Improving 3D Cellular Positioning Integrity With Bayesian RAIM

Liqin Ding, Gonzalo Seco-Granados, *Fellow, IEEE*, Hyowon Kim, *Member, IEEE*, Russ Whiton,
Erik G. Ström, *Fellow, IEEE*, Jonas Sjöberg, *Member, IEEE*, and Henk Wymeersch, *Fellow, IEEE*

***Abstract*—Ensuring positioning integrity amid faulty measurements is crucial for safety-critical applications, making receiver autonomous integrity monitoring (RAIM) indispensable. This paper introduces a Bayesian RAIM algorithm with a streamlined architecture for 3D cellular positioning. Unlike traditional frequentist-type RAIM algorithms, it computes the exact posterior probability density function (PDF) of the position vector as a Gaussian mixture (GM) model using efficient message passing along a factor graph. This Bayesian approach retains all crucial information from the measurements, eliminates the need to discard faulty measurements, and results in tighter protection levels (PLs) in 3D space and 1D/2D subspaces that meet target integrity risk (TIR) requirements. Numerical simulations demonstrate that the Bayesian RAIM algorithm significantly outperforms a baseline algorithm, achieving over 50% PL reduction at a comparable computational cost.**

***Index Terms*—Bayesian inference, cellular positioning, factor graph, positioning integrity, receiver autonomous integrity monitoring (RAIM).**

I. INTRODUCTION

Ensuring integrity in positioning systems is vital for safety-critical applications such as autonomous driving, drones, and industrial automation, where position errors can have catastrophic consequences [2], [3]. Integrity refers to the trustworthiness of the position information and the system's ability to issue timely warnings when errors exceed acceptable thresholds [4, Chapter 7.5]. An important measure of integrity is integrity risk (IR), which is the probability that the error in the provided position information exceeds an acceptable tolerance without warning the user in a given period of time [2]. See [2, Section II] for additional integrity-related parameters and definitions.

With advanced cellular technologies like 5 G and beyond, cellular network-based positioning has become a promising alternative or complement to global navigation satellite systems (GNSS), particularly in environments where GNSS is unreliable, such as indoors and dense urban areas [5], [6]. While cellular positioning has been extensively studied, with numerous algorithms developed to improve positioning accuracy [7], [8], [9], integrity issues remain underexplored. Robust methods are needed to ensure cellular positioning systems meet the stringent integrity requirements of safety-critical applications [10], [11], [12].

Receiver autonomous integrity monitoring (RAIM) was originally developed for GNSS for aviation navigation and enables receivers to autonomously detect and mitigate faulty measurements [13], [14]. Traditional RAIM methods typically employ frequentist statistical approaches for fault detection and exclusion (FDE). These methods rely on measurement redundancy and perform consistency tests either in the measurement domain, such as analyzing measurement residuals [13], [14], [15], [16], or in the position domain using solution separation testing [17]. After FDE, the protection level (PL) is computed using error bounding methods to ensure that the integrity risk remains within the acceptable target integrity risk (TIR).

Applying traditional RAIM methods to cellular positioning systems presents significant challenges. On one hand, in typical cellular positioning scenarios, the number of available base stations (BSs) is often limited, reducing the measurement redundancy needed for effective FDE [18]. Additionally, the challenging propagation environments, such as urban canyons and indoor settings, introduce multipath effects and non-line-of-sight (NLoS) conditions, complicating the signal measurements [19]. Inexpensive hardware may also introduce clock biases and other signal impairments [20]. As a result, faults are more prevalent in measurements using cellular signals, and these faults often differ in nature from those in GNSS. Traditional FDE methods

---

Received 29 July 2024; revised 1 June 2025; accepted 13 August 2025. Date of publication 16 September 2025; date of current version 13 February 2026. This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme under Grant 101006664, in part by the Swedish Research Council (VR) under Grant 2023-03821, in part by the Catalan Government through the ICREA Academia Program and Grant 2021SGR00737, in part by MICIU/AEI/10.13039/501100011033 and ERDF/EU through Spanish Project under Grant PID2023-152820OB-I00, in part by the National Research Foundation of Korea funded by the Ministry of Science and ICT under Grant RS-2025-25415939, and in part by the Regional Innovation System and Education Program funded by the South Korean Ministry of Education and Digital Media City under Grant 2025-RISE-06-012. An earlier version of this paper entitled "Bayesian Integrity Monitoring for Cellular Positioning—A Simplified Case Study", was presented in part at 2023 IEEE International Conference on Communication Workshops (ICC Workshops) [DOI: 10.1109/ICCWorkshops57953.2023.10283608]. The review of this article was coordinated by Prof. Jiawen Kang. *(Corresponding author: Liqin Ding.)*

Liqin Ding was with the Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden. She is now with Ericsson Research, Ericsson AB, 223 63 Lund, Sweden (e-mail: liqin.ding@ericsson.com).

Gonzalo Seco-Granados is with the Department of Telecommunications and Systems Engineering, Universitat Autonoma de Barcelona, CERES-IEEC, 08193 Bellaterra, Spain.

Hyowon Kim is with the Department of Electronics Engineering, Chungnam National University, Daejeon 34134, South Korea.

Russ Whiton was with the Volvo Car Corporation, SE-405 31 Gothenburg, Sweden. He is now with the European Space Agency, 2200 AG Noordwijk, Netherlands.

Erik G. Ström, Jonas Sjöberg, and Henk Wymeersch are with the Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden.

Digital Object Identifier 10.1109/TVT.2025.3610075

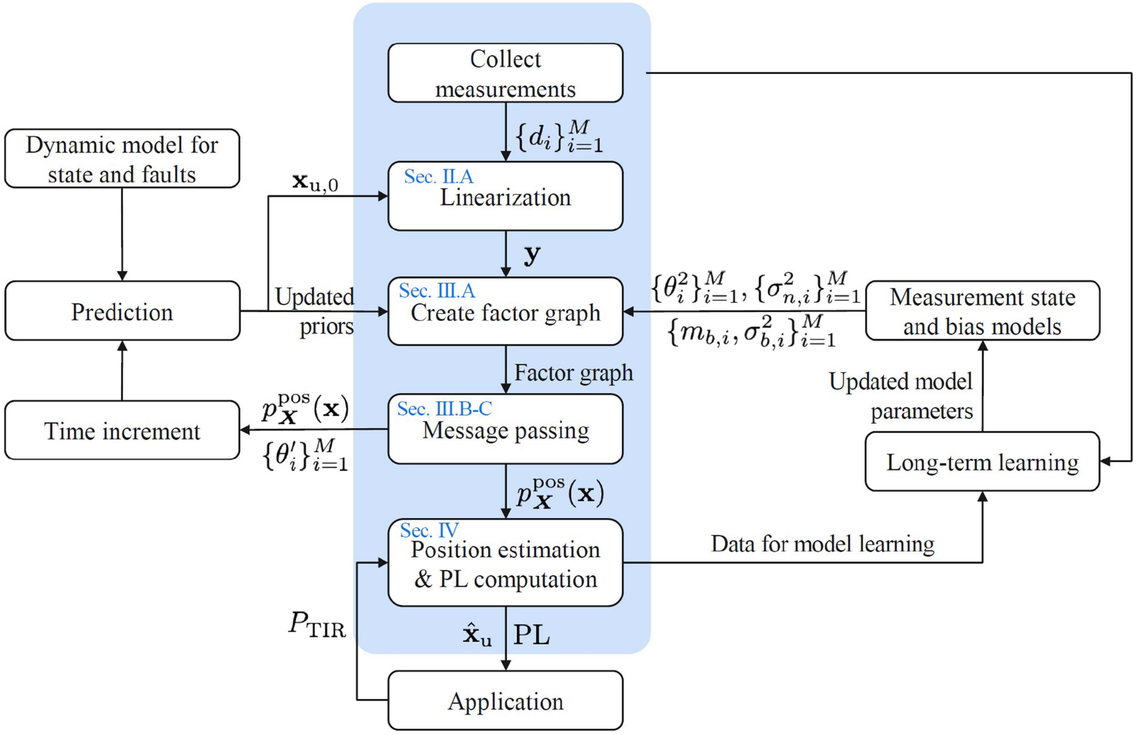


Fig. 1. Illustration of the Bayesian RAIM framework. A long-term learning module uses hyper-priors to learn fault probabilities and bias models over time. For a single positioning epoch (as highlighted in the blue box and the focus of this paper), these fault probabilities and bias models are considered known. By incorporating dynamic models for both user equipment (UE) state and faults, the factor graph approach naturally extends to dynamic scenarios, enabling sequential belief updates for UE and fault tracking. This provides initial UE state estimates and updated fault priors for each positioning epoch.

may struggle to detect and exclude such faults effectively. On the other hand, traditional RAIM methods in GNSS compute separate PLs for horizontal and vertical directions, which is critical for aviation, while cellular positioning applications require PL computation in full 3D space. Land vehicles operate on varied terrains, and accurate vertical positioning is essential in environments with stacked highways, overpasses, or multi-level structures. Drones and other airborne vehicles inherently require 3D positioning integrity [21]. Therefore, extending PL computation to 3D space, as well as to arbitrary 2D or 1D subspaces, is necessary for these applications. Moreover, excluding faulty measurements can discard valuable information, which is particularly detrimental when measurement redundancy is limited. This can lead to conservative PLs, reducing the availability of the positioning solution, as the system may declare the solution unsafe even when it is acceptable.

These challenges can potentially be addressed by the Bayesian RAIM approach [22], [23], which integrates prior information and all measurements (including potentially faulty ones) into the positioning solution. By modeling faults probabilistically within the estimation process, this approach avoids fault exclusion and yields tighter PLs, thereby improving availability. However, existing Bayesian RAIM methods often depend on computationally intensive techniques such as particle filters or Monte Carlo sampling [23], [24], [25], making them unsuitable for real-time applications.

The Bayesian RAIM framework, after adapting our proposed assumptions and methodologies for times-of-arrival (ToA) based cellular positioning, is shown in Fig. 1. It supports long-term learning of measurement state and bias models through hyper-priors and allows integration of dynamic models for cross-epoch prediction. Focusing on a single epoch (the blue box in Figure 1), in previous work [1], we prototyped a Bayesian RAIM algorithm in a 1D setting using efficient message passing on factor graphs. The computed PLs were significantly tighter than those obtained by a traditional method [17]. In this paper, we extend this methodology to 3D positioning, addressing new challenges in message passing and PL computation in higher dimensions. Specifically, we focus on (i) accurate computation of Gaussian mixture (GM) model weights, which is often ignored in previous works [26], [27] but is crucial for precise posterior estimation; (ii) efficient handling of degenerate Gaussian densities in message passing, which is essential for computational efficiency and accuracy in multi-dimensional problems; (iii) precise computation of the probability that a Gaussian random vector lies within an arbitrary sphere in $\mathbb{R}^n$ for $n \geq 2$, which allows for application-specific integrity requirements under various scenarios. Our main technical contributions are as follows:

- *Explicit Message Passing Rules for Degenerate Densities:* We introduce efficient computational rules for Gaussian message passing to ensure precise scaling factor calculations for multiple lower-dimensional measurements of a random vector via linear mapping, including rules for the inverse of linear mappings and the product of multiple, potentially degenerate, Gaussian densities.
- *Tight 2D/3D PL Computation:* We develop a numerical integration-based method to accurately assess the probability of a Gaussian-distributed random vector residing within an arbitrary ellipsoid in $\mathbb{R}^n$. This advancement enables

precise 2D/3D PL computations based on the probability density function (PDF) of the position, whether in Gaussian or GM model forms.

- *Performance Evaluation and Comparison:* Using the above methods, we develop a Bayesian RAIM algorithm for 3D positioning. Monte-Carlo simulations show that our algorithm provides significantly tighter PLs compared to a baseline RAIM algorithm adapted from [17], at a comparable computational cost using a PL overestimation method. With the developed precise PL computation method, it achieves further PL reduction, albeit with increased computational complexity.

The remainder of this paper is organized as follows. Section II presents the system assumptions and formally defines the n-dimensional PL. Section III describes the construction of the factor graph and details the message passing procedure used to derive the posterior position distribution. Section IV presents different methods for computing the PL. Section V presents a baseline RAIM algorithm that is employed for performance benchmarking. Numerical simulation results are presented in Section VI. Finally, Section VII offers the conclusions of this work.

Notations: We use uppercase letters like X for random scalars and boldface uppercase letters like $\boldsymbol{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ for random vectors. Lowercase letters like x denote deterministic scalars, bold lowercase letters like $\mathbf{x} = (x_1, \ldots, x_n)^{\mathrm{T}}$ denote deterministic vectors, and uppercase sans-serif letters like A denote deterministic matrices. We write $\mathsf{A} \succeq 0$ and $\mathsf{A} \succ 0$ for symmetric positive semidefinite and definite matrices, respectively. $\boldsymbol{X} \sim \mathcal{N}(\mathbf{x}; \mathbf{m}, \Sigma)$ indicates that $\boldsymbol{X}$ is a Gaussian vector with mean $\mathbf{m}$ and covariance $\Sigma \succeq 0$, and $X \sim \mathcal{N}(x; m, \sigma^2)$ indicates that X is a Gaussian variable with mean m and variance σ^2.

II. PROBLEM FORMULATION

A. System Assumptions and the Bayesian RAIM Framework

We consider a cellular network where M time-synchronized BSs collaborate in downlink positioning of a UE. The positions of the M BSs in the network coordinate reference system: $\mathbf{x}_i = (x_{i,1}, x_{i,2}, x_{i,3})^{\mathrm{T}} \in \mathbb{R}^3$, $i = 1, \ldots, M$, are known. In a single positioning epoch, the BSs send coordinated positioning reference signals (PRSs) to the UE to estimate M ToAs for the line-of-sight (LoS) paths. These ToAs convert to pseudoranges as $d_i = \|\mathbf{x}_i - \mathbf{x}_{\mathrm{u}}\| + x_c + b_i + n_i$, $i = 1, \ldots, M$, where $\mathbf{x}_{\mathrm{u}} = (x_{\mathrm{u},1}, x_{\mathrm{u},2}, x_{\mathrm{u},3})^{\mathrm{T}} \in \mathbb{R}^3$ and $x_c \in \mathbb{R}$ are the UE's 3D position and clock bias, n_i is measurement noise, and b_i represents measurement bias caused by faults.

As is commonly assumed, an initial position estimate $\mathbf{x}_{\mathrm{u},0}$ is available. As illustrated in Fig. 1, it can be predicted based on the dynamic state model. Linearizing $\|\mathbf{x}_i - \mathbf{x}_{\mathrm{u}}\|$ around $\mathbf{x}_{\mathrm{u},0}$ using a first-order Taylor expansion yields [20], [28], [29]

$$d_i \approx \|\mathbf{x}_i - \mathbf{x}_{\mathrm{u},0}\| + \mathbf{g}_i^{\mathrm{T}}(\mathbf{x}_{\mathrm{u}} - \mathbf{x}_{\mathrm{u},0}) + x_c + b_i + n_i, \qquad (1)$$

where $\mathbf{g}_i \triangleq (\mathbf{x}_{\mathrm{u},0} - \mathbf{x}_i)/\|\mathbf{x}_{\mathrm{u},0} - \mathbf{x}_i\|$ is the unit vector from $\mathbf{x}_{\mathrm{u},0}$ to $\mathbf{x}_i$. Letting $\mathbf{h}_i = [\mathbf{g}_i^{\mathrm{T}} \; 1]^{\mathrm{T}}$ and $\mathbf{x} = [\mathbf{x}_{\mathrm{u}}^{\mathrm{T}} \; x_c]^{\mathrm{T}}$, we establish a linear measurement model

$$y_i = \mathbf{h}_i^{\mathrm{T}}\mathbf{x} + b_i + n_i, \quad i = 1, \ldots, M, \qquad (2)$$

which serves as an approximation to $d_i - \|\mathbf{x}_i - \mathbf{x}_{\mathrm{u},0}\| + \mathbf{g}_i^{\mathrm{T}}\mathbf{x}_{\mathrm{u},0}$. Arranging the M row vectors $\mathbf{h}_i^{\mathrm{T}}$ into an $M \times 4$ matrix H and forming y_i, b_i, and n_i into length-M column vectors $\mathbf{y}$, $\mathbf{b}$, and $\mathbf{n}$, the model can be expressed in matrix form

$$\mathbf{y} = \mathsf{H}\mathbf{x} + \mathbf{b} + \mathbf{n}. \qquad (3)$$

The initial position estimate error $\mathbf{e}_0 \triangleq \mathbf{x}_{\mathrm{u}} - \mathbf{x}_{\mathrm{u},0}$ introduces approximation errors through the approximation (1), affecting the performance of any method that relies on this model.

We treat $\mathbf{x} \in \mathbb{R}^4$, b_i and n_i as realizations of random vector/variable $\boldsymbol{X} = [X_1, X_2, X_3, X_4]^{\mathrm{T}}$, B_i and N_i, respectively. To indicate the measurement state of y_i, we introduce a latent random variable Λ_i with realization λ_i, which follows the Bernoulli probability mass function (PMF) $p_{\Lambda_i}(\lambda_i) = \theta_i^{\lambda_i}(1 - \theta_i)^{(1-\lambda_i)}$ where $0 < \theta_i \ll 1$. The value $\Lambda_i = 0$ denotes a fault-free measurement with $B_i \sim \mathcal{N}(0, 0)$, and $\Lambda_i = 1$ indicates a faulty measurement with $B_i \sim \mathcal{N}(m_{b,i}, \sigma_{b,i}^2)$. Measurement noises are zero-mean Gaussian: $N_i \sim \mathcal{N}(0, \sigma_{n,i}^2)$. The random variables $\{\Lambda_i\}$, $\{B_i\}$ and $\{N_i\}$ are considered independent within each set. By applying hyper-priors under the Bayesian framework, we can continually learn and refine both these models and their parameters over time. In the context of a single positioning epoch, we that assume these models and their parameters are known.

B. n-Dimensional Protection Level

To measure the integrity of positioning results, PLs are computed under the specified TIR requirements. In the context of 3D positioning, PLs can be determined for the full 3D space and its 1D or 2D subspaces. We define n-dimensional PL ($n = 1, 2, 3$) using the 3D positioning error vector $\mathbf{e}_{\mathrm{3D}} \triangleq \mathbf{x}_{\mathrm{u}} - \hat{\mathbf{x}}_{\mathrm{u}} \in \mathbb{R}^3$, where $\hat{\mathbf{x}}_{\mathrm{u}} \triangleq \hat{\mathbf{x}}_{1:3}$ is the position estimate. Consider two orthogonal unit vectors $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2 \in \mathbb{R}^3$. The 1D subspace spanned by $\tilde{\mathbf{v}}_1$ is $\mathcal{L}(\tilde{\mathbf{v}}_1) \triangleq \{s\tilde{\mathbf{v}}_1 \mid s \in \mathbb{R}\}$, and the 2D subspace spanned by both vectors is $\mathcal{P}(\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2) \triangleq \{s\tilde{\mathbf{v}}_1 + t\tilde{\mathbf{v}}_2 \mid s, t \in \mathbb{R}\}$. The projections of $\mathbf{e}_{\mathrm{3D}}$ onto these subspaces are $\mathbf{e}_{\mathrm{1D}} = \tilde{\mathbf{v}}_1^{\mathrm{T}}\mathbf{e}_{\mathrm{3D}}$ and $\mathbf{e}_{\mathrm{2D}} = (\tilde{\mathbf{v}}_1^{\mathrm{T}}\mathbf{e}_{\mathrm{3D}}, \tilde{\mathbf{v}}_2^{\mathrm{T}}\mathbf{e}_{\mathrm{3D}})^{\mathrm{T}}$.

Definition 1: The n-dimensional PL for the position estimate $\hat{\mathbf{x}}_{\mathrm{u}}$ in $\mathcal{L}(\tilde{\mathbf{v}}_1)$ ($n = 1$), $\mathcal{P}(\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2)$ ($n = 2$), or $\mathbb{R}^3$ ($n = 3$), for a TIR of P_{TIR}, is the minimum distance r such that the actual IR, $\Pr\{\|\mathbf{e}_{n\mathrm{D}}\| > r\}$, does not exceed P_{TIR}

$$\mathrm{PL}_{n\mathrm{D}}(P_{\mathrm{TIR}}) = \min\{r \mid \Pr\{\|\mathbf{e}_{n\mathrm{D}}\| > r\} < P_{\mathrm{TIR}}\}. \qquad (4)$$

Geometrically, $\mathrm{PL}_{n\mathrm{D}}(P_{\mathrm{TIR}})$ is the radius of the smallest interval, circle, or sphere centered on the estimated position that encompasses $\mathbf{e}_{n\mathrm{D}}$ with a probability of at least $1 - P_{\mathrm{TIR}}$. For $n = 2$ or 3, $\mathrm{PL}_{n\mathrm{D}}(P_{\mathrm{TIR}})$ bounds $\mathbf{e}_{n\mathrm{D}}$ in any direction within the specified n-dimensional space. If the actual IR for a given r can be precisely or approximately computed for 1D subspaces, an overestimate of the 2D/3D PL can be determined using the following lemma.

Lemma 1: For $n = 2$ or 3, given $\mathbf{e}_{n\mathrm{D}} = (e_1, \ldots, e_n)^{\mathrm{T}}$ and weights[1] $\{w_1, \ldots, w_n\}$ such that $0 < w_i < 1$ for all i and

---

[1] Choosing $w_i = 1/n$ for all i is pragmatic when positioning error distributions are similar across coordinate directions. Adjusting these weights can refine the overestimate, leading to tighter results. Conversely, setting w_i close to 0 can significantly increase $\mathrm{PL}_{\mathrm{1D},i}$ and consequently enlarge $\mathrm{PL}_{n\mathrm{D}}^{U}$.

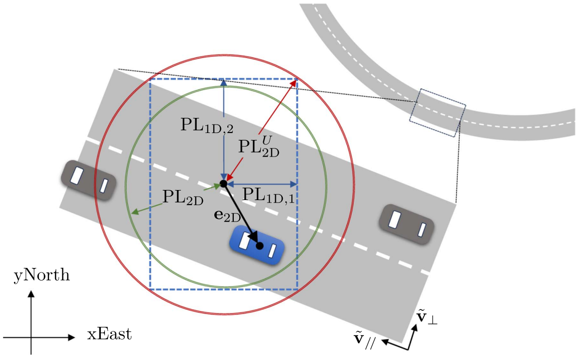


Fig. 2. Illustration of 2D PL computation on the horizontal plane.

$\sum_{i=1}^{n} w_i = 1$, if $\{\mathrm{PL}_{1\mathrm{D},1}, \ldots, \mathrm{PL}_{1\mathrm{D},n}\}$ are obtained such that

$$\mathrm{PL}_{1\mathrm{D},i} = \min\{r \mid \Pr\{|e_i| > r\} < w_i\, P_{\mathrm{TIR}}\}, \tag{5}$$

then $\Pr\{\|\mathbf{e}_{n\mathrm{D}}\| > \mathrm{PL}_{n\mathrm{D}}^{U}\} < P_{\mathrm{TIR}}$ can be guaranteed, where

$$\mathrm{PL}_{n\mathrm{D}}^{U} = \left(\sum_{i=1}^{n} \mathrm{PL}_{1\mathrm{D},i}^{2}\right)^{1/2}. \tag{6}$$

Proof: See Appendix A. □

Example 1: Fig. 2 shows a scenario where the UE is equipped in a vehicle on a curved road in the horizontal-plane (x-y plane). To ensure the vehicle remains within its lane, the position error perpendicular to the road ($\tilde{\mathbf{v}}_{\perp}$ direction) needs to stay within a margin at a confidence level of $1 - P_{\mathrm{TIR}}$. As the vehicle navigates the curve, $\tilde{\mathbf{v}}_{\perp}$ changes. If $\tilde{\mathbf{v}}_{\perp}$ is known in real-time, the positioning system can compute $\mathrm{PL}_{1\mathrm{D}}(P_{\mathrm{TIR}})$ for $\mathcal{L}(\tilde{\mathbf{v}}_{\perp})$. Otherwise, it can provide an overestimated 2D PL $\mathrm{PL}_{2\mathrm{D}}^{U} = (\mathrm{PL}_{1\mathrm{D},1}^{2} + \mathrm{PL}_{1\mathrm{D},2}^{2})^{1/2}$, where $\mathrm{PL}_{1\mathrm{D},1}$ and $\mathrm{PL}_{1\mathrm{D},2}$ are 1D PLs computed for $\tilde{\mathbf{v}}_1 = (1, 0, 0)^{\mathrm{T}}$ and $\tilde{\mathbf{v}}_2 = (0, 1, 0)^{\mathrm{T}}$, with $w_1 = w_2 = 0.5$. It is conceivable that by slightly decreasing w_1 and increasing w_2, the blue rectangle can become closer to a square, resulting in a tighter overestimate.

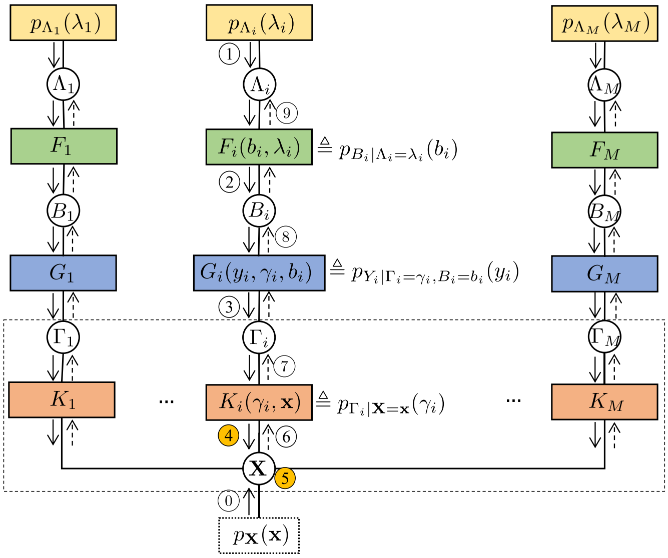
PL requirements vary by application and must align with specific operational demands, see e.g. [30, Table II]. In Fig. 2, PL_{2D}^{U} might suffice for placing a vehicle on the correct road, suitable for automatic tolling, but it cannot ensure accurate lane positioning, which is crucial for automated driving.

III. BAYESIAN RAIM PART I: MESSAGE PASSING

The linear Gaussian measurement model (2) allows a precise computation of the posterior probability distribution of $\boldsymbol{X}$, $p_{\boldsymbol{X}}^{\mathrm{pos}}(\mathbf{x})$, for each positioning epoch. This section describes the process for obtaining this distribution using message passing on a factor graph. The section concludes with several remarks, including a discussion of computational complexity and an explanation of why measurement exclusion is not recommended.

A. Factor Graph Construction and Message Passing Schedule

First, we introduce an auxiliary random variable $\Gamma_i = \mathbf{h}_i^{\mathrm{T}} \boldsymbol{X}$ for $i = 1, \ldots, M$, so that we can rewrite the measurement model (2) as $y_i = \gamma_i + b_i + n_i$, where γ_i is a realization of Γ_i. We form the sets of random variables $\{Y_i\}$, $\{B_i\}$, $\{\Gamma_i\}$, and $\{\Lambda_i\}$ into random vectors $\boldsymbol{Y}$, $\boldsymbol{B}$, $\boldsymbol{\Gamma}$, and $\boldsymbol{\Lambda}$, and their realizations into vectors $\mathbf{y}$, $\mathbf{b}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\lambda}$. As shown in Fig. 3, we factorize the joint posterior probability of $\boldsymbol{\Gamma}$, $\boldsymbol{X}$, $\boldsymbol{B}$, and $\boldsymbol{\Lambda}$ in equation $(*)$ following the assumptions in Section II-A, and form a cycle-free factor graph with M branches to represent the factorization. Each term in equation $(*)$ corresponds to a factor (function) node in the graph. For ease of description, we define



$$\begin{aligned} & p_{\boldsymbol{\Gamma},\mathbf{X},\mathbf{B},\boldsymbol{\Lambda}|\mathbf{Y}}(\boldsymbol{\gamma}, \mathbf{x}, \mathbf{b}, \boldsymbol{\lambda}) \\ \propto\, & p_{\mathbf{Y}|\boldsymbol{\Gamma}=\boldsymbol{\gamma},\mathbf{X}=\mathbf{x},\mathbf{B}=\mathbf{b},\boldsymbol{\Lambda}=\boldsymbol{\lambda}}(\mathbf{y})\, p_{\boldsymbol{\Gamma},\mathbf{X},\mathbf{B},\boldsymbol{\Lambda}}(\boldsymbol{\gamma}, \mathbf{x}, \boldsymbol{b}, \boldsymbol{\lambda}) \\ =\, & p_{\mathbf{Y}|\boldsymbol{\Gamma}=\boldsymbol{\gamma},\mathbf{B}=\mathbf{b}}(\mathbf{y})\, p_{\boldsymbol{\Gamma}|\mathbf{X}=\mathbf{x}}(\boldsymbol{\gamma})\, p_{\mathbf{X}}(\mathbf{x})\, p_{\mathbf{B}|\boldsymbol{\Lambda}=\boldsymbol{\lambda}}(\mathbf{b})\, p_{\boldsymbol{\Lambda}}(\boldsymbol{\lambda}) \qquad (*) \\ =\, & p_{\mathbf{X}}(\mathbf{x}) \prod\nolimits_{i=1}^{M} \underbrace{p_{Y_i|\Gamma_i=\gamma_i,B_i=b_i}(y_i)}_{G_i(y_i,\gamma_i,b_i)} \underbrace{p_{\Gamma_i|\mathbf{X}=\mathbf{x}}(\gamma_i)}_{K_i(\gamma_i,\mathbf{x})} \underbrace{p_{B_i|\Lambda_i=\lambda_i}(b_i)}_{F_i(b_i,\lambda_i)}\, p_{\Lambda_i}(\lambda_i) \end{aligned}$$

Fig. 3. The factor graph used by the Bayesian RAIM algorithm, constructed according to the joint posterior probability factorization given by $(*)$. The order of message computation and passing is given by the numbers in circles (shown only on the ith branch, but is the same for all branches), while the arrows indicate the passing direction.

$$G_i(y_i, \gamma_i, b_i) \triangleq p_{Y_i|\Gamma_i=\gamma_i,B_i=b_i}(y_i), \tag{7a}$$

$$K_i(\gamma_i, \mathbf{x}) \triangleq p_{\Gamma_i|\boldsymbol{X}=\mathbf{x}}(\gamma_i) = \delta(\gamma_i - \mathbf{h}_i^{\mathrm{T}} \mathbf{x}), \tag{7b}$$

$$F_i(b_i, \lambda_i) \triangleq p_{B_i|\Lambda_i=\lambda_i}(b_i). \tag{7c}$$

In equation $(*)$, $p_{\mathbf{X}}(\mathbf{x})$ represents the prior PDF of $\mathbf{X}$. Since we are focusing on a single epoch, we will omit $p_{\mathbf{X}}(\mathbf{x})$ from message passing.

Applying the sum-product rule [31] on this cycle-free factor graph leads to a straightforward message passing schedule: Messages are passed from variable node Λ_i to variable node $\mathbf{X}$ in steps ①-④ along all M branches in parallel. Then in step ⑤, variable node $\mathbf{X}$ computes the product of all received messages, which after normalization is the posterior PDF of $\mathbf{X}$. These steps are sufficient for position estimation and PL computation of the current positioning epoch. Messages can be further passed back to variable node Λ_i in steps ⑥-⑨ along each branch to obtain the posterior PMF of Λ_i.[2]

B. General Gaussian Message Computation Rules

Message computation in steps ④-⑥ is challenging because the messages consist of GM models, i.e. weighted sums of Gaussians. They require precise computation of weights, and the Gaussian densities may be degenerate with rank-deficient covariance matrices. Specifically, step ④ involves inferring a message of $\boldsymbol{X}$ from a lower-dimensional subspace GM model of Y_i, which yields degenerate Gaussian densities. Steps ⑤ and ⑥ involve computing the product of multiple such messages. Appendix B, rigorously addresses the computational rules for these steps in the following general problem setting.

Problem 1: For $i = 1, \ldots, K$, given a linear mapping $\boldsymbol{Y}_i = \mathsf{A}_i\boldsymbol{X}$, where $\boldsymbol{X} \in \mathbb{R}^n$ and $\boldsymbol{Y}_i \in \mathbb{R}^{m_i}$ are Gaussian random vectors, each matrix $\mathsf{A}_i \in \mathbb{R}^{m_i \times n}$ is full-rank[3], with $\mathrm{rank}(\mathsf{A}_i) = m_i \leq n$. The objective is to infer the PDF of $\boldsymbol{X}$ from the known non-degenerate PDF of $\boldsymbol{Y}_i$ for all i, and then compute the product of all inferred PDFs of $\boldsymbol{X}$.

By letting $\mathsf{A}_i = \mathbf{h}_i^{\mathrm{T}}$, our problem becomes a special case of this general problem. A general Gaussian distribution for a random vector $\boldsymbol{X} \in \mathbb{R}^n$ is characterized by a symmetric positive semi-definite covariance matrix $\Sigma \succeq 0$ and a mean vector $\mathbf{m} \in \mathbb{R}^n$, or equivalently, by $\mathsf{V} \triangleq \Sigma^{+}$ (the pseudo-inverse of Σ) and $\mathbf{u} \triangleq \mathsf{V}\mathbf{m}$. Notably, $\mathrm{rank}(\Sigma) = \mathrm{rank}(\mathsf{V}) \leq n$, with the $<$ relation occurring when the distribution is degenerate. The PDF expressed using $(\mathbf{m}, \Sigma)$ is denoted $f_{\boldsymbol{X}}(\mathbf{x}; \mathbf{m}, \Sigma)$ (see (35)), and when using $(\mathbf{u}, \mathsf{V})$, it is denoted $f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x}; \mathbf{u}, \mathsf{V})$ (see (37)). Moreover,

$$\alpha_{\boldsymbol{X}} \triangleq -\frac{1}{2}\mathbf{m}^{\mathrm{T}}\Sigma^{+}\mathbf{m} \equiv -\frac{1}{2}\mathbf{u}^{\mathrm{T}}\mathsf{V}^{+}\mathbf{u} \tag{8}$$

is defined to simplify expressions. The computation rules are summarized in Lemma 2, Lemma 3, and Remark 1.

Lemma 2 (Inverse of linear mapping): Consider a linear mapping $\boldsymbol{Y} = \mathsf{A}\boldsymbol{X}$, where $\boldsymbol{X} \in \mathbb{R}^n$ and $\boldsymbol{Y} \in \mathbb{R}^m$ are Gaussian random vectors, $\mathsf{A} \in \mathbb{R}^{m \times n}$, and $\mathrm{rank}(\mathsf{A}) = m \leq n$. Given a message of $\boldsymbol{Y}$: $\mu_{\boldsymbol{Y}}(\mathbf{y}) = f^{\mathrm{E}}_{\boldsymbol{Y}}(\mathbf{y}; \mathbf{u}_{\boldsymbol{Y}}, \mathsf{V}_{\boldsymbol{Y}})$ with $\mathsf{V}_{\boldsymbol{Y}} \succ 0$, the inferred message of $\boldsymbol{X}$ following the sum-product rule is given by

$$\begin{aligned}\mu_{\boldsymbol{X}}(\mathbf{x}) &= \int \delta(\mathbf{y} - \mathsf{A}\mathbf{x})\mu_{\boldsymbol{Y}}(\mathbf{y})\mathrm{d}\mathbf{y} = f^{\mathrm{E}}_{\boldsymbol{Y}}(\mathsf{A}\mathbf{x}; \mathbf{u}_{\boldsymbol{Y}}, \mathsf{V}_{\boldsymbol{Y}}), \\ &= s^{\boldsymbol{X}}_{\boldsymbol{Y}} \cdot f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}}, \mathsf{V}_{\boldsymbol{X}}),\end{aligned} \tag{9a}$$

where

$$\mathbf{u}_{\boldsymbol{X}} = \mathsf{A}^{\mathrm{T}}\mathbf{u}_{\boldsymbol{Y}}, \quad \mathsf{V}_{\boldsymbol{X}} = \mathsf{A}^{\mathrm{T}}\mathsf{V}_{\boldsymbol{Y}}\mathsf{A}, \tag{9b}$$

and the scaling factor $s^{\boldsymbol{X}}_{\boldsymbol{Y}}$ is given by

$$s^{\boldsymbol{X}}_{\boldsymbol{Y}} = (|\mathsf{V}_{\boldsymbol{Y}}|/|\mathsf{V}_{\boldsymbol{X}}|_{+})^{1/2}. \tag{9c}$$

Proof: See Appendix B-B. ☐

For $m = n$, $\mathsf{V}_{\boldsymbol{Y}} \succ 0$ ensures $\mathsf{V}_{\boldsymbol{X}} \succ 0$ (since A is full-rank), making the inverse mapping of $\boldsymbol{Y} = \mathsf{A}\boldsymbol{X}$ unique and straightforward to compute. In this case, (9a)-(9c) can be directly obtained following [32, Eq. (356), (357)].

Lemma 3 (Product of multiple Gaussian densities): The product of K Gaussian PDFs for $\boldsymbol{X}$: $f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_i}, \mathsf{V}_{\boldsymbol{X}_i})$, where $\mathrm{rank}(\mathsf{V}_{\boldsymbol{X}_i}) = k_i$, for $i = 1, \ldots, K$, is given by

$$\prod_{i=1}^{K} f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_i}, \mathsf{V}_{\boldsymbol{X}_i}) = s_{\boldsymbol{X}_{1:K}} \cdot f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}}, \mathsf{V}_{\boldsymbol{X}}) \tag{10a}$$

where

$$\mathbf{u}_{\boldsymbol{X}} = \sum_{i=1}^{K} \mathbf{u}_{\boldsymbol{X}_i}, \quad \mathsf{V}_{\boldsymbol{X}} = \sum_{i=1}^{K} \mathsf{V}_{\boldsymbol{X}_i}, \tag{10b}$$

and the scaling factor $s_{\boldsymbol{X}_{1:K}}$ is, with $k = \mathrm{rank}(\mathsf{V}_{\boldsymbol{X}})$,

$$s_{\boldsymbol{X}_{1:K}} = \frac{\prod_{i=1}^{K} |\mathsf{V}_{\boldsymbol{X}_i}|_{+}^{1/2}}{(2\pi)^{(\sum_{i=1}^{K} k_i - k)/2}|\mathsf{V}_{\boldsymbol{X}}|_{+}^{1/2}} \exp\left(-\alpha_{\boldsymbol{X}} + \sum_{i=1}^{K} \alpha_{\boldsymbol{X}_i}\right). \tag{10c}$$

Proof: See Appendix B-C. ☐

Remark 1: The objective of Problem 1 can be achieved by applying Lemma 2 and Lemma 3, with the scaling factor computations (9c) and (10c) replaced respectively by

$$s^{\boldsymbol{X}}_{\boldsymbol{Y}} = |\mathsf{V}_{\boldsymbol{Y}}|_{+}^{1/2} \tag{11}$$

$$s_{\boldsymbol{X}_{1:K}} = \frac{\exp(-\alpha_{\boldsymbol{X}} + \sum_{i=1}^{K} \alpha_{\boldsymbol{X}_i})}{(2\pi)^{(\sum_{i=1}^{K} k_i - k)/2}|\mathsf{V}_{\boldsymbol{X}}|_{+}^{1/2}}. \tag{12}$$

These replacements avoid the computation of $|\mathsf{V}_{\boldsymbol{X}_i}|_{+}$ while preserving the results.

C. Message Passing Algorithm

Applying the above computation rules, a detailed description of steps ①-⑤, and a brief description of the optional steps ⑥-⑨, are given in Appendix C. Steps ⑧ and ⑨ are identical to steps ⑤ and ⑥ in [1, Section IV.A] where more details can be found. Additionally, a few remarks should be made regarding step ⑤.

- First, when $M \geq 4$ and the measurement vectors $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_M$ are linearly independent, the matrices $\mathsf{V}^{(l)}_{\boldsymbol{X}}$, $l = 1, \ldots, L$, calculated in (45a), are guaranteed to be positive definite. Consequently, the pseudo-inverse and pseudo-determinant of $\mathsf{V}^{(l)}_{\boldsymbol{X}}$ simplify to the regular inverse and determinant, making the computation of $|\mathsf{V}^{(l)}_{\boldsymbol{X}}|_{+} \equiv |\mathsf{V}^{(l)}_{\boldsymbol{X}}|$ in (45c) straightforward. Moreover, the covariance matrix $\Sigma^{(l)}_{\boldsymbol{X}} = (\mathsf{V}^{(l)}_{\boldsymbol{X}})^{+} \equiv (\mathsf{V}^{(l)}_{\boldsymbol{X}})^{-1}$ and mean vector $\mathbf{m}^{(l)}_{\boldsymbol{X}} = (\mathsf{V}^{(l)}_{\boldsymbol{X}})^{-1}\mathbf{u}$ can also be easily computed for all l. Thus, we can rewrite (44) using Gaussian PDFs $f_{\boldsymbol{X}}(\mathbf{x}; \mathbf{m}^{(l)}_{\boldsymbol{X}}, \Sigma^{(l)}_{\boldsymbol{X}})$ instead.

---

[2] In a real-life positioning system, measurement states in consecutive epochs are likely correlated, making it reasonable to update $p_{\Lambda_i}(\lambda_i)$ for the next positioning epoch using the obtained posterior PMF. However, as this is beyond the scope of this paper, so steps ⑥-⑨ are not discussed further.

[3] This full-rank assumption does not lose generality. If A is rank-deficient, i.e., $k = \mathrm{rank}(\mathsf{A}) < \min(n, m)$, a (rank) decomposition $\mathsf{A} = \mathsf{BC}$ can be obtained (e.g., using compact singular value decomposition (SVD)) where $\mathsf{B} \in \mathbb{R}^{m \times k}$ and $\mathsf{C} \in \mathbb{R}^{k \times n}$ have full rank ($= k$). Thus, $\boldsymbol{Y} = \mathsf{A}\boldsymbol{X} = \mathsf{B}(\mathsf{C}\boldsymbol{X})$ can be treated as two consecutive linear mapping via full-rank matrices.

- Second, if a prior PDF $p_{\boldsymbol{X}}(\mathbf{x})$ is available, it should be multiplied to $\mu_{\boldsymbol{X}}(\mathbf{x})$. This computation can be easily performed if $p_{\boldsymbol{X}}(\mathbf{x})$ is a Gaussian distribution or GM.
- Third, the weights $\{w_{\boldsymbol{X}}^{(l)}\}$ given by (45b) do not necessarily sum to one, so a normalization step is required. We reuse $w_{\boldsymbol{X}}^{(l)}$ for the normalized weight to avoid introducing extra symbols. The exact posterior PDF of $\boldsymbol{X}$ can be obtained from $\mu_{\boldsymbol{X}}(\mathbf{x})$ as follows:

$$p_{\boldsymbol{X}}^{\text{pos}}(\mathbf{x}) = \sum_{l=1}^{L} w_{\boldsymbol{X}}^{(l)}\, f_{\boldsymbol{X}}(\mathbf{x}; \mathbf{m}_{\boldsymbol{X}}^{(l)}, \Sigma_{\boldsymbol{X}}^{(l)}). \tag{13}$$

1) Complexity Discussion: The computational complexity of the message passing process scales with M. The main computational load comes from step ④, which is executed in parallel for each measurement, and step ⑤, which is performed once. Among the optional steps, step ⑥ is the most demanding, followed by step ⑦. However, most operations in these steps are simple vector or matrix additions and multiplications, and many intermediate results can be reused between steps ⑤ and ⑥.

2) Measurement Exclusion Not Recommended: In [1], we proposed excluding a measurement y_i if its posterior fault probability exceeds a threshold θ_{T} and using the posterior PDF of $\boldsymbol{X}$ computed with the remaining measurements, i.e., $p_{\boldsymbol{X}}^{\text{ex}}(\mathbf{x}) \propto \prod_{j\in\mathcal{I}_{\text{ex}}^{c}} \mu_{K_j\to\boldsymbol{X}}(\mathbf{x})$, where $\mathcal{I}_{\text{ex}}^{c} \triangleq \{1 \le i \le M : \theta_i' \le \theta_{\text{T}}\}$, instead of (13) for PL computation. However, this approach can compromise integrity requirements. Indeed, measurement exclusion is unnecessary since all information about the UE position and measurement states is captured in $p_{\boldsymbol{X}}^{\text{pos}}(\mathbf{x}) \propto \prod_{j=1}^{M} \mu_{K_j\to\boldsymbol{X}}(\mathbf{x})$. In fact, if exclusion is performed, the conditional pointwise mutual information (C-PMI) between $\boldsymbol{X}$ and the excluded measurements, given the remaining measurements, must be considered to maintain integrity requirements. To elaborate, we revisit the nature of the problem: In each positioning epoch, the position $\mathbf{x}\in\mathbb{R}^n$ and the measurement $\mathbf{y}\in\mathbb{R}^M$ are generated according to their joint probability PDF $p_{\boldsymbol{X},\boldsymbol{Y}}(\mathbf{x},\mathbf{y})$. Based on the posterior PDF $p_{\boldsymbol{X}|\boldsymbol{Y}=\mathbf{y}}(\mathbf{x})$, a n-dimensional ball $\mathcal{B}(\mathbf{y})$, centered at $\hat{\mathbf{x}}$ with radius PL, is determined to satisfy the requirement $\int p_{\boldsymbol{X}|\boldsymbol{Y}=\mathbf{y}}(\mathbf{x}) I(\mathbf{x}\in\mathcal{B}(\mathbf{y}))\mathrm{d}\mathbf{x} \ge 1 - P_{\text{TIR}}$, where $I(\cdot)$ stands for the indicator function. The integrity probability, which is the reciprocal of the integrity risk and can be expressed as

$$\begin{aligned} P_{\text{I}} &= \int\int p_{\boldsymbol{X},\boldsymbol{Y}}(\mathbf{x},\mathbf{y}) I(\mathbf{x}\in\mathcal{B}(\mathbf{y}))\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y} \\ &= \int p_{\boldsymbol{Y}}(\mathbf{y}) \left(\int p_{\boldsymbol{X}|\boldsymbol{Y}=\mathbf{y}}(\mathbf{x}) I(\mathbf{x}\in\mathcal{B}(\mathbf{y}))\mathrm{d}\mathbf{x} \right) \mathrm{d}\mathbf{y} \end{aligned}$$

is ensured to be $\ge 1 - P_{\text{TIR}}$.

As an example, let us consider that for a set of measurements, we have a rule to exclude the first measurement y_1. Then, in the realizations where the rule establishes that measurement y_1 has to be removed, a n-dimensional ball $\bar{\mathcal{B}}(\bar{\mathbf{y}})$ is obtained based on the the posterior PDF $p_{\boldsymbol{X}|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x})$, where $\bar{\mathbf{y}} \triangleq [y_2,\ldots,y_M]^{\text{T}}$. This exclusion rule corresponds to region $\mathcal{D}\in\mathbb{R}^M$: If $\mathbf{y}\in\mathcal{D}$, y_1 is excluded. Letting $\mathcal{D}^{\text{c}} \triangleq \mathbb{R}^M \setminus \mathcal{D}$, the integrity probability decomposes as $P_{\text{I}} = P_{\text{I},1} + P_{\text{I},2}$ where $P_{\text{I},1} = \int_{\mathbf{y}\in\mathcal{D}^{\text{c}}} p_{\boldsymbol{Y}}(\mathbf{y}) (\int p_{\boldsymbol{X}|\boldsymbol{Y}=\mathbf{y}}(\mathbf{x}) I(\mathbf{x}\in\mathcal{B}(\mathbf{y}))\mathrm{d}\mathbf{x})\mathrm{d}\mathbf{y}$ and $P_{\text{I},2} = \int_{\mathbf{y}\in\mathcal{D}} p_{\boldsymbol{Y}}(\mathbf{y}) (\int p_{\boldsymbol{X}|\boldsymbol{Y}=\mathbf{y}}(\mathbf{x}) I(\mathbf{x}\in\bar{\mathcal{B}}(\bar{\mathbf{y}}))\mathrm{d}\mathbf{x})\mathrm{d}\mathbf{y}$. Since

$$\begin{aligned} &\int p_{\boldsymbol{X}|\boldsymbol{Y}=\mathbf{y}}(\mathbf{x}) I(\mathbf{x}\in\bar{\mathcal{B}}(\bar{\mathbf{y}}))\mathrm{d}\mathbf{x} \\ &= \int \underbrace{\frac{p_{\boldsymbol{X},Y_1|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x},y_1)}{p_{\boldsymbol{X}|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x}) p_{Y_1|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(y_1)}}_{\text{C-PMI}} p_{\boldsymbol{X}|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x}) I(x\in\bar{\mathcal{B}}(\bar{\mathbf{y}}))\mathrm{d}\mathbf{x}, \end{aligned}$$

and the C-PMI term, which measures the dependence between $\mathbf{X}$ and Y_1 given $\overline{\boldsymbol{Y}} = \bar{\mathbf{y}}$, can be either greater or less than 1, $P_{\text{I},1} + P_{\text{I},2} \ge 1 - P_{\text{TIR}}$ is *generally not guaranteed*[4] given that $\bar{\mathcal{B}}(\bar{\mathbf{y}})$ meets the requirement $\int p_{\boldsymbol{X}|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x}) I(\mathbf{x}\in\bar{\mathcal{B}}(\bar{\mathbf{y}}))\mathrm{d}\mathbf{x} \ge 1 - P_{\text{TIR}}$. Intuitively, the issue is that we use the information in $y_1,\ldots,y_M$ to exclude y_1 but then use $p_{\boldsymbol{X}|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x})$ for PL computation as if y_1 never existed, but $p_{\boldsymbol{X}|\overline{\boldsymbol{Y}}=\bar{\mathbf{y}}}(\mathbf{x})$ is not the actual posterior PDF of $\mathbf{X}$ because y_1 existed and the whole vector $\mathbf{y}$ was such that it triggered the exclusion rule.

IV. BAYESIAN RAIM PART II: PL COMPUTATION

This section introduces the exact and overestimate methods for the n-dimensional PL based on the posterior PDF (13).

A. Exact 1D PL and 2D/3D PL Overestimates

Since this work does not focus on position estimation methods, the Bayesian RAIM algorithm simply computes the weighted mean, $\overline{\mathbf{m}}_{\boldsymbol{X}} \triangleq \sum_{l=1}^{L} w_{\boldsymbol{X}}^{(l)} \mathbf{m}_{\boldsymbol{X}}^{(l)}$, as the estimate of $\mathbf{x}$. Thus, the 3D position estimate is given by $\hat{\mathbf{x}}_{\text{u}} = [\overline{\mathbf{m}}_{\boldsymbol{X}}]_{1:3} = \sum_{l=1}^{L} w_{\boldsymbol{X}}^{(l)} [\mathbf{m}_{\boldsymbol{X}}^{(l)}]_{1:3}$. Based on (13), we can immediately obtain the PDFs of the positioning error vectors $\mathbf{e}_{n\text{D}}$ for any $\mathcal{L}(\tilde{\mathbf{v}}_1)$ ($n=1$), $\mathcal{P}(\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2)$ ($n=2$), or $\mathbb{R}^3$ ($n=3$), which are all GMs with the same number of terms and weights as (13). Specifically, for the 3D positioning error vector $\mathbf{e}_{3\text{D}}$:

$$p_{\boldsymbol{E}_{3\text{D}}}(\mathbf{e}_{3\text{D}}) = \sum_{l=1}^{L} w_{\boldsymbol{X}}^{(l)}\, f_{\boldsymbol{E}_{3\text{D}}}(\mathbf{e}_{3\text{D}}; \mathbf{m}_{3\text{D}}^{(l)}, \Sigma_{3\text{D}}^{(l)}), \tag{14}$$

where, for $l = 1,\ldots,L$,

$$\mathbf{m}_{3\text{D}}^{(l)} = [\mathbf{m}_{\boldsymbol{X}}^{(l)}]_{1:3} - \hat{\mathbf{x}}_{\text{u}}, \quad \Sigma_{3\text{D}}^{(l)} = [\Sigma_{\boldsymbol{X}}^{(l)}]_{1:3,1:3}. \tag{15}$$

Letting $\widetilde{\mathrm{V}}_{1\text{D}} = [\tilde{\mathbf{v}}_1]$ and $\widetilde{\mathrm{V}}_{2\text{D}} = [\tilde{\mathbf{v}}_1\ \tilde{\mathbf{v}}_2]$, the PDFs of $\mathbf{e}_{1\text{D}}$ and $\mathbf{e}_{2\text{D}}$ are given by

$$p_{\boldsymbol{E}_{n\text{D}}}(\mathbf{e}_{n\text{D}}) = \sum_{l=1}^{L} w_{\boldsymbol{X}}^{(l)}\, f_{\boldsymbol{E}_{n\text{D}}}(\mathbf{e}_{n\text{D}}; \mathbf{m}_{n\text{D}}^{(l)}, \Sigma_{n\text{D}}^{(l)}), \tag{16}$$

where $n = 1, 2$, and for $l = 1,\ldots,L$,

$$\mathbf{m}_{n\text{D}}^{(l)} = \widetilde{\mathrm{V}}_{n\text{D}}^{\text{T}} \mathbf{m}_{3\text{D}}^{(l)}, \quad \Sigma_{n\text{D}}^{(l)} = \widetilde{\mathrm{V}}_{n\text{D}}^{\text{T}} \Sigma_{3\text{D}}^{(l)} \widetilde{\mathrm{V}}_{n\text{D}}. \tag{17}$$

To determine the exact 1D PL for the subspace $\mathcal{L}(\tilde{\mathbf{v}}_1)$, we use (16). Given that $\mathbf{e}_{1\text{D}}$, $\mathbf{m}_{1\text{D}}^{(l)}$, and $\Sigma_{1\text{D}}^{(l)}$ are scalars, we denote them as $e_{1\text{D}}$, $m_{1\text{D}}^{(l)}$, and $[\sigma_{1\text{D}}^{(l)}]^2$, respectively. Using the Q function, the

---

[4] Actually, it was observed also experimentally that the inequality is violated when a exclusion rule is implemented.

actual IR associated with $\hat{\mathbf{x}}_{\mathrm{u}}$ and any r in $\mathcal{L}(\tilde{\mathbf{v}}_1)$ is expressed as:

$$\begin{aligned}&\Pr\{|e_{\mathrm{1D}}|>r\}=\Pr\{e_{\mathrm{1D}}<-r\}+\Pr\{e_{\mathrm{1D}}>r\}\\&=\sum_{l=1}^{L}w_{\boldsymbol{X}}^{(l)}\left[1-Q\left(\frac{-r-m_{\mathrm{1D}}^{(l)}}{\sigma_{\mathrm{1D}}^{(l)}}\right)+Q\left(\frac{r-m_{\mathrm{1D}}^{(l)}}{\sigma_{\mathrm{1D}}^{(l)}}\right)\right]\end{aligned}\quad(18)$$

The smallest r ensuring $\Pr\{|e_{\mathrm{1D}}|>r\}<P_{\mathrm{TIR}}$ is found using bisection search, yielding $\mathrm{PL}_{\mathrm{1D}}$. This value represents the most stringent 1D PL for the position estimate $\hat{\mathbf{x}}_{\mathrm{u}}$.

To obtain overestimates for the 2D PL for $\mathcal{P}(\tilde{\mathbf{v}}_1,\tilde{\mathbf{v}}_2)$ and the 3D PL for $\mathbb{R}^3$ using Lemma 1, we select $w_i=1/n$ for $i=1,\ldots,n$, and find the smallest r_i that ensures

$$\sum_{l=1}^{L}w_{\boldsymbol{X}}^{(l)}\left[1-Q\left(\frac{-r_i-[\mathbf{m}_{n\mathrm{D}}^{(l)}]_i}{[\Sigma_{n\mathrm{D}}^{(l)}]_{i,i}^{1/2}}\right)+Q\left(\frac{r_i-[\mathbf{m}_{n\mathrm{D}}^{(l)}]_i}{[\Sigma_{n\mathrm{D}}^{(l)}]_{i,i}^{1/2}}\right)\right]<w_iP_{\mathrm{TIR}}$$

using bisection search and assigned to PL_i. Finally, the overestimate $\mathrm{PL}_{n\mathrm{D}}^{U}$ is computed following (6).

B. Exact 2D/3D PL Computation

As discussed in Section II-B, computing the exact 2D/3D PL equivalent to finding the minimum radius of a 2D circle or 3D sphere centered at the origin that includes the positioning error vector $\mathbf{e}_{n\mathrm{D}}$ with a probability of at least $1-P_{\mathrm{TIR}}$. This is a complex problem. In Appendix D, we derive Theorem 1, which provides a formulation for the minimum radius and forms the basis for our PL searching algorithm.

Theorem 1: Given that the n-dimensional positioning error vector $\mathbf{e}_{n\mathrm{D}}$ follows a GM distribution with the PDF (14) for $n=3$ and (16) for $n=2$, the PL computation problem defined by (4) can be reformulated as:

$$\mathrm{PL}_{n\mathrm{D}}=\min\left\{r\,\middle|\,\sum_{l=1}^{L}w_{\boldsymbol{X}}^{(l)}\left[1-F_{Z_l}(r^2)\right]<P_{\mathrm{TIR}}\right\},\quad(19)$$

where $F_{Z_l}(z)\triangleq\Pr\{Z_l\le z\}$, $l=1,\ldots,L$, represents the cumulative density function (CDF) of a random variable Z_l, given by a weighted sum of n independent noncentral chi-square distributed random variables $W_{l,1},\ldots,W_{l,n}$, each with one degree of freedom and noncentrality parameter $\nu_{l,i}^2$. Specifically,

$$Z_l=\sum_{i=1}^{n}\omega_{l,i}W_{l,i},\quad W_{l,i}\sim\chi^2\left(1,\nu_{l,i}^2\right),\ \forall i=1,\ldots,n;\quad(20)$$

and $\{\omega_{l,i}\}$ and $\{\nu_{l,i}^2\}$ are determined by $\Sigma_{n\mathrm{D}}^{(l)}$ and $\mathbf{m}_{n\mathrm{D}}^{(l)}$ in the following way: Perform eigendecomposition to obtain $\Sigma_{n\mathrm{D}}^{(l)}=\mathsf{P}_l\Omega_l\mathsf{P}_l^{\mathrm{T}}$, where $\mathsf{P}_l\in\mathbb{R}^{n\times n}$ is an orthogonal matrix and Ω_l is a diagonal matrix. The diagonal elements of Ω_l are $\omega_{l,1},\ldots,\omega_{l,n}$ (since $\Sigma_{n\mathrm{D}}^{(l)}\succ0,\omega_{l,i}>0$ for $i=1,\ldots,n$); and the noncentrality parameters $\nu_{l,1},\ldots,\nu_{l,n}$ are computed by $(\nu_{l,1},\ldots,\nu_{l,n})^{\mathrm{T}}\triangleq\mathsf{P}_l^{\mathrm{T}}[\Sigma_{n\mathrm{D}}^{(l)}]^{-\frac{1}{2}}\mathbf{m}_{n\mathrm{D}}^{(l)}$.

Proof: See Appendix D. □

There is no closed-form expression for the CDF of a generalized chi-squared variable, but numerical methods are available. In Appendix E, we describe the Imhof method [33], a numerical method that can achieve arbitrary accuracy. The CDF $F_{Z_l}(z)$ can be approximated by:

$$\bar{F}_{Z_l}(z,U_l)=\frac{1}{2}-\frac{1}{\pi}\int_0^{U_l}\frac{\sin\beta(u,z)}{u\kappa(u)}\mathrm{d}u,\quad(21)$$

where $\beta(u,z)$ and $\kappa(u)$ are given by (51a) and (51b). The approximation error is bounded by

$$\left|F_{Z_l}(z)-\bar{F}_{Z_l}(z,U_l)\right|\le\Xi(U_l)\quad(22)$$

where $\Xi(\cdot)$ is a decreasing function provided in (52). For any desired accuracy, a sufficiently large U_l can be found to ensure $\Xi(U_l)$ meets the requirements. For PL computation, the smallest U_l satisfying $\Xi(U_l)\le\zeta_1P_{\mathrm{TIR}}$, where $\zeta_1\ll1$, is chosen. To incorporate the approximation error, the constraint in (19) is modified to:

$$\sum_{l=1}^{L}w_{\boldsymbol{X}}^{(l)}\left[1-\bar{F}_{Z_l}(r^2,U_l)\right]<(1-\zeta_1)P_{\mathrm{TIR}}.\quad(23)$$

With this numerical tool, $\mathrm{PL}_{n\mathrm{D}}$ in (19) can be obtained using a search process similar to the bisection search in Section V-D. In particular, the overestimate $\mathrm{PL}_{n\mathrm{D}}^{U}$ serves as a good initial value for R_{up}. Choosing an initial value for R_{low} is trivial. During each iteration, the numerical integral (21) needs to be computed L times, where $L=2^M$. This can be computationally expensive. To reduce the computational complexity, we sort $\{w_{\boldsymbol{X}}^{(l)}\}$ in non-increasing order, denoted by $\{w_{\boldsymbol{X}}^{(l(1))},w_{\boldsymbol{X}}^{(l(2))},\ldots,w_{\boldsymbol{X}}^{(l(L))}\}$. Then we find the smallest integer J such that

$$\sum_{j=J+1}^{L}w_{\boldsymbol{X}}^{(l(j))}\le\zeta_2\,P_{\mathrm{TIR}},\quad(24)$$

where $\zeta_2\ll1$. Since $\sum_{l=1}^{L}w_{\boldsymbol{X}}^{(l)}[1-F_{Z_l}(r^2)]<\sum_{j=1}^{J}w_{\boldsymbol{X}}^{(l(j))}[1-F_{Z_{l(j)}}(r^2)]+\sum_{j=J+1}^{L}w_{\boldsymbol{X}}^{(l(j))}$, we can replace the constraint in (19) with a stronger one:

$$\sum_{j=1}^{J}w_{\boldsymbol{X}}^{(l(j))}\left[1-F_{Z_{l(j)}}(r^2)\right]<(1-\zeta_2)P_{\mathrm{TIR}}.\quad(25)$$

When using (21) for approximation and choosing U_l as described, (23) becomes

$$\sum_{j=1}^{J}w_{\boldsymbol{X}}^{(l(j))}\left[1-\bar{F}_{Z_{l(j)}}(r^2,U_{l(j)})\right]<(1-\zeta_1-\zeta_2)P_{\mathrm{TIR}}.$$

The PL search algorithm developed based on the above discussions is summarized in Algorithm 1. Due to conservative approximations, the output is expected to be looser than the optimal PL. The gap depends on the choice of ζ_1 and ζ_2, the larger they are, the looser the computed PL.

Remark 2 (Complexity discussion): Computing the exact 1D PL using bisection search is efficient, as is the process of overestimating the 2D/3D PL. For a 1D search with $N_{\mathrm{it}}>0$ iterations, the computational complexity is $\mathcal{O}(2N_{\mathrm{it}}L)$. In contrast, obtaining the exact 2D/3D PL using Algorithm 1 is more computationally intensive due to the numerical integration required in (21). To ensure accuracy, the upper integration limit U_l is typically set high by choosing a small $\zeta_1\ll1$. Our numerical

Algorithm 1: Exact n-Dimensional PL Computation, $n = 2, 3$.

```
Input: $P_{\text{TIR}}$, $p_{\boldsymbol{E}_{n\text{D}}}(\mathbf{e}_{n\text{D}})$ given by (14) or (16), $\zeta_1$, $\zeta_2$,
  search error tolerance $r_{\text{tol}}$.
Output: $\text{PL}_{n\text{D}}$ defined in (19) within error tolerance $r_{\text{tol}}$.
Compute the overestimate $\text{PL}_{n\text{D}}^{U}$ following Section IV-A;
Sort $\{w_{\boldsymbol{X}}^{(l)}\}$ into non-increasing order and find the smallest $J$ satisfying (24);
for $j = 1, \ldots, J$ do                    ▷ Parameter preparation
  Compute $\{\omega_{l(j),i}\}$ and $\{\nu_{l(j),i}^2\}$ following Theorem 1;
  Find $U_{l(j)}$ such that $\Xi(U_{l(j)}) = \zeta_1 P_{\text{TIR}}$ based on (52);
end for
$r_{\text{up}} \leftarrow \text{PL}_{n\text{D}}^{U}$;          ▷ Starting upper limit for $\text{PL}_{n\text{D}}$
$r_{\text{low}} \leftarrow$ some small value;     ▷ Starting lower limit
while $|r_{\text{up}} - r_{\text{low}}| > r_{\text{tol}}$ do
  $r_{\text{mid}} \leftarrow (r_{\text{up}} + r_{\text{low}})/2$;
  $P_{\text{mid}} \leftarrow \sum_{j=1}^{J} w_{\boldsymbol{X}}^{(l(j))}[1 - \bar{F}_{Z_{l(j)}}(r_{\text{mid}}^2, U_{l(j)})]$ using (21);
  if $P_{\text{mid}} < (1 - \zeta_1 - \zeta_2)P_{\text{TIR}}$ then
    $r_{\text{up}} \leftarrow r_{\text{mid}}$, $P_{\text{up}} \leftarrow P_{\text{mid}}$;
  else
    $r_{\text{low}} \leftarrow r_{\text{mid}}$, $P_{\text{low}} \leftarrow P_{\text{mid}}$;
  end if
end while
$\text{PL}_{n\text{D}} \leftarrow r_{\text{up}}$.
```

studies show that the integration process is also sensitive to the parameters of the generalized chi-squared variables in (20), which are dependent on the measurement bias distributions. Additionally, discarding GM terms with negligible weights, as described in (24), can significantly reduce the computational complexity across all methods. However, increasing the number of discarded terms (i.e., using a larger ζ_2) can cause conservative PL values. Therefore, ζ_1 and ζ_2 should be carefully selected to balance computational complexity and the tightness of the PL.

V. BASELINE RAIM ALGORITHM

To benchmark performance, we adapt the advanced RAIM algorithm from [17]. We refer to the adapted algorithm as the *Baseline RAIM*[5] algorithm. It uses a hypothesis testing framework for FDE. The following inputs are required: $P_{\text{TIR,H}}$ and $P_{\text{TIR,V}}$, the TIR requirements for the horizontal-plane and the vertical direction, and $P_{\text{FA,H}}$ and $P_{\text{FA,V}}$, the respective false alarm budgets. A false alarm occurs if the algorithm detects faults in fault-free measurements. The outputs include a position estimate and PLs for both horizontal and vertical subspaces. The main steps of Baseline RAIM are as follows.

1) *Fault mode determination:* Identify a set of *fault modes* (hypotheses) for monitoring, where each fault mode assumes a subset of measurements is faulty and the rest are fault-free. These fault modes are exclusive, meaning only one can occur.
2) *SS test for fault detection:* Compute an SS test statistic for each fault mode and position coordinate to evaluate the closeness of different position estimates obtained from assumed fault-free measurements by the fault modes.
3) *Fault exclusion (if SS test fails):* If the SS test fails, attempt fault exclusion by repeatedly applying fault detection to subsets of assumed fault-free measurements for each fault mode, starting with the one with the highest probability of occurrence (see Remark 3).
4) *Position estimate and PL calculation:* If fault exclusion is successful or not needed, calculate a position estimate and compute two PLs following [17]: one for the horizontal plane and one for the vertical direction.

In this process, the following components are involved: (i) all-in-view position estimation; (ii) fault mode identification; (iii) solution separation (SS) testing; and (iv) PL computations. The following subsections provide detailed explanations of each component, and conclude with a discussion of computational complexity.

A. All-in-View Position Estimation

The all-in-view position assumes all measurements are fault-free. In the linear model (3), when $\mathbf{b} = \mathbf{0}$ and $\mathbf{n}$ follows $\mathcal{N}(\mathbf{0}, \Sigma)$, where Σ is the diagonal covariance matrix with elements $\Sigma_{i,i} = \sigma_{n,i}^2$, $i = 1, \ldots, M$, the weighted least squares (WLS) estimate of $\mathbf{x}$, also the maximum likelihood (ML) estimate under under the fault-free assumption, is given by

$$\hat{\mathbf{x}}^{(0)} = (\mathsf{H}^{\mathrm{T}}\Sigma^{-1}\mathsf{H})^{-1}\mathsf{H}^{\mathrm{T}}\Sigma^{-1}\mathbf{y} = \mathsf{A}^{(0)}\mathbf{y}, \tag{26}$$

where $\mathsf{A}^{(0)} \triangleq (\mathsf{H}^{\mathrm{T}}\Sigma^{-1}\mathsf{H})^{-1}\mathsf{H}^{\mathrm{T}}\Sigma^{-1}$. The positioning error vector $\mathbf{x} - \hat{\mathbf{x}}^{(0)} = \mathsf{A}^{(0)}\mathbf{n}$ follows a zero-mean Gaussian distribution with covariance matrix $\Phi^{(0)} \triangleq (\mathsf{H}^{\mathrm{T}}\Sigma^{-1}\mathsf{H})^{-1}$.

B. Fault Modes Identification

To enable SS testing for fault exclusion, an eligible fault mode must contain at least 5 assumed fault-free measurements. Baseline RAIM monitors all eligible modes for optimal performance. The total number of monitored fault modes (with at least one fault) is given by $N_{\text{FM}} = \sum_{j=1}^{M-5}\binom{M}{j}$. For fault mode $k = 1, \ldots, N_{\text{FM}}$, let $\mathcal{I}_k$ represent the index set of assumed fault-free measurements, and $\mathcal{I}_k^c \triangleq \{1, 2, \ldots, M\} \setminus \mathcal{I}_k$ represent the index set of faulty measurements. The probability of occurrence of fault mode k is $p_{\text{FM},k} = \prod_{i\in\mathcal{I}_k}(1 - \theta_i)\prod_{i\in\mathcal{I}_k^c}\theta_i$. For convenience, we assume that fault modes are sorted in decreasing order of probability of occurrence, thus $p_{\text{FM},k} \geq p_{\text{FM},k+1}$, for $1 \leq k < N_{\text{FM}}$.

---

[5] The dropped term "advanced" emphasizes its ability to detect and exclude multiple faults, which is more advanced than the earliest RAIM algorithm (which could handle only one fault).

C. Solution Separation Testing

For fault mode k, define Σ_k as the diagonal matrix derived from Σ in the following way

$$[\Sigma_k^{-1}]_{i,i} = \begin{cases} 0, & i \in \mathcal{I}_k^c, \\ 1/\sigma_{n,i}^2, & i \in \mathcal{I}_k. \end{cases} \tag{27}$$

The WLS estimate of $\mathbf{x}$ using measurements indexed by $\mathcal{I}_k$ is

$$\hat{\mathbf{x}}^{(k)} = \mathsf{A}^{(k)}\mathbf{y} \tag{28}$$

where $\mathsf{A}^{(k)} \triangleq (\mathsf{H}^{\mathrm{T}}\Sigma_k^{-1}\mathsf{H})^{-1}\mathsf{H}^{\mathrm{T}}\Sigma_k^{-1}$. If all these measurements are fault-free, the positioning error $\mathbf{x} - \hat{\mathbf{x}}^{(k)} = \mathsf{A}^{(k)}\mathbf{n}$ follows a zero-mean Gaussian distribution with covariance matrix $\Phi^{(k)} \triangleq (\mathsf{H}^{\mathrm{T}}\Sigma_k^{-1}\mathsf{H})^{-1}$, and difference between the all-in-view and fault mode k estimates

$$\Delta\hat{\mathbf{x}}^{(k)} \triangleq \hat{\mathbf{x}}^{(0)} - \hat{\mathbf{x}}^{(k)} \tag{29}$$

follows a zero-mean Gaussian distribution with covariance matrix given by $(\mathsf{A}^{(k)} - \mathsf{A}^{(0)})\Sigma(\mathsf{A}^{(k)} - \mathsf{A}^{(0)})^{\mathrm{T}}$. The SS test statistics used by the Baseline RAIM algorithm are

$$\tau_{n,k} \triangleq \left|\Delta\hat{\mathbf{x}}_n^{(k)}\right|, \quad n = 1, 2, 3,\ k = 1, \ldots, N_{\mathrm{FM}}. \tag{30}$$

The SS test compares $\tau_{n,k}$ with a test threshold $T_{n,k}$ for each position coordinate and fault mode. Only if $\tau_{n,k} \leq T_{n,k}$ for all $n = 1, 2, 3$ and $k = 1, \ldots, N_{\mathrm{FM}}$, are the M measurements considered fault-free, and the algorithm outputs $\hat{\mathbf{x}}_{\mathrm{u}} = \hat{\mathbf{x}}_{1:3}^{(0)}$ as the 3D position estimate. If $\tau_{n,k} > T_{n,k}$ for any n, k, the algorithm proceeds to fault exclusion. The test thresholds are determined based on the false alarm budgets $P_{\mathrm{FA,H}}$ and $P_{\mathrm{FA,V}}$[6]. The algorithm distributes $P_{\mathrm{FA,H}}$ evenly between the two horizontal position coordinates x_1 and x_2, and then distributes each coordinate's false alarm budget evenly across the N_{FM} fault modes. For $k = 1, \ldots, N_{\mathrm{FM}}$, the thresholds are

$$T_{n,k} = \begin{cases} \sigma_{ss,n}^{(k)}\, Q^{-1}\left(\frac{P_{\mathrm{FA,H}}}{4N_{\mathrm{FM}}}\right), & n = 1, 2 \\ \sigma_{ss,n}^{(k)}\, Q^{-1}\left(\frac{P_{\mathrm{FA,V}}}{2N_{\mathrm{FM}}}\right), & n = 3 \end{cases} \tag{31}$$

where $\sigma_{ss,n}^{(k)} \triangleq [(\mathsf{A}^{(k)} - \mathsf{A}^{(0)})\Sigma(\mathsf{A}^{(k)} - \mathsf{A}^{(0)})^{\mathrm{T}}]_{n,n}^{1/2}$, and $Q^{-1}(\cdot)$ is the inverse of the Q function: $Q(u) = \frac{1}{\sqrt{2\pi}}\int_u^{+\infty} e^{-\frac{t^2}{2}}\,\mathrm{d}t$.

Remark 3 (Fault exclusion process): When the SS test fails, the algorithm attempts to exclude the faulty measurements by reapplying the fault mode identification and SS testing to subsets of assumed fault-free measurements specified by the N_{FM} fault modes, ordered by decreasing probability of occurrence. This continues until the SS test passes or all fault modes are checked, indicating fault exclusion failure. Upon reaching fault mode k in this process, the subset of measurements indexed by $\mathcal{I}_k$ are checked, and a new list of $N_{\mathrm{FM},k} = \sum_{j=1}^{|\mathcal{I}_k|-5}\binom{|\mathcal{I}_k|}{j}$ fault modes is identified. If the SS testing is successful, fault mode k is considered true, and the algorithm returns the position estimate ($\hat{\mathbf{x}}_{\mathrm{u}} = \hat{\mathbf{x}}_{1:3}^{(k)}$) and PLs computed using these $|\mathcal{I}_k|$ measurements. If fault exclusion fails, the algorithm declares that the system cannot provide reliable positioning results (system unavailable).

D. Protection Level Computation

The Baseline RAIM algorithm can compute 1D PLs for each of the three coordinate directions. It can also obtain PL overestimates in the three coordinate planes and the full 3D space using Lemma 1. Below we summarize the formulation of the 1D PL for the vertical direction and the 2D PL overestimate for the horizontal plane as described in [17]. Assume that the SS test passes for the M measurements, so $\hat{\mathbf{x}}_{\mathrm{u}} = \hat{\mathbf{x}}_{1:3}^{(0)}$, and we define $\sigma_n^{(k)} \triangleq [\Phi^{(k)}]_{n,n}^{1/2}$ for $n = 1, 2, 3$ and $k = 0, 1, \ldots, N_{\mathrm{FM}}$.

In the vertical direction, the actual IR associated with an arbitrary r is given by $\Pr\{|x_3 - \hat{x}_3^{(0)}| > r\}$. An overbound of the vertical PL PL_{V} is given by the minimum r satisfying the following condition

$$2Q\left(\frac{r}{\sigma_3^{(0)}}\right) + \sum_{k=1}^{N_{\mathrm{FM}}} p_{\mathrm{FM},k}\, Q\left(\frac{r - T_{3,k}}{\sigma_3^{(k)}}\right) < P_{\mathrm{TIR,V}}. \tag{32}$$

A bisection search is used to find the smallest r within a given error tolerance r_{tol}. The detailed formulation process can be found in [17, Appendix H]. In the horizontal plane, $P_{\mathrm{TIR,H}}$ is evenly divided for the x and y directions (i.e., $w_1 = w_2 = 0.5$ in (5)). For $n = 1, 2$, an overbound of the PL PL_n is given by the minimum r_n that satisfies

$$2Q\left(\frac{r_n}{\sigma_n^{(0)}}\right) + \sum_{k=1}^{N_{\mathrm{FM}}} p_{\mathrm{FM},k}\, Q\left(\frac{r_n - T_{n,k}}{\sigma_n^{(k)}}\right) < \frac{P_{\mathrm{TIR,H}}}{2}. \tag{33}$$

Again, bisection search is used to find these minimum values within error tolerance r_{tol}. Following (6), the horizontal PL overestimate is given by $\mathrm{PL}_{\mathrm{H}} = (\mathrm{PL}_1^2 + \mathrm{PL}_2^2)^{1/2}$.

If the SS test fails for the M measurements and the kth fault mode is accepted during the fault exclusion process, then $\hat{\mathbf{x}}_{\mathrm{u}} = \hat{\mathbf{x}}_{1:3}^{(k)}$, and in (32) and (33), $p_{\mathrm{FM},k}$, $T_{n,k}$, and $\sigma_n^{(k)}$ should be replaced with their counterparts computed for the $N_{\mathrm{FM},k}$ *new* fault modes. The left-hand sides of (32) and (33) are loose upper bounds of the actual IR related to the 1D positioning error variables [17]. Moreover, PL_{H}, by formulation, is an overestimate of the 2D PL for the horizontal plane. Due to these, the values obtained for PL_{V} and PL_{H} tend to be large. This intrinsic drawback will be addressed by the proposed Bayesian RAIM algorithm.

Remark 4 (Complexity discussion): The Baseline RAIM algorithm has a variable computational complexity due to the varying number of SS tests required (from 1 in the best case and N_{FM} in the worst case). Most of the computational cost for SS tests comes from the matrix operations in (26)–(31), with matrix sizes depending on the number of assumed fault-free measurements, ranging between M and 5. To reduce complexity, fault modes with occurrence probabilities significantly lower than the TIR can be excluded from monitoring [17]. PL computation via bisection search adds little to the total cost, as confirmed by our numerical study. This iterative algorithm requires a variable number of iterations, denoted by $N_{\mathrm{it}} > 0$, and its complexity

---

[6] Reducing $P_{\mathrm{FA,H}}$ and $P_{\mathrm{FA,V}}$ increases the test thresholds computed by (31), thus lowering the likelihood of issuing warning for potential faults. This reduces computational costs but raises the risk of misdetection. Conversely, higher $P_{\mathrm{FA,H}}$ and $P_{\mathrm{FA,V}}$ trigger more fault exclusion attempts, raising computational costs but enhancing positioning quality.

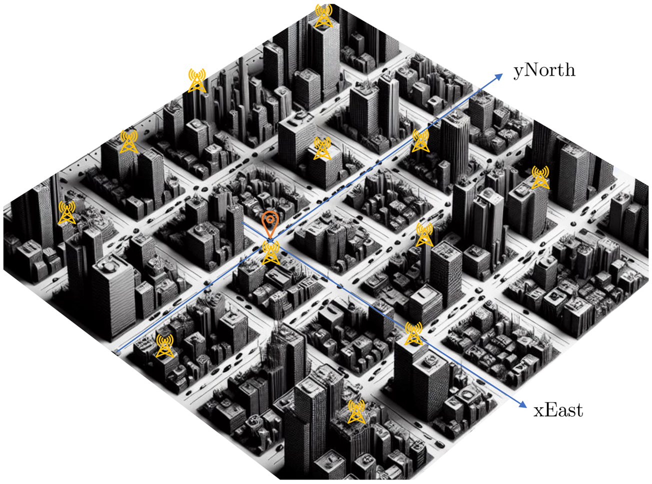


Fig. 4. Illustration of the dense urban grid scenario used in the simulation study. BSs are shown in yellow, while the UE location is marked with a red location pin.

is $\mathcal{O}(N_{\text{it}}N_{\text{FM}})$, since both (32) and (33) involve about N_{FM} Q-function evaluations per iteration.

VI. NUMERICAL STUDY

We conduct numerical studies with two main objectives: (i) to compare the performance and computational complexity (in terms of running time) of the proposed Bayesian RAIM algorithm against the baseline RAIM algorithm; and (ii) to assess how sensitive the measurement model linearization is to the initial positioning error, $\mathbf{e}_0$.

A. Simulation Description

1) Scenario and Parameters: The simulation study is conducted in a dense urban grid covering approximately 1000 × 1000 m^2, divided into 12 cells of 400 × 250 m^2 each. A total of $M = 12$ BSs are deployed, one per cell, each placed on a building rooftop at a random height between 10 and 30 meters. The scenario is illustrated in Fig. 4. The position of the target UE is fixed at $\mathbf{x}_{\text{u}} = (0, 0, 0)$ with zero clock bias ($x_c = 0$). ToA measurement noise is identically distributed across all BSs with zero mean, $\sigma_{n,i} = 0.5$ m [34, Annex B2], and a prior fault probability $\theta_i = 0.05$. Two fault types are considered: one due to strong NLoS signals, with $m_{b,i}$ between 1 and 20 m and $\sigma_{b,i} = 1$ m for all i, and another due to clock synchronization errors, with $m_{b,i} = 0$ m and $\sigma_{b,i} = 10$ m for all i. Note that the NLOS conditions were not generated to match the building locations in Fig. 4. A TIR of $P_{\text{TIR}} = 10^{-3}$ is used for all PL computations.

2) Simulation Setup: For performance comparison, $N_{\text{sim}} = 5 \times 10^5$ positioning epochs are simulated for each fault type, using an error-free initial UE position estimate, $\mathbf{x}_{\text{u},0} = \mathbf{x}_{\text{u}}$. This ensures there is no linearization errors in (1). The Baseline RAIM algorithm computes PLs for both the horizontal-plane (H-plane) and the vertical-direction (V-direction), with false alarm probabilities set to $P_{\text{FA,H}} = P_{\text{FA,V}} = 10^{-2}$. The Bayesian RAIM algorithm computes the horizontal PL overestimate, PL_{H}^{U}, using the method described in Section IV-A, a near-optimal PL_{H} via the *exact PL computation* algorithm with $\zeta_1 = 0.1$ and $\zeta_2 = 0.002$, and the optimal PLs for the vertical and for the 45° directions in the x-y plane, where unit vector is $\tilde{\mathbf{v}}_{45^\circ} = (\cos 45^\circ, \sin 45^\circ, 0)^{\text{T}}$. Additionally, two variations of the Bayesian RAIM algorithm are run for benchmarking. The first, termed the *fault-ignorant variation*, assumes zero fault probability for all measurements, yielding an identical Gaussian posterior PDF of $\boldsymbol{X}$ in all epochs. The PL derived from this PDF corresponds to the positioning accuracy at the $1 - \text{TIR}$ percentile. The second, termed the *genie variation*, uses perfect knowledge about the fault state, setting θ_i to 0 or 1 based on the actual fault state. This also results in a Gaussian posterior PDF, but it varies across epochs. Results from this set of simulation are discussed in Section VI-B.

Another set of simulations is conducted to evaluate the sensitivity of the two RAIM methods to initial positioning error. An initial error $\mathbf{e}_0$ is introduced such that $\mathbf{x}_{\text{u},0} = \mathbf{x}_{\text{u}} + \mathbf{e}_0$. Two error models are considered: a horizontal error, $\mathbf{e}_{0,\text{H}} = (E_{\text{H}} \cos\phi, E_{\text{H}} \sin\phi, 0)^{\text{T}}$, where ϕ is uniformly distributed over $[0, 2\pi)$, and a vertical error, $\mathbf{e}_{0,\text{V}} = (0, 0, E_{\text{V}})^{\text{T}}$. E_{H} ranges from 0 to 5 m in 0.5 m increments, while E_{V} ranges from −10 to 10 m in 2 m steps. For each value of E_{H} or E_{V}, N_{sim} positioning epochs are simulated for each fault type, using identical noise and bias realizations. For Bayesian RAIM, we focus on the vertical direction and the horizontal $\tilde{\mathbf{v}}_{45^\circ}$ direction. Results from this set of simulation are discussed in Section VI-C.

3) Performance Evaluation Metrics: The algorithms are evaluated using the following metrics:

1) *Simulated IR:* Rather than adopting the traditional, requirement-specific definition of IR, which depends on both the PL threshold and time-to-alert (TTA) constraints, we use a requirement-independent definition of IR in this study:

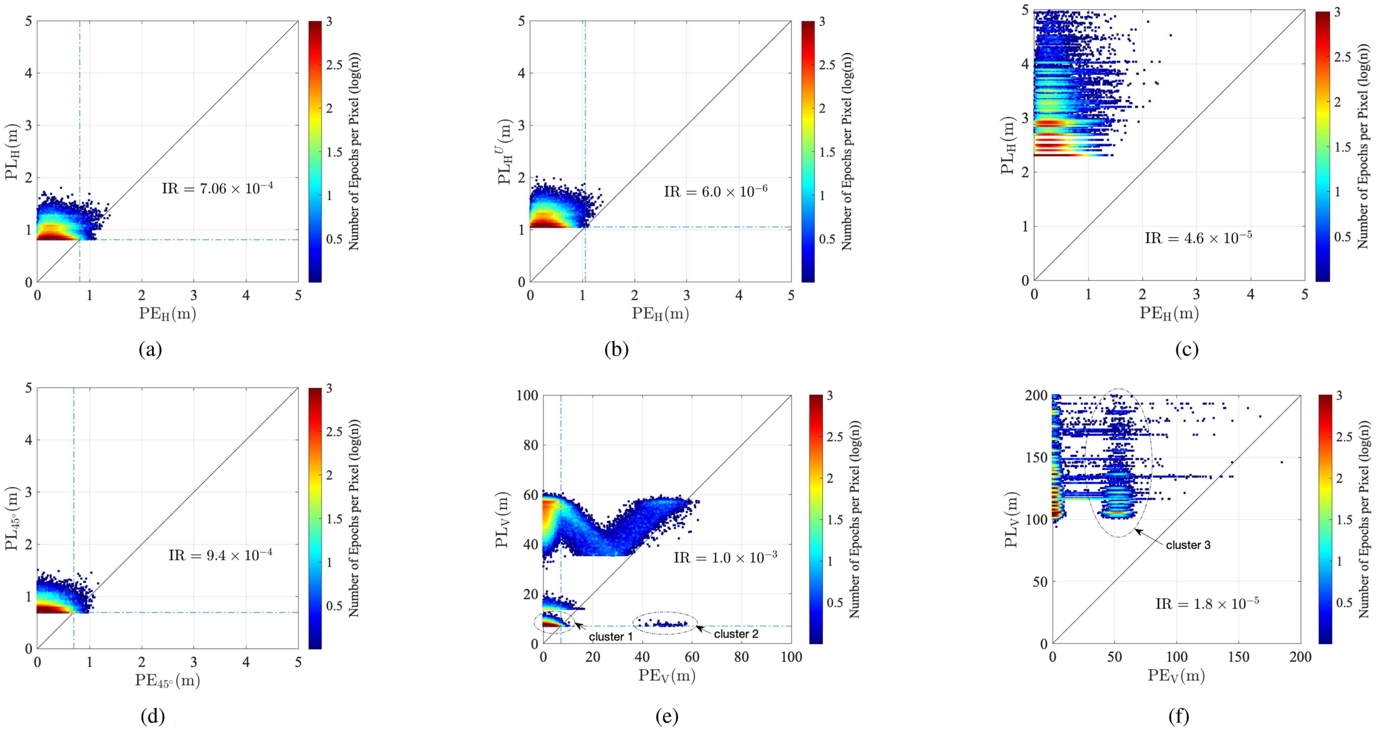
$$\text{IR}_{n\text{D}} = \frac{\sum_{k=1}^{N_{\text{sim}}} I(\text{PE}_{n\text{D},k} > \text{PL}_{n\text{D},k})}{N_{\text{sim}}}, \tag{34}$$

where $\text{PE}_{n\text{D},k} \triangleq \|\mathbf{e}_{n\text{D},k}\|$ is the actual positioning error (PE) and $\text{PL}_{n\text{D},k}$ is the computed PL for the kth epoch. The indicator function $I(\cdot)$ returns 1 if $\text{PE}_{n\text{D},k} > \text{PL}_{n\text{D},k}$, and 0 otherwise.

2) *PL tightness:* We assess how closely the PL track the actual errors using the *Stanford diagram* [12], which plots $\text{PE}_{n\text{D}}$ versus $\text{PL}_{n\text{D}}$ for each epoch on a PE-PL plane.[7] Points below the diagonal indicate integrity failures; ideally, points cluster in the lower-left for tight PLs and low errors. PL tightness is further quantified using empirical CDFs, with PL values reported at the 50th (median), 95th, and 99th percentiles. These percentiles are chosen to represent typical and extreme cases, and the empirical CDF curves in Fig. 7 illustrate the position of these PL values within the overall distribution.

---

[7] The resolution and color intensity of a Stanford diagram depends on the density of pixels, each of which contains a small square area.



(a) (b) (c)

(d) (e) (f)

Fig. 5. Performances of the RAIM algorithms in the form of Stanford diagrams under NLoS type fault conditions, with same TIR requirement $P_{\mathrm{TIR}} = 10^{-3}$ for all subspaces. In (a)-(d), a pixel stands for 0.01×0.01 m^2; in (e) and (f), a pixel stands for 0.2×0.2 m^2. (a) Bayesian RAIM, H-plane Alg. 1. (b) Bayesian RAIM, H-plane overestimate. (c) Baseline RAIM, H-plane. (d) Bayesian RAIM, $\tilde{\mathbf{v}}_{45^\circ}$-direction. (e) Bayesian RAIM, V-direction. (f) Baseline RAIM, V-direction.

TABLE I

Empirical Percentile PL Values (v) and Reduction (r, Computed by $r = 1 - \mathrm{PL_{Bayesian}}/\mathrm{PL_{Baseline}}$) by Bayesian RAIM Compared to Baseline RAIM: **N** Denotes NLoS-Type Fault, and **C** Denotes Clock-Type Fault

		PL@50%		PL@95%		PL@99%	
		v[m]	r[%]	v[m]	r[%]	v[m]	r[%]
$\mathrm{PL_H^U}$	N	1.10	52.4	1.32	59.1	1.46	62.5
	C	1.14	50.6	1.44	51.8	1.80	51.7
$\mathrm{PL_H}$	N	0.84	63.5	1.07	66.7	1.22	68.7
	C	0.88	61.7	1.18	60.5	1.55	58.4
$\mathrm{PL_V}$	N	7.32	93.1	56.53	58.5	57.48	65.9
	C	34.12	67.7	58.63	56.4	67.81	55.0

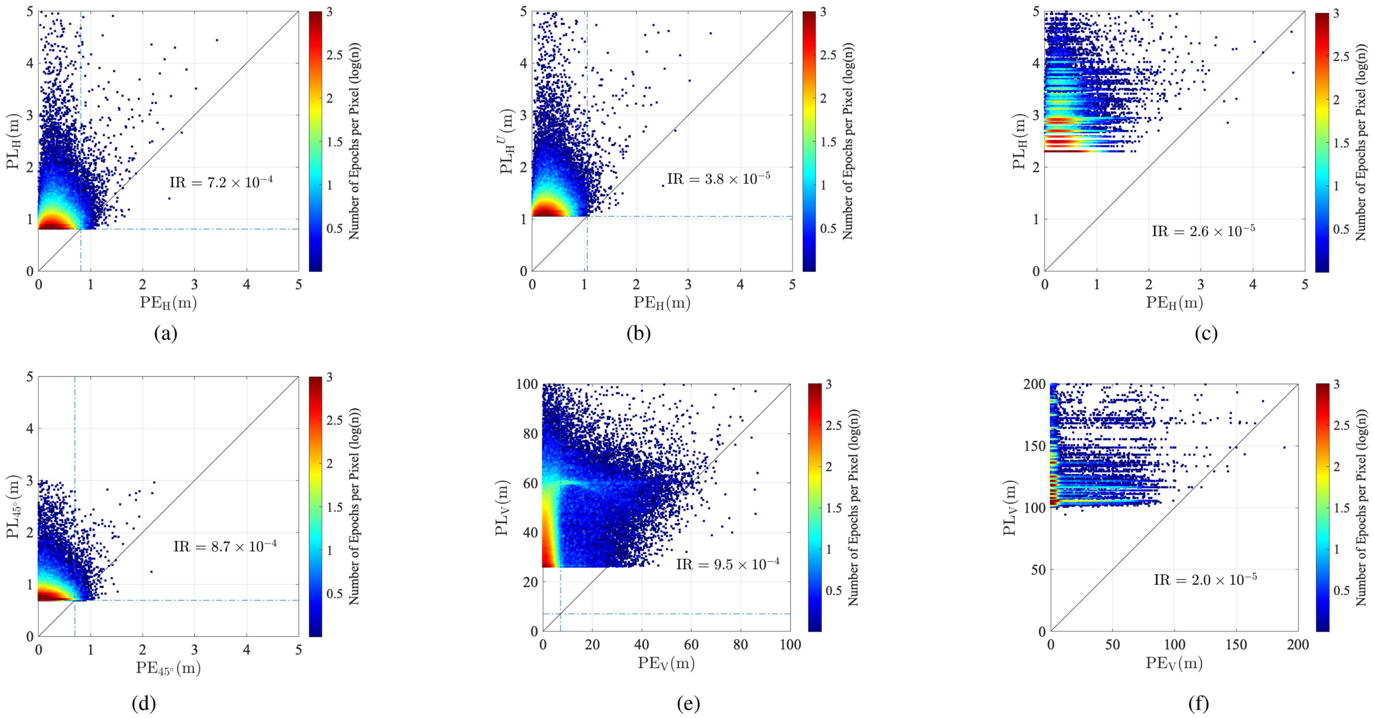
3) *Running time:* Computational complexity is measured by the running time in Matlab on a MacBook Pro with an Intel Core i7 processor. For the Bayesian RAIM algorithm, all nine message passing steps described in Section III-C are performed.

B. Performance Comparison Without Linearization Error

Figs. 5 and 6 show Stanford diagrams for NLoS- and clock-type faults, respectively, with pixel sizes specified in the captions. Both algorithms achieve simulated IRs below the $P_{\mathrm{TIR}} = 10^{-3}$ limit in all subspaces, demonstrating their reliability. Bayesian RAIM consistently provides tighter PLs than Baseline RAIM, even though the point cloud shapes vary across subspaces and fault types. Percentile values of the horizontal overestimate $\mathrm{PL_H^U}$ and the near-optimal $\mathrm{PL_H}$, along with their reduction over Baseline RAIM, are summarized in Table I. In the Bayesian RAIM diagrams, two blue dash-dotted lines are shown, one aligned with the PE-axis and one with the PL-axis, which indicate the PL value obtained from the fault-ignorant variation. This value (corresponding to the accuracy at the 99.9% percentile) serves as the theoretical lower bound for the PL computed at any epoch, as demonstrated in our simulation results. The dots appearing to the right of the vertical blue line represent epochs in which the actual PE exceeds this accuracy threshold. The percentage of such occurrences is significantly higher than the simulated IR, highlighting the necessity of accounting for potential faults rather than relying solely on nominal accuracy.

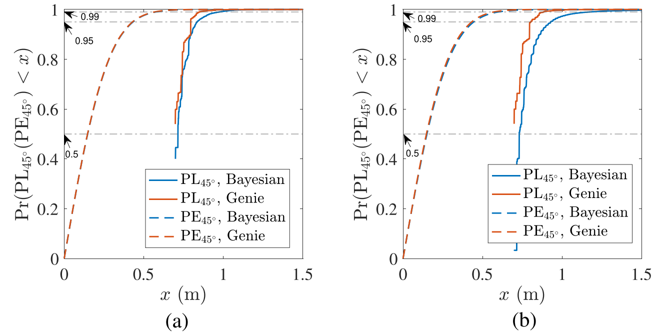
1) *H-Plane:* The well-distributed BSs around the UE provide strong positioning capability on the H-plane. In Fig. 5 and Fig. 6, subfigures (a)-(c), Baseline RAIM achieves simulated IRs in the order of 10^{-5} for both fault types. Bayesian RAIM achieves similar IRs using PL overestimation and just below P_{TIR} using the exact PL computation algorithm. While fault types minimally affect Baseline RAIM, clock-type faults causes larger PLs in Bayesian RAIM, spreading the point clouds upwards due to the inseparable small bias realizations from noise. Table I shows that Bayesian RAIM achieved over 50% PL reduction for all percentiles with the overestimation method and over 60% with the exact PL computation method, except for the 99th percentile value under clock-type faults.

2) $\tilde{\mathbf{v}}_{45^\circ}$*-Direction:* In both Figs. 5 and 6, subfigure (d) demonstrates tighter PLs and simulated IRs closer to P_{TIR} than subfigure (a), illustrating the benefit of Bayesian RAIM when the direction of interest is known. Fig. 7 compares the empirical CDFs of PL and PE values with those from the genie variation.



(a) (b) (c) (d) (e) (f)

Fig. 6. Performances of the RAIM algorithms in the form of Stanford diagrams under clock synchronization type fault conditions, with same TIR requirement $P_{\mathrm{TIR}} = 10^{-3}$ for all subspaces. In (a)-(d), a pixel stands for 0.01×0.01 m^2; in (e) and (f), a pixel stands for 0.2×0.2 m^2. (a) Bayesian RAIM, H-plane Alg. 1. (b) Bayesian RAIM, H-plane overestimate. (c) Baseline RAIM, H-plane. (d) Bayesian RAIM, $\tilde{\mathbf{v}}_{45^\circ}$-direction. (e) Bayesian, V-direction. (f) Baseline, V-direction.



(a) (b)

Fig. 7. Empirical CDF of PL and PE obtained by Bayesian RAIM between (1) our implementation and (2) the genie variation. (a) NLOS fault. (b) Clock synchronization fault.

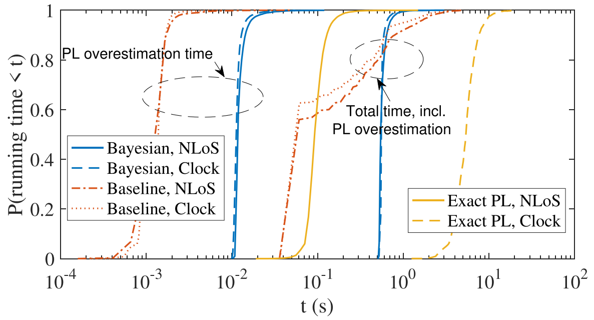


Fig. 8. Empirical CDF of running times, including the total and PL computation times for Baseline RAIM and Bayesian RAIM with PL overestimation, along with the running time for Algorithm 1.

The PE CDFs are nearly identical, and the PL CDFs are also very close, particularly for NLoS faults.

3) V-Direction: In both figures, subfigures (e) and (f) show Baseline RAIM achieving simulated IRs in the order of 10^{-5}, while Bayesian RAIM remains close to 10^{-3}. The relatively low BS heights reduce vertical positioning capability, resulting in significantly larger PL values compared to the H-plane. Point clouds vary significantly in shape depending on the algorithm and fault type, influenced by BS geometry. Notably, they cluster into distinct groups under NLoS-type fault conditions in Fig. 5(e) and (f), due to the proximity of the 7th BS to the UE, making its large measurement biases influential. Specifically, in Fig. 5(e), cluster 1 is generated by approximately 3.44×10^5 epochs where the 7th measurement is fault-free; and cluster 2 consists of 51 epochs where the 7th measurement is faulty but has a low posterior fault probability θ_i' (46), resulting in too small PLs to upper bound the PEs. In Fig. 5(f), cluster 3 consists mainly of epochs with fault modes where the 7th measurement is incorrectly excluded or faulty but not detected. Despite this, Bayesian RAIM consistently produces much lower PL values compared to Baseline RAIM for both fault types, with reductions ranging from 55.0% to 93.1%, as shown in Table I.

4) Running Time: Fig. 8 shows empirical CDFs for the total and PL computation times of Baseline RAIM and Bayesian RAIM with PL overestimation. Fault type shows little effect on these results. Baseline RAIM's total runtime is more variable and can exceed that of Bayesian RAIM with overestimation, which is consistently around 0.55 seconds (median). PL overestimation adds minimal overhead to both algorithms. Figure 8 also presents CDFs for exact PL computation time, which is much longer than the overestimation method and highly dependent on fault type: about 0.1 seconds for NLoS faults and several seconds for

TABLE II

BAYESIAN RAIM PERFORMANCE WITH INITIAL POSITIONING ERRORS: (1) 45°-DIRECTION + NLOS-FAULT, (2) 45°-DIRECTION + CLOCK-FAULT, (3) V-DIRECTION + NLOS-FAULT, (4) V-DIRECTION + CLOCK-FAULT

		PL@50% [m]			Simulated IR [$\times 10^{-3}$]		
		min	max	CV	min	max	CV
E_H	1	0.71	0.71	0.01%	0.94	1.0	1.6%
	2	0.73	0.73	0.02%	0.87	0.96	2.4%
	3	7.32	7.52	0.96%	0.96	1.1	3.3%
	4	34.0	34.1	0.06%	0.78	0.95	5.9%
E_V	1	0.70	0.74	1.8%	0.84	1.1	7.5%
	2	0.72	0.76	1.4%	0.87	1.0	4.5%
	3	5.31	11.04	25.7%	0.94	5.4	77.0%
	4	16.7	47.6	34.6%	0.76	1.2	13.2%

TABLE III

BASELINE RAIM PERFORMANCE WITH INITIAL POSITIONING ERRORS: (1) H-PLANE + NLOS-FAULT, (2) H-PLANE + CLOCK-FAULT, (3) V-DIRECTION + NLOS-FAULT, (4) V-DIRECTION + CLOCK-FAULT

		PL@50% [m]			Simulated IR [$\times 10^{-5}$]		
		min	max	CV	min	max	CV
E_H	1	2.31	2.34	0.38%	4.4	4.6	2.3%
	2	2.31	2.33	0.35%	2.4	2.8	5.9%
	3	105.5	106.5	0.32%	1.6	1.8	5.3%
	4	105.5	106.4	0.27%	1.8	2.6	13.7%
E_V	1	2.20	2.35	2.4%	1.8	5.8	31.3%
	2	2.20	2.35	2.4%	2.6	4.6	17.1%
	3	55.5	148.4	33.6%	0.4	3.8	54.5%
	4	55.5	148.3	33.6%	2.0	3.4	22.5%

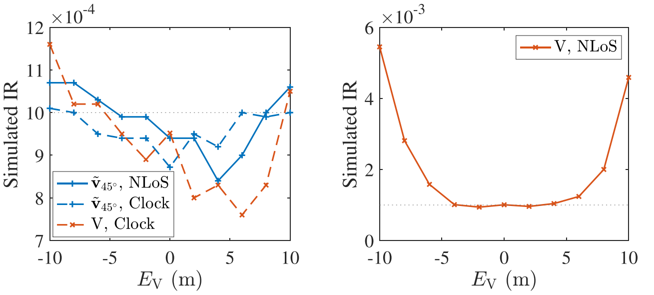
clock faults. This difference arises from the GM posterior PDF weights: with NLoS faults, only a few terms are significant, while clock faults involve many terms. For $\zeta_2 = 0.002$ (see (24)), the GM model typically reduces to fewer than 10 terms for NLoS faults, but several hundred for clock faults; even with $\zeta_2 = 0.1$, around 100 terms may remain.

C. Evaluation of Sensitivity to Linearization Error

We evaluate the sensitivity of the algorithms to initial positioning errors using the 50th percentile PL and the simulated IR. As the results turn show relative insensitivity as E_H varying in the $[0, 5]$ m range and E_V in $[-10, 10]$ m range, we show them as tables as opposed to plots. Specifically, Table II presents their minimum, maximum, and coefficient of variation (CV, defined as the ratio of the standard deviation to the mean) for Bayesian RAIM, while Table III provides these metrics for Baseline RAIM.

1) Bayesian RAIM: As Table II shows, the horizontal initial positioning error E_H has minimal effect on the 50th percentile PL for both the H-plane (cases 1-2) and the V-direction (cases 3-4). However, it can cause the maximum simulated IR (case 3) to exceed the TIR requirement. In contrast, the vertical initial positioning error E_V significantly affects these metrics, especially in the V-direction. Under NLoS fault conditions, the simulated IR in the V-direction (case 3) exceeds five times the TIR. Fig. 9 displays the curves of the simulated IR over E_V, showing that the IR increases with $|E_\mathrm{V}|$. No clear trends were observed for other metrics, so they are not included here.

2) Baseline RAIM: Table III shows that initial positioning errors similarly affect the performance of Baseline RAIM. However, because it computes much looser PLs, the simulated IRs remain in the order of 10^{-5}. Additionally, the 50th percentile PL values under both NLoS and Clock-type faults are almost identical, indicating the insensitivity of Baseline RAIM to the fault distribution. This contrasts with the Bayesian RAIM, which exhibits sensitivity to fault distribution.



(a) (b)

Fig. 9. Simulated IR performance of Bayesian RAIM with vertical initial positioning error. (a) Cases 1,2 and 4. (b) Case 3.

3) Discussion and Implications: Bayesian RAIM computes the exact posterior PDFs of the UE's position based on the linear measurement model (2) and the distributions of bias and noise, making it sensitive to model mismatches. Due to the relative low heights of the BS, the already small third entry of $\mathbf{g}_i = (\mathbf{x}_{\mathrm{u},0} - \mathbf{x}_i)/\|\mathbf{x}_{\mathrm{u},0} - \mathbf{x}_i\| \approx (\mathbf{x}_\mathrm{u} - \mathbf{x}_i + \mathbf{e}_{0,\mathrm{V}})/\|\mathbf{x}_\mathrm{u} - \mathbf{x}_i\|$, is significantly affected by high vertical error magnitude $|E_\mathrm{V}|$. With an average BS height of just 19.2 meters in our simulation, a maximum $|E_\mathrm{V}|$ value of 10 meters can cause a substantial error in the unit vector. The results reaffirm that effective positioning performance, including robust integrity assurance, critically hinges on the geometric placement of BSs. Additionally, investigating methods to incorporate potential model errors into the problem formulation emerges as a promising direction for future research.

VII. CONCLUSION

In this paper, we developed a Bayesian RAIM algorithm for ToA-based, snapshot 3D cellular positioning to handle measurement faults. Using message passing on a factor graph and a linearized measurement model, the algorithm efficiently computes the posterior PDF of the UE position. We introduced computational rules for Gaussian messages to ensure accurate scaling and robust handling of degenerate cases. This approach fully leverages all available measurements, enabling exact, low-cost computation of 1D PLs in any direction, as well as immediate overestimates for 2D or 3D PLs. We also developed an exact 2D/3D PL search by relating the probability of a Gaussian vector inside an ellipsoid to the CDF of a generalized chi-squared variable.

For performance evaluation, we adapted an advanced RAIM algorithm [17] as a baseline. Monte Carlo simulations show that our Bayesian RAIM achieves over 50% tighter PLs in all cases (Table I), even with the efficient PL overestimation approach, and closely meets the TIR requirement with exact PL computation. PL tightness is further confirmed by comparison with the genie variation. Results also highlight the importance of good BS geometry: under favorable conditions, Bayesian RAIM remains robust to initial position errors during linearization, making it an effective and reliable solution for 3D cellular positioning integrity.

In this paper, our primary goal is to demonstrate the Bayesian RAIM principle using simplified, idealized models as a first step. However, several extensions are possible. First, incorporating non-Gaussian noise models (e.g., Laplacian or heavy-tailed distributions) and nonlinear measurements (e.g., unlinearized ToA and angle-of-arrival (AoA)) could better capture real-world effects. Sensitivity analyses with respect to noise model mismatches would further elucidate the robustness of the proposed algorithms. Furthermore, validation using real-world datasets or synthetic datasets generated by deterministic (e.g., ray-tracing) or standardized stochastic (e.g., 3GPP TR 38.901) channel models would provide a more comprehensive assessment of performance under practical conditions. Finally, extensions to dynamic scenarios involving user and fault tracking are natural, as the factor graph framework inherently supports sequential belief updates across time steps.

APPENDIX A
PROOF OF LEMMA 1

Lemma 1 follows since

$$\Pr\{\|\mathbf{e}_{n\mathrm{D}}\| > \mathrm{PL}_{n\mathrm{D}}^{U}\} \overset{(a)}{\leq} \Pr\{\cup_{i=1}^{n} |e_i| > \mathrm{PL}_{1\mathrm{D},i}\}$$
$$\overset{(b)}{\leq} \sum_{i=1}^{n} \Pr\{|e_i| > \mathrm{PL}_{1\mathrm{D},i}\} \overset{(c)}{<} \sum_{i=1}^{n} w_i\, P_{\mathrm{TIR}} = P_{\mathrm{TIR}}.$$

The inequality (a) can be verified from geometry: $\cup_{i=1}^{n} |e_i| > \mathrm{PL}_{1\mathrm{D},i}$ describes the event when the error vector $\mathbf{e}_{n\mathrm{D}}$ is located outside the 2D/3D box with edge lengths given by $\{2\mathrm{PL}_{1\mathrm{D},i}\}$; while $\|\mathbf{e}_{n\mathrm{D}}\| > \mathrm{PL}_{n\mathrm{D}}^{U}$ is the event when $\mathbf{e}_{n\mathrm{D}}$ is located outside the 2D circle/3D sphere with radius $\mathrm{PL}_{n\mathrm{D}}^{U}$, which encloses the box. The inequality (b) follows the union bound, and (c) follows from the assumption (5).

APPENDIX B
MESSAGE PASSING RULES FOR PROBLEM 1

B. *General Gaussian Distribution*

A rigorous definition of a general Gaussian random vector, following [35, Definition 23.1.1], is given below. An equivalent definition can be found in [36, Definition 2.1].

Definition 2: (Gaussian Random Vector) A real random vector $\boldsymbol{X} = (X_1, \ldots, X_n)^{\mathrm{T}}$ is said to be Gaussian if there exists a deterministic matrix $\mathsf{A} \in \mathbb{R}^{n\times m}$ and a deterministic vector $\mathbf{m} \in \mathbb{R}^{n}$ such that the distribution of $\boldsymbol{X}$ is equal to that of $\mathsf{A}\boldsymbol{Z} + \mathbf{m}$, where $\mathbf{Z}$ is a standard Gaussian vector with m components. The distribution is denoted by $\boldsymbol{X} \sim \mathcal{N}(\mathbf{x}; \mathbf{m}, \Sigma)$, where $\Sigma \triangleq \mathsf{A}\mathsf{A}^{\mathrm{T}} \succeq 0$ is the covariance matrix. The PDF of $\mathbf{X}$ is given by

$$f_{\boldsymbol{X}}(\mathbf{x}) = \frac{\exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{m})^{\mathrm{T}}\Sigma^{+}(\mathbf{x}-\mathbf{m})\right]}{(2\pi)^{k/2}|\Sigma|_{+}^{1/2}}, \tag{35}$$

where $k = \mathrm{rank}(\Sigma)$, and Σ^{+} and $|\Sigma|_{+}$ are the pseudo-inverse[8] and pseudo-determinant[9] of Σ, respectively.

In this definition, n can be greater than, equal to, or smaller than m; $k \leq \min(m, n)$. When $\Sigma \succ 0$, Σ^{+} and $|\Sigma|_{+}$ coincide with the regular inverse Σ^{-1} and determinant $|\Sigma|$. Expanding (35), we have

$$f_{\boldsymbol{X}}(\mathbf{x}) = \frac{\exp\left(-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\Sigma^{+}\mathbf{x} + \mathbf{x}^{\mathrm{T}}\Sigma^{+}\mathbf{m} - \frac{1}{2}\mathbf{m}^{\mathrm{T}}\Sigma^{+}\mathbf{m}\right)}{(2\pi)^{k/2}|\Sigma|_{+}^{1/2}} \tag{36}$$

$$= \frac{|\mathsf{V}|_{+}^{1/2}\exp\left(-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathsf{V}\mathbf{x} + \mathbf{x}^{\mathrm{T}}\mathbf{u} - \frac{1}{2}\mathbf{u}^{\mathrm{T}}\mathsf{V}^{+}\mathbf{u}\right)}{(2\pi)^{k/2}}, \tag{37}$$

where $\mathsf{V} \triangleq \Sigma^{+}$, $\mathbf{u} \triangleq \mathsf{V}\mathbf{m}$, and it can be easily verified that $\mathbf{m}^{\mathrm{T}}\Sigma^{+}\mathbf{m} = \mathbf{u}^{\mathrm{T}}\mathsf{V}^{+}\mathbf{u}$. This demonstrates that the Gaussian distribution can be equivalently parameterized using $(\mathbf{m}, \Sigma)$ or $(\mathbf{u}, \mathsf{V})$. Note that $\mathrm{rank}(\Sigma) = \mathrm{rank}(\mathsf{V})$. To avoid ambiguity, we denote the PDF expression in (35) by $f_{\boldsymbol{X}}(\mathbf{x}; \mathbf{m}, \Sigma)$ and in (37) by $f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}, \mathsf{V})$. Finally, recall $\alpha_{\boldsymbol{X}} \triangleq -\frac{1}{2}\mathbf{m}^{\mathrm{T}}\Sigma^{+}\mathbf{m} \equiv -\frac{1}{2}\mathbf{u}^{\mathrm{T}}\mathsf{V}^{+}\mathbf{u}$, as defined in (8).

B. *Proof of Lemma 2*

Consider $\mathbf{u}_{\boldsymbol{X}}$ and $\mathsf{V}_{\boldsymbol{X}}$ given in (9b). Since $\mathsf{A} \in \mathbb{R}^{m\times n}$ has full row rank, we have $\mathrm{rank}(\mathsf{V}_{\boldsymbol{X}}) = \mathrm{rank}(\mathsf{V}_{\boldsymbol{Y}}) = m$ and $\mathsf{A}\mathsf{A}^{+} = \mathsf{I}_m$. Therefore $\mathbf{u}_{\boldsymbol{X}}^{\mathrm{T}}\mathsf{V}_{\boldsymbol{X}}^{+}\mathbf{u}_{\boldsymbol{X}} = \mathbf{u}_{\boldsymbol{Y}}^{\mathrm{T}}\mathsf{A}\mathsf{A}^{+}\mathsf{V}_{\boldsymbol{Y}}^{+}(\mathsf{A}\mathsf{A}^{+})^{\mathrm{T}}\mathbf{u}_{\boldsymbol{Y}} = \mathbf{u}_{\boldsymbol{Y}}^{\mathrm{T}}\mathsf{V}_{\boldsymbol{Y}}^{+}\mathbf{u}_{\boldsymbol{Y}}$, where the property $(\mathsf{A}\mathsf{B})^{+} = \mathsf{B}^{+}\mathsf{A}^{+}$ is used. Thus, $\alpha_{\mathbf{X}} = \alpha_{\mathbf{Y}}$. Given this, we confirms the correctness of (9a)–(9c):

$$f_{\boldsymbol{Y}}^{\mathrm{E}}(\mathsf{A}\mathbf{x}; \mathbf{u}_{\boldsymbol{Y}}, \mathsf{V}_{\boldsymbol{Y}})$$
$$= \frac{|\mathsf{V}_{\boldsymbol{Y}}|_{+}^{1/2}}{(2\pi)^{m/2}}\exp\left[-\frac{1}{2}(\mathsf{A}\mathbf{x})^{\mathrm{T}}\mathsf{V}_{\boldsymbol{Y}}\mathsf{A}\mathbf{x} + (\mathsf{A}\mathbf{x})^{\mathrm{T}}\mathbf{u}_{\boldsymbol{Y}} + \alpha_{\mathbf{Y}}\right]$$
$$= \underbrace{\frac{|\mathsf{V}_{\boldsymbol{Y}}|_{+}^{1/2}}{|\mathsf{V}_{\boldsymbol{X}}|_{+}^{1/2}}}_{s_{\mathbf{Y}}^{\mathbf{X}}}\underbrace{\frac{|\mathsf{V}_{\boldsymbol{X}}|_{+}^{1/2}}{(2\pi)^{m/2}}\exp\left(-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathsf{V}_{\boldsymbol{X}}\mathbf{x} + \mathbf{x}^{\mathrm{T}}\mathbf{u}_{\boldsymbol{X}} + \alpha_{\mathbf{X}}\right)}_{f_{\mathbf{X}}^{\mathrm{E}}(\mathbf{x};\mathbf{u}_{\boldsymbol{X}},\mathsf{V}_{\boldsymbol{X}})}.$$

C. *Proof of Lemma 3*

The following lemma is need for the proof of Lemma 3. It provides the computation rule for the product of two arbitrary, possibly degenerate, Gaussian densities.

Lemma 4 (Product of two Gaussian densities): Given two messages of $\boldsymbol{X}$ in the form of Gaussian densities: $\mu_{\boldsymbol{X},1}(\mathbf{x}) = f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_1}, \mathsf{V}_{\boldsymbol{X}_1})$ and $\mu_{\boldsymbol{X},2}(\mathbf{x}) = f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_2}, \mathsf{V}_{\boldsymbol{X}_2})$, where $\mathrm{rank}(\mathsf{V}_{\boldsymbol{X}_1}) = k_1$ and $\mathrm{rank}(\mathsf{V}_{\boldsymbol{X}_2}) = k_2$, their product is given by

$$f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_1}, \mathsf{V}_{\boldsymbol{X}_1}) f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_2}, \mathsf{V}_{\boldsymbol{X}_2}) = s_{\boldsymbol{X}_{1:2}} f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}}, \mathsf{V}_{\boldsymbol{X}}), \tag{38a}$$

where $\mathbf{u}_{\boldsymbol{X}} = \mathbf{u}_{\boldsymbol{X}_1} + \mathbf{u}_{\boldsymbol{X}_2}$ and $\mathsf{V}_{\boldsymbol{X}} = \mathsf{V}_{\boldsymbol{X}_1} + \mathsf{V}_{\boldsymbol{X}_2}$, and the scaling factor $s_{\boldsymbol{X}_{1:2}}$ is given by ($k = \mathrm{rank}(\mathsf{V}_{\boldsymbol{X}})$)

$$s_{\boldsymbol{X}_{1:2}} = \frac{|\mathsf{V}_{\boldsymbol{X}_1}|_{+}^{1/2}|\mathsf{V}_{\boldsymbol{X}_2}|_{+}^{1/2}}{(2\pi)^{(k_1+k_2-k)/2}|\mathsf{V}_{\boldsymbol{X}}|_{+}^{1/2}}\exp(\alpha_{\boldsymbol{X}_1} + \alpha_{\boldsymbol{X}_2} - \alpha_{\boldsymbol{X}}). \tag{38b}$$

Proof: Using the PDF expression (37), we immediately have

$$f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_1}, \mathsf{V}_{\boldsymbol{X}_1}) f_{\boldsymbol{X}}^{\mathrm{E}}(\mathbf{x}; \mathbf{u}_{\boldsymbol{X}_2}, \mathsf{V}_{\boldsymbol{X}_2}) = \frac{|\mathsf{V}_{\boldsymbol{X}_1}|_{+}^{1/2}}{(2\pi)^{k_1/2}}\frac{|\mathsf{V}_{\boldsymbol{X}_2}|_{+}^{1/2}}{(2\pi)^{k_2/2}}.$$

---

[8] The pseudo-inverse (Moore–Penrose inverse) of A is the matrix A^{+} that satisfies [32, Section 3.6]: (i) $\mathsf{A}\mathsf{A}^{+}\mathsf{A} = \mathsf{A}$; (ii) $\mathsf{A}^{+}\mathsf{A}\mathsf{A}^{+} = \mathsf{A}^{+}$; (iii) $\mathsf{A}\mathsf{A}^{+}$ is symmetric; (iv) $\mathsf{A}^{+}\mathsf{A}$ is symmetric. The pseudo-inverse is unique and always exists, with $(\mathsf{A}^{+})^{+} = \mathsf{A}$. When A has full-rank, it has explicit expressions: for full row rank, $\mathsf{A}^{+} = \mathsf{A}^{\mathrm{T}}(\mathsf{A}\mathsf{A}^{\mathrm{T}})^{-1}$; for full column rank, $\mathsf{A}^{+} = (\mathsf{A}^{\mathrm{T}}\mathsf{A})^{-1}\mathsf{A}^{\mathrm{T}}$.

[9] The pseudo-determinant of $\Sigma \succeq 0$ is given by $|\Sigma|_{+} \triangleq \prod_{i=1}^{k}\gamma_i^{+}$, where $\gamma_1^{+}, \ldots, \gamma_k^{+}$ are the positive eigenvalues of Σ.

$$\exp\left[-\frac{1}{2}\mathbf{x}^{\mathrm{T}}(\mathsf{V}_{\boldsymbol{X}_1}+\mathsf{V}_{\boldsymbol{X}_2})\mathbf{x}+\mathbf{x}^{\mathrm{T}}(\mathbf{u}_{\boldsymbol{X}_1}+\mathbf{u}_{\boldsymbol{X}_2})+\alpha_{\boldsymbol{X}_1}+\alpha_{\boldsymbol{Y}_2}\right]$$

$$= \underbrace{\frac{|\mathsf{V}_{\boldsymbol{X}_1}|_+^{1/2}|\mathsf{V}_{\boldsymbol{X}_2}|_+^{1/2}}{(2\pi)^{(k_1+k_1-k)/2}|\mathsf{V}_{\boldsymbol{X}}|_+^{1/2}}\exp(\alpha_{\boldsymbol{X}_1}+\alpha_{\boldsymbol{X}_2}-\alpha_{\boldsymbol{X}})\cdot}_{s_{\boldsymbol{X}_{1:2}}}$$

$$\underbrace{\frac{|\mathsf{V}_{\boldsymbol{X}}|_+^{1/2}}{(2\pi)^{k/2}}\exp\left(-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathsf{V}_{\boldsymbol{X}}\mathbf{x}+\mathbf{x}^{\mathrm{T}}\mathbf{u}_{\boldsymbol{X}}+\alpha_{\boldsymbol{X}}\right)}_{f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x};\mathbf{u}_{\boldsymbol{X}},\mathsf{V}_{\boldsymbol{X}})}.$$

If $\mathsf{V}_{\boldsymbol{X}_1} \succ 0$ and $\mathsf{V}_{\boldsymbol{X}_2} \succ 0$, (38a)–(38b) can be obtained by using [32, Eq. (358)-Eq. (364)]. □

Based Lemma 4, the computation rule for the product of multiple Gaussian densities in Lemma 3 can be verified by successively performing the product of two Gaussian densities.

APPENDIX C
MESSAGE PASSING ALGORITHM

In the following description, a Gaussian random variable is treated as a special case ($n = 1$) of a Gaussian random vector and the same notations are used.

Step ①: The message $p_{\Lambda_i}(\lambda_i)$ is sent from the leaf node to the variable node Λ_i and then directly to F_i.

Step ②: Factor node F_i sends the following message, 1D GM consisting of two terms, to variable node B_i:

$$\mu_{F_i\to B_i}(b_i) = \sum_{\lambda_i=0,1} p_{B_i|\Lambda_i=\lambda_i}(b_i)\,p_{\Lambda_i}(\lambda_i)$$
$$= (1-\theta_i)\,f_{B_i}(b_i;0,0)+\theta_i\,f_{B_i}(b_i;m_{b,i},\sigma^2_{b,i}). \quad (39)$$

This prior PDF of B_i is passed directly to factor node G_i.

Step ③: Factor node G_i sends the following message to variable node Γ_i:

$$\mu_{G_i\to\Gamma_i}(\gamma_i) = \int G_i\cdot\mu_{F_i\to B_i}(b_i)\,\mathrm{d}b_i,$$

where $G_i = p_{Y_i|\Gamma_i=\gamma_i,B_i=b_i}(y_i) = f_{Y_i}(y_i;\gamma_i+b_i,\sigma^2_{n,i}) = f_{B_i}(b_i;y_i-\gamma_i,\sigma^2_{n,i})$. Following [1, Eqs. (15)-(16)], the integration results in a 1D GM density with two terms for Γ_i:

$$\mu_{G_i\to\Gamma_i}(\gamma_i) = (1-\theta_i)f_{\Gamma_i}(\gamma_i;y_i,\sigma^2_{n,i})$$
$$+\,\theta_i f_{\Gamma_i}(\gamma_i;y_i-m_{b,i},\sigma^2_{n,i}+\sigma^2_{b,i}). \quad (40)$$

This message is passed directly to factor node K_i.

Step ④: Factor node K_i sends the following message to variable node $\boldsymbol{X}$:

$$\mu_{K_i\to\boldsymbol{X}}(\mathbf{x}) = \int\delta(\gamma_i-\mathbf{h}_i^{\mathrm{T}}\mathbf{x})\cdot\mu_{G_i\to\Gamma_i}(\gamma_i)\,\mathrm{d}\gamma_i.$$

Applying the computations rules (9a), (9b), and (11),

$$\mu_{K_i\to\boldsymbol{X}}(\mathbf{x}) \propto \sum_{l=1,2} w^{(l)}_{\boldsymbol{X}_i}\,f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x};\mathbf{u}^{(l)}_{\boldsymbol{X},i},\mathsf{V}^{(l)}_{\boldsymbol{X}_i}) \quad (41)$$

where

$$w^{(1)}_{\boldsymbol{X}_i} = \frac{1-\theta_i}{\sigma_{n,i}},\quad w^{(2)}_{\boldsymbol{X}_i} = \frac{\theta_i}{(\sigma^2_{n,i}+\sigma^2_{b,i})^{1/2}}, \quad (42a)$$

$$\mathbf{u}^{(1)}_{\boldsymbol{X}_i} = \frac{y_i}{\sigma^2_{n,i}}\mathbf{h}_i,\quad \mathbf{u}^{(2)}_{\boldsymbol{X}_i} = \frac{y_i-m_{b,i}}{\sigma^2_{n,i}+\sigma^2_{b,i}}\mathbf{h}_i \quad (42b)$$

$$\mathsf{V}^{(1)}_{\boldsymbol{X}_i} = \frac{1}{\sigma^2_{n,i}}\mathbf{h}_i\mathbf{h}_i^{\mathrm{T}},\quad \mathsf{V}^{(2)}_{\boldsymbol{X}_i} = \frac{1}{\sigma^2_{n,i}+\sigma^2_{b,i}}\mathbf{h}_i\mathbf{h}_i^{\mathrm{T}}, \quad (42c)$$

Since the α parameter of $\boldsymbol{X}_i$ is the same as that of K_i, we have

$$\alpha^{(1)}_{\boldsymbol{X}_i} = -\frac{1}{2}\frac{y_i^2}{\sigma^2_{n,i}},\quad \alpha^{(2)}_{\boldsymbol{X}_i} = -\frac{1}{2}\frac{(y_i-m_{b,i})^2}{\sigma^2_{n,i}+\sigma^2_{b,i}} \quad (43)$$

The GM model (41) for $\boldsymbol{X}\in\mathbb{R}^4$ consists of two degenerate Gaussian densities in a 1D subspace.

Step ⑤: Node $\boldsymbol{X}$ computes the product of all the messages sent to it: $\mu_{\boldsymbol{X}}(\mathbf{x}) \propto \prod_{j=1}^M \mu_{K_j\to\boldsymbol{X}}(\mathbf{x})$. Since each message is a GM of two terms, the result is a GM of $L \triangleq 2^M$ terms:

$$\mu_{\boldsymbol{X}}(\mathbf{x}) \propto \prod_{j=1}^{M}\left(\sum\nolimits_{l_j=1,2} w^{(l_j)}_{\boldsymbol{X}_j}\,f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x};\mathbf{u}^{(l_j)}_{\boldsymbol{X}_j},\mathsf{V}^{(l_j)}_{\boldsymbol{X}_j})\right)$$
$$\propto \sum_{l=1}^{L} w^{(l)}_{\boldsymbol{X}}\,f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x};\mathbf{u}^{(l)}_{\boldsymbol{X}},\mathsf{V}^{(l)}_{\boldsymbol{X}}). \quad (44)$$

Each term of (44) is computed as following: For $l_j\in\{1,2\}$, $j=1,\ldots,M$, $l\leftarrow 1+\sum_{j=1}^{M}(l_j-1)\,2^{j-1}$, and

$$w^{(l)}_{\boldsymbol{X}}\,f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x};\mathbf{u}^{(l)}_{\boldsymbol{X}}\mathsf{V}^{(l)}_{\boldsymbol{X}}) \propto \prod\nolimits_{j=1}^{M} w^{(l_j)}_{\boldsymbol{X}_j}\,f^{\mathrm{E}}_{\boldsymbol{X}}(\mathbf{x};\mathbf{u}^{(l_j)}_{\boldsymbol{X}_j},\mathsf{V}^{(l_j)}_{\boldsymbol{X}_j}),$$

where, following (10a), (10b), and (12),

$$\mathbf{u}^{(l)}_{\boldsymbol{X}} = \sum_{j=1}^{M}\mathbf{u}^{(l_j)}_{\boldsymbol{X}_j},\quad \mathsf{V}^{(l)}_{\boldsymbol{X}} = \sum_{j=1}^{M}\mathsf{V}^{(l_j)}_{\boldsymbol{X}_j}, \quad (45a)$$

and

$$w^{(l)}_{\boldsymbol{X}} = s^{(l)}_{\boldsymbol{X}}\prod_{j=1}^{M} w^{(l_j)}_{\boldsymbol{X}_j}, \quad (45b)$$

where, after omitting the constant $(2\pi)^{(\sum_{i=1}^{K}k_i-k)/2}$ in (12), the scaling factor is given by

$$s^{(l)}_{\boldsymbol{X}} = \frac{1}{|\mathsf{V}^{(l)}_{\boldsymbol{X}}|_+^{1/2}}\exp\left(-\alpha^{(l)}_{\boldsymbol{X}}+\sum_{j=0}^{M}\alpha^{(l_j)}_{\boldsymbol{X}_j}\right). \quad (45c)$$

Step ⑥: Variable node $\boldsymbol{X}$ sends a message $\mu_{\boldsymbol{X}\to K_i}(\mathbf{x}) \propto \prod_{j\neq i}\mu_{K_j\to\boldsymbol{X}}(\mathbf{x})$ to factor node K_i. The computation follows the same rules as step ⑤ and result in a GM of 2^{M-1} terms.

Step ⑦: Factor node K_i converts $\mu_{\boldsymbol{X}\to K_i}(\mathbf{x})$ to $\mu_{K_i\to\Gamma_i}(\gamma_i)$, a 1D GM model of Γ_i. Each term of $\mu_{\boldsymbol{X}\to K_i}(\mathbf{x})$ is converted to a Gaussian density of Γ_i following the linear mapping[10] $\Gamma_i = \mathbf{h}_i^{\mathrm{T}}\boldsymbol{X}$, while retaining its weight. This message is sent to Γ_i and then directly to factor node G_i.

---

[10] The Gaussian message passing rule for the linear mapping $\boldsymbol{Y} = \mathsf{A}\boldsymbol{X}$ is trivial. Given $\mu_{\boldsymbol{X}}(\mathbf{x}) = f_{\boldsymbol{X}}(\mathbf{x};\mathbf{m}_{\boldsymbol{X}},\Sigma_{\boldsymbol{X}})$, the inferred message for $\boldsymbol{Y}$ is $\mu_{\boldsymbol{Y}}(\mathbf{y}) = f_{\boldsymbol{Y}}(\mathbf{y};\mathbf{m}_{\boldsymbol{Y}},\Sigma_{\boldsymbol{Y}})$ where $\mathbf{m}_{\boldsymbol{Y}} = \mathsf{A}\mathbf{m}_{\boldsymbol{X}}$ and $\Sigma_{\boldsymbol{Y}} = \mathsf{A}\Sigma_{\boldsymbol{X}}\mathsf{A}^{\mathrm{T}}$.

Step ⑧: Factor node G_i sends $\mu_{G_i \to B_i}(b_i)$, a 1D GM model of 2^{M-1} terms, to variable node B_i and then to F_i.

Step ⑨: Factor node F_i sends a message $\mu_{F_i \to \Lambda_i}(\lambda_i)$, which is the posterior PMF of Λ_i after normalization, to variable node Λ_i. Specifically, the posterior probability of measurement y_i being faulty is given by

$$\theta'_i \triangleq \frac{\mu_{F_i \to \Lambda_i}(\lambda_i = 1)}{\mu_{F_i \to \Lambda_i}(\lambda_i = 0) + \mu_{F_i \to \Lambda_i}(\lambda_i = 1)}. \tag{46}$$

APPENDIX D
PROOF OF THEOREM 1

We first show that the probability of a Gaussian distributed vector in $\mathbb{R}^N$ lying within an arbitrary ellipsoid in $\mathbb{R}^N$ can be calculated by computing the CDF of a generalized chi-squared random variable. An ellipsoid in $\mathbb{R}^N$ centered at $\mathbf{c} \in \mathbb{R}^N$ is defined as $\mathcal{E}(\mathsf{A}, \mathbf{c}, \rho) = \{\mathbf{x} \in \mathbb{R}^N : (\mathbf{x} - \mathbf{c})^{\mathrm{T}}\mathsf{A}(\mathbf{x} - \mathbf{c}) \leq \rho\}$, where $\mathsf{A} \in \mathbb{R}^{N \times N}$ is symmetric positive definite, and ρ is a positive real value. The eigendecomposition of A is $\mathsf{A} = \mathsf{Q}\mathsf{D}\mathsf{Q}^{\mathrm{T}}$, where $\mathsf{D} = \mathrm{diag}(d_1, \ldots, d_N)$ and $\mathsf{Q} \in \mathbb{R}^{N \times N}$ is orthogonal. The lengths of the N semi-axes are given by $\rho/\sqrt{d_1}, \ldots, \rho/\sqrt{d_N}$. Consider a random vector $\boldsymbol{X} \in \mathbb{R}^N$ that follows a Gaussian distribution $\mathcal{N}(\mathbf{x}; \mathbf{m}, \Sigma)$ where $\Sigma \succ 0$. Let $Z = (\boldsymbol{X} - \mathbf{c})^{\mathrm{T}}\mathsf{A}(\boldsymbol{X} - \mathbf{c})$. The probability of $\boldsymbol{X}$ lying within $\mathcal{E}(\mathsf{A}, \mathbf{c}, \rho)$ is

$$\int_{\mathcal{E}(\mathsf{A}, \mathbf{c}, \rho)} f_{\boldsymbol{X}}(\mathbf{x}; \mathbf{m}, \Sigma) \mathrm{d}\mathbf{x} = F_Z(\rho), \tag{47}$$

where $F_Z(z) \triangleq \Pr(Z \leq z)$ is the CDF of Z. We will show that Z is a generalized chi-squared random variable determined by $\mathbf{m}$, Σ, A, $\mathbf{c}$.

Given $\mathsf{A} \succ 0$ and $\Sigma \succ 0$, we have

$$\Sigma^{\frac{1}{2}}\mathsf{A}\Sigma^{\frac{1}{2}} = \mathsf{P}\Omega\mathsf{P}^{\mathrm{T}}, \tag{48}$$

where $\Omega = \mathrm{diag}(\omega_1, \ldots, \omega_N)$ with $\omega_i > 0$, and $\mathsf{P} \in \mathbb{R}^{N \times N}$ is orthogonal. It can be shown that $\boldsymbol{Y} = \mathsf{P}^{\mathrm{T}}\Sigma^{-\frac{1}{2}}(\boldsymbol{X} - \mathbf{c}) \sim \mathcal{N}(\mathbf{y}; \boldsymbol{\nu}, \mathsf{I}_N)$, where

$$\boldsymbol{\nu} \triangleq \mathsf{P}^{\mathrm{T}}\Sigma^{-\frac{1}{2}}(\mathbf{m} - \mathbf{c}). \tag{49}$$

Letting Y_i be the i-th element of $\boldsymbol{Y}$ and ν_i be the i-th entry of $\boldsymbol{\nu}$. Then Y_i^2 follows a noncentral chi-square distribution with one degree of freedom and noncentrality parameter ν_i^2 (i.e., $Y_i^2 \sim \chi^2(1, \nu_i^2)$). We can rewrite Z as

$$Z = \boldsymbol{Y}^{\mathrm{T}}\Omega\boldsymbol{Y} = \sum_{i=1}^{N} \omega_i Y_i^2. \tag{50}$$

This shows that Z is a weighted sum of N independent noncentral chi-square random variables. Summarizing the above, we have the following lemma.

Lemma 5: The probability that a Gaussian random vector $\boldsymbol{X} \in \mathbb{R}^N \sim \mathcal{N}(\mathbf{x}; \mathbf{m}, \Sigma)$ lies within an ellipsoid $\mathcal{E}(\mathsf{A}, \mathbf{c}, \rho) \in \mathbb{R}^N$ is given by $F_Z(\rho)$, the CDF of a generalized chi-squared random variable Z evaluated at ρ. Specifically, $Z = \sum_{i=1}^{N} \omega_i W_i$ where $W_i \sim \chi^2(1, \nu_i^2)$, $i = 1 \ldots, N$, and the weights $\{\omega_i\}$ and noncentrality parameters $\{\nu_i^2\}$ are determined by $\mathbf{m}$, Σ, A, $\mathbf{c}$ via (48) and (49).

By setting $\mathsf{A} = \mathsf{I}_n$, $\mathbf{c} = \mathbf{0}$, and $\rho = r^2$, the ellipsoid $\mathcal{E}(\mathsf{A}, \mathbf{c}, \rho)$ becomes a circle (for $n = 2$) or an sphere (for $n = 3$) centered at the origin with radius r. Using the PDFs of $\mathbf{e}_{n\mathrm{D}}$ given by (14) and (16), the probability that $\mathbf{e}_{n\mathrm{D}}$ lies within a circle/sphere of radius r is $\Pr\{\mathbf{e}_{n\mathrm{D}} \in \mathcal{E}(\mathsf{I}_n, \mathbf{0}, r^2)\} = \sum_{l=1}^{L} w_{\boldsymbol{X}}^{(l)} F_{Z_l}(r^2)$, where $F_{Z_l}(r^2)$ is the CDF of the generalized chi-squared random variable Z_l given by (20) computed at r^2. With this, the proof of Theorem 1 follows directly from Lemma 5.

APPENDIX E
NUMERICAL COMPUTATION METHOD FOR THE CDF OF A GENERALIZED CHI-SQUARED RANDOM VARIABLE

There is no closed-form expression for the PDF, CDF, and inverse CDF of a generalized chi-squared variable $Z = \sum_{i=1}^{N} \omega_i W_i$, for $W_i \sim \chi^2(k_i, \theta_i)$, where $\chi^2(k_i, \theta_i)$ denotes a noncentral chi-square distribution with k_i degrees of freedom and noncentral parameter θ_i. (Setting $k_i = 1$ and $\theta_i = \nu_i^2$ yields (50).) Numerical methods can be used instead. The Imholf method computes $F_Z(z)$, the CDF of Z, as a numerical integral $F_Z(z) \approx \frac{1}{2} - \frac{1}{\pi}\int_0^U \frac{\sin\beta(u,z)}{u\kappa(u)}\mathrm{d}u$, where

$$\beta(u, z) = \frac{1}{2}\sum_{i=1}^{N}\left(k_i \arctan(\omega_i u) + \frac{\theta_i \omega_i u}{1 + \omega_i^2 u^2}\right) - \frac{1}{2}zu, \tag{51a}$$

$$\kappa(u) = \exp\left(\frac{1}{4}\sum_{i=1}^{N} k_i \ln(1 + \omega_i^2 u^2) + \frac{1}{2}\sum_{i=1}^{N}\frac{\theta_i \omega_i^2 u^2}{1 + \omega_i^2 u^2}\right). \tag{51b}$$

The error for terminating at U is given by $\xi(U) = \frac{1}{\pi}\int_U^{\infty} \frac{\sin\beta(u,y)}{u\kappa(u)}\mathrm{d}u$. It is shown in [37] that $|\xi(U)| \leq \Xi(U)$, where

$$\Xi(U) = \left[\pi K U^K \prod_{i=1}^{N} |\omega_i|^{\frac{k_i}{2}} \exp\left(\frac{1}{2}\sum_{i=1}^{n}\frac{\theta_i \omega_i^2 U^2}{1 + \omega_i^2 U^2}\right)\right]^{-1} \tag{52}$$

with $K = \frac{1}{2}\sum_{i=1}^{N} k_i$. Thus, for any required accuracy ϵ, a sufficiently large U can be found to ensure $\Xi(U) \leq \epsilon$.

ACKNOWLEDGMENT

The authors would like to thank all partners within Hi-Drive for their cooperation and valuable contribution.

REFERENCES

[1] L. Ding et al., "Bayesian integrity monitoring for cellular positioning—A simplified case study," in *Proc. IEEE Int. Conf. Commun. Workshops*, 2023, pp. 1050–1056.

[2] N. Zhu, J. Marais, D. Bétaille, and M. Berbineau, "GNSS position integrity in urban environments: A review of literature," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 2762–2778, Sep. 2018.

[3] O. K. Isik, J. Hong, I. Petrunin, and A. Tsourdos, “Integrity analysis for GPS-based navigation of UAVs in urban environment,” *Robotics*, vol. 9, no. 3, 2020, Art. no. 66.

[4] E. D. Kaplan and C. J. Hegarty, *Understanding GPS: Principles and Applications*, 2nd ed., Norwood, MA, USA: Artech House, 2006.

[5] C. De Lima et al., “Convergent communication, sensing and localization in 6 G systems: An overview of technologies, opportunities and challenges,” *IEEE Access*, vol. 9, pp. 26902–26 925, 2021.

[6] P. Hammarberg, J. Vinogradova, G. Fodor, R. Shreevastav, S. Dwivedi, and F. Gunnarsson, “Architecture, protocols, and algorithms for location-aware services in beyond 5G networks,” *IEEE Commun. Standards Mag.*, vol. 6, no. 4, pp. 88–95, Dec. 2022.

[7] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados, “Survey of cellular mobile radio localization methods: From 1G to 5G,” *IEEE Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 1124–1148, Second Quarter 2017.

[8] C. Laoudias, A. Moreira, S. Kim, S. Lee, L. Wirola, and C. Fischione, “A survey of enabling technologies for network localization, tracking, and navigation,” *IEEE Commun. Survsys Tuts.*, vol. 20, no. 4, pp. 3607–3644, Fourth Quarter 2018.

[9] H. Xue, H. Zhu, S. Cao, S. Chang, and J. Cao, “UPS: Combatting urban vehicle localization with cellular-aware trajectories,” in *Proc. IEEE Global Commun. Conf.*, 2016, pp. 1–7.

[10] S. Bartoletti et al., “Positioning and sensing for vehicular safety applications in 5G and beyond,” *IEEE Commun. Mag.*, vol. 59, no. 11, pp. 15–21, Nov. 2021.

[11] A. Behravan et al., “Positioning and sensing in 6G: Gaps, challenges, and opportunities,” *IEEE Veh. Technol. Mag.*, vol. 18, no. 1, pp. 40–48, Mar. 2023.

[12] R. Whiton, “Cellular localization for autonomous driving: A function pull approach to safety-critical wireless localization,” *IEEE Veh. Technol. Mag.*, vol. 17, no. 4, pp. 2–11, Dec. 2022.

[13] B. W. Parkinson and P. Axelrad, “Autonomous GPS integrity monitoring using the pseudorange residual,” *Navigation*, vol. 35, no. 2, pp. 255–274, 1988.

[14] R. G. Brown, “Receiver autonomous integrity monitoring,” *Glob. Positioning Syst.: Theory Appl.*, vol. 2, pp. 143–165, 1996.

[15] G. Schroth, A. Ene, J. Blanch, T. Walter, and P. Enge, “Failure detection and exclusion via range consensus,” in *Proc. Eur. Navigation Conf*, 2008, pp. 1–10.

[16] G. Castaldo, A. Angrisano, S. Gaglione, and S. Troisi, “P-RANSAC: An integrity monitoring approach for GNSS signal degraded scenario,” *Int. J. Navigation Observ.*, vol. 2014, Sep. 2014, Art. no. 173818.

[17] J. Blanch et al., “Baseline advanced RAIM user algorithm and possible improvements,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 1, pp. 713–732, Jan. 2015.

[18] E. Rastorgueva-Foi et al., “Networking and positioning co-design in multi-connectivity industrial MMW systems,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15842–15856, Dec. 2020.

[19] A. Xhafa, J. A. del Peral-Rosado, J. A. López-Salcedo, and G. Seco-Granados, “Evaluation of 5 G positioning performance based on UTDoA, AoA and base-station selective exclusion,” *Sensors*, vol. 22, no. 1, 2021, Art. no. 101.

[20] M. Koivisto et al., “Joint device positioning and clock synchronization in 5G ultra-dense networks,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 2866–2881, May 2017.

[21] C. Xu, Z. Wang, Y. Wang, Z. Wang, and L. Yu, “Three passive TDOA-AOA receivers-based flying-UAV positioning in extreme environments,” *IEEE Sensors J.*, vol. 20, no. 16, pp. 9589–9595, Aug. 2020.

[22] P. Ober, “Integrity according to bayes,” in *Proc. IEEE Position Location Navigation Symp.*, 2000, pp. 325–332.

[23] H. Pesonen, “A framework for bayesian receiver autonomous integrity monitoring in urban navigation,” *Navigation*, vol. 58, no. 3, pp. 229–240, 2011.

[24] S. Gupta and G. X. Gao, “Particle RAIM for integrity monitoring,” in *Proc. 32nd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2019, pp. 811–826.

[25] J. Gabela, A. Kealy, M. Hedley, and B. Moran, “Case study of bayesian RAIM algorithm integrated with spatial feature constraint and fault detection and exclusion algorithms for multi-sensor positioning,” *Navigation*, vol. 68, no. 2, pp. 333–351, 2021.

[26] H.-A. Loeliger, “An introduction to factor graphs,” *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Jan. 2004.

[27] H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang, “The factor graph approach to model-based signal processing,” *Proc. IEEE*, vol. 95, no. 6, pp. 1295–1322, Jun. 2007.

[28] I. Guvenc, S. Gezici, and Z. Sahinoglu, “Fundamental limits and improved algorithms for linear least-squares wireless position estimation,” *Wireless Commun. Mob. Comput.*, vol. 12, no. 12, pp. 1037–1052, 2012.

[29] S. Zhu and Z. Ding, “A simple approach of range-based positioning with low computational complexity,” *IEEE Trans. Wireless Commun.*, vol. 8, no. 12, pp. 5832–5836, Dec. 2009.

[30] H. Jing, Y. Gao, S. Shahbeigi, and M. Dianati, “Integrity monitoring of GNSS/INS based positioning systems for autonomous vehicles: State-of-the-art and open challenges,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14166–14187, Sep. 2022.

[31] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[32] K. B. Petersen and M. S. Pedersen, “The matrix cookbook, version,” Tech. Univ. Denmark, Nov. 2012.

[33] D. A. Bodenham and N. M. Adams, “A comparison of efficient approximations for a weighted sum of CHI-squared random variables,” *Statist. Comput.*, vol. 26, no. 4, pp. 917–928, 2016.

[34] 3GPP, “Study on expanded and improved NR positioning,” *3rd Gener. Partnership Project (3GPP)*, Tech. Rep. (TR) 38.859, Dec 2022.

[35] A. Lapidoth, *A Foundation in Digital Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[36] V. A. Nguyen, D. Kuhn, and P. Mohajerin Esfahani, “Distributionally robust inverse covariance estimation: The wasserstein shrinkage estimator,” *Operations Res.*, vol. 70, no. 1, pp. 490–515, 2022.

[37] J.-P. Imhof, “Computing the distribution of quadratic forms in normal variables,” *Biometrika*, vol. 48, no. 3/4, pp. 419–426, 1961.

Liqin Ding received the Ph.D. degree in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2017. She is currently a Senior Researcher with Ericsson Research, Ericsson AB, Lund, Sweden. From 2017 to 2019, she was a Postdoctoral Researcher with the School of Electronic and Information Engineering, Harbin Institute of Technology, Shenzhen, China. Between 2019 and 2024, she was a Marie Skłodowska-Curie Fellow (MSCA-IF) and Researcher with the Communication Systems Group, Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden. Her research interests include various areas in signal processing and wireless communication, with a particular focus on antenna array-based communication technologies.

Gonzalo Seco-Granados (Fellow, IEEE) received the Ph.D. degree in telecommunications engineering from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2000, and the M.B.A. degree from IESE Business School, Barcelona, in 2002. From 2002 to 2005, he was with the European Space Agency, where he contributed to the design of the Galileo system. He is currently a Professor with the Department of Telecommunications, Universitat Autònoma de Barcelona (UAB), Bellaterra, Spain, and also the Director with the Centre for Space Studies and Research. In 2015, 2019, and 2022, respectively, he was a Fulbright Visiting Scholar with the University of California, Irvine, CA, USA. He has contributed to the design of two Galileo services, the open service navigation message authentication and the signal authentication service, which are meant to improve the security of the Galileo system. His research interests include signal processing for GNSS, LEO-PNT, and 5G/6G integrated communications, localization, and sensing. From 2018 to 2025, he was the President with the Spanish Chapter of the IEEE Aerospace and Electronic Systems Society. Dr. Seco-Granados was the recipient of the 2021 IEEE Signal Processing Society’s Best Paper Award. He is also with the Institute of Space Studies of Catalonia and is an ICREA Academia Fellow. He has been a member of the IEEE Signal Processing Society Sensor Array and Multichannel Technical Committee (2019–2024) and the EURASIP Signal Processing for Multisensor Systems Technical Committee since 2022.

Hyowon Kim (Member, IEEE) received the Ph.D. degree from the Department of Electronic Engineering, Hanyang University, Seoul, Korea, in 2021. He is currently an Assistant Professor with the Department of Electronics Engineering, Chungnam National University, Daejeon, Korea. From 2021 to 2023, he was a Marie Skłodowska-Curie Fellow/Postdoctoral Researcher with the Department of Electrical Engineering, Chalmers University of Technology, Sweden. His research interests mainly include wireless communications and integrated sensing/localization/communications, Beyond 5G and 6G communication systems.

Russ Whiton received the Ph.D. degree in electrical engineering from Lund University, Lund, Sweden, in 2024. He is currently a System Engineer with European Space Agency, Noordwijk, Netherlands. He was an Engineer with Volvo Cars, Volvo Group Trucks Technology, Sony Mobile Communications, and Qualcomm. His research interests include navigation and wireless systems. Dr. Whiton is a member of the Institute of Navigation and the Nordic Institute of Navigation.

Erik G. Ström (Fellow, IEEE) received the M.S. degree in electrical engineering from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 1990, and the Ph.D. degree in electrical engineering from the University of Florida, Gainesville, FL, USA, in 1994. Since 1990, he has acted as a Consultant with the Educational Group for Individual Development, Stockholm, Sweden. In 1995, he accepted a Postdoctoral Position with the Department of Signals, Sensors, and Systems, KTH, where he was appointed Assistant Professor in Feb. 1996. In June 1996, he joined the Chalmers University of Technology, Göteborg, Sweden, where he has been a Professor of communication systems since June 2003. He currently Heads the Division of Communications, Antennas, and Optical Networks and also the Director of Chalmers' Area-of-Advance Information and Communication Technology. His research interests include signal processing and communication theory in general, and constellation labelings, channel estimation, synchronization, multiple access, medium access, multiuser detection, wireless positioning, and vehicular communications in particular. He is also a Senior Editor of IEEE TRANSACTION ON INTELLIGENT TRANSPORT SYSTEMS, a Contributing Author and Associate Editor for Roy. Admiralty Publishers FesGas-series, and was a Co-Guest Editor of PROCEEDINGS OF THE IEEE special issue on Vehicular Communications in 2011, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS special issues on Signal Synchronization in Digital Transmission Systems in 2001, and on Multiuser Detection for Advanced Communication Systems and Networks in 2008. Dr. Ström was a member of the board of the IEEE VT/COM Swedish Chapter 2000–2006. He was the recipient of the Chalmers Pedagogical Prize in 1998, Chalmers Ph.D. Supervisor of the Year award in 2009, and the Chalmers Area of Advance Award in 2020.

Jonas Sjöberg (Member, IEEE) received the M.Sc. degree in applied physics from Uppsala University, Uppsala, Sweden, in 1989, and the Ph.D. degree from Linkoping University, Sweden, in 1995. After Postdoc with ETH Zurich, he became Assistant Professor with Chalmers, and after research visits with TU Wien, Vienna, Austria, and also with Technion, Haifa, Israel, he became a Professor of mechatronics with the Chalmers University of Technology, Gothenburg, Sweden, in 2001. His research interests include in mechatronics, and mechatronic related fields, such as signal processing, and control, and focuses on model-based methods, simulations, system identification, and optimization for design and product development of mechatronic systems. In 2015, he was Co-Main Chair with FASTzero symposium, and Main Chair with IEEE Intelligent Vehicles Symposium in 2016. Between 2017 and 2023, he was a BOG Member of IEEE Intelligent Transportation Systems Society. In 2024, he was Program Chair for the IEEE Intelligent Vehicles Symposium. He was the recipient of the Volvo Cars Technology Award 2011, and the Håkan Frisinger Award 2017.

Henk Wymeersch (Fellow, IEEE) received the Ph.D. degree in electrical engineering/applied sciences from Ghent University, Ghent, Belgium, in 2005. From 2005 to 2009, he was a Postdoctoral Researcher with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently a Professor of communication systems with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden, and also a Distinguished Visiting Professor with Tsinghua University, Beijing, China. His research interests include the convergence of communication and sensing, in a 5 G and Beyond 5 G context. Prof. Wymeersch was Associate Editor for IEEE COMMUNICATION LETTERS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and IEEE TRANSACTIONS ON COMMUNICATIONS. He is also a Senior Member of the IEEE Signal Processing Magazine Editorial Board. During 2019–2021, he was an IEEE Distinguished Lecturer with the Vehicular Technology Society.