00975.x>
47. Y. Sun, L. Fu, A new threat for pseudorange-based RAIM: adversarial attacks on GNSS positioning. *IEEE Access* **7**, 126051–126058 (2019). <https://doi.org/10.1109/ACCESS.2019.2939141>
48. C. Tanil, S. Khanafesh, M. Joerger et al. Sequential integrity monitoring for kalman filter innovations-based detectors, in *Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018)* (2018), pp. 2440–2455
49. R. Van Der Merwe, A. Doucet, N. De Freitas et al. The unscented particle filter, in *Advances in Neural Information Processing Systems* (2001), pp. 584–590
50. J. Vila, P. Schnitter, Expectation-maximization Gaussian-mixture approximate message passing, in *2012 46th Annual Conference on Information Sciences and Systems (CISS)* (IEEE, Princeton, NJ, USA, 2012), pp 1–6. <https://doi.org/10.1109/CISS.2012.6310932>
51. E. Wang, T. Pang, M. Cai et al., Fault detection and isolation for GPS RAIM based on genetic resampling particle filter approach. *TELKOMNIKA Indones. J. Electr. Eng.* **12**(5), 3911–3919 (2014). <https://doi.org/10.11591/telkomnika.v12i5.4879>
52. E. Wang, T. Pang, Z. Zhang et al., GPS receiver autonomous integrity monitoring algorithm based on improved particle filter. *J. Comput.* **9**(9), 2066–2074 (2014). <https://doi.org/10.4304/jcp.9.9.2066-2074>
53. E. Wang, C. Jia, G. Tong et al., Fault detection and isolation in GPS receiver autonomous integrity monitoring based on chaos particle swarm optimization-particle filter algorithm. *Adv. Sp. Res.* (2018). <https://doi.org/10.1016/j.asr.2017.12.016>
54. L. Yu, Research on GPS RAIM algorithm based on SIR particle filtering state estimation and smoothed residual. *Appl. Mech. Mater.* **422**, 196–203 (2013). <https://doi.org/10.4028/www.scientific.net/amm.422.196>
55. N. Zhu, J. Marais, D. Betaille et al., GNSS position integrity in urban environments: a review of literature. *IEEE Trans. Intell. Transp. Syst.* **19**(9), 2762–2778 (2018). <https://doi.org/10.1109/TITS.2017.2766768>

Publisher's Note

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