Integrity Risk-Based Model Predictive Control for Mobile Robots

Osama Abdul Hafez, Guillermo Duenas Arana, *student member IEEE*, and Matthew Spenko, *member IEEE*

***Abstract*— This paper presents a Model Predictive Controller (MPC) that uses navigation integrity risk as a constraint. Navigation integrity risk accounts for the presence of faults in localization sensors and algorithms, an increasingly important consideration as the number of robots operating in life and mission-critical situations is expected to increase dramatically in near future (e.g. a potential influx of self-driving cars). Specifically, the work uses a local nearest neighbor integrity risk evaluation methodology that accounts for data association faults as a constraint in order to guarantee localization safety over a receding horizon. Moreover, state and control-input constraints have also been enforced in this work. The proposed MPC design is tested using real-world mapped environments, showing that a robot is capable of maintaining a predefined minimum level of localization safety while operating in an urban environment.**

I. Introduction

Traditionally, pose estimation performance is quantified using a covariance matrix or particle spread [1], [2]. However, these methods only take into account non-faulted cases, and thus may be insufficient for life- and mission-critical applications, such as self-driving cars or other co-robots [3]. Faults, such as GNSS-clock errors [4] or mis-associating features extracted from the environment (e.g. from lidar, radar, or camera data) with the wrong landmark may occur regularly in mobile robot navigation. While some faults may be easily detected and removed, others remain undetected, and not accounting for the impact of those faults can lead to an increased safety risk.

To address this issue, prior work has taken inspiration from the aviation industry to evaluate a robot's integrity risk, or the probability that a robot's pose estimate lies outside predefined limits, a safety metric that considers both faulted and non-faults situations [5], [6], [7], [8], [9], [10], [11]. This paper builds upon that work to present a Model Predictive Controller (MPC) that uses navigation integrity risk as a constraint.

Several methods have been developed to predict the integrity risk in GNSS-based aviation applications [12], [13]. However, these methods cannot be directly applied to mobile robots because ground vehicles operate under sky-obstructed areas where GNSS signals can be altered or blocked by buildings and trees. Therefore, additional sensors are required to localize ground vehicles, such as lidar. An integrity monitoring algorithm for lidar-based localization using an Extended Kalman Filter (EKF) coupled with a global nearest neighbor data association algorithm has been developed in prior work [6], [9]. However, the computational complexity of the global nearest neighbor algorithms restricts its online applicability. In response, a local nearest neighbor-based integrity risk evaluation methodology was developed [8]. This paper uses that implementation and investigates the use of a local nearest neighbor integrity risk metric as a constraint for mobile robot based MPC to guarantee a predefined minimum level of localization safety.

There has been relatively little work in this area. [14], [15], [16], [17] employ the notion of "safe driving envelope" in designing MPCs for autonomous vehicles, but the approach concentrates mostly on collision avoidance, and presents neither rigorous proof of integrity nor practical safety levels. [18] uses the concept of "collision risk assessment" in designing a lidar-based predictive safety control algorithm for excavators. The approach is promising, but does not account for possible faults. Recently, there has been research on tube-based model predictive control, which tries to find the solution that keeps a robot as far as possible from obstacles or tracks a road's center line in the presence of disturbances [19], [20], [21]. But again, the approach assumes that all disturbances are zero mean random variables (non-faulted). In fact, most of the previous work uses state estimation error covariance as the basis for risk assessment, which is proven to not be sufficient [22], [23]. In contrast, the MPC presented here generates control actions that guarantee a minimum level of localization safety during the receding horizon while taking into account possible faults in feature/landmark data associations.

The paper is organized as follows. Section II provides a mathematical background to the problem while Section III introduces the proposed model predictive controller. Results based on experimental data collected in an urban campus are discussed in Section IV. Finally, Section V presents conclusions and future work.

II. Background

This section introduces the necessary mathematical notation and explains the integrity monitoring algorithm used in the paper.

A. State Evolution Model

This work is motivated by its potential benefit and applicability to self-driving cars. As such, a no-slip kinematic bicycle model is used to model the robot (see Fig. 1). However, the work is general enough to be applied to any

*This work is supported by NSF Grant #1637899

O. A. Hafez, G. D. Arana, and M. Spenko are with the Mechanical, Materials, and Aerospace Engineering Department, Illinois Institute of Technology, Chicago, IL, USA oabdulhafez@hawk.iit.edu, gdueasar@hawk.iit.edu, mspenko@iit.edu

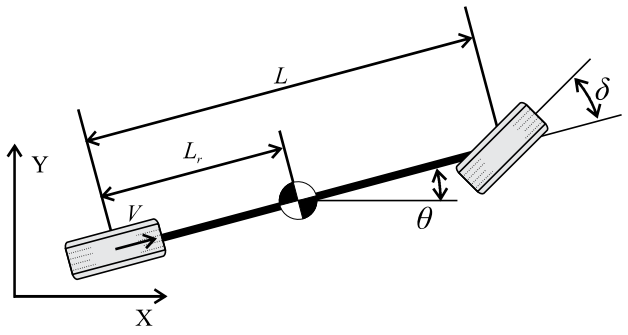

Fig. 1. Bicycle model schematic

mobile robot motion model. The model is given as:

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} V\left(\cos(\theta) - \frac{L_r \sin(\theta)\tan(\delta)}{L}\right) \\ V\left(\sin(\theta) + \frac{L_r \cos(\theta)\tan(\delta)}{L}\right) \\ \frac{V\tan(\delta)}{L} \end{pmatrix} \quad (1)$$

where X and Y represent the center of mass position in the inertial frame, V is the rear wheel velocity, and L_r is the distance between the rear wheel and the center of mass. The vehicle state is given as $\mathbf{x} = [X, Y, \theta]^T$, and the control input as $\mathbf{u} = [V, \delta]^T$ where δ is the steering angle.

The state evolution for the kinematic bicycle model is modeled as a non-linear function of the prior states, $\mathbf{x}_k \in \mathbb{R}^m$, and control-inputs, $\mathbf{u}_k \in \mathbb{R}^n$, with additive noise:

$$\mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k \quad \text{where} \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{W}_k) \quad (2)$$

where $\mathbf{g}(\cdot,\cdot)$ is the discretized version of (1), $\mathbf{u}_k$ is the control-input vector (determined by the MPC), and $\mathbf{w}_k$ is white Gaussian noise with known covariance $\mathbf{W}_k$.

B. Measurement Model

In feature-based navigation, raw sensor data is reduced to a number of distinguishable features. This work assumes access to a map consisting of n_L landmarks from which n_k^F features are extracted at time k, such that $n_k^F \leq n_L$, where each feature is assumed to provide the same number of measurements, m_F. Measurements belonging to each extracted feature $\mathbf{z}_k^i \in \mathbb{R}^{m_F}$ are concatenated into the measurement vector $\mathbf{z}_k \in \mathbb{R}^{n_k}$, which comprises a total of $n_k = n_k^F m_F$ measurements at time k. The feature extracted from landmark t is denoted i_t, whereas the landmark from which feature i has been extracted is denoted t_i. The measurement model for a single extracted feature i is:

$$\mathbf{z}_k^i = \mathbf{h}_{t_i}(\mathbf{x}_k) + \mathbf{v}_k^i \quad (3)$$

where $\mathbf{h}_{t_i}(\cdot)$ is the measurement function of landmark t_i and $\mathbf{v}_k^i$ is the white Gaussian noise with known covariance matrix such that $\mathbf{v}_k^i \sim \mathcal{N}(\mathbf{0}, \mathbf{V}_k^i)$.

C. Innovation Vector

The discrepancy between a feature's measurements and the expected measurements from a landmark is the innovation. The innovation vector of feature i and landmark t is:

$$\boldsymbol{\gamma}_k^{i,t} = \mathbf{z}_k^i - \mathbf{h}_t(\bar{\mathbf{x}}_k) \quad (4)$$

where $\bar{\mathbf{x}}_k$ is the pose estimate mean after the EKF prediction step. Joerger et. al [6] proved that the innovations are normally distributed as:

$$\boldsymbol{\gamma}_k^{i,t} \sim \mathcal{N}\left(\bar{\mathbf{y}}_k^{t_i,t}, \mathbf{Y}_k^i\right) \quad (5)$$

where $\mathbf{Y}_k^i \triangleq \mathbf{H}_k^{t_i}\bar{\mathbf{P}}_k{\mathbf{H}_k^{t_i}}^T + \mathbf{V}_k^i$ is the innovation covariance matrix. The innovation mean measures the separation between the expected measurements of landmark t and the expected measurements of the actual landmark corresponding to extracted feature i, which is unknown:

$$\bar{\mathbf{y}}_k^{t_i,t} \triangleq \mathbf{h}_{t_i}(\bar{\mathbf{x}}_k) - \mathbf{h}_t(\bar{\mathbf{x}}_k) \quad (6)$$

Thus, the innovation is only zero mean when the correct association between feature and landmark is selected, i.e. $t = t_i$, in which case the r.h.s. of (6) becomes zero.

The next section briefly presents the local nearest neighbor data association process, which employs the innovation vectors as criteria to select correct associations.

D. Data Association

Given a set of n_F extracted features and n_L previously mapped landmarks, each feature i is associated to landmark t^* if the following criteria is met:

$$\|\boldsymbol{\gamma}_{i,t^*}\|_{\mathbf{Y}_i^{-1}} < T \qquad \text{where} \quad t^* = \underset{t}{\operatorname{argmin}} \ \|\boldsymbol{\gamma}_{i,t}\|_{\mathbf{Y}_i^{-1}} \quad (7)$$

Threshold T is a user defined parameter and $\|\boldsymbol{\gamma}_{i,t}\|_{\mathbf{Y}_i^{-1}} = \sqrt{\boldsymbol{\gamma}_{i,t}^T \mathbf{Y}_i^{-1} \boldsymbol{\gamma}_{i,t}}$ is the weighted norm of the individual innovation vector of extracted feature i and landmark t. Note that an incorrect association occurs when the criteria in (7) is met for $t^* \neq t_i$, which means that feature i will be associated to a landmark, t^*, other than t_i.

In the next section, the integrity risk of the system is evaluated when incorrect associations occur.

E. Hazardous Misleading Information

In this work, the system's integrity risk is evaluated as the probability of Hazardous Misleading Information (HMI). HMI occurs when the estimate error in the state of interest exceeds a predefined alert limit (see Fig. 2), i.e.:

$$HMI_k = \left|\boldsymbol{\alpha}^T \hat{\boldsymbol{\epsilon}}_k\right| > l \quad (8)$$

where $\boldsymbol{\alpha} \in \mathbb{R}^m$ is a vector selects the state of interest, l is the permissible error limit, and $\hat{\boldsymbol{\epsilon}}_k = \hat{\mathbf{x}}_k - \mathbf{x}_k$ is the EKF update estimate error.

The system's integrity risk must be evaluated under both faulted and fault-free hypotheses. This work focus on data association faults, which occur when extracted features are associated to the incorrect landmarks as explained in Section II-D. An incorrect association event, IA_k, at time k occurs as soon as one feature is incorrectly associated while a correct association, CA_k, indicates that all extracted features at time k are correctly associated. Similarly, situations where at least one incorrect association has occurred up to and including the current time are considered faulted while only when all features have been correctly associated, it is considered a fault-free condition. Then, denoting the fault free condition

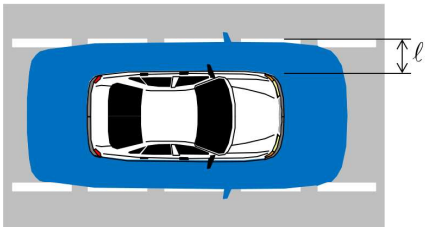

Fig. 2. Integrity risk representation for autonomous vehicle applications. The integrity risk is the probability of the car being outside the alert limit requirement box (blue shaded area) when it was estimated to be inside the box. If the primary concern is the lateral deviation, then the alert limit is the distance l between the edge of the car and the edge of the lane.

as $CA_K = \{CA_1, \ldots, CA_k\}$ and its complementary event as IA_K, the integrity risk is:

$$P(HMI_k) = P(HMI_k, CA_K) + P(HMI_k, IA_K) \quad (9)$$

where an uppercase time subscript K denotes all time up to and including time k. Rewriting (9) using the total probability theorem and bounding $P(HMI_k \mid IA_K)$ by one, we define an upper bound on the integrity risk as:

$$P(HMI_k) \leq 1 + (P(HMI_k \mid CA_K) - 1)\, P(CA_K) \quad (10)$$

The first term in the right hand side of (10) is the integrity risk under fault-free conditions, which is evaluated by integrating the tails of the pose estimate distribution in the state of interest as:

$$P(HMI_k \mid CA_K) = P\left(\left|\boldsymbol{\alpha}^T \hat{\boldsymbol{\epsilon}}_k\right| > l \mid CA_K\right) = 2\Phi\left[-\frac{l}{\hat{\sigma}_k}\right] \quad (11)$$

where $\Phi[\cdot]$ is the standard normal CDF and $\hat{\sigma}_k = \sqrt{\boldsymbol{\alpha}^T \hat{\mathbf{P}}_k \boldsymbol{\alpha}}$ is the standard deviation in the state of interest.

The second term in (10) evaluates the probability of every landmark up to and including k being correctly associated, which can be recursively computed as:

$$P(CA_K) = P(CA_k \mid CA_{K-1}) P(CA_{K-1}) \quad (12)$$

where $P(CA_0) = 1$. A lower bound on the probability of correct associations at time k is presented in [8]:

$$P(CA_k \mid CA_{K-1}) \geq 1 - n_k^F + \sum_{i=1}^{n_k^F} X^2_{m+m_F}\left[\max\left\{\frac{1}{4}\left\|\bar{\mathbf{y}}_k^{t_i}\right\|^2_{\mathbf{Y}_k^{i\,-1}}, \left(\left\|\bar{\mathbf{y}}_k^{t_i}\right\|_{\mathbf{Y}_k^{i\,-1}} - T\right)^2\right\}\right] \quad (13)$$

where $X^2_a[\cdot]$ is the chi-squared CDF with a degrees of freedom and:

$$\left\|\bar{\mathbf{y}}_k^{t_i}\right\|_{\mathbf{Y}_k^{i\,-1}} = \min_{t \neq t_i} \left\|\bar{\mathbf{y}}_k^{t_i,t}\right\|_{\mathbf{Y}_k^{i\,-1}} \quad (14)$$

evaluates the minimum separation between landmark t_i and the rest of the map.

The next section develops a model predictive controller that uses integrity risk as a constraint to guarantee localization safety.

III. Controller Design

Model predictive control has become increasingly popular in robotics [24], [25], [26], and a detailed description of the technique can be found in [27]. In mobile robotics, most model predictive controllers predict future system states using a linear kinematic model [24]. Model predictive controllers based on a linear time varying representation of the system have been applied to trajectory tracking in [28], [29]. Nonlinear formulations for trajectory tracking have been shown in [30], [31].

A. Cost Function

In this work, the model predictive controller is designed to find the control input sequence that minimizes a quadratic cost function with a finite Receding Horizon N:

$$J(k) = \sum_{i=1}^{N} \Big(\delta\bar{\mathbf{x}}_{k+i|k}^T \; \mathbf{R}(\theta_{k+i}^{ref})^T \; \mathbf{Q} \; \mathbf{R}(\theta_{k+i}^{ref}) \; \delta\bar{\mathbf{x}}_{k+i|k} + \delta\mathbf{u}_{k+i-1|k}^T \; \mathbf{R} \; \delta\mathbf{u}_{k+i-1|k} \Big) \quad (15)$$

where $\delta\bar{\mathbf{x}}_{k+i|k} = \bar{\mathbf{x}}_{k+i|k} - \mathbf{x}_{k+i}^{ref}$ is the difference between the predicted state given $\hat{\mathbf{x}}_k$ (estimated state at time k using EKF) and the reference state (predicted tracking error) expressed in the inertial frame; $\delta\mathbf{u}_{k+i-1|k} = \mathbf{u}_{k+i-1|k} - \mathbf{u}_{k+i-1}^{ref}$ is the difference between the predicted optimal control given $\hat{\mathbf{x}}_k$ and the reference control determined by the path (predicted control error); $\mathbf{R}(\theta_{k+i}^{ref})$ is a rotation matrix that transforms the predicted tracking error from the inertial frame to the path reference frame (x-axis is aligned with the tangent of the path); and $\mathbf{R} \in \mathbb{R}^{n \times n}$ and $\mathbf{Q} \in \mathbb{R}^{m \times m}$ are positive semi-definite weighting matrices.

MPC formulation requires converting the optimal control problem to a static optimization problem expressed as a function of the receding horizon control sequence. Therefore, every variable in the optimal control problem needs to be written in terms of the predicted control error over the horizon. The receding horizon predicted control and tracking error, respectively, are defined as:

$$\delta\mathbf{U}_k = \left[\; \delta\mathbf{u}_{k|k}^T \; \delta\mathbf{u}_{k+1|k}^T \; \ldots \; \delta\mathbf{u}_{k+N-1|k}^T \;\right]^T \quad (16)$$

$$\delta\bar{\mathbf{X}}_k = \left[\; \delta\bar{\mathbf{x}}_{k+1|k}^T \; \delta\bar{\mathbf{x}}_{k+2|k}^T \; \ldots \; \delta\bar{\mathbf{x}}_{k+N|k}^T \;\right]^T \quad (17)$$

where $\delta\mathbf{U}_k \in \mathbb{R}^{nN}$ and $\delta\bar{\mathbf{X}}_k \in \mathbb{R}^{mN}$. It can be proven that $\delta\bar{\mathbf{X}}_k$ can be expressed as a function of $\delta\mathbf{U}_k$ and $\delta\hat{\mathbf{x}}_{k|k}$ using the state evolution model (2):

$$\delta\bar{\mathbf{X}}_k = \boldsymbol{\Phi}_k \delta\hat{\mathbf{x}}_{k|k} + \boldsymbol{\Gamma}_k \delta\mathbf{U}_k \quad (18)$$

where $\boldsymbol{\Gamma}_k \in \mathbb{R}^{mN \times nN}$ and $\boldsymbol{\Phi}_k \in \mathbb{R}^{mN \times m}$ are defined in [24]. By defining $\boldsymbol{\Psi} \in \mathbb{R}^{nN \times nN}$ and $\boldsymbol{\Omega}_k \in \mathbb{R}^{mN \times mN}$ as a block matrices with $\mathbf{R}$ and $\mathbf{R}(\theta_{k+i}^{ref})^T \; \mathbf{Q} \; \mathbf{R}(\theta_{k+i}^{ref})$ (arranged from $i = 1, ..., N$) as a diagonal elements, respectively, the cost function can be expressed as:

$$\begin{aligned} J(k) &= \delta\bar{\mathbf{X}}_k^T \boldsymbol{\Omega}_k \delta\bar{\mathbf{X}}_k + \delta\mathbf{U}_k^T \boldsymbol{\Psi} \delta\mathbf{U}_k \\ &= (\boldsymbol{\Phi}_k \delta\hat{\mathbf{x}}_{k|k} + \boldsymbol{\Gamma}_k \delta\mathbf{U}_k)^T \boldsymbol{\Omega}_k (\boldsymbol{\Phi}_k \delta\hat{\mathbf{x}}_{k|k} + \boldsymbol{\Gamma}_k \delta\mathbf{U}_k) \\ &\quad + \delta\mathbf{U}_k^T \; \boldsymbol{\Psi} \; \delta\mathbf{U}_k \end{aligned} \quad (19)$$

The cost function can be written in its standard form by expanding (19) then removing the terms that do not contain $\delta\mathbf{U}_k$ (as they don't affect the optimal solution):

$$J(k) = \frac{1}{2}\delta\mathbf{U}_k^T\mathbf{G}_k\delta\mathbf{U}_k + \mathbf{f}_k^T\delta\mathbf{U}_k \tag{20}$$

where $\mathbf{G}_k \in \mathbb{R}^{nN\times nN}$ and $\mathbf{f}_k \in \mathbb{R}^{nN}$ are defined as:

$$\mathbf{G}_k = 2(\mathbf{\Gamma}_k^T\mathbf{\Omega}_k\mathbf{\Gamma}_k + \mathbf{\Psi}) \tag{21}$$

$$\mathbf{f}_k = 2\mathbf{\Gamma}_k^T\mathbf{\Omega}_k\mathbf{\Phi}_k\delta\hat{\mathbf{x}}_{k|k} \tag{22}$$

B. Control input and State constraints

In this work, inequality constraints on control input and state have been enforced, expressed in terms of the optimization variable $\delta\mathbf{U}_k$:

$$\mathbf{U}_{min} - \mathbf{U}_k^{ref} \leq \delta\mathbf{U}_k \leq \mathbf{U}_{max} - \mathbf{U}_k^{ref} \tag{23}$$

where $\mathbf{U}_k^{ref}$ is the reference control input over the horizon, at which the state evolution model is linearized.

A covariance propagation model over the receding horizon needs to be specified, in order to build a probabilistically valid state inequality constraints. EKF covariance propagation equations can be used to propagate uncertainty over the horizon (without measurements) by using the linearized state and measurement matrices around the reference path. The linearized state and measurement matrices at time k and prediction step r is:

$$\mathbf{A}_{k+r} = \left.\frac{\partial\mathbf{g}(\mathbf{x}_{k+r},\mathbf{u}_{k+r})}{\partial\mathbf{x}_{k+r}}\right|_{\mathbf{u}_{k+r}^{ref},\mathbf{x}_{k+r}^{ref}} \tag{24}$$

$$\mathbf{H}_{k+r} = \left.\frac{\partial\mathbf{h}(\mathbf{x}_{k+r})}{\partial\mathbf{x}_{k+r}}\right|_{\mathbf{x}_{k+r}^{ref}} \tag{25}$$

where the size of $\mathbf{h}(\mathbf{x}_{k+r}) \in \mathbb{R}^{n_{k+r}}$ is determined by the predicted number of landmarks in the field of view (FoV) at time k and prediction step r. Extended Kalman Filter-based covariance propagation during the receding horizon can be written as:

$$\bar{\mathbf{P}}_{k+r|k} = \mathbf{A}_{k+r-1}\hat{\mathbf{P}}_{k+r-1|k}\mathbf{A}_{k+r-1}^T + \mathbf{W}_{k+r-1} \tag{26}$$

$$\mathbf{Y}_{k+r} = \mathbf{H}_{k+r}\bar{\mathbf{P}}_{k+r|k}\mathbf{H}_{k+r}^T + \mathbf{V}_{k+r} \tag{27}$$

$$\hat{\mathbf{P}}_{k+r|k} = \left(\mathbf{I} - \bar{\mathbf{P}}_{k+r|k}\mathbf{H}_{k+r}^T\mathbf{Y}_{k+r}^{-1}\mathbf{H}_{k+r}\right)\bar{\mathbf{P}}_{k+r|k} \tag{28}$$

where $\hat{\mathbf{P}}_{k+r|k}$ is the predicted covariance matrix at time k and prediction step r, given the covariance at k and assuming that all FoV landmarks are detected from k to $k+r$. In this work, the state inequality constraints are applied on the positioning part of the state. Due to the fact that rotational states generally have smaller uncertainty than positioning states, a tight probabilistic bound on positioning state can be derived by solving the eigenvalue problem for $\hat{\mathbf{P}}_{k+r|k}$:

$$\hat{\mathbf{P}}_{k+r|k} = \mathbf{Q}_{k+r}\mathbf{D}_{k+r}\mathbf{Q}_{k+r}^T \tag{29}$$

where $\mathbf{D}_{k+r}$ is the eigenvalues (diagonal) matrix, and $\mathbf{Q}_{k+r}$ is a matrix its columns are the normalized eigenvectors. The w-sigma probabilistic bound on positioning state at time k and prediction step r is defined as:

$$\mathbf{c}_{k+r} = w\,\mathbf{max}\left[\mathbf{diag}\left(\mathbf{D}_{k+r}^{1/2}\right)\right]\mathbf{1} \tag{30}$$

where $\mathbf{c}_{k+r} \in \mathbb{R}^{m_{xyz}}$, m_{xyz} is the number of positioning states ($m_{xyz} = 2$ for bicycle model), and $\mathbf{1} \in \mathbb{R}^{m_{xyz}}$ is a vector of 1 as elements. The probabilistic bound vector at time k over the receding horizon is defined as:

$$\mathbf{C}_k = [\ \mathbf{c}_{k+1}^T\ \mathbf{c}_{k+2}^T\ ...\ \mathbf{c}_{k+N}^T\ ]^T \tag{31}$$

By defining $\Upsilon \in \mathbb{R}^{Nm_{xyz}\times Nm}$ as a block matrix with the positioning states extraction matrix $\mathbf{L} \in \mathbb{R}^{m_{xyz}\times m}$ as a diagonal elements, the positioning state inequality constraints can be written as a linear combination of the optimization variable δU_k using (18) as follows:

$$\Upsilon\left(\mathbf{X}_{min} - \mathbf{X}_k^{ref}\right) + \mathbf{C}_k \leq \Upsilon\delta\bar{\mathbf{X}}_k \leq \Upsilon\left(\mathbf{X}_{max} - \mathbf{X}_k^{ref}\right) - \mathbf{C}_k \tag{32}$$

$$\begin{aligned}\Upsilon\left(\mathbf{X}_{min} - \mathbf{X}_k^{ref} - \mathbf{\Phi}_k\delta\hat{\mathbf{x}}_{k|k}\right) + \mathbf{C}_k \leq \Upsilon\mathbf{\Gamma}_k\delta\mathbf{U}_k \leq \\ \Upsilon\left(\mathbf{X}_{max} - \mathbf{X}_k^{ref} - \mathbf{\Phi}_k\delta\hat{\mathbf{x}}_{k|k}\right) - \mathbf{C}_k\end{aligned} \tag{33}$$

where $\mathbf{X}_k^{ref}$ is the reference state over the receding horizon, at which the state evolution model is linearized, and $\delta\hat{\mathbf{x}}_{k|k}$ is the difference between EKF state estimate and reference state at the beginning of the horizon.

C. Integrity Risk Constraint

Unlike control input and state constraints, integrity risk constraints cannot be expressed as a linear combination of the optimization variable $\delta\mathbf{U}_k$. Therefore, a proper formulation must be defined in order to relate this complex metric to the control variation over the receding horizon. To this end, the integrity risk constraint is defined as:

$$P(HMI_{k+r|k}) \leq I_{req} \quad \forall r = 1, 2, ..., N \tag{34}$$

where $P(HMI_{k+r|k})$ is the predicted integrity risk at time k and prediction step r given the state estimate and covariance at k, and I_{req} is the integrity risk requirement. This requirement is intended to be determined by some regulating agency. For example, the Federal Aviation Administrations (FAA) navigation integrity requirement for an aircraft precision approach is between 10^{-7} to 10^{-9} [5].

The integrity risk constraint formulation as a function of the optimization variable, $\delta\mathbf{U}_k$, is illustrated in Algorithm 1. At the first prediction step, the non-linear state evolution model is used to predict the next states $\bar{\mathbf{x}}_{k+1|k}$ as a function of the estimated states $\bar{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_k$ (EKF state estimate with real measurement) and the optimization variable $\delta\mathbf{u}_{k|k}$. Then, the predicted (EKF) next state covariance matrix $\bar{\mathbf{P}}_{k+1|k}$ is found by propagating the estimated (EKF) state covariance matrix $\bar{\mathbf{P}}_{k|k} = \hat{\mathbf{P}}_k$ using (26). Afterwards, the predicted measurements for every FoV landmark are computed. Subsequently, the predicted innovation mean between every pair of FoV landmarks is evaluated followed by the predicted minimum innovation mean norm for every FoV landmark.

Thereafter, the upper-bound on the correct association probability $P(CA_{K+1|k})$ is evaluated followed by EKF covariance update $\hat{\mathbf{P}}_{k+1|k}$ (without measurements) using (28). After that, the predicted integrity risk in the first prediction step is evaluated as a function of the optimization variable $\delta\mathbf{u}_{k|k}$, and augmented in the non-linear constraint vector $\mathbf{c}$. Finally, the process repeats by propagating the predicted state $\bar{\mathbf{x}}_{k+1|k}$ and covariance matrix $\hat{\mathbf{P}}_{k+1|k}$ in the second prediction step, and so on (N times).

Algorithm 1 Integrity risk constraint formulation

Given
- Landmark map
- Correct Association probability at k $P(CA_K)$
- Predefined integrity risk requirement I_{req}
- State estimate and covariance matrix at k ($\hat{\mathbf{x}}_k$, $\hat{\mathbf{P}}_k$)
- Linearized state matrices over the receding horizon $\{\mathbf{A}_k, ..., \mathbf{A}_{k+N-1}\}$
- Linearized control matrices over the receding horizon $\{\mathbf{B}_k, ..., \mathbf{B}_{k+N-1}\}$
- Optimization variables $\{\delta\mathbf{u}_{k|k}, ..., \delta\mathbf{u}_{k+N-1|k}\}$
- Reference control inputs over the receding horizon $\{\mathbf{u}^{ref}_k, ..., \mathbf{u}^{ref}_{k+N-1}\}$

for every prediction step r **do**
- State prediction $\bar{\mathbf{x}}_{k+r|k} = \mathbf{g}(\bar{\mathbf{x}}_{k+r-1|k}, \delta\mathbf{u}_{k+r-1|k} + \mathbf{u}^{ref}_{k+r-1})$
- EKF covariance prediction $\bar{\mathbf{P}}_{k+r|k}$ using (26)
- Calculate the predicted measurements for every FoV landmark
- Evaluate the predicted innovation mean between every pair of FoV landmarks $\bar{\mathbf{y}}^{t_i,t_j}_{k+r}$ $(t_i \neq t_j \quad \forall ij)$ using (6)
- Find the predicted minimum innovation mean norm for every FoV landmark $\left\|\bar{\mathbf{y}}^{t_i}_{k+r}\right\|_{\mathbf{Y}^{i}_{k+r}{}^{-1}}$ using (14)
- Calculate $P(CA_{K+r|k})$ by evaluating $P(CA_{k+r|k}|CA_{K+r-1|k})$ using (13) then substituting the result in (12)
- EKF covarience update $\hat{\mathbf{P}}_{k+r|k}$ (without measurements) using (28)
- Compute the non-faulted integrity risk $P(HMI_{k+r|k} \mid CA_{K+r|k})$ using (11)
- Find the integrity risk $P(HMI_{k+r|k})$ by substituting $P(HMI_{k+r|k} \mid CA_{K+r|k})$ and $P(CA_{K+r|k})$ in (10)
- Augment $P(HMI_{k+r|k}) - I_{req}$ into the non-linear constraints vector $\mathbf{c}$

end for

Return c

IV. Results

This section presents the results of applying the proposed MPC to a simulated mobile robot (modeled using eq. (1)) navigating in an experimentally mapped urban environment (see Fig. 3). A feature extractor, based on [32], is implemented to extract tree trunks and light posts. For life-critical

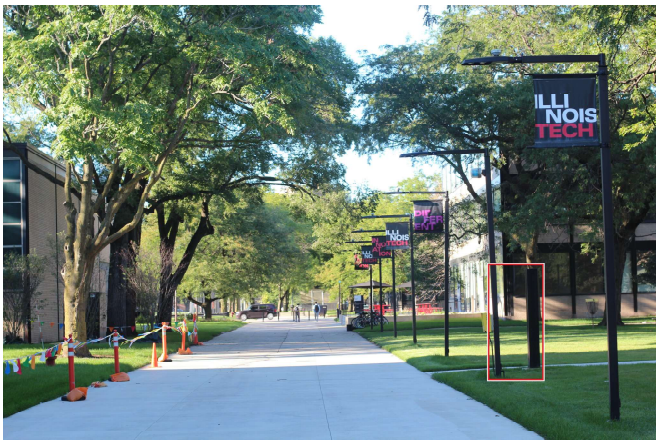

Fig. 3. The experimentally mapped environment used for testing. Note the two landmarks highlighted by the red rectangle on the right side.

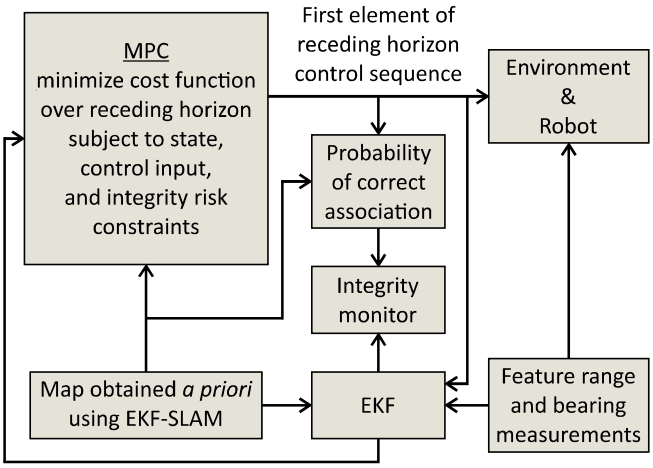

Fig. 4. The proposed Control Scheme

co-robotics applications, ignoring the impact of misassociations may result in an unpredictable positioning errors that are unbounded by the 3σ covariance envelope (bounds the estimate error with 0.997 probability). For example, if the mobile robot were a self-driving car, the state estimate may be in the wrong lane, which is extremely dangerous. In the testing environment, landmarks are well-spaced (which leads to low $P(HMI)$) except for a light post and a column, as shown enclosed in red rectangle in the lower right side of Fig. 3. Mapping is done by fusing GPS positioning measurements, Inertial Measurement Unit (IMU), and lidar features using EKF-SLAM in a loose coupling fashion [33]. The experimental setup is comprised of two Velodyne VLP-16 lidars, a STIM-300 IMU, and a Novatel SPAN-CPT GPS reciever.

The control scheme is illustrated in Fig. 4. A simulated lidar provides range and bearing measurements to the mapped landmarks ($m_F = 2$). Lidar measurements are disturbed by Gaussian white noise (see Table I). Constant velocity is assumed, thus creating a path tracking problem (the penalization of longitudinal error in the path reference frame is set to zero). The path planner specifies the path to

TABLE I

SIMULATION PARAMETERS

Std. dev. lidar range, bearing	$0.1\,\text{m}, 2^\circ$
$[X(m), Y(m), \theta(^\circ)]_{min,max}$	$[(10, 80)\,(-2.5, 2.5)\,(-45, 45)]$
$[\delta(^\circ), V(m/s)]_{min,max}$	$[(-45, 45)(4, 4)]$
Goal states $[X(m), Y(m), \theta(^\circ)]$	$\begin{bmatrix}0 & -1.5 & 0\end{bmatrix}$
R	$\begin{bmatrix}5 & 0\\ 0 & 5\end{bmatrix}$
Q	$\begin{bmatrix}0 & 0 & 0\\ 0 & 20 & 0\\ 0 & 0 & 20\end{bmatrix}$
W	$\begin{bmatrix}0.05 & 0 & 0\\ 0 & 0.05 & 0\\ 0 & 0 & 0.002\end{bmatrix}$
lidar range	15 m
Sampling time	0.1 s
Prediction horizon	20 steps
Alert limit	1 m
I_{req}	10^{-8}

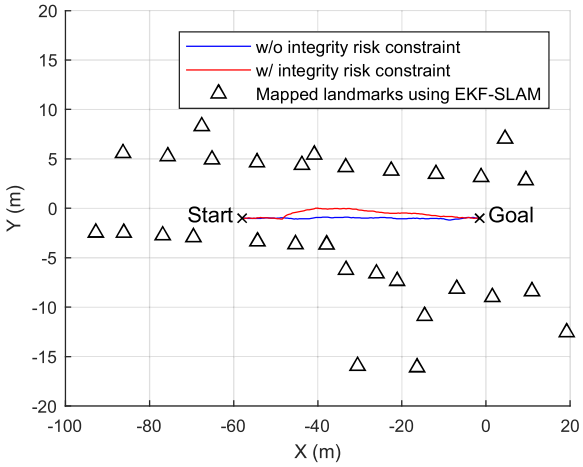

Fig. 5. Mobile Robot trajectory

be a straight line from the current estimated state to the goal state. Therefore, the linearization, during the horizon, will be based on the path specified at the beginning of the horizon.

A comparison study between MPC with and without the integrity risk constraint has been conducted to investigate its impact on localization safety. Fig. 5 shows the robot's trajectory for MPC with and without the integrity risk constraint. With the integrity risk constraint, the mobile robot turns to the left when it reaches $X \approx -50m$, because the MPC predicts that staying on the original course would lead to an integrity risk that exceeds I_{req} after 2 s (prediction horizon Ndt) due to the chance of mis-associating the lamppost and column shown in Fig. 3 .

Specifically, the model predictive controller increases the distance between the two landmarks in the measurement space (range and bearing). Consequently, the probability of correct associations drops more for the case of MPC without the integrity-risk constraint than with the integrity risk constraint (≈ 0.9999994 vs ≈ 0.9999999975 after passing the critical region), which is two orders of magnitude smaller. Therefore, the integrity risk exceeds the localization safety

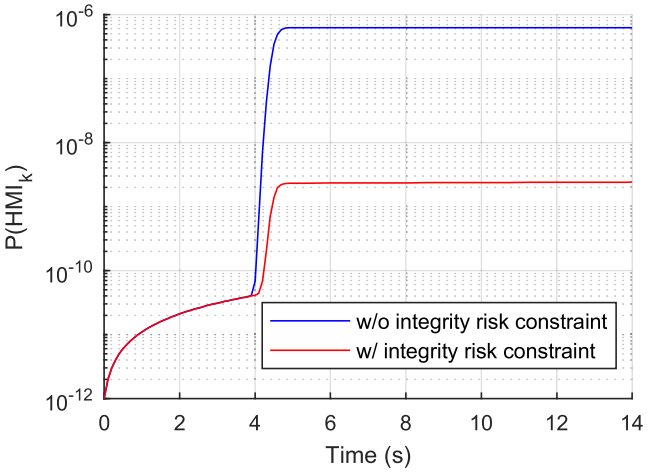

Fig. 6. Integrity Risk $P(HMI_k)$

threshold without constraining the integrity ($\approx 6 \times 10^{-7} > I_{req}$), whereas constraining integrity preserves the integrity risk below the localization safety threshold ($\approx 2.5 \times 10^{-9} < I_{req}$), as shown in Fig. 6.

V. CONCLUSION AND FUTURE WORK

This paper investigates the application of integrity risk as a constraint in a model predictive controller for mobile robot tracking problems. The results show that the MPC is able to perform maneuvers needed to maintain a minimum level of localization safety. The work has implications in life- and mission-critical mobile robot applications, where not accounting for the probability of undetected faults can have a significant impact on safety. In future work, we will investigate the derivation of the integrity risk for obstacle avoidance problems.

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] N. A. Othman and H. Ahmad, "The analysis of covariance matrix for kalman filter based slam with intermittent measurement," in *Proc. Int. Conf. Systems, Control, and Informatics*, 2013.

[4] T. Walter and J. Blanch, "Characterization of gps clock and ephemeris errors to support araim," 2015.

[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, "Landmark selection and unmapped obstacle detection in lidar-based navigation," *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017)*, Sep 2017. [Online]. Available: http://par.nsf.gov/biblio/10072545

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

[9] M. Joerger and B. Pervan, "Quantifying safety of laser-based navigation," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 1–1, 2018.