

Efficient Integrity Monitoring for KF-based Localization

Guillermo Duenas Arana¹, Mathieu Joerger² and Matthew Spenko¹

Abstract—This paper presents a new method to efficiently monitor localization safety in mobile robots. Localization safety is quantified by measuring the system’s integrity risk, which is a well-known aviation performance metric. However, aviation integrity monitoring solutions almost exclusively rely on the Global Navigation Satellite System (GNSS) while robot navigation usually needs the additional information provided by a state evolution model and/or relative positioning sensors, which makes previously established approaches impractical. In response, this paper develops an efficient integrity monitoring methodology applicable to Kalman Filter-based localization. The work is intended for life- or mission-critical operations such as co-robot applications where ignoring the impact of faults can jeopardize human safety.

I. INTRODUCTION

Precise localization is paramount for autonomous navigation applications, especially in life- or mission- critical applications where localization faults may result in human endangerment. Most localization algorithms rely on performance metrics based on the estimate variance, which only consider nominal cases, and as such are insufficient when unmodeled faults occur. In response, this paper applies prior work evaluating integrity for aviation applications to develop a new method to effectively monitor localization safety in mobile robots. Integrity is a quantifiable performance metric used to set certifiable requirements on individual system components to achieve and prove a level of safety for the overall system. More precisely, navigation integrity risk is the probability that a robot’s pose estimation lies outside pre-defined acceptable limits while no alarm is triggered.

Multiple GNSS Receiver Integrity Autonomous Monitoring (RAIM) methodologies have been developed to quantify navigation integrity risk [1]–[4]. Unfortunately most of these approaches rely on a snapshot least squares solution obtained entirely from absolute positioning sensor measurements. In contrast, the Kalman Filter (KF) provides a recursive solution that also employs information about prior estimates and the state evolution model and/or other types of relative positioning sensor information, e.g. inertial measurements or odometry. Thus, the ability to monitor integrity risk using a KF can be of great interest, especially for ground robots in which the additional information added by the state evolution model may be critical to obtaining an accurate pose estimate.

*This work was supported by NSF Grant #1637899

¹G. D. Arana, *student member IEEE*, and M. Spenko, *member IEEE*, are with the Mechanical, Materials and Aerospace Engineering Dept., Illinois Institute of Technology, Chicago, IL, USA gduesar@hawk.iit.edu

²M. Joerger, *member IEEE*, is with the Dept. of Aerospace and Mechanical Engineering, University of Arizona, Tucson, AZ, USA

There has been some previous work in integrity monitoring using a recursive filter. For example, [5] presents a solution separation-based method with a bank of KFs. However, the number of parallel KFs increases as the number of measurement sources increases over time, which typically makes the problem intractable outside GNSS-based localization. [6] and [7] also present a KF integrity monitoring methods, but the former becomes quickly intractable and the latter is inefficient in all but short length missions. Other recursive fault detector methods are presented in [8]–[10], but the impact of undetected faults is not addressed.

In contrast, this paper introduces an efficient methodology to evaluate integrity risk when using a KF solution. First, faults occurring prior to the current time are encapsulated into an estimate bias. Then, the KF update equations are reformulated as a least squares fitting problem by including the previous state estimate as an extra measurement. Given this formulation, localization integrity is evaluated employing traditional aviation RAIM algorithms, which allows for efficient monitoring of both current and previous sensor faults in a common frame. While this method might have significant disadvantages when applied to missions with limited measurement sources, it has practical benefits, especially for systems with low computing power. A more suitable method for high computing power applications is presented in [11].

The paper is organized as follows. Section II presents the background and assumptions. Section III extends the definition of the integrity risk to a format that can be recursively evaluated. Section IV derives an alternative KF-equivalent formulation to evaluate the integrity risk as presented in the prior section. Finally, Section V discuss the results obtained when simulating a robot self-localizing in a landmark map and Section VI presents conclusions and future work.

II. BACKGROUND

This section presents the mathematical framework and states the key assumptions used throughout this paper.

A. State Evolution Model

In this paper, the state evolution model is assumed to be linear and disturbed by zero-mean white Gaussian noise as:

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{u}_k + \mathbf{w}_k \quad \text{where} \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{W}_k) \quad (1)$$

The vector $\mathbf{x}_k \in \mathbb{R}^m$ includes the m estimated states of the system, the matrices Φ_k and Γ_k are assumed known at all time epochs k , \mathbf{u}_k is an input to the system and \mathbf{w}_k represents the system model’s error. The notation $\mathbf{a} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ indicates that the random vector \mathbf{a} is normally distributed with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.

B. Measurement Model

The sensor measurement model is also assumed to be linear and disturbed by zero-mean white Gaussian noise, but in contrast to the state evolution model, it can also be disturbed by sensor faults, i.e.:

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{f}_{z,k} + \mathbf{v}_k \quad \text{where} \quad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{V}_k) \quad (2)$$

The vector $\mathbf{z}_k \in \mathbb{R}^{n_k}$ includes the n_k measurements obtained from absolute positioning sensors at time k , the matrix \mathbf{H}_k is assumed known, and the vector $\mathbf{f}_{z,k}$ contains zeros except for those measurements that are faulted whose components are unknown.

Note that the linear and normality assumptions can be relaxed by making use of linearization and noise over-bounding methods such as the nonlinear extended KF or the CDF Gaussian over-bounding theorem presented in [12]. However, this paper presents the general algorithm and these extensions will not be further discussed.

C. Hazardously Misleading Information

Localization safety is quantified as the integrity risk of the system, which is evaluated as the probability of Hazardously Misleading Information (HMI). HMI occurs when the error on the state (or linear combination of states) of interest exceeds a predefined threshold or *alert limit* and the fault detector does not trigger the alarm, i.e.:

$$HMI_k \triangleq |\hat{\epsilon}_{x,k}| > l \cap q_{D,k} < T_{D,k} \quad (3)$$

where $\hat{\epsilon}_{x,k} = \hat{x}_k - x_k$ is the error in the state of interest (e.g. lateral positioning error in autonomous vehicles navigation), l is the alert limit, $q_{D,k}$ is the fault detector, and $T_{D,k}$ is a threshold such that if $q_{D,k} \geq T_{D,k}$ an alarm is triggered. In this paper, the notations $\bar{\mathbf{x}}_k$, $\hat{\mathbf{x}}_k$ and \mathbf{x}_k refer to the KF prediction and measurement update state estimate, and the actual (unknown) state respectively. The error in the state of interest, $\hat{\epsilon}_{x,k}$, is selected by the vector $\boldsymbol{\alpha}_k \in \mathbb{R}^m$ such that:

$$\hat{\epsilon}_{x,k} = \boldsymbol{\alpha}_k^T \hat{\epsilon}_{x,k} = \boldsymbol{\alpha}_k^T (\hat{\mathbf{x}}_k - \mathbf{x}_k) \quad (4)$$

Note that $\boldsymbol{\alpha}_k$ can be a vector that selects one state of interest. For example, if $m = 3$ and the state of interest is the second component of \mathbf{x} , then $\boldsymbol{\alpha} = [0 \ 1 \ 0]^T$.

In this work, localization integrity risk is evaluated under both faulted and fault-free hypotheses. Fault hypotheses indicate the faulted components in $\mathbf{f}_{k,z}$ that are non-zero and the time k at which this faults occur. For example, a fault hypothesis might indicate that two measurements are faulted at epoch 5 and one measurement is faulted at epoch 7, while the current time is $k = 10$. Given a set of mutually exclusive, jointly exhaustive fault hypotheses, $\{H_0, \dots, H_{n_H}\}$, the probability of HMI or integrity risk at time k can be computed as:

$$P(HMI_k) = \sum_{h=0}^{n_H} P(HMI_k | H_h) P(H_h) \quad (5)$$

The next two sections will upper bound the right hand side of (5) to obtain a conservative measure of navigation safety.

III. FAULT HYPOTHESES PROBABILITIES

In this work, hypotheses with prior sensor faults are distinguished from those without. Denoting as H_0^p the hypothesis with no faults prior to current time k and H_1^p as its complementary event (at least one fault occurring prior to k), the integrity risk can be expressed as:

$$P(HMI_k) = \sum_{h=0}^{n_H} P(HMI_k | H_h, H_0^p) P(H_h | H_0^p) P(H_0^p) + P(HMI_k | H_h, H_1^p) P(H_h | H_1^p) P(H_1^p) \quad (6)$$

Similarly, the probability of a sensor failure at each epoch is considered independent. Thus, for independent sensors, each with probability of failure P_i :

$$P(H_0^p) = \prod_{j=1}^{k-1} \prod_{i=1}^{n_k} (1 - P_i) \quad (7)$$

which can be recursively evaluated. The complementary event is the probability of at least one previous fault occurring, i.e. $P(H_1^p) = 1 - P(H_0^p)$.

Faults at previous epochs (before k) are accounted for by H_0^p and H_1^p and thus, in (6), $\{H_0, \dots, H_{n_H}\}$ form a set of mutually exclusive, jointly exhaustive fault hypotheses for the current measurements (at k). By the independence assumption between epochs, $P(H_h | H_0^p) = P(H_h | H_1^p) = P(H_h)$. Every H_h depicts a scenario in which a different subset of measurements is faulted; hence, every H_h occurs with different probability. For example, hypothesis H_h with faulted measurements i_1, \dots, i_r :

$$P(H_h) = P(H_0) \prod_{s=1}^r \frac{P_{i_s}}{1 - P_{i_s}} \quad (8)$$

where H_0 is the current-time fault-free hypothesis, i.e. no measurement is faulted at the current time; it is computed as $P(H_0) = \prod_{i=1}^{n_k} (1 - P_i)$.

Monitoring all possible hypotheses would be impractical because hypotheses with a large number of faulted measurements are very rare. Thus, we determine the maximum number of simultaneous sensor faults that need to be monitored, n_{max} , such that the probability of more than n_{max} simultaneous sensor faults is less than I_H . Blanch et al. [4] (Appendix C) shows that n_{max} is obtained as the maximum integer r for which the next expression holds:

$$\frac{(\sum_{i=1}^{n_k} P_i)^r}{r!} \leq I_H \quad (9)$$

Thus, the risk of failures due to unmonitored faulted subsets will be accounted for by adding I_H in (6). One important limitation of integrity monitoring methods in general is that the number of measurements impacted by faults cannot be larger than the number of redundant measurements ($n - m$); otherwise the fault might be undetectable. Therefore, the inequality $n_{max} \leq n - m$ must hold or some fault hypotheses with $P(H_h) > I_H$ will not be monitored. We will assume that this condition holds for the rest of the paper.

Finally, summing over the n_H hypotheses with at most n_{max} simultaneous sensor faults and accounting for the unmonitored subsets with I_H , the integrity risk can be upper-bounded as:

$$P(HMI_k) \leq \sum_{h=0}^{n_H} P(H_h) \left[P(HMI_k | H_{h,0}^p) P(H_0^p) + P(HMI_k | H_{h,1}^p) P(H_1^p) \right] + I_H \quad (10)$$

where the simplified notation $H_{h,0}^p \equiv \{H_h, H_0^p\}$ and $H_{h,1}^p \equiv \{H_h, H_1^p\}$ has been employed.

This section showed how to compute the fault hypotheses prior probabilities. The next section presents a KF-equivalent formulation that enables the computation of (10).

IV. KALMAN FILTER-LEAST SQUARES RAIM

In this section, efficient expressions for $P(HMI_k | H_{h,0}^p)$ and $P(HMI_k | H_{h,1}^p)$ in (10) are derived. First, an alternative formulation that includes the state evolution information into the measurement model is presented and the least squares solution to this formulation is proven to be equivalent to the KF update. Second, the residual vector and the fault detector in (3) are defined, and the distribution of both the estimate error and the detector are specified for a given fault hypothesis. Then, the integrity risk is evaluated for the worst-case fault, which is derived in Section IV-E. Finally, a summary of equations is given for ease of implementation.

A. Kalman Filtering as Least Squares

Given the linear state evolution and measurement models presented in (1) and (2), the KF prediction estimate has mean and covariance matrix:

$$\bar{\mathbf{x}}_{k+1} = \Phi_k \bar{\mathbf{x}}_k + \Gamma_k \mathbf{u}_k \quad \text{and} \quad \bar{\mathbf{P}}_{k+1} = \Phi_k \hat{\mathbf{P}}_k \Phi_k^T + \mathbf{W}_k \quad (11)$$

respectively. Previous sensor faults result in a unknown estimate fault, $\bar{\mathbf{f}}_x$, that increases the estimate error, $\bar{\boldsymbol{\epsilon}}_x$:

$$\bar{\boldsymbol{\epsilon}}_x \triangleq \bar{\mathbf{x}} - \mathbf{x} = \bar{\mathbf{f}}_x + \bar{\boldsymbol{\epsilon}} \quad (12)$$

where $\bar{\boldsymbol{\epsilon}} \sim \mathcal{N}(\mathbf{0}, \bar{\mathbf{P}})$ is the error arising from the sensor and system model's Gaussian errors. Note that the time index has been removed to lighten notation.

Hewitson et al. [8], [9] proposes a least squares formulation for the KF update step. First, equations (2) and (12) are combined as:

$$\underbrace{\begin{bmatrix} \mathbf{z} \\ \bar{\mathbf{x}} \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \mathbf{H} \\ \mathbf{I} \end{bmatrix}}_{\mathbf{D}} \mathbf{x} + \underbrace{\begin{bmatrix} \mathbf{f}_z \\ \bar{\mathbf{f}}_x \end{bmatrix}}_{\mathbf{f}_y} + \underbrace{\begin{bmatrix} \mathbf{v} \\ \bar{\boldsymbol{\epsilon}} \end{bmatrix}}_{\boldsymbol{\delta}} \quad (13)$$

where \mathbf{I} denotes the identity matrix. Definitions for each term are underbraced such that:

$$\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{f}_y + \boldsymbol{\delta} \quad (14)$$

where the noise term is zero mean normally distributed as:

$$\boldsymbol{\delta} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Delta}) \quad \text{where} \quad \boldsymbol{\Delta} = \begin{bmatrix} \mathbf{V} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{P}} \end{bmatrix} \quad (15)$$

Assuming $\mathbf{f}_y = \mathbf{0}$ in (14), the least squares solution provides the best unbiased estimate whose mean and variance are:

$$\hat{\mathbf{x}} = \mathbf{S}\mathbf{y} \quad \text{and} \quad \hat{\mathbf{P}} = \mathbf{S}\boldsymbol{\Delta}\mathbf{S}^T \quad (16)$$

respectively, and $\mathbf{S} = (\mathbf{D}^T \boldsymbol{\Delta}^{-1} \mathbf{D})^{-1} \mathbf{D}^T \boldsymbol{\Delta}^{-1}$. The appendix shows that the estimate in (16) is equivalent to the one obtained employing the KF measurement update equations.

B. Residual and Fault Detector

The residual vector measures the discrepancy between the actual, \mathbf{y} , and the estimated measurements using the least squares solution, $\mathbf{D}\hat{\mathbf{x}}$, i.e.:

$$\mathbf{r} = \mathbf{y} - \mathbf{D}\hat{\mathbf{x}} \quad (17)$$

This augmented residual vector is a good indicator of possible system faults because it provides a direct (independent of the current state) measure of the system's error—it can be proven that $\mathbf{r} = (\mathbf{I} - \mathbf{D}\mathbf{S})(\mathbf{f}_y + \boldsymbol{\delta})$ [13]. As in most residual-based RAIM snapshot solutions, the weighted norm of the residual is employed as fault detector:

$$q_D^2 \triangleq \|\mathbf{r}\|_{\boldsymbol{\Delta}^{-1}}^2 = \mathbf{r}^T \boldsymbol{\Delta}^{-1} \mathbf{r} \quad (18)$$

Garcia-Crespillo [14] proves that this augmented residual norm is in fact the weighted norm of the KF measurement update innovation vector, $\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}$, which is computed when employing the KF equations. Thus, the detector can be computed without the need of the updated estimate in (17).

The next section specifies the distributions of the two random variables required to calculate the integrity risk: the fault detector, q_D , and state of interest estimate error, $\hat{\boldsymbol{\epsilon}}_x$.

C. Distribution of fault detector and estimate error

The weighted norm squared of the residual vector is proven to be non-central chi-square distributed with n degrees of freedom and non-centrality parameter λ_D^2 :

$$\|\mathbf{r}\|_{\boldsymbol{\Delta}^{-1}}^2 \sim \chi_{n, \lambda_D^2}^2 \quad \text{where} \quad \lambda_D^2 = \mathbf{f}_y^T \boldsymbol{\Delta}^{-1} (\mathbf{I} - \mathbf{D}\mathbf{S}) \mathbf{f}_y \quad (19)$$

and the estimate error in (4) is normally distributed as:

$$\hat{\boldsymbol{\epsilon}}_x \sim \mathcal{N}(\hat{\mathbf{f}}_x, \hat{\sigma}^2) \quad \text{where} \quad \hat{\mathbf{f}}_x = \boldsymbol{\alpha}^T \hat{\mathbf{f}}_x = \boldsymbol{\alpha}^T \mathbf{S}\mathbf{f}_y \quad (20)$$

and $\hat{\sigma}^2 = \boldsymbol{\alpha}^T \hat{\mathbf{P}} \boldsymbol{\alpha}$. Because of the least squares solution of the KF-equivalent alternative formulation, Joerger et al. [13] showed that the random parts of $\hat{\boldsymbol{\epsilon}}_x$ and $\|\mathbf{r}\|_{\boldsymbol{\Delta}^{-1}}$ are independent. A detailed derivation of (19) and (20) for non-recursive integrity monitoring (snapshot RAIM) can also be found in [13].

D. Evaluation of $P(HMI)$

The independence between the random parts of $\hat{\boldsymbol{\epsilon}}_x$ and $\|\mathbf{r}\|_{\boldsymbol{\Delta}^{-1}}$ is employed to separately evaluate the probability of the two events in (3) as:

$$P(HMI|H_{h,b}^p) = P(|\hat{\boldsymbol{\epsilon}}_x| > l | H_{h,b}^p) P(\|\mathbf{r}\|_{\boldsymbol{\Delta}^{-1}} < T_D | H_{h,b}^p) \quad (21)$$

where b can be either 0 or 1. The detector threshold (T_D) is set to limit the probability of false alarms to a predefined continuity risk allocation, I_C . False alarms occur when the

alarm is triggered ($q_D > T_D$), but no faults have occurred. Then, given the distribution of the detector in (19) with $\lambda_D = 0$, the detector threshold is set as:

$$T_D = \sqrt{X_n^{-2} [1 - I_C]} \quad (22)$$

where $X_n^{-2} [\cdot]$ is the inverse chi-squared CDF with n degrees of freedom.

At this point, the only unknown in (10) is the augmented measurement fault, \mathbf{f}_y , which appears in both (19) and (20). The next section finds the fault that maximizes integrity risk.

E. Worst-case Fault

In this work, faults are modeled as an unknown deterministic quantities. Thus, the augmented fault vector is chosen to maximize the integrity risk in (10) to obtain a conservative bound on the system's safety. This section presents a method to compute such worst-case fault. Unlike previous methods [5], [6] where a subset of past-time measurements could be assumed fault-free, the proposed method imposes that all past-time measurements can be faulted. In addition, this proposed method can be implemented using small-size matrices, it does not require a bank of KFs as in [5], nor does it require batch-type matrix operations as in [6].

Under hypothesis $\{H_h, H_b^p\}$, only certain measurements in \mathbf{y} are faulted; therefore, the matrix $\mathbf{E}_{h,b}$ composed of zeros and ones in specified locations, can be employed to extract only the faulted measurements. For example, if $n = 6$, $m = 2$ and measurements 2 and 5 are faulted in hypothesis H_h :

$$\mathbf{E}_{h,b}\mathbf{y} \quad \text{and} \quad \mathbf{E}_{h,b}\mathbf{f}_y \quad (23)$$

where:

$$\mathbf{E}_{h,0} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad \text{under } H_0^p \quad (24)$$

$$\mathbf{E}_{h,1} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{under } H_1^p \quad (25)$$

extract the corresponding faulted measurements and nonzero fault components respectively. Note that under H_1^p the m terms in \mathbf{y} corresponding to the prediction estimate mean, $\bar{\mathbf{x}}$, are extracted as faulted (see the lower right 2×2 identity matrix in (25)).

Angun [15] showed that the fault direction that maximizes (21) is the one that results in the highest slope defined as the rate between the estimate error mean squared and the fault detector non-centrality parameter, i.e.:

$$g_F^2 = \frac{\hat{f}_x^2}{\lambda_D^2} \quad (26)$$

Therefore, given the estimate error mean in (20), the non-centrality parameter in (19) and the extraction matrix in (23), the worst-case fault direction under hypothesis $\{H_h, H_b^p\}$ is:

$$\check{\mathbf{f}}_{y_h} = \underset{\mathbf{f}_y}{\operatorname{argmax}} \frac{\mathbf{f}_y^T \mathbf{E}_{h,b}^T \mathbf{E}_{h,b} \mathbf{S}^T \boldsymbol{\alpha} \boldsymbol{\alpha}^T \mathbf{S} \mathbf{E}_{h,b}^T \mathbf{E}_{h,b} \mathbf{f}_y}{\mathbf{f}_y^T \mathbf{E}_{h,b}^T \mathbf{E}_{h,b} \boldsymbol{\Delta}^{-1} (\mathbf{I} - \mathbf{DS}) \mathbf{E}_{h,b}^T \mathbf{E}_{h,b} \mathbf{f}_y} \quad (27)$$

which has been proven in [13] to be:

$$\check{\mathbf{f}}_{y_h} = \mathbf{E}_{h,b}^T [\mathbf{E}_{h,b} \boldsymbol{\Delta}^{-1} (\mathbf{I} - \mathbf{DS}) \mathbf{E}_{h,b}^T]^{-1} \mathbf{E}_{h,b} \mathbf{S}^T \boldsymbol{\alpha} \quad (28)$$

Finally, the worst-case fault magnitude is numerically obtained by a linear search in (21) as:

$$|\mathbf{f}_{y_h}| = \underset{q}{\operatorname{argmax}} P \left(|Z_{\hat{f}_x(q), \hat{\sigma}}| > l \right) X_{n, \lambda_D^2(q)}^2 [T_D] \quad (29)$$

where $Z_{\mu, \sigma}$ is a Gaussian random variable with mean μ and standard deviation σ , and $X_{a,b}^2 [\cdot]$ is the CDF of the non-central chi-squared distribution with a degrees of freedom and non-centrality parameter b . The state of interest error mean and the non-centrality parameter are functions of the fault magnitude as:

$$\hat{f}_x(q) = q \boldsymbol{\alpha}^T \mathbf{S} \check{\mathbf{f}}_{y_h} \quad \text{and} \quad \lambda_D^2(q) = q^2 \check{\mathbf{f}}_{y_h}^T \boldsymbol{\Delta}^{-1} (\mathbf{I} - \mathbf{DS}) \check{\mathbf{f}}_{y_h} \quad (30)$$

Finally the worst-case fault is obtained as: $\mathbf{f}_{y_h} = |\mathbf{f}_{y_h}| \check{\mathbf{f}}_{y_h}$.

F. Summary

This section presents an step by step implementation of the proposed KF integrity monitoring. At each epoch:

- 1) Recursively evaluate the probability of previous faults, $P(H_b^p)$ for $b = 1, 2$, using (7).
- 2) Obtain the maximum number of simultaneous sensor faults, n_{max} , using (9).
- 3) Form the n_H hypotheses and compute their probabilities, $P(H_h)$, using (8).
- 4) Calculate the detector threshold, T_D , using (22).
- 5) Obtain the worst-case fault, \mathbf{f}_{y_h} , for each hypothesis using (28) and (29).
- 6) Compute $P(HMI | H_{h,b}^p)$ for $b = 1, 2$ and $h = 1, \dots, n_H$ using (21).
- 7) Substitute $P(HMI | H_{h,b}^p)$, $P(H_h)$ and $P(H_b^p)$ into (10) to obtain an upper bound on the integrity risk.
- 8) Repeat for the next time epoch.

This section has derived the KF residual-based integrity monitoring in detail. The next section implements this methodology to monitor navigation integrity risk in a simple ground robot and discusses the differences between the proposed methodology and a snapshot solution where the state evolution model information is not employed; thus relying only on absolute positioning sensors.

V. SIMULATION RESULTS

This section simulates a robot navigating in a planar environment (see Fig. 1). The robot's state evolution and sensor measurement model are linear and given by (1) and (2) respectively where both process and sensor noise covariance matrices are diagonal with equal variances, i.e.: $\mathbf{W} = \sigma_w^2 \mathbf{I}$ and $\mathbf{V} = \sigma_v^2 \mathbf{I}$ where \mathbf{I} is the identity matrix. The state vector includes the robot's x and y coordinates ($m = 2$), the measurement vector is composed of the relative distance in both x and y directions to six unevenly spaced landmarks ($n = 12$), and the input to the system in (1) is the robot's instant velocity. Note that this simple linear measurement model is chosen to validate the methodology, more

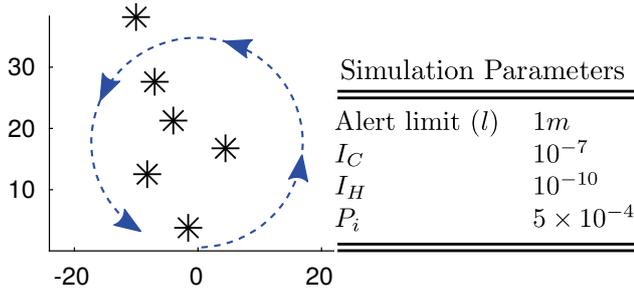


Fig. 1: On the left, the simulation environment. Black marks denote landmark locations and the robot path is in blue. Axis are in meters. On the right, the simulation parameters.

TABLE I: Kalman Filter (KF) and snapshot (SN) root mean squared errors (RMSE) and 3σ envelopes on the state of interest. On the left, the table corresponding to Section V-A and on the right the ones corresponding to Section V-B.

σ_v	KF ($\sigma_w = 0.3$)		SN		KF ($\sigma_v = 0.3$)		
	RMSE	$3\hat{\sigma}$	RMSE	$3\hat{\sigma}$	σ_w	RMSE	$3\hat{\sigma}$
0.10	0.045	0.004	0.045	0.005	0.1	0.112	0.024
0.20	0.087	0.018	0.090	0.020	0.2	0.117	0.034
0.25	0.106	0.028	0.113	0.031	0.3	0.124	0.039
0.30	0.124	0.039	0.136	0.045	0.4	0.128	0.041
0.40	0.156	0.064	0.181	0.080	∞	0.136	0.045

realistic sensor models involve non-linearities which are not considered in this paper. In this simulation, a sensor failure is considered to be a spurious measurement wrongly associated to a landmark and thus, two components—corresponding to the x and y distances to such landmark—in the measurement vector are simultaneously faulted, resulting in the ones in (24) and (25) being substituted by 2×2 identity matrices. No faults are injected in the simulation and associations between measurements and the map of landmarks is assumed known; incorrect association faults are covered in [16].

The proposed methodology is compared to a snapshot integrity monitoring solution [17], [18] in which the estimate at each epoch is obtained by exclusively using the sensor measurements at such epoch; prior estimates are not exploited. This is the traditional approach in aviation, where the measurement vector is not augmented as in (13) and thus, faults occurring at time prior to k do not affect the current estimate. The following subsections analyze the algorithms' response to changes in both sensor and process noise separately. Simulation parameters are given in Fig 1.

A. Effect of Sensor Noise

To quantify the effect of sensor noise, the process noise standard deviation is kept constant ($\sigma_w = 0.3m$) as the sensor noise varies ($\sigma_v = 0.1m-0.4m$). Table I shows the estimate error and 3σ estimate variance on the state of interest. Fig. 2 shows the integrity risk bound for the recursive KF and the snapshot solutions. Note that the KF's integrity risk increases over time while the snapshot's stays constant throughout the simulation. This occurs because the probability of previous faults employed in the proposed KF solution, H_1^p , increases as more measurements are obtained over time; in practice

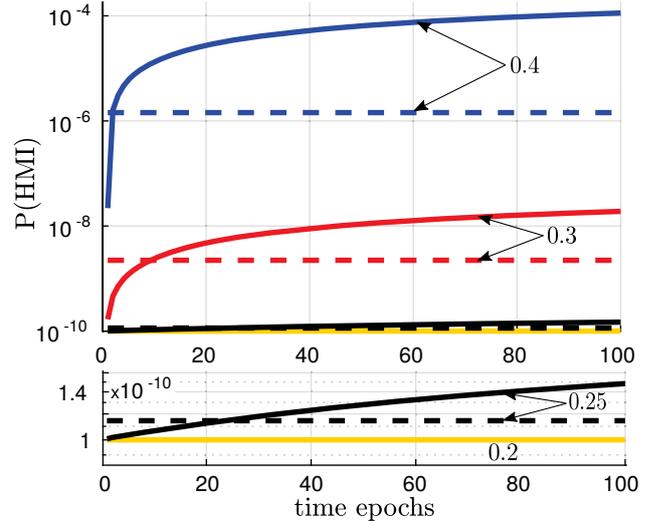


Fig. 2: Integrity risk comparison between the proposed KF (solid) and the snapshot (dashed) solutions with sensor noise standard deviation, σ_v , as indicated in the figure in meters. The bottom figure zooms into the low integrity risk region. Note that for $\sigma_v \leq 0.2$ the risk equals I_H .

only a window of time must be considered to compute $P(H_1^p)$ or it will increase without bound.

Two conclusions can be inferred from the results: 1) both the proposed KF and snapshot solutions variances and integrity risk bounds rapidly increase with the sensor noise as expected and, 2) the KF solution presents a lower error and variance in this unfaulted scenario; however, the integrity risk tends to be higher, especially after a few epochs (this point is further developed in the next section). Similar responses are obtained when decreasing the number of landmarks or increasing the probability of sensor faults.

B. Effect of Process Noise

In this simulation the sensor and process noise are generated with $\sigma_v = 0.3m$ and $\sigma_w = 0.1m$ respectively. However, while the sensor data is filtered using the nominal $\sigma_v = 0.1m$, the state evolution data is filtered with increasing $\sigma_w = 0.1-\infty$. This means that the trust in the state evolution model is deliberately decreased to evaluate the algorithm response to the same data inputs.

Fig. 3 shows that, initially, the integrity risk reduces as the trust in the model is decreased (σ_w increased) and then, as σ_w is further increased, the integrity risk asymptotically approaches the constant value resulting from the KF approach with $\sigma_w = \infty$. This an interesting result because, in a fault-free scenario like the one simulated, the estimate error is minimal when the state evolution model information is filtered with its nominal $\sigma_w = 0.1$ (see Table I). However, the higher the trust on the state evolution, the larger the impact of prior faults on the current estimate; this, after some time, makes the KF's integrity risk surpass the snapshot's. Note that the constant snapshot integrity risk is lower than the KF risk with $\sigma_w = \infty$ even when they both compute the

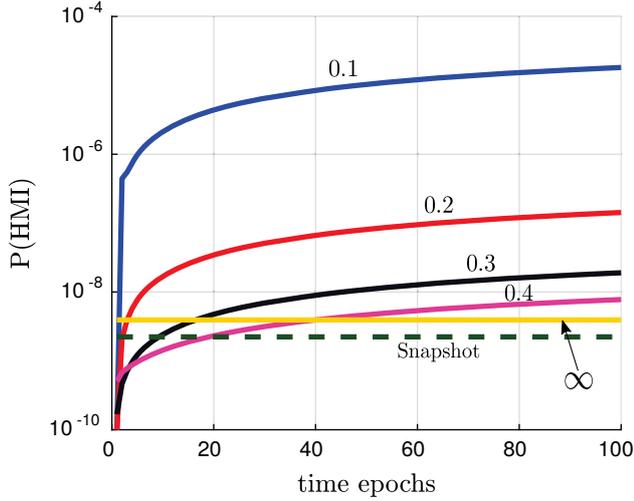


Fig. 3: Integrity risk for the same data employing the σ_w parameter indicated in meters (solid) in the proposed KF method and for the snapshot solution, which is independent of process noise.

same estimate. The reason for this increment in the integrity risk is the greater number of degrees of freedom in the KF detector (n in KF vs. $n - m$ in snapshot), which makes it less effective when prior information is not employed.

C. Discussion

The KF offers a smaller variance estimate than the snapshot's by recursively including the previous state information. Unfortunately, including prior state information also entails accounting for previously occurring faults. The simulations showed that the impact of those previous faults greatly affect the integrity risk when the process noise is small relative to the sensor noise. This occurs because of the conservative approach taken in this work—prior faults result in the worst-case fault in the estimate. One possible solution is to increase the process noise variance used in the KF until a satisfactory trade-off between safety (integrity risk) and accuracy (variance) is reached. In addition, if a window of time is employed to calculate $P(H_1^p)$, an optimal process noise value can be found such that the proposed method continuously provides lower integrity risk and variance than the snapshot solution.

There are at least two possible applications for the proposed methodology: 1) find the optimal process noise value for which the KF's integrity risk falls below the snapshot's, or 2) if the KF integrity risk is lower than a predefined safety requirement (e.g. 10^{-7} in aviation), use the nominal process noise to obtain the optimal filter.

VI. CONCLUSIONS AND FUTURE WORK

This paper presents a new Kalman Filter integrity monitoring method that effectively evaluates localization safety for a mobile robot with minimal computation requirements. The methodology is derived and validated in simulation. Simulation results show that the method offers a trade-off

between safety (integrity risk) and accuracy (variance) that can be exploited by varying the filter parameters.

Quantifying safety will be critical as more robots operate near and among humans. Future work will reduce the impact of previous faults on integrity risk by evaluating fault hypotheses in a window of time; thus, introducing a trade-off between computational complexity and performance.

APPENDIX

EQUIVALENCE BETWEEN LEAST SQUARES AND KALMAN FILTER UPDATE EQUATIONS

This appendix proves that the least squares solution to the alternative formulation presented in (16) is equivalent to the KF update step. The KF update equations are:

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{L}(\mathbf{z} - \mathbf{H}\bar{\mathbf{x}}) \quad \text{and} \quad \hat{\mathbf{P}} = \left(\bar{\mathbf{P}}^{-1} + \mathbf{H}^T \mathbf{V}^{-1} \mathbf{H} \right)^{-1} \quad (31)$$

where the Kalman gain is:

$$\mathbf{L} = \hat{\mathbf{P}} \mathbf{H}^T \mathbf{V}^{-1} \quad (32)$$

To prove the equivalence of the covariance matrix expressions, we expand (16) using the definition of \mathbf{S} and simplify:

$$\hat{\mathbf{P}} = \mathbf{S} \mathbf{\Delta} \mathbf{S}^T = \left(\mathbf{D}^T \mathbf{\Delta}^{-1} \mathbf{D} \right)^{-1} \quad (33)$$

Using \mathbf{D} and $\mathbf{\Delta}$ from (13), the matrix inside the inverse is:

$$\mathbf{D}^T \mathbf{\Delta}^{-1} \mathbf{D} = \begin{bmatrix} \mathbf{H}^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{-1} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{P}}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{H} \\ \mathbf{I} \end{bmatrix} = \bar{\mathbf{P}}^{-1} + \mathbf{H}^T \mathbf{V}^{-1} \mathbf{H} \quad (34)$$

Substituting (34) into (33) proves the covariance update (31).

The mean update equivalence is similarly proven by expanding (16) and substituting (33):

$$\hat{\mathbf{x}} = \mathbf{S} \mathbf{y} = \hat{\mathbf{P}} \mathbf{D}^T \mathbf{\Delta}^{-1} \mathbf{y} \quad (35)$$

Substituting the definitions of each term again:

$$\hat{\mathbf{x}} = \hat{\mathbf{P}} \begin{bmatrix} \mathbf{H}^T & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{-1} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{P}}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \bar{\mathbf{x}} \end{bmatrix} = \hat{\mathbf{P}} \mathbf{H}^T \mathbf{V}^{-1} \mathbf{z} + \hat{\mathbf{P}} \bar{\mathbf{P}}^{-1} \bar{\mathbf{x}} \quad (36)$$

The first term in the right hand side of this expression contains the Kalman gain definition in (32) and substituting $\bar{\mathbf{P}}^{-1}$ from the proven KF covariance update in (31):

$$\hat{\mathbf{x}} = \mathbf{L} \mathbf{z} + \hat{\mathbf{P}} \left(\bar{\mathbf{P}}^{-1} - \mathbf{H}^T \mathbf{V}^{-1} \mathbf{H} \right) \bar{\mathbf{x}} = \mathbf{L} \mathbf{z} + \bar{\mathbf{x}} - \hat{\mathbf{P}} \mathbf{H}^T \mathbf{V}^{-1} \mathbf{H} \bar{\mathbf{x}} \quad (37)$$

Recognizing the Kalman gain definition from the third term and grouping terms we obtain the KF update equation (31).

REFERENCES

- [1] Y. C. Lee, "Analysis of Range and Position Comparison Methods as a Means to Provide GPS Integrity in the User Receiver," in *Proceedings of the 42nd Annual Meeting of The Institute of Navigation*, 1986.
- [2] R. G. Brown, "A baseline gps raim scheme and a note on the equivalence of three raim methods," *Navigation*, vol. 39, no. 3, 1992.
- [3] —, *GPS RAIM: Calculation of Thresholds and Protection Radius Using Chi-square Methods; a Geometric Approach*. Radio Technical Commission for Aeronautics, 1994.

- [4] J. Blanch, T. Walter, P. Enge, Y. Lee, B. Pervan, M. Rippl, and A. Spletter, "Advanced raim user algorithm description: Integrity support message processing, fault detection, exclusion, and protection level calculation," in *ION GNSS*, 2012, pp. 2828–2849.
- [5] M. Brenner, "Integrated gps/inertial fault detection availability," *Navigation*, vol. 43, no. 2, pp. 111–130, 1996.
- [6] M. Joerger and B. Pervan, "Integrity risk of kalman filter-based raim," in *ION GNSS*, 2011, pp. 3856–3867.
- [7] M. Joerger, G. D. Arana, M. Spenko, and B. Pervan, "A new approach to unwanted-object detection in gnss/lidar-based navigation," *Sensors*, vol. 18, no. 8, p. 2740, 2018.
- [8] S. Hewitson and J. Wang, "Gnss receiver autonomous integrity monitoring with a dynamic model," *Navigation*, vol. 60, no. 2, 2007.
- [9] —, "Extended receiver autonomous integrity monitoring (e raim) for gnss/ins integration," *Journal of Surveying Engineering*, vol. 136, no. 1, pp. 13–22, 2010.
- [10] H.-Q. Mu and K.-V. Yuen, "Novel outlier-resistant extended kalman filter for robust online structural identification," *Journal of Engineering Mechanics*, vol. 141, no. 1, 2014.
- [11] G. D. Arana, O. A. Hafez, M. Joerger, and M. Spenko, "Recursive integrity monitoring for mobile robot localization safety," *ICRA*, 2019.
- [12] J. Rife, S. Pullen, P. Enge, and B. Pervan, "Paired Overbounding for Nonideal LAAS and WAAS Error Distributions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 4, 2006.
- [13] M. Joerger, F.-C. Chan, and B. Pervan, "Solution Separation Versus Residual-Based RAIM," *Navigation*, vol. 61, no. 4, 2014.
- [14] O. Garcia-Crespillo, "Personal Conversation," 2017.
- [15] J. Angus, "Raim with multiple faults," *Navigation*, vol. 53, no. 4, pp. 249–257, 2006.
- [16] G. D. Arana, M. Joerger, and M. Spenko, "Local nearest neighbor integrity risk evaluation for robot navigation," *ICRA*, 2018.
- [17] B. W. Parkinson and P. Axelrad, "Autonomous GPS Integrity Monitoring Using the Pseudorange Residual," *Navigation*, vol. 35, no. 2, pp. 225–274, 1988.
- [18] T. Walter and P. Enge, "Weighted raim for precision approach," in *PROCEEDINGS OF ION GPS*, vol. 8. Institute of Navigation, 1995, pp. 1995–2004.