2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

Recursive Integrity Monitoring for Mobile Robot Localization Safety

Guillermo Duenas Arana[1], Osama Abdul Hafez[1], Mathieu Joerger[2], and Matthew Spenko[1]

Abstract— **This paper presents a new methodology to quantify robot localization safety by evaluating integrity risk, a performance metric widely used in open-sky aviation applications that has been recently extended to mobile ground robots. Here, a robot is localized by feeding relative measurements to mapped landmarks into an Extended Kalman Filter while a sequence of innovations is evaluated for fault detection. The main contribution is the derivation of a sequential chi-squared integrity monitoring methodology that maintains constant computation requirements by employing a preceding time window and, at the same time, is robust against faults occurring prior to the window. Additionally, no assumptions are made on either the nature or shape of the faults because safety is evaluated under the worst possible combination of sensor faults.**

I. Introduction

Robotic localization algorithms traditionally quantify pose estimation performance with a covariance matrix or particle spread [1], [2], [3]. However, these only consider non-faulted cases, making them insufficient for life-critical applications, such as self-driving cars, where ignoring the probability that an *undetected* fault (e.g. Global Navigation Satellite Systems (GNSS) clock errors, misassociations among extracted features and database landmarks) occurs can lead to a localization error with catastrophic consequences.

To account for the probability of an undetected fault occurring, prior work extended the concept of localization integrity, the probability that a robot's pose estimation lies within pre-defined acceptable limits, from aviation applications [4], [5] to robotics [6], [7], [8]. This paper builds upon that work by introducing a new method to monitor localization integrity risk for mobile robots that localize using feature extraction and data association algorithms, which may become faulted when incorrectly extracted unmapped objects are associated to mapped landmarks.

A. Related Work

This paper leverages prior work evaluating localization integrity for GNSS-based aviation applications, which are instrumental in ensuring the safety of pilots, crew, and passengers [9], [10], [11], [12], [13], [14]. Unfortunately, these methods do not directly apply to mobile robots because they often operate in GNSS-denied environments. The additional sensors required to localize ground robots, such as lidar, introduce new integrity monitoring challenges.

Two variants of Receiver Autonomous Integrity Monitoring (RAIM) have been the primary techniques to quantify localization safety in aviation applications: solution separation and chi-squared RAIM [13]. The former can be employed for integrity monitoring in sequential implementations [15]. However, it requires banks of Extended Kalman Filters (EKFs) for each possible fault hypothesis, which become impractical when applied to even sparse landmark maps. The latter was developed for snapshot estimators [10], [16] in which the pose estimate is the least squares solution computed from the new measurements obtained at each epoch and previous knowledge of the state is ignored. Several methods extend chi-squared RAIM to sequential implementations [17], [18], but they either lack a recursive computation of the worst-case fault or make assumptions about the nature of faults that are not applicable to landmark-based localization.

[6], [19] quantified the risk of data association faults when the feature extractor correctly extracts all mapped landmarks, and [8] expanded this technique to cases where only a subset of the mapped landmarks is detected. [20] quantifies the risk reduction in data association gained by incorporating an inertial measurement unit. These methodologies are expanded in [7] and [21] to account for the possibility of a single landmark being continuously incorrectly associated to an unmapped object, assuming that the same landmarks are extracted at all times.

In contrast, the method presented here can monitor multiple faults in multiple landmarks at different times. Thus, to the best of our knowledge, this paper presents the first fully recursive integrity monitoring methodology for mobile robot localization without the need of unrealistic assumptions in either the nature or the shape of faults.

B. Overview

In this work, we build upon the chi-squared integrity monitoring methodology presented in [22]. An EKF localizes a robot moving in a previously mapped environment, where a landmark map is assumed, and a sequence of EKF innovations within a preceding window of time is employed for fault detection. Although other popular methods, such as particle filters or graph optimization algorithms [23], may show better consistency under certain scenarios [24], an EKF-based localization approach was chosen because it is widely known and is the most popular localization technique in aviation applications from which this work is inspired.

This work evaluates localization integrity risk when undetected measurement faults occur. The estimate error and fault detector are analyzed, and their distributions are derived as a

*This work is supported by NSF Grant #1637899

[1]G. D. Arana, *student member IEEE*, O. A. Hafez, and M. Spenko, *member IEEE*, are with the Mechanical, Materials, and Aerospace Engineering Department, Illinois Institute of Technology, Chicago, IL, USA gdueasar@hawk.iit.edu

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

978-1-5386-6027-0/19/$31.00 ©2019 IEEE

function of the faults within a preceding horizon (including the current time) and a prior estimate bias that accounts for faults occurring previous to the preceding horizon. Faults are modeled as unknown deterministic shifts in the measurements' means; the reason is that faults are rarely occurring events and therefore, cannot be modeled statistically. Integrity monitoring can be then formulated as an optimization problem: finding the worst-case fault that maximizes the estimation error while going undetected. The worst-case fault is computed by efficiently solving this complex optimization problem.

The paper is organized as follows. Section II introduces the necessary background, presents the measurement model with deterministic faults, and mathematically defines integrity risk. Section III derives the probability of each landmark's failure, which is later employed to compute the probability of the fault hypotheses. Estimate error and fault detector probabilities are derived in Section IV as a function of the worst-case fault determined in Section V. Section VI shows both simulated and experimental results. Finally, Section VII presents conclusions and future work.

II. Background

This section presents the background necessary to evaluate a robot's integrity including the state evolution and measurement models, the notation for the Extended Kalman Filter equations, and the definition of integrity risk.

A. State Evolution and Measurement Models

Given m states, the state vector at time k is denoted as $\mathbf{x}_k \in \mathbb{R}^m$. The robot's state evolution is:

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

is white Gaussian noise with known covariance matrix $\mathbf{W}_k$, $\mathbf{u}_k$ are the system inputs (e.g. odometry/IMU readings), and $\mathbf{g}(\cdot,\cdot)$ is a known function.

The sensor measurements corresponding to each extracted feature, $\mathbf{z}_{k,i} \in \mathbb{R}^{m_F}$, are stacked into the measurement vector, $\mathbf{z}_k \in \mathbb{R}^{n_k}$, which includes a total of $n_k = n_k^F m_F$ measurements corresponding to the n_k^F extracted features at time k, i.e.: $\mathbf{z}_k = \begin{bmatrix}\mathbf{z}_{k,1}^T & \cdots & \mathbf{z}_{k,n_k^F}^T\end{bmatrix}^T$. The measurements are thus modeled as:

$$\mathbf{z}_k = \mathbf{h}\left(\mathbf{x}_k\right) + \mathbf{v}_k + \mathbf{f}_k \quad \text{where} \quad \mathbf{v}_k \sim \mathcal{N}\left(\mathbf{0}, \mathbf{V}_k\right) \tag{2}$$

is white Gaussian noise with known covariance matrix $\mathbf{V}_k$ and $\mathbf{h}(\cdot)$ is a known function.

Faults, $\mathbf{f}_k$, are modeled as unknown deterministic terms whose elements are only nonzero when measurements are faulted. Thus, in the nominal (non-faulted) case, the fault vector is null and the measurement error only includes Gaussian noise, $\mathbf{v}_k$. Faults, which are rare, are assumed to occur when wrongly extracted features are associated with mapped landmarks. Examples include dynamic objects in front of landmarks or new objects since the map was made.

B. The Extended Kalman Filter (EKF)

The EKF prediction (3) & (4) and update (5) & (6) equations are given as reference for later derivations.

$$\bar{\mathbf{x}}_k = \mathbf{g}\left(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}\right) \tag{3}$$

$$\bar{\mathbf{P}}_k = \mathbf{\Phi}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{\Phi}_{k-1}^T + \mathbf{W}_{k-1} \tag{4}$$

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{L}_k\boldsymbol{\gamma}_k \tag{5}$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{L}_k\mathbf{H}_k)\bar{\mathbf{P}}_k \tag{6}$$

where $\mathbf{L}_k = \bar{\mathbf{P}}_k\mathbf{H}_k^T\mathbf{Y}_k^{-1}$ is the Kalman gain, $\boldsymbol{\gamma}_k$ is the innovation:

$$\boldsymbol{\gamma}_k = \mathbf{z}_k - \mathbf{h}\left(\bar{\mathbf{x}}_k\right), \tag{7}$$

$\mathbf{Y}_k = \mathbf{H}_k\bar{\mathbf{P}}_k\mathbf{H}_k^T + \mathbf{V}_k$ is its covariance matrix, and the state evolution and measurement model function Jacobians are $\mathbf{\Phi}_k \triangleq \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1}}$ and $\mathbf{H}_k \triangleq \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\bar{\mathbf{x}}_k}$, respectively.

Faults affect the estimate mean resulting in a biased Gaussian estimate with unknown (nonzero) mean:

$$\delta\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k - \mathbf{x}_k \sim \mathcal{N}\left(\hat{\mathbf{f}}_{x_k}, \hat{\mathbf{P}}_k\right) \tag{8}$$

where $\delta\hat{\mathbf{x}}_k$ is the updated estimate error, $\mathbf{x}_k$ is the actual unknown state, and $\hat{\mathbf{f}}_{x_k}$ is the unknown estimate bias introduced by the faults. Thus, the estimate variance, which the fault does not directly affect, is an insufficient safety metric when faults occur [8], [6].

C. Hazardous Misleading Information

Integrity risk is evaluated as the probability of Hazardous 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.:

$$HMI_k = |\delta\hat{x}_k| > \ell \,\cap\, q_{D_k} < T_{D_k} \tag{9}$$

where $\delta\hat{x}_k = \hat{x}_k - x_k$ is the error in the state of interest (e.g. lateral positioning error in autonomous vehicles applications), ℓ is the alert limit, q_{D_k} is the fault detector, and T_{D_k} is a threshold such that when $q_{D_k} \geq T_{D_k}$ an alarm is triggered. Vector $\mathbf{t}_k \in \mathbb{R}^m$ extracts the state of interest as:

$$\delta\hat{x}_k = \mathbf{t}_k^T\delta\hat{\mathbf{x}}_k \sim \mathcal{N}\left(\hat{f}_{x_k}, \sigma_k^2\right) \tag{10}$$

where $\hat{f}_{x_k} = \mathbf{t}_k^T\hat{\mathbf{f}}_{x_k}$ is the estimate bias in the state of interest, $\hat{\sigma}_k^2 = \mathbf{t}_k^T\hat{\mathbf{P}}_k\mathbf{t}_k$, and $\mathbf{t}_k$ extracts a linear combination of states. For example, if $m = 3$ and the state of interest is the second component of $\mathbf{x}_k$, then $\mathbf{t}_k = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T$.

The probability of HMI, $P\left(HMI\right)$, is evaluated under different fault hypotheses (H_h). Since both the estimate error and detector at time k are affected by previous and current faults, these hypotheses include faults occurring at epochs up to and including epoch k. For example, a hypothesis at $k = 5$ might indicate that $\mathbf{z}_{3,2}$ (epoch 3, feature 2) was faulted. Then, given a set of mutually exclusive, jointly exhaustive fault hypotheses, $\{H_0, \ldots, H_{n_H}\}$, the $P\left(HMI\right)$, or integrity risk, at epoch k is computed as:

$$P(HMI_k) = \sum_{h=0}^{n_H} P\left(HMI_k \mid H_h\right) P(H_h) \tag{11}$$

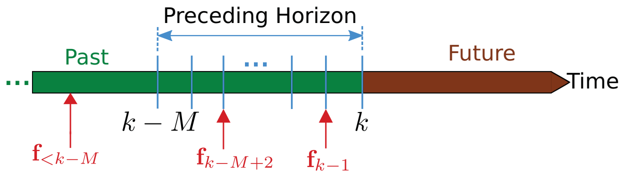

Fig. 1. Preceding horizon. An example depicts two faults occurring within the preceding horizon and one fault at some previous time $< k-M$.

The remainder of the paper will derive an upper bound on the right hand side of (11) to obtain a conservative measure of localization safety. The next section determines the hypotheses probabilities, $P(H_h)$.

III. Hypotheses Probabilities

This section computes the probability of occurrence of every fault hypothesis, $P(H_h)$. To do this, we need the individual probability of failure of every landmark association, P_t, which is the probability that the measurements associated to a specific landmark are faulted—this may occur when a feature that has not been extracted from such landmark is associated to it. We assume that P_t is obtained from limited experimentation and that it is usually low ($\approx 10^{-3}$).

Given the fault probability of all landmarks associated until time k, the probability of hypothesis H_h with the set of faulted landmark associations $t_1, \dots, t_r$ at epochs up to and including k is [12]:

$$P(H_h) = P(H_0)\prod_{s=1}^{r}\frac{P_{t_s}}{1-P_{t_s}} \quad \text{where } P(H_0) = \prod_{t=1}^{n^L_{1:k}}(1-P_t) \tag{12}$$

is the fault-free hypothesis probability and $n^L_{1:k} = \sum_{j=1}^{k} n^L_j$. Note, it is assumed that each landmark fault occurs independently, i.e. one faulted landmark association does not affect the probability of others being faulted.

A. Preceding Horizon

Monitoring all faults occurring at any time $\leq k$ is generally intractable. Thus, a *preceding horizon* of M epochs reduces complexity such that only faults occurring from $k-M$ to k (see Fig. 1), both included, are monitored; the effect of faults prior to $k-M$ are addressed in Section IV. Then, hypotheses $1, \dots, n_H$ only contain faults occurring within the preceding horizon and the second part of (12) becomes:

$$P(H_0) = \prod_{t=1}^{n_k^{L(M)}} (1-P_t) \quad \text{where} \quad n_k^{L(M)} = \sum_{j=k-M}^{k} n_j^L \tag{13}$$

is the number of associated landmarks within the preceding horizon. The first part of (12) remains the same, but t_s only indexes landmark associations within the preceding horizon. The preceding horizon size is user-defined; larger horizons will obtain a better (lower) upper bound on the system's integrity risk, but are computationally more expensive.

B. Hypotheses Reduction

Even for small M, the number of hypotheses can be computationally intractable. Thus, the number of hypotheses, n_H, can be reduced by only monitoring those hypotheses with a probability higher than a predefined threshold, I_H. The maximum number of simultaneous faults that need to be monitored, n_{max}, such that the probability of more than n_{max} simultaneous landmark faults is less than I_H is obtained from Appendix C in [12] as the maximum integer r for which the next expression holds: $\left(\sum_{t=1}^{n_k^{L(M)}} P_t\right)^r / r! \leq I_H$. Finally, given a set of mutually exclusive, jointly exhaustive hypotheses, $\{H_0, \dots, H_{n_H}\}$, each of which consists of at most n_{max} simultaneous feature faults, and their probabilities from (12) and (13), the integrity risk is upper bounded as:

$$P(HMI_k) \leq \sum_{h=0}^{n_H} P\left(HMI_k \mid H_h\right) P(H_h) + I_H \tag{14}$$

where I_H is added to account for the risk of unmonitored failure modes that are not included in the summation.

This section derived the upper bounds on the hypotheses probabilities, $P(H_h)$. The next section determines the distributions of the estimate error and the fault detector in the presence of faults needed to compute $P(HMI_k \mid H_h)$.

IV. Estimate Error and Detector Distributions in the Presence of Faults

This section analyzes the estimate error and the fault detector in the presence of faults. The terms in (14) are calculated under faulted, $H_{h\neq 0}$, and fault-free, H_0, conditions as:

$$P(HMI_k \mid H_h) = P\left(|\delta\hat{x}_k| > l, \; q_{D_k} < T_{D_k} \mid H_h\right) \tag{15}$$

Recall that faults occurring prior to the preceding horizon are not explicitly monitored and thus faults are separated between those occurring prior to the preceding horizon and those within the horizon. This section determines the statistical distributions' parameters of $\delta\hat{x}_k$ and q_{D_k} as an explicit function of the faults occurring within the preceding horizon and as a function of a prior estimate bias, which encompasses the faults occurring prior to the preceding horizon.

A. Estimate Error Distribution

The estimate error is defined in (8) as the difference between the estimated and actual state. In the presence of faults, the estimate error is Gaussian with unknown mean $\hat{\mathbf{f}}_{x_k}$, which is the estimate bias induced by the faults. Appendix I shows that the estimate bias at k can be linearized as a function of the faults occurring within the preceding horizon and the estimate bias at time $k-M-1$ as:

$$\hat{\mathbf{f}}_{x_k} = \mathbf{A}_k^{(M)} \mathbf{f}_k^{(M)} \tag{16}$$

where $\mathbf{A}_k^{(M)}$ is defined in (35) and:

$$\mathbf{f}_k^{(M)} = \left[\mathbf{f}_k^T \;\; \mathbf{f}_{k-1}^T \;\; \dots \;\; \mathbf{f}_{k-M}^T \;\; \hat{\mathbf{f}}_{x_{k-M-1}}^T\right]^T \tag{17}$$

Note that $\mathbf{f}_k^{(M)}$ includes all faults occurring within the preceding horizon plus the estimate bias prior to it. A recursive computation of $\mathbf{A}_k^{(M)}$ is presented in Appendix I. Finally, the state of interest's estimate error's distribution in (15) is:

$$\delta\hat{x}_k \sim \mathcal{N}\left(\hat{f}_{x_k}, \hat{\sigma}_k^2\right) \quad \text{where} \quad \hat{f}_{x_k} = \quad \mathbf{t}_k^T \mathbf{A}_k^{(M)} \mathbf{f}_k^{(M)} \tag{18}$$

and $\hat{\sigma}_k^2 = \mathbf{t}_k^T \hat{\mathbf{P}}_k \mathbf{t}_k$.

B. Fault Detector

The fault detector, which peaks when a fault occurs, is the innovation norm sequence within the preceding horizon:

$$q_{D_k} = \sum_{j=k-M}^{k} \|\boldsymbol{\gamma}_j\|^2_{\mathbf{Y}_j^{-1}} \tag{19}$$

where $\|\mathbf{a}\|^2_{\mathbf{A}} = \mathbf{a}^T\mathbf{A}\mathbf{a}$. Innovations measure the difference between the actual sensor measurements and the EKF-predicted measurements [25], [26].

Innovations defined in (7) at different epochs are independent [22]. Additionally, each innovation norm in the summation of (19) is non-central chi-squared distributed with n_k degrees of freedom (DOF). Thus, the detector, q_{D_k}, follows a non-central chi-squared distribution with $n_k^{(M)}$ DOF and non-centrality parameter $\lambda_k^{(M)}$, i.e.:

$$q_{D_k} \sim \chi^2_{n_k^{(M)}, \lambda_k^{(M)}} \tag{20}$$

where $n_k^{(M)} = \sum_{j=k-M}^{k} n_j$ and $n_k = m_F n_k^F$ [22]. Appendix II shows that the non-centrality parameter is a function of the faults occurring within the preceding horizon and the estimate bias prior to the horizon is:

$$\lambda_k^{(M)} = \mathbf{f}_k^{(M)\,T} \mathbf{M}_k^{(M)} \mathbf{f}_k^{(M)} \tag{21}$$

where $\mathbf{M}_k^{(M)}$ is defined in (44) and can be efficiently computed using Algorithms 2 & 3 in Appendix II.

C. Evaluation of $P(HMI_k)$

One advantage of selecting an innovation-based detector is that the random elements of the estimate error and fault detector are independent [13]. Thus, eq. (15) becomes:

$$P(HMI_k \mid H_h) = P\left(|\delta\hat{x}_k| > l \mid H_h\right) P\left(q_{D_k} < T_{D_k} \mid H_h\right) \tag{22}$$

The next section obtains a conservative bound on integrity risk by finding the fault that maximizes integrity risk or *worst-case fault* which will be applied to both estimate error and detector distributions.

V. Worst-case Fault

A bound on the integrity risk is guaranteed by calculating $P\left(HMI\right)$ under the worst-case fault, $\mathbf{f}^{(M)}_{k,h_{worst}}$. The worst-case fault is applied to (18) and (21) to determine the estimate error and detector distributions needed to compute (22). The time index k is removed in this section to lighten notation.

The worst-case fault vector elements are the sensor faults within the preceding horizon and the estimate bias at $k - M - 1$, computed by optimizing (22), i.e.:

$$\mathbf{f}^{(M)}_{h_{worst}} = \underset{\mathbf{f}^{(M)}}{\operatorname{argmax}} \; P\left(|\delta\hat{x}| > l \mid H_h\right) P\left(q_D < T_D \mid H_h\right) \tag{23}$$

The terms on the right hand side depend on the faults through the distributions of $\delta\hat{x}_k$ in (18), and q_{D_k} in (20) & (21).

Only a subset of the landmark associations is faulted under each hypothesis, H_h. Thus, we extract the faulted elements using an extraction matrix as $\mathbf{E}_h\mathbf{f}^{(M)}$ where $\mathbf{E}_h$'s elements are zeros and ones. For example, given three associations in the preceding horizon where the 1st and 3rd are faulted in H_h yields:

$$\mathbf{E}_h = \begin{bmatrix} \mathbf{I}_{m_F} & \mathbf{0}_{m_F} & \mathbf{0}_{m_F} & \mathbf{0}_{m_F\times m} \\ \mathbf{0}_{m_F} & \mathbf{0}_{m_F} & \mathbf{I}_{m_F} & \mathbf{0}_{m_F\times m} \\ \mathbf{0}_{m\times m_F} & \mathbf{0}_{m\times m_F} & \mathbf{0}_{m\times m_F} & \mathbf{I}_m \end{bmatrix} \tag{24}$$

Note that the estimate bias at $k - M - 1$ is extracted as faulted under every hypothesis (lower right identity matrix).

Next, we determine the worst-case fault direction, $\breve{\mathbf{f}}^{(M)}_{h_{worst}}$, and magnitude, $\left\|\mathbf{f}^{(M)}_{h_{worst}}\right\|$, such that:

$$\mathbf{f}^{(M)}_{h_{worst}} = \breve{\mathbf{f}}^{(M)}_{h_{worst}} \left\|\mathbf{f}^{(M)}_{h_{worst}}\right\| \tag{25}$$

A. Worst-Case Fault Direction & Magnitude

[27] showed that the fault direction that maximizes (23) also maximizes the failure mode slope, the ratio between the estimated error mean squared and the fault detector non-centrality parameter, $\hat{f}^2_{x_k}/\lambda_k^{(M)}$. [13] proved that the fault vector that maximizes the fault slope under H_h is:

$$\breve{\mathbf{f}}^{(M)}_{h_{worst}} = \mathbf{E}_h^T \left[\mathbf{E}_h \mathbf{M}^{(M)} \mathbf{E}_h^T\right]^{-1} \mathbf{E}_h \; \mathbf{A}^{(M)\,T} \mathbf{t} \tag{26}$$

which defines the worst-case fault direction.

Given the worst-case fault direction, its magnitude is numerically determined from (22) as:

$$\left\|\mathbf{f}_{h^{(M)}_{worst}}\right\| = \underset{y}{\operatorname{argmax}} \, P\left(|Z_{y\breve{f}_{x_h}, \hat{\sigma}}| > l\right) \left(X^2_{n^{(M)}, y^2 \breve{\lambda}_h^{(M)}} \left[T_D\right]\right) \tag{27}$$

where $Z_{a,b}$ is a Gaussian random variable with mean a and standard deviation b, $X^2_{a,b}[\cdot]$ is the CDF of a chi-square with a degrees of freedom and non-centrality parameter b, and $\breve{f}_{x_h} = \boldsymbol{\alpha}^T \mathbf{A}^{(M)} \breve{\mathbf{f}}^{(M)}_{h_{worst}}$ and $\breve{\lambda}_h^{(M)} = \breve{\mathbf{f}}^{(M)\,T}_{h_{worst}} \mathbf{M}^{(M)} \breve{\mathbf{f}}^{(M)}_{h_{worst}}$ are the estimate error mean and the non-centrality parameter of the fault detector distribution given the worst-case fault direction. Substituting (26) and (27) into (25), we obtain the worst-case fault, which is used in (22) to calculate an upper bound on $P(HMI_k \mid H_h)$ and thus, to fully determine (14).

VI. Results

In this section, localization safety is evaluated in both simulated (Fig. 2) and experimentally mapped (Fig. 3) environments. In the simulation, a robot moving from left to right is localized in a landmark map with two defined sections: one with landmarks laterally spaced $30\,\mathrm{m}$ apart and another with the spacing reduced to $8\,\mathrm{m}$. Fig. 2, bottom, shows the lateral error integrity risk for three preceding horizon sizes. The smallest preceding horizon has the largest integrity risk, since larger horizons reduce the impact of the conservative assumption that unmonitored previous faults result in the worst possible estimate bias at $k - M - 1$.

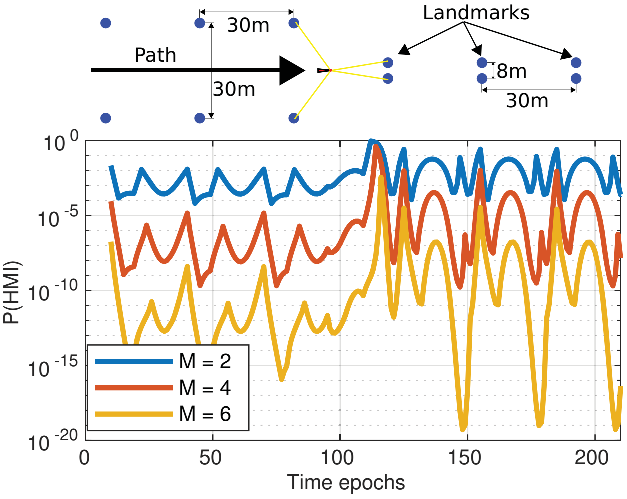

Fig. 2. Simulation environment (above) and integrity risk results for different preceding horizon sizes (below). The standard deviation on lidar range and bearing are $0.3m$ and $2°$. The lidar range is $25\,\mathrm{m}$, the sampling interval is $0.1\,\mathrm{s}$, and the alert limit is $1\,\mathrm{m}$.

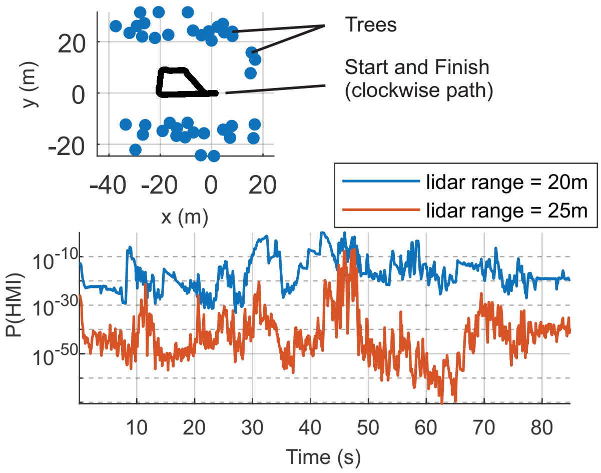

Fig. 3. Landmark map, estimated path (bottom) and integrity risk with $M = 2$ when the lidar range is limited to 20 or $25\,\mathrm{m}$ (top).

Moreover, note that the integrity risk peaks when the robot transitions between the two sections, (epochs ≈100–120), as the relative landmark geometry is not good enough to ensure the robot lateral position with high confidence in that region.

The experimental map is generated using an EKF-SLAM algorithm that fuses IMU, RTK DGPS, and lidar data from a mobile platform. Tree trunks are used as landmarks (Fig. 3). To calculate integrity risk, DGPS is removed and the lidar range is artificially reduced. The integrity risk for a preceding horizon of $M = 2$ epochs and a $25\,\mathrm{m}$ range-limited lidar is shown in red in the bottom figure, where the high integrity risk between 40 and $50\,\mathrm{s}$ can be explained by the lack of visible landmarks at the left-most region of the path. In addition, the integrity risk for a $20\,\mathrm{m}$ range-limited lidar is shown in blue. This decreases the number of landmarks in the view resulting in a significant increase in localization risk.

VII. Conclusions and Future Work

This paper derives the first constant-time localization integrity risk evaluation methodology without assumptions on the form or nature of the faults. The work becomes particularly important in safety-critical applications where undetected faults may have catastrophic consequences, such as mobile robots operating among humans. Simulated and experimental results show the evaluation of localization safety in two different scenarios. Future work entails extending this methodology to graph optimization techniques.

Appendix I
Estimate Bias as a Function of Faults

This appendix shows that the estimate bias at k can be expressed as a function of prior faults from time $k - M$ to k and the estimate bias at time $k - M - 1$. First, the estimate error at k is expanded as a function of the estimate bias at $k - 1$ and the faults at k. Then, this relation is generalized for some preceding horizon of M epochs.

First, we substitute (2) and (7) into (5):

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{L}_k\left(\mathbf{h}\left(\mathbf{x}_k\right) + \mathbf{v}_k + \mathbf{f}_k - \mathbf{h}\left(\bar{\mathbf{x}}_k\right)\right) \tag{28}$$

Then, function $\mathbf{h}\left(\cdot\right)$ is linearized using a first order Taylor expansion as: $\mathbf{h}\left(\mathbf{x}_k\right) \approx \mathbf{h}\left(\bar{\mathbf{x}}_k\right) + \mathbf{H}_k\left(\mathbf{x}_k - \bar{\mathbf{x}}_k\right)$, and substituted into (28), which is rewritten as:

$$\hat{\mathbf{x}}_k = \underbrace{\left(\mathbf{I} - \mathbf{L}_k\mathbf{H}_k\right)}_{\mathbf{L}'_k}\bar{\mathbf{x}}_k + \mathbf{L}_k\mathbf{H}_k\mathbf{x}_k + \mathbf{L}_k\left(\mathbf{v}_k + \mathbf{f}_k\right) \tag{29}$$

The estimate error is obtained subtracting the unknown true state, $\mathbf{x}_k$, from both sides:

$$\delta\hat{\mathbf{x}}_k = \mathbf{L}'_k\left(\bar{\mathbf{x}}_k - \mathbf{x}_k\right) + \mathbf{L}_k\left(\mathbf{v}_k + \mathbf{f}_k\right) \tag{30}$$

Substituting (1) and (3) and again linearizing $\mathbf{g}(\cdot,\cdot)$ using a first order Taylor expansion as:

$$\mathbf{g}\left(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}\right) \approx \mathbf{g}\left(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}\right) + \mathbf{\Phi}_{k-1}\left(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}\right) \tag{31}$$

the estimate error becomes:

$$\delta\hat{\mathbf{x}}_k = \underbrace{\mathbf{L}'_k\mathbf{\Phi}_{k-1}}_{\mathbf{L}''_k}\delta\hat{\mathbf{x}}_{k-1} - \mathbf{L}'_k\mathbf{w}_{k-1} + \mathbf{L}_k\left(\mathbf{v}_k + \mathbf{f}_k\right) \tag{32}$$

Finally, taking the expected value of both sides:

$$\hat{\mathbf{f}}_{x_k} = \mathbf{L}''_k\hat{\mathbf{f}}_{x_{k-1}} + \mathbf{L}_k\mathbf{f}_k \tag{33}$$

Expanding (33) as a function of previous faults until any desired preceding horizon time $k - M$:

$$\begin{aligned}\hat{\mathbf{f}}_{x_k} &= \mathbf{L}_k\mathbf{f}_k + \mathbf{L}''_k\mathbf{L}_{k-1}\mathbf{f}_{k-1} + \mathbf{L}''_k\mathbf{L}''_{k-1}\mathbf{L}_{k-2}\mathbf{f}_{k-2} + \ldots \\ &\ldots + \mathbf{L}''_k\ldots\mathbf{L}''_{k-(M-1)}\mathbf{L}_{k-M}\mathbf{f}_{k-M} + \ldots \\ &\ldots + \mathbf{L}''_k\ldots\mathbf{L}''_{k-(M-1)}\mathbf{L}''_{k-M}\hat{\mathbf{f}}_{x_{k-M-1}}\end{aligned} \tag{34}$$

and expressing the previous equation in matrix form we arrive at (16), where $\mathbf{f}_k^{(M)}$ and $\mathbf{A}_k^{(M)}$ are defined in (17) and (35). Additionally, given a fixed preceding horizon of M epochs, $\mathbf{A}_k^{(M)}$ is recursively computed from $\mathbf{A}_{k-1}^{(M)}$ using Algorithm 1. Note that subindexes *end* and *end-1* refer to the last and second to last block matrices in $\mathbf{A}_k^{(M)}$ respectively.

Algorithm 1 Recursive evaluation of matrix $\mathbf{A}_k^{(M)}$

```
1: function A_EVALUATION ($\mathbf{A}_{k-1}^{(M)}, \mathbf{L}_k, \mathbf{L}''_k, \mathbf{L}''_{k-M-1}$)
2:     $\mathbf{A}_k^{(M)} = \left[\mathbf{L}_k \quad \mathbf{L}''_k \mathbf{A}_{k-1}^{(M)}\right]$
3:     $\left(\mathbf{A}_k^{(M)}\right)_{\text{end}-1} = [\,]$   // remove block matrix
4:     $\left(\mathbf{A}_k^{(M)}\right)_{\text{end}} = \left(\mathbf{A}_k^{(M)}\right)_{\text{end}} \left(\mathbf{L}''_{k-M-1}\right)^{-1}$
5: end function
```

Appendix II
Non-centrality parameter as a function of faults

This appendix shows that the non-centrality parameter of the fault detector distribution at k can be expressed as a function of prior measurement faults from time $k-M$ to k and the estimate bias at time $k-M-1$. First, the detector non-centrality parameter in (20) is rewritten in matrix form:

$$\lambda_k^{(M)} \triangleq \mathbb{E}\left[\boldsymbol{\gamma}_k^{(M)}\right]^T \mathbf{Y}_k^{(M)\,-1}\, \mathbb{E}\left[\boldsymbol{\gamma}_k^{(M)}\right] \tag{36}$$

where:

$$\boldsymbol{\gamma}_k^{(M)} = \begin{bmatrix} \boldsymbol{\gamma}_k \\ \dots \\ \boldsymbol{\gamma}_{k-M} \end{bmatrix} \quad \text{and} \quad \mathbf{Y}_k^{(M)} = \begin{bmatrix} \mathbf{Y}_k & & \\ & \ddots & \\ & & \mathbf{Y}_{k-M} \end{bmatrix} \tag{37}$$

are the augmented innovation and its covariance matrix.

To find $\mathbb{E}\left[\boldsymbol{\gamma}_k^{(M)}\right]$ as a function of $\mathbf{f}_k^{(M)}$, the innovation mean at time k is expanded as a function of the estimate bias at $k-1$ and the faults at k. Substituting (2) into (7), and approximating $\mathbf{h}\left(\mathbf{x}_k\right)$ as in (29):

$$\boldsymbol{\gamma}_k = -\mathbf{H}_k\left(\mathbf{x}_k - \bar{\mathbf{x}}_k\right) + \mathbf{v}_k + \mathbf{f}_k \tag{38}$$

Substituting (3) and (1) into (38), linearizing $\mathbf{g}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ using (31) and taking the expected value of both sides:

$$\mathbb{E}\left[\boldsymbol{\gamma}_k\right] = \mathbf{f}_k - \mathbf{H}_k \boldsymbol{\Phi}_{k-1} \hat{\mathbf{f}}_{x_{k-1}} \tag{39}$$

Substituting (16) with a preceding horizon of $(M-1)$ epochs into (39) and using matrix notation:

$$\mathbb{E}\left[\boldsymbol{\gamma}_k\right] = \underbrace{\left[\mathbf{I}_{n_k} \quad -\mathbf{H}_k \boldsymbol{\Phi}_{k-1} \mathbf{A}_{k-1}^{(M-1)}\right]}_{\mathbf{B}_k^{(M)}} \underbrace{\begin{bmatrix} \mathbf{f}_k \\ \mathbf{f}_{k-1}^{(M-1)} \end{bmatrix}}_{\mathbf{f}_k^{(M)}} \tag{40}$$

Eq. (40) shows that the innovation mean at k is again a function of the faults occurring within the preceding horizon and the estimate bias at $k-M-1$. Following a similar process for $1 \leq i \leq M$:

$$\mathbb{E}\left[\boldsymbol{\gamma}_{k-i}\right] = \left[\mathbf{0}_{n_{k-i} \times n_k^{(i-1)}} \quad \mathbf{B}_{k-i}^{(M-i)}\right] \underbrace{\begin{bmatrix} \mathbf{f}_{k:k-i} \\ \mathbf{f}_{k-i-1}^{(M-i-1)} \end{bmatrix}}_{\mathbf{f}_k^{(M)}} \tag{41}$$

where $b:a$ for $b \geq a$ include all subindexes from b to a, and $\mathbf{B}_{k-i}^{(M-i)} \in \mathbb{R}^{n_{k-i} \times \left[n_{k-i}^{(M-i)} + m\right]}$ is defined as:

$$\mathbf{B}_{k-i}^{(M-i)} = \left[\mathbf{I}_{n_{k-i}} \quad -\mathbf{H}_{k-i} \boldsymbol{\Phi}_{k-i-1} \mathbf{A}_{k-i-1}^{(M-i-1)}\right] \tag{42}$$

Algorithm 2 presents recursive method to evaluate $\mathbf{A}_{k-i-1}^{(M-i-1)}$ given $\mathbf{A}_{k-i}^{(M-i)}$ and $\mathbf{L}''_{k-i}$. Note that subindex 1 in line 3 refers to the first block matrix in $\mathbf{A}_k^{(M)}$. From (40) and (41), the mean of the augmented innovation vector can be expressed in matrix notation as:

Algorithm 2 Recursive evaluation of matrix $\mathbf{A}_{k-i-1}^{(M-i-1)}$

```
1: function A_PREVIOUS ($\mathbf{A}_{k-i}^{(M-i)}, \mathbf{L}''_{k-i}$)
2:     $\mathbf{A}_{k-i-1}^{(M-i-1)} = \left(\mathbf{L}''_{k-i}\right)^{-1} \mathbf{A}_{k-i}^{(M-i)}$
3:     $\left(\mathbf{A}_{k-i-1}^{(M-i-1)}\right)_1 = [\,]$   // remove block matrix
4:     return: $\mathbf{A}_{k-i-1}^{(M-i-1)}$
5: end function
```

$$\mathbb{E}\left[\boldsymbol{\gamma}_k^{(M)}\right] = \underbrace{\begin{bmatrix} \mathbf{B}_k^{(M)} & \rule{3cm}{0.4pt} & \\ & \mathbf{B}_{k-1}^{(M-1)} & \rule{2cm}{0.4pt} \\ & & \ddots \\ & & & \mathbf{B}_{k-M}^{(0)} \end{bmatrix}}_{\bar{\mathbf{B}}_k^{(M)}} \mathbf{f}_k^{(M)} \tag{43}$$

where $\bar{\mathbf{B}}_k^{(M)} \in \mathbb{R}^{n_k^{(M)} \times \left[n_k^{(M)} + m\right]}$ is an upper triangular matrix that can be recursively computed using Algorithm 3. Finally, substituting (43) into (36):

$$\lambda_k^{(M)} = \mathbf{f}_k^{(M)\,T} \underbrace{\bar{\mathbf{B}}_k^{(M)\,T}\, \mathbf{Y}_k^{(M)\,-1}\, \bar{\mathbf{B}}_k^{(M)}}_{\mathbf{M}_k^{(M)}} \mathbf{f}_k^{(M)} \tag{44}$$

where $\mathbf{M}_k^{(M)}$ is a $\left[n_k^{(M)} + m\right]$ square matrix of rank $n_k^{(M)}$.

Algorithm 3 Recursive evaluation of matrix $\bar{\mathbf{B}}_k^{(M)}$

```
1: function B̄_EVAL($\mathbf{A}_k^{(M)}, \mathbf{H}_{k-M:k}, \boldsymbol{\Phi}_{k-M-1:k-1}, \mathbf{L}''_{k-M:k}$)
2:     for j = M to 0 do
3:         $\mathbf{A}_{k-j-1}^{(M-j-1)}$ = A_PREVIOUS ($\mathbf{A}_{k-j}^{(M-j)}, \mathbf{L}''_{k-j}$)
4:         $\mathbf{B}_{k-j}^{(M-j)} = \left[\mathbf{I}_{n_{k-j}} \quad -\mathbf{H}_{k-j} \boldsymbol{\Phi}_{k-j-1} \mathbf{A}_{k-j-1}^{(M-j-1)}\right]$
5:         $\bar{\mathbf{B}}_k^{(M)} = \begin{bmatrix} \bar{\mathbf{B}}_k^{(M)} \\ \left[\mathbf{0}_{n_{k-j} \times n_k^{(j-1)}} \quad \mathbf{B}_{k-j}^{(M-j)}\right] \end{bmatrix}$
6:     end for
7: end function
```

$$\mathbf{A}_k^{(M)} = \left[\mathbf{L}_k \quad \mathbf{L}''_k \mathbf{L}_{k-1} \quad [\dots] \quad \mathbf{L}''_k \dots \mathbf{L}''_{k-(M-1)} \mathbf{L}_{k-M} \quad \mathbf{L}''_k \dots \mathbf{L}''_{k-M}\right] \tag{35}$$

REFERENCES

[1] G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem," *IEEE Trans. on Robotics Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[2] J. Leonard and H. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, 1992.

[3] S. Williams, G. Dissanayake, and H. Durrant-Whyte, "An efficient approach to the simultaneous localisation and mapping problem," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2002.

[4] R. Kelly and J. Davis, "Required navigation performance (rnp) for prescision approach and landing with gnss application," *NAVIGATION*, vol. 41, no. 1, pp. 1–30, 1997.

[5] R. T. C. for Aeronautics Special Committee 159, "Minimum Aviation System Performance Standards for the Local Area Augmentation System (LAAS)," Document RTCA/DO-245, 2004.

[6] M. Joerger, M. Jamoom, M. Spenko, and B. Pervan, "Integrity of laser-based feature extraction and data association," in *2016 IEEE/ION PLANS*, April 2016, pp. 557–571.

[7] M. Joerger, G. D. Arana, M. Spenko, and B. Pervan, "A new approach to unwanted-object detection in gnss/lidar-based navigation," in *Sensors*, 2018.

[8] G. D. Arana, M. Joerger, and M. Spenko, "Local nearest neighbor integrity risk evaluation for robot navigation," *ICRA*, 2018.

[9] R. G. Brown, "A baseline gps raim scheme and a note on the equivalence of three raim methods," *Navigation*, vol. 39, no. 3, 1992.

[10] B. W. Parkinson and P. Axelrad, "Autonomous GPS Integrity Monitoring Using the Pseudorange Residual," *Navigation*, vol. 35, no. 2, pp. 225–274, 1988.

[11] R. F. SC-159, *Minimum Aviation System Performance Standards for the Local Area Augmentation System (LAAS)*. RTCA, 2004. [Online]. Available: https://books.google.com/books?id=eZXWtgAACAAJ

[12] 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.

[13] M. Joerger, F.-C. Chan, and B. Pervan, "Solution Separation Versus Residual-Based RAIM," *Navigation*, vol. 61, no. 4, 2014.

[14] W. G. C. A. T. Subgroup, "Milestone 3 Report," EU-US Cooperation on Satellite Navigation, Tech. Rep., 2015.

[15] M. Brenner, "Integrated gps/inertial fault detection availability," *Navigation*, vol. 43, no. 2, pp. 111–130, 1996.

[16] T. Walter and P. Enge, "Weighted raim for precision approach," in *PROCEEDINGS OF ION GPS*, vol. 8. Institute of Navigation, 1995, pp. 1995–2004.

[17] M. Joerger and B. Pervan, "Kalman Filter-Based Integrity Monitoring Against Sensor Faults," *AIAA Journal of Guidance, Control and Dynamics*, vol. 36, no. 2, pp. 349–361, 2013.

[18] C. Tanil, M. Joerger, S. Khanafseh, and B. Pervan, "Sequential integrity monitoring for kalman filter innovation-based detectors," *ION-GNNS [accepted for publication]*, 2018.

[19] M. Joerger and B. Pervan, "Quantifying safety of laser-based navigation (submitted)," *in press, IEEE Transactions on Aerospace and Electronic Systems*, 2018.

[20] A. Hassani, G. D. Arana, M. Spenko, and M. Joerger, "Lidar data association risk reduction using tight integration with ins," *ION-GNNS*, 2018.

[21] C. Tanil, M. Joerger, S. Khanafseh, and B. Pervan, "A sequential integrity monitoring for kalman filter innovations-based detectors," *ION GNSS+*, September 2018.

[22] C. Tanil, *Detecting GNSS spoofing attacks using INS coupling*. Illinois Institute of Technology, 2016.

[23] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.

[24] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the ekf-slam algorithm," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2006, pp. 3562–3568.

[25] M. C. Graham, "Robust bayesian state estimation and mapping," Ph.D. dissertation, Massachusetts Institute of Technology, 2015.

[26] R. Isermann, *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media, 2006.

[27] J. Angus, "Raim with multiple faults," *Navigation*, vol. 53, no. 4, pp. 249–257, 2006.