LMI Methods for Extended $\mathcal{H}_\infty$ Filters for Landmark-based Mobile Robot Localization

Julio Fajardo, Victor Ferman, Jabes Guerra, Antonio Ribas Neto and Eric Rohmer

***Abstract*—Localization is still one of the most fundamental tasks for autonomous navigation of mobile robots. However, the existing methods lack robustness when dealing with uncertainties without assuming some characteristics about noise inputs and the non-linearity of the measurement models. In this work, a theoretical basis for designing two separate extended robust filters based on linear matrix inequalities is proposed to solve the localization problem. The first approach is based on the design of an $\mathcal{H}_\infty$ observer-based filter through a two-step prediction correction structure. In this way, a convex optimization problem needs to be solved at each time step to determine the observer-gain that corrects the predicted pose of a differential wheeled robot. The second approach considers the advantages of a full-order filter which guarantees a better performance under the $\mathcal{H}_\infty$ robust requirements. Besides, satisfactory results that validate theoretical remarks were performed in real and virtual scenarios through simulation frameworks.**

***Index Terms*—Localization, Optimization and Optimal Control, Robust Control of Robotic Systems.**

I. Introduction

In many mobile robots' applications, the localization problem plays a fundamental task in autonomous navigation systems, from simple tasks such as vacuum cleaners and floor mops to self-driving cars and delivery robots. Thus, this task has been studied over decades, facilitating a variety of approaches to address this problem. These approaches differ mainly due to the techniques involved in dealing with the problem, specifically the methods to represent the belief about the current pose of the robot and the different sensors used to acquire information about its surroundings. In this way, to achieve full autonomy, only onboard sensors must be used to perform the robot's localization [1]. Thus, one of the most common methods used to predict a mobile robot's current pose is the dead reckoning technique through the use of encoders [2,3]. Nevertheless, the prediction error increases as the robot travel, degrading the pose estimation of the platform, especially for long trajectories and even worse on slippery surfaces [4].

All authors are with Department of Computer Engineering and Industrial Automation, FEEC, UNICAMP, 13083-852 Campinas, SP, Brazil. {julioef,vferman,eric}@dca.fee.unicamp.br

Julio Fajardo and Jabes Guerra is with Turing Research Laboratory, FISICC, Galileo University, Guatemala City, Guatemala. {julio.fajardo,jabes.guerra}@galileo.edu

Antonio Ribas Neto is with Federal Institute of Education, Science and Technology Catarinense, 89609-000 Luzerna, SC, Brazil. PhD Candidate. antonio.ribas@ifc.edu.br

On the other hand, various types of vision sensors (e.g., LiDARs, cameras, and radars) are currently used for mobile robot localization. Data provided by specific sensors are processed to measure some specific parameters of the robots environment. In this way, the system can detect specific landmarks that help the robot to be located within the surrounding environment [3,5]. The main disadvantage of these techniques relies on their dependence on the characteristics of the environment leading to erroneous interpretation of the provided measurements [1].

The most common approach to deal with the problem of robot localization is the probabilistic one which is based on the Bayesian estimation. Classical algorithms like the Kalman Filter (KF) and the extended Kalman Filter (EKF) make stochastic assumptions about the process and sensor noises, treating them as additive Gaussian noise [6]–[8]. However, it is well known that in real applications, the probability distributions are multimodal, and the nonlinearities of the system degrade the performance of these methods as well. Furthermore, other methods like the Monte Carlo approach and the Markov-chain-Monte-Carlo-based methods deal with the problem without making any assumption about noise characteristics [9]–[12]. Nowadays, other mainstream techniques are based on the fast laser scan matching approach, which is based on the iterative closest points (ICP) and the normal distribution transform (NCP) algorithms [13]–[16]. However, some drawbacks are present due to issues with the beam sensor model. Modern optimization-based techniques include incremental constrained smoothing for state estimation, mainly to solve the simultaneous localization and mapping (SLAM) problem [17]. Its limitation is the need for a fixed linearization point for older states, making it unsuitable for highly nonlinear problems. Also, run-time performance can be further improved by exploiting sparsity in constraint jacobians.

This work presents the theoretical basis of two more conservative and robust methods to the probabilistic approach and their proper reformulation to the landmark-based mobile robot localization problem. These methods are based on the robust extended $\mathcal{H}_\infty$ filtering methods as described in [1,18]. Therefore, these approaches differ because they take advantage of robust filtering and linear matrix inequalities (LMIs) methodologies, one of the most powerful tools to formulate control systems as described in [19,20]. Thus, our methodologies consider nonlinear systems with unknown noise inputs through general noise vectors that only require to be energy-bounded and that are also difficult to solve analytically [21].

In addition to that, the landmark-based methodologies proposed throughout this work ensure that the energy gain from the noise inputs to the estimation error ratio is limited by an upper-bound limit, which guarantees the convergence of its solution. The first technique is based on the design of an $\mathcal{H}_\infty$ observer-based filter under a two-step prediction-correction structure. In contrast, the second approach considers a full-order filter $\mathcal{H}_\infty$, which guarantees better robust performance requirements than the previous one. In both methods, a convex optimization problem has to be solved at each time step to determine the filter parameters that estimate a differential wheeled robot's pose. This leads to a better solution to more complex estimation problems than the analytical way (finding feasible suboptimal solutions by solving the Riccati equation, as done in works [1,18]), allowing the model of the process and measurement's noise effects in a more general and robust way [22,23].

The notation used throughout this work is as follows: capital and small bold letters stand for matrices and vectors, respectively; the rest denote scalars. For vectors and matrices, $(^T)$ indicates transpose. For symmetric matrices, $\mathbf{P} > 0$ indicates that $\mathbf{P}$ is positive definite; similarly with $\mathbf{P} \geq 0$ denoting it as non-negative definite. For a transfer function $H(z)$ analytic for $|z| \geq 0$, $\left\|H(z)\right\|_2$ and $\left\|H(z)\right\|_\infty$ denote the standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms, correspondingly. For the sake of easing the notation of partitioned symmetric matrices, the symbol $\star$ indicates, generically, each of its symmetric blocks.

The rest of this paper is structured as follows: Section II elaborates on the discrete kinematics of a differential drive for two-wheel mobile robots. Besides, the laser-based measurement model is described, assuming a well-known method for associating correspondences between known landmarks and the output provided by a feature extraction algorithm. Section III proposes a discrete-time $\mathcal{H}_\infty$ observer to handle unknown measurements and process noises, as well as a method to determine its optimal and robust gain through the utilization of LMI methods. While Section IV proposes another LMI-based method to find a full order robust filter that guarantees a lower $\mathcal{H}_\infty$ norm. Finally, experimental results and conclusions are presented in Sections V and VI.

II. Problem Statement

Localization of mobile robots in a two-dimensional space requires a mathematical model to describe the robot's position as a rigid body on wheels moving across a horizontal plane. Hence, the discrete kinematic model of the differential drive for mobile robots is considered as follows [24].

$$\mathbf{x}_{k+1} = \begin{bmatrix} x_k + \Delta T v_k \cos\left(\theta_k + \Delta\omega_k/2\right) \\ y_k + \Delta T v_k \sin\left(\theta_k + \Delta\omega_k/2\right) \\ \theta_k + \Delta T \omega_k \end{bmatrix} \tag{1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the robot state vector at time k. Thus, $\mathbf{x}_k = [x_k,\ y_k,\ \theta_k]^{\mathrm{T}}$, where x_k and y_k are the Cartesian coordinates of the main axis midpoint between the two driving wheels and θ_k is the orientation of the robot respect to the inertial frame, while ΔT is the sampling period of the process. Besides, the input control vector $\mathbf{u}_k \in \mathbb{R}^p$ is defined as $\mathbf{u}_k = [v_k,\ \omega_k]^{\mathrm{T}}$ where v_k and ω_k are the linear and angular velocities of the robot, respectively.

In this way, the kinematic model described in the Eq. (1) can be rewritten in the following form

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \Delta T v_k \cos\left(\theta_k + \Delta\omega_k/2\right) \\ \Delta T v_k \sin\left(\theta_k + +\Delta\omega_k/2\right) \\ \Delta T \omega_k \end{bmatrix}$$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \tilde{f}(\mathbf{x}_k, \mathbf{u}_k) \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the process matrix and $\tilde{f}(\mathbf{x}_k, \mathbf{u}_k)$ represents the nonlinear effect of the control input that also depends on the orientation of the robot θ_k, as is shown in Equation (2). So, for the sake of simplicity in the notation, the nonlinear term will be denoted by the matrix $\mathbf{B}_{1,k}$. Also, it is essential to emphasize that this term does not affect the dynamic of the filters proposed in Sections III and IV.

Furthermore, a mathematical model of the sensor that acquires information about the environment is required as well. Thus, for this particular case, the 2-D laser-based measurement model was adopted as follows

$$\mathbf{y}_k = g(\mathbf{x}_k) = \begin{bmatrix} \sqrt{(x_m - x_k)^2 + (y_m - y_k)^2} \\ \arctan\left(\dfrac{y_m - y_k}{x_m - x_k}\right) - \theta_k \end{bmatrix} \tag{3}$$

where $\mathbf{y}_k \in \mathbb{R}^q$ is the measured output vector, $x_m \in \mathbb{R}$ and $y_m \in \mathbb{R}$ are the Cartesian coordinates of one landmark (for N landmarks, the length of the output vector has to be equal to $2N$). Furthermore, $\mathbf{y}_k = [\ \rho_k \quad \phi_k\ ]^{\mathrm{T}}$, where $\rho_k \in \mathbb{R}$ and $\phi_k \in \mathbb{R}$ are the euclidean distance and the angle from the robot position to a landmark. Besides, well-known methods for feature extraction (corners) and landmark correspondence are assumed [14]. However, occupied cells from an occupancy grid map can be considered as individual landmarks, and the correspondence with each laser measurement can be performed using a bi-linear interpolation [13].

Since the measurement model $g(\mathbf{x}_k)$ is nonlinear and time-variant, this expression can be expanded in a Taylor series about the operating point $\hat{\mathbf{x}}_k \in \mathbb{R}^n$ as follows

$$\mathbf{C}_k = \left.\begin{bmatrix} -\dfrac{(x_m - x_k)}{\rho_k} & -\dfrac{(y_m - y_k)}{\rho_k} & 0 \\ \dfrac{(y_m - y_k)}{\rho_k^2} & -\dfrac{(x_m - x_k)}{\rho_k^2} & -1 \end{bmatrix}\right|_{\mathbf{x}_k = \hat{\mathbf{x}}_k} \tag{4}$$

Thus, to solve the localization problem under the methods presented in the following Sections, it is necessary to linearize the measurement model $g(\mathbf{x}_k)$ at each time step k. This implies solving a convex optimization problem in each iteration of the algorithm, precisely a semi-definite programming problem via interior point methods [25].

III. Discrete-time $\mathcal{H}_\infty$ Full-State Observer

Throughout this section, a robust localization algorithm based on LMI methodologies is presented. In general, considering a linear time-invariant (LTI) system at a sampling time k where noise inputs corrupt the state and measured vectors, it is described as follows

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}_{1,k}\mathbf{u}_k + \mathbf{B}_2\mathbf{w}_k \tag{5}$$

$$\mathbf{y}_k = \mathbf{C}_k\mathbf{x}_k + \mathbf{D}_1\mathbf{v}_k + \mathbf{D}_2\mathbf{w}_k \tag{6}$$

where $\mathbf{w}_k \in \mathbb{R}^r$ and $\mathbf{v}_k \in \mathbb{R}^s$ are the process and measurement noise vectors, respectively. Besides, $\mathbf{B}_{1,k} \in \mathbb{R}^{n\times p}$ is the control input matrix at time k (which does not affect the error dynamic of the filter), $\mathbf{C} \in \mathbb{R}^{q\times n}$ is the measured output matrix and $\mathbf{B}_2 \in \mathbb{R}^{n\times r}$ is the process noise matrix, $\mathbf{D}_1 \in \mathbb{R}^{q\times s}$ and $\mathbf{D}_2 \in \mathbb{R}^{q\times r}$ are the process and sensor noise matrices affecting the output. So, by defining a general noise vector, $\tilde{\mathbf{w}}_k = [\mathbf{w}_k \;\; \mathbf{v}_k]^T$, a two step prediction-correction observer-based filter can be described by

$$\bar{\mathbf{x}}_k = f(\hat{\mathbf{x}}) = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}_{1,k}\mathbf{u}_k \tag{7}$$

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_k - \mathbf{K}(\mathbf{y}_k - \hat{\mathbf{y}}_k) \tag{8}$$

where $\bar{\mathbf{x}}_k \in \mathbb{R}^n$ is the predicted state and $\hat{\mathbf{x}}_k \in \mathbb{R}^n$ is the estimated one; $\hat{\mathbf{y}}_k \in \mathbb{R}^n$ is the estimated output and $\mathbf{K}$, the observer gain.

Since the initial conditions of the estimated state $\hat{\mathbf{x}}_\mathbf{0}$ are equal to those of the initial state $\mathbf{x}_0 = [0\ 0\ 0]^T$. Thus, from expressions (5)-(8), the filtering error dynamic is given by

$$\mathbf{e}_{k+1} = \mathbf{A}_\mathbf{o}\mathbf{e}_k + \mathbf{B}_\mathbf{o}\tilde{\mathbf{w}}_k \tag{9}$$

$$\tilde{\mathbf{y}}_k = \mathbf{C}_\mathbf{o}\mathbf{e}_k + \mathbf{D}_\mathbf{o}\tilde{\mathbf{w}}_k \tag{10}$$

with

$$\mathbf{A}_\mathbf{o} = \mathbf{A} + \mathbf{K}\mathbf{C}_k \qquad \mathbf{B}_\mathbf{o} = [\mathbf{B}_2 + \mathbf{K}\mathbf{D}_2 \quad \mathbf{K}\mathbf{D}_1]$$

$$\mathbf{C}_\mathbf{o} = \mathbf{C}_\mathbf{k} \qquad \mathbf{D}_\mathbf{o} = [\mathbf{D}_2 \quad \mathbf{D}_1]$$

The main goal is to find an optimal robust observer-based filter for the system composed by (5) and (6), where the error filtering, $\mathbf{e}_k$, has to satisfy that $\|\mathbf{e}_k\|_2 \leq \gamma(\|\mathbf{w}_k\|_2 + \|\mathbf{v}_k\|_2)$, with the robustness level $\gamma \in \mathbb{R}$ s.t. $\gamma > 0$. Therefore, from the bounded-real lemma and given the transfer function $H(z)$ in the complex frequency-domain, the norm $\mathcal{H}_\infty$ can be characterized using $\nu(\mathbf{x_k}) = \mathbf{x}_k^T\mathbf{P}\mathbf{x}_k$ as Lyapunov function [26], imposing that

$$\left\|H(z)\right\|_\infty < \gamma \Leftrightarrow \exists \mathbf{P} \in \mathbb{R}^{n\times n} \text{ s.t. } \mathbf{P} = \mathbf{P}^\mathbf{T} \geq 0 \tag{11}$$

where

$$\begin{bmatrix} \mathbf{P} & \mathbf{A}_\mathbf{o}^\mathbf{T}\mathbf{P} & \mathbf{0} & \mathbf{C}_\mathbf{o}^\mathbf{T} \\ \star & \mathbf{P} & \mathbf{P}\mathbf{B}_\mathbf{o} & \mathbf{0} \\ \star & \star & \mathbf{I}_r & \mathbf{D}_\mathbf{o}^\mathbf{T} \\ \star & \star & \star & \gamma^2\mathbf{I}_q \end{bmatrix} > \mathbf{0} \tag{12}$$

Hence, an observer meeting the requirements as mentioned above can be successfully established if a solution to the following convex optimization problem can be found

$$\min_{\mathbf{Z},\mathbf{P}=\mathbf{P}^\mathbf{T}>0} \gamma \tag{13}$$

which is subject to the following LMI

$$\begin{bmatrix} \mathbf{P} & \mathbf{A}^\mathbf{T}\mathbf{P} + \mathbf{C}_k^\mathbf{T}\mathbf{Z}^\mathbf{T} & \mathbf{0}_{n\times r} & \mathbf{0}_{n\times r} & \mathbf{C}_k^\mathbf{T} \\ \star & \mathbf{P} & \mathbf{P}\mathbf{B}_2 + \mathbf{Z}\mathbf{D}_2 & \mathbf{Z}\mathbf{D}_1 & \mathbf{0}_{n\times q} \\ \star & \star & \mathbf{I}_r & \mathbf{0}_{r\times r} & \mathbf{D}_2^\mathbf{T} \\ \star & \star & \star & \mathbf{I}_r & \mathbf{D}_1^\mathbf{T} \\ \star & \star & \star & \star & \gamma^2\mathbf{I}_q \end{bmatrix} > \mathbf{0} \tag{14}$$

where the matrices $\mathbf{Z} \in \mathbb{R}^{n\times q}$ and $\mathbf{P}$ are the variables of the problem [19]. Also, $\mathbf{K} \in \mathbb{R}^{n\times q}$ can be recovered using the following expression

$$\mathbf{K} = \mathbf{P}^{-1}\mathbf{Z} \tag{15}$$

To further improve the system's robustness, a slack variable $\mathbf{G} \in \mathbb{R}^{n\times n}$ can be incorporated such that

$$\min_{\mathbf{Z},\mathbf{G},\mathbf{P}=\mathbf{P}^\mathbf{T}>0} \gamma \tag{16}$$

subjected to the following LMI

$$\begin{bmatrix} \mathbf{P} & \mathbf{A}^\mathbf{T}\mathbf{G} + \mathbf{C}_k^\mathbf{T}\mathbf{Z}^\mathbf{T} & \mathbf{0}_{n\times r} & \mathbf{0}_{n\times r} & \mathbf{C}_k^\mathbf{T} \\ \star & \mathbf{G} + \mathbf{G}^\mathbf{T} - \mathbf{P} & \mathbf{G}^\mathbf{T}\mathbf{B}_2 + \mathbf{Z}\mathbf{D}_2 & \mathbf{Z}\mathbf{D}_1 & \mathbf{0}_{n\times q} \\ \star & \star & \mathbf{I}_r & \mathbf{0}_{r\times r} & \mathbf{D}_2^\mathbf{T} \\ \star & \star & \star & \mathbf{I}_r & \mathbf{D}_1^\mathbf{T} \\ \star & \star & \star & \star & \gamma^2\mathbf{I}_q \end{bmatrix} > \mathbf{0} \tag{17}$$

Moreover, since $\mathbf{G} + \mathbf{G}^\mathbf{T} > \mathbf{P} > 0$, it implies that $\mathbf{G}$ is non-singular [19], resulting in $\mathbf{K}$ being able to be recovered using the following expression

$$\mathbf{K} = (\mathbf{G}^\mathbf{T})^{-1}\mathbf{Z} \tag{18}$$

IV. Discrete-time $\mathcal{H}_\infty$ Filter

On the other hand, the problem of designing a full order robust filter (i.e., a $n = n_f$) is addressed. This method guarantees a lower robustness level γ, that upper bounds the maximum magnitude of the transfer function from the noise inputs to the estimation error. This way, considering an LTI discrete-time system at sampling time k (as shown in Fig. 1) in the following form

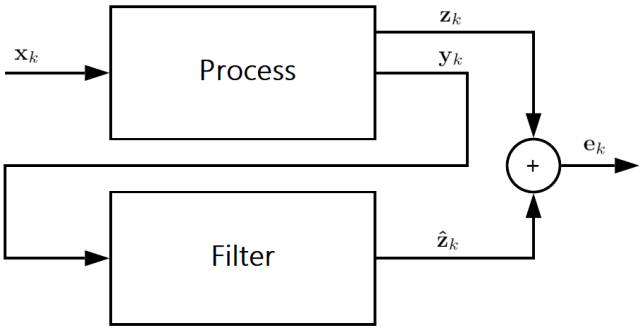

Fig. 1: Discrete-time system with a full order filter scheme.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}_{1,k}\mathbf{u}_k + \mathbf{B_2}\mathbf{w}_k \tag{19}$$

$$\mathbf{z}_k = \mathbf{C}_1\mathbf{x}_k + \mathbf{D}_{11}\mathbf{w}_k \tag{20}$$

$$\mathbf{y}_k = \mathbf{C}_{2,k}\mathbf{x}_k + \mathbf{D}_{21}\mathbf{v}_k + \mathbf{D}_{22}\mathbf{w}_k \tag{21}$$

where $\mathbf{z}_k \in \mathbb{R}^n$ is the output reference vector, $\mathbf{C}_1 \in \mathbb{R}^{p\times n}$ is the output reference matrix, $\mathbf{C}_{2,k} \in \mathbb{R}^{q\times n}$ is the measured output matrix, $\mathbf{D}_{11} \in \mathbb{R}^{p\times r}$ and $\mathbf{D}_{22} \in \mathbb{R}^{q\times r}$ are process noise matrices; and $\mathbf{D}_{21} \in \mathbb{R}^{q\times r}$ is the sensor noise matrix. Besides, $\mathbf{x}_k$, $\mathbf{y}_k$, $\mathbf{w}_k$, $\mathbf{v}_k$, $\mathbf{A}$ and $\mathbf{B}_2$ are defined as was done in expressions (5) and (6). Since the measurement model $g(\mathbf{x}_k)$ is nonlinear and time-variant, $\mathbf{C}_{2,k}$ needs to be equal to its Taylor series expansion $\mathbf{C}_k$.

Hence, using the general noise vector $\tilde{\mathbf{w}}_k$ and the performance requirements previously defined in Section III, the dynamic of the optimal guaranteed robust $\mathcal{H}_\infty$ filter can be described by

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A_f}\hat{\mathbf{x}}_k + \mathbf{B_f}\mathbf{y}_k \tag{22}$$

$$\hat{\mathbf{z}}_k = \mathbf{C_f}\hat{\mathbf{x}}_k + \mathbf{D_f}\mathbf{y}_k \tag{23}$$

where $\hat{\mathbf{x}}_k \in \mathbb{R}^{n_f}$ is the estimated state, $\hat{\mathbf{z}}_k \in \mathbb{R}^n$ is the estimated output. So, the matrices $\mathbf{A_f} \in \mathbb{R}^{n\times n}$, $\mathbf{B_f} \in \mathbb{R}^{n\times q}$, $\mathbf{C_f} \in \mathbb{R}^{p\times n}$ and $\mathbf{D_f} \in \mathbb{R}^{p\times q}$ are to be determined [20]. Thus, from expressions (19)-(23) and considering $\mathbf{e}_k = \mathbf{z}_k - \hat{\mathbf{z}}_k$ as well as the control input does not affect the dynamic of the filter, an augmented state dynamic is given by

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{A_a}\tilde{\mathbf{x}}_k + \mathbf{B_a}\tilde{\mathbf{w}}_k \tag{24}$$

$$\mathbf{e}_k = \mathbf{C_a}\tilde{\mathbf{x}}_k + \mathbf{D_a}\tilde{\mathbf{w}}_k \tag{25}$$

with

$$\mathbf{A_a} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{B_f}\mathbf{C}_2 & \mathbf{A_f} \end{bmatrix} \quad \mathbf{B_a} = \begin{bmatrix} \mathbf{B}_2 & \mathbf{0} \\ \mathbf{B_f}\mathbf{D}_{22} & \mathbf{D_f}\mathbf{D}_{21} \end{bmatrix}$$

$$\mathbf{C_a} = \begin{bmatrix} \mathbf{C}_1 - \mathbf{D_f}\mathbf{C}_2 & -\mathbf{C_f} \end{bmatrix}$$

$$\mathbf{D_a} = \begin{bmatrix} \mathbf{D}_{11} - \mathbf{D_f}\mathbf{D}_{22} & \mathbf{D_f}\mathbf{D}_{21} \end{bmatrix}$$

Therefore, the optimal guaranteed robust filter that satisfies the requirements described in Eq. 11, can be successfully characterized if and only if there exists a symmetric matrix $\mathbf{P} = \mathbf{P}^T > 0$ such that

$$\begin{bmatrix} \mathbf{P} & \mathbf{P}\mathbf{A_a} & \mathbf{P}\mathbf{B_a} & \mathbf{0} \\ \star & \mathbf{P} & \mathbf{0} & \mathbf{C_a^T} \\ \star & \star & \mathbf{I}_r & \mathbf{D_a^T} \\ \star & \star & \star & \gamma^2\mathbf{I}_q \end{bmatrix} > \mathbf{0} \tag{26}$$

In this way, following the same methodology applied in [20], where the matrices $\mathbf{P} \in \mathbb{R}^{2n\times 2n}$ and its inverse are partitioned into $n \times n$ blocks to convert the nonlinear matrix inequality into an LMI as follows

$$\mathbf{P} = \begin{bmatrix} \mathbf{X} & \mathbf{I}_n \\ \mathbf{I}_n & \tilde{\mathbf{X}} \end{bmatrix} \quad \mathbf{P}^{-1} = \begin{bmatrix} \mathbf{Y} & \mathbf{V^T} \\ \mathbf{V} & \tilde{\mathbf{Y}} \end{bmatrix}$$

The robust filter can be characterized if and only if there exist positive defined symmetric matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{n\times n}$, $\tilde{\mathbf{X}}, \tilde{\mathbf{Y}} \in \mathbb{R}^{n\times n}$, with $\mathbf{Z} = \mathbf{Y}^{-1}$; the matrix $\mathbf{V} \in \mathbb{R}^{n\times n}$ as well as the following matrices $\mathbf{F} \in \mathbb{R}^{p\times n}$, $\mathbf{G} \in \mathbb{R}^{n\times n}$, $\mathbf{B_f}$ and $\mathbf{D_f}$, that minimize the robustness level γ subjected to the following LMI

$$\begin{bmatrix} \mathbf{Z} & \mathbf{Z} & \mathbf{ZA} & \mathbf{ZA} \\ \star & \mathbf{X} & \mathbf{XA} + \kappa\mathbf{B_f}\mathbf{C}_2 + \mathbf{G} & \mathbf{XA} + \kappa\mathbf{B_f}\mathbf{C}_2 \\ \star & \star & \mathbf{Z} & \mathbf{Z} \\ \star & \star & \star & \mathbf{X} \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \star & \star & \star \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{ZB}_2 & \mathbf{0}_{n\times r} & \mathbf{0}_{n\times p} \\ \mathbf{XB}_2 + \kappa\mathbf{B_f}\mathbf{D}_{22} & \kappa\mathbf{D_f}\mathbf{D}_{21} & \mathbf{0}_{n\times p} \\ \mathbf{0}_{n\times r} & \mathbf{0}_{n\times r} & \mathbf{C}_1^\mathbf{T} - \mathbf{C}_2^\mathbf{T}\mathbf{D_f}^\mathbf{T} - \mathbf{F}^\mathbf{T} \\ \mathbf{0}_{n\times r} & \mathbf{0}_{n\times r} & \mathbf{C}_1^\mathbf{T} - \mathbf{C}_2^\mathbf{T}\mathbf{D_f}^\mathbf{T} \\ \mathbf{I}_r & \mathbf{0}_{r\times r} & \mathbf{D}_{11}^\mathbf{T} - \mathbf{D}_{22}^\mathbf{T}\mathbf{D_f}^\mathbf{T} \\ \star & \mathbf{I}_r & \mathbf{D}_{21}^\mathbf{T}\mathbf{D_f}^\mathbf{T} \\ \star & \star & \gamma^2\mathbf{I}_p \end{bmatrix} > \mathbf{0} \tag{27}$$

where $\kappa \in \mathbb{R}^n$ s.t. $\kappa > 0$ can be freely selected by the designer. Besides, the remaining matrices that describe the dynamics of the filter are given by

$$\mathbf{A_f} = \frac{1}{\kappa}\mathbf{G}(\mathbf{VZ})^{-1} \qquad \mathbf{C_f} = \mathbf{F}(\mathbf{VZ})^{-1} \tag{28}$$

with $\mathbf{V} = \frac{1}{\kappa}(\mathbf{I}_n - \mathbf{XZ}^{-1})$, defined to simplify the notation. Finally, it is important to note that matrices $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{F}$, $\mathbf{G}$, as well as the matrices $\mathbf{B_f}$ and $\mathbf{D_f}$ who also describe the dynamics of the filter are variables of the problem. The main difference with the methods presented in Section III is that unlike finding an observation gain we are finding a full order dynamic filter given by the matrices $\mathbf{A_f}$, $\mathbf{B_f}$, $\mathbf{C_f}$ and $\mathbf{B_f}$.

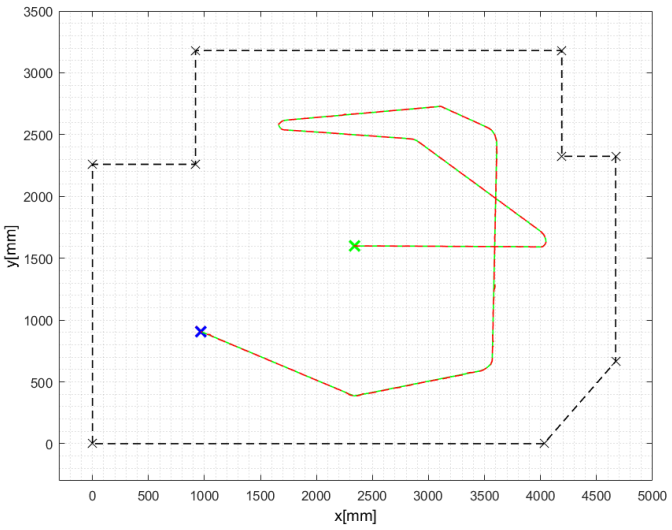

Fig. 2: Robot trajectory under a real controlled environment. Green cross is the robot's initial position, blue cross is the final estimated position, while green and doted red line are the ground-truth and the estimated trajectory, respectively.

V. Results

The experiments to test and validate the methods mentioned above were carried out using the Pioneer 3-DX robot from Adept MobileRobots in real and virtual scenarios. The real robot is equipped with a Sick LMS100 2D LiDAR, while the virtual model utilizes the fast Hokuyo 2D LiDAR available in the CoppeliaSim simulator [27]. In addition, although the algorithms were tested in a large environment using the simulator, a small static environment which is composed of flat surfaces (i.e., walls) was built as shown in Fig. 2, both in a real scenario and under the virtual robotic simulation framework. This environment was built within an area of 4.67×3.18 meters. Thus, nine landmarks (corners) were placed at Cartesian coordinates (x_m, y_m) in meters, as follows: $\mathcal{L} = \{(0.0, 0.0),\ (0.0, 2.26),\ (0.92, 2.26),\ (0.92, 3.18),\ (4.19, 3.18),\ (4.19, 2.325),\ (4.67, 2.325),\ (4.67, 0.665),\ (4.03, 0.0)\}$. This environment was more suitable to validate, evaluate, and compare both algorithms' performance concerning the EKF algorithm.

On the other hand, MATLAB, YALMIP and MOSEK [28]–[30] were used to implement the robust $\mathcal{H}_\infty$ the proposed localization methods, in addition to solve the convex optimization problems in terms subjected to the LMIs described in Eqs. (14), (17) and (27) via interior point methods [25]. This, with the aim to determine the parameters of the $\mathcal{H}_\infty$-based filters described in Eqs. (15), (18) and (28). Furthermore, successful results were obtained with both methodologies. However, since the results were quite similar, only the results obtained by the filter gain described by the Eq. (18) are shown in Fig. 2, where the solid green line illustrates the ground-truth trajectory of the robot, while the red dashed line illustrates its estimated position. The green and blue crosses represent the initial and final position of the robot, respectively. In contrast, the black dashed line and small crosses illustrate the static environment and the landmarks' real positions. Therefore, performance parameters for this scenario, such as the robot pose estimation errors for each iteration, as well as the approximated robustness level γ, are shown in Fig. 3. These results were obtained using the two proposed methods and the EKF from a data collection of 500 samples at a sampling frequency of about $10\ Hz$. Each sample is composed of the control input vector $\mathbf{u}_k$ and a measurement vector with the 241 laser measurements at time k. As was previously mentioned, well-known feature extraction (corners) and landmark correspondence algorithms were employed. Each solution's complexity depends on the number of detected landmarks in the environment, which was reflected in the number of variables to solve in each iteration. However, both proposed methods provide convergence even if only two landmarks are observed at any time-step. For the first proposed method, the amount of variables oscillates between 34 and 82 at each iteration, and its behavior is described in the same way as the Fig. 3(b) for the second method, the number of variables remain constant (40). In all the tests, the observer-based filter gain matrix and the dynamic filter matrices were calculated below a time interval of approximately $0.1\ s$.

Furthermore, both methodologies present small state estimation errors. The Fig. 3(a) shows the state estimation error from the observer-based filter using slack variables, as proposed in Section III and tested under a real scenario; while the Fig. 3(c) presents the estimation error from the EKF, also tested under a real scenario. The root mean square error (RMSE) for each of the states are $[1.645\ mm, 1.482\ mm, 1.874°]^{\mathrm{T}}$ and $[6.339\ mm, 4.363\ mm, 4.311°]^{\mathrm{T}}$, respectively. For the method proposed in Section IV, the RMSE is about $[1.213\ mm, 1.028\ mm, 0.754°]^{\mathrm{T}}$, showing a better result than the previous methods.

Finally, as it is shown in Figs. 3(b) and 3(d) the approximated value of γ depends on the amount of landmarks detected. This, because the length of the measured output matrix $\mathbf{C}_k$ depends directly on the number of variables involved in the convex optimization problem.

VI. Conclusions

Both theoretical remarks for landmark-based localization approaches were successfully validated for small and static environments, as shown in Fig. 2. More extensive performance tests have to be done to validate its functionality in large environments. However, the methodologies' performance was almost unaffected when a smaller amount of landmarks was detected since both filters converged even when only two landmarks were observed. Nevertheless, the computational load seems expensive for large amounts of landmarks, for example, when using all laser measurements as landmarks (polynomial-time complexity, at least $\mathcal{O}(n^6)$ for general purpose solvers) [31,32]. The computational load rises due to the linearization step through Taylor series expansion about an operating point, which implies solving an optimization

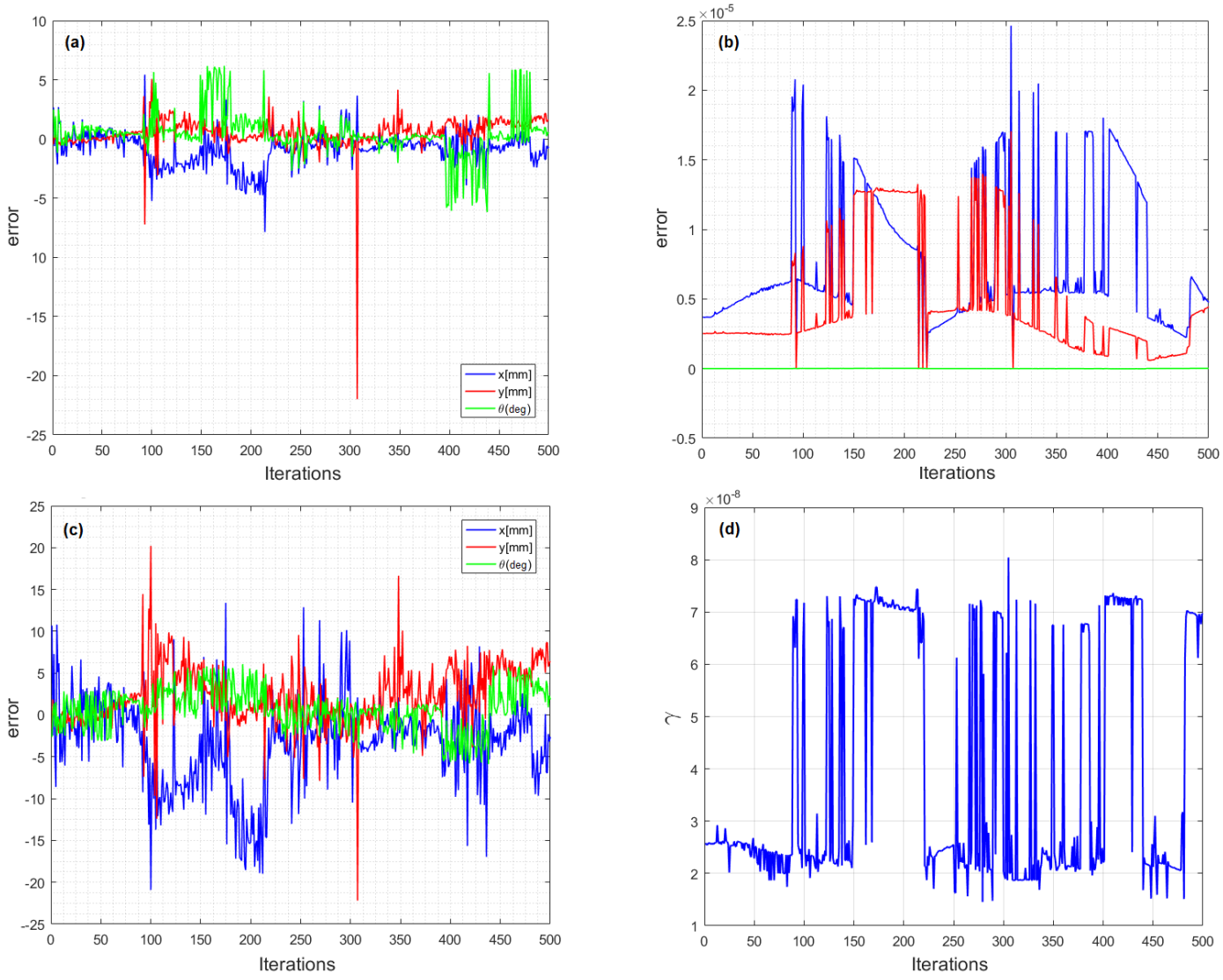

Fig. 3: Robot pose estimation error throughout each iteration using the method proposed in Section III under a real scenario, and using the method proposed in Section IV under the CoppeliaSim environment are shown in sub-figures (a) and (b). (c) Robot pose estimation error throughout each iteration using the Extended Kalman Filter (EKF) under a real scenario. (d) Robustness level γ at each iteration using the method proposed in Section IV and under the CoppeliaSim environment.

problem in each algorithm's iteration. However, both methods present good results by only observing two landmarks. It is also interesting to explore other linearization methodologies. This implies modeling the non-linearities as linear parameter-varying (LPV) systems or even using Takagi-Tsugeno fuzzy models to reduce the computational load but also to determine better LMIs characterizations to solve this specific problem in a better way. Therefore, the problem's complexity could become a more challenging problem, but the filter parameters would not have to be calculated at each iteration as be done. On the other hand, as shown in Fig. 3, the second approach guaranteed a lower RMSE and norm than the first one, providing encouraging results for future large scale implementations, validation with other types of sensors and kinematic models; and for possible experimentation by merging with other methodologies. Therefore, the full-order robust filter performed better than the other two compared methods, not only in terms of RMSE but also for the level of robustness achieved. This means that the upper-bound that limits the energy gain from the noise inputs to the estimation error ratio is lower than the others, guaranteeing better performance and robustness (more conservative), as no assumptions were made about the noise. Finally, since LMI-based methodologies have been growing during the last decades and the computational power needed to handle this approach, reliable and user-friendly LMI machinery to solve the convex optimization problems related to robust filtering are needed in order to get better solutions for large scale problems.

REFERENCES

[1] F. Yang, Z. Wang, S. Lauria, and X. Liu, "Mobile robot localization using robust extended H_∞ filtering," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 223, no. 8, pp. 1067–1080, 2009.

[2] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Transactions on robotics and automation*, vol. 12, no. 6, pp. 869–880, 1996.

[3] E. Kiriy and M. Buehler, "Three-state extended kalman filter for mobile robot localization," *McGill University., Montreal, Canada, Tech. Rep. TR-CIM*, vol. 5, p. 23, 2002.

[4] C. M. Wang, "Location estimation and uncertainty analysis for mobile robots," in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*. IEEE, 1988, pp. 1231–1235.

[5] M. Betke and L. Gurvits, "Mobile robot localization using landmarks," *IEEE transactions on robotics and automation*, vol. 13, no. 2, pp. 251–263, 1997.

[6] L. Jetto, S. Longhi, and G. Venturini, "Development and experimental validation of an adaptive extended kalman filter for the localization of mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 219–229, 1999.

[7] L. Jetto, S. Longhi, and D. Vitali, "Localization of a wheeled mobile robot by sensor data fusion based on a fuzzy logic adapted kalman filter," *Control Engineering Practice*, vol. 7, no. 6, pp. 763–771, 1999.

[8] E. Fabrizi, G. Oriolo, S. Panzieri, and G. Ulivi, "A KF-based localization algorithm for nonholonomic mobile robots," in *Theory and Practice of Control and Systems*. World Scientific, 1998, pp. 130–135.

[9] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *IEE proceedings F (radar and signal processing)*, vol. 140, no. 2. IET, 1993, pp. 107–113.

[10] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American statistical association*, vol. 94, no. 446, pp. 590–599, 1999.

[11] D. Fox, W. Burgard, and S. Thrun, "Markov localization for mobile robots in dynamic environments," *Journal of artificial intelligence research*, vol. 11, pp. 391–427, 1999.

[12] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte carlo localization for mobile robots," *Artificial intelligence*, vol. 128, no. 1-2, pp. 99–141, 2001.

[13] S. Kohlbrecher, O. Von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation," in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*. IEEE, 2011, pp. 155–160.

[14] C.-C. Peng, Y.-T. Wang, and C.-L. Chen, "LIDAR based scan matching for indoor localization," in *2017 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2017, pp. 139–144.

[15] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2d range scans," *Journal of Intelligent and Robotic systems*, vol. 18, no. 3, pp. 249–275, 1997.

[16] P. Biber and W. Straßer, "The normal distributions transform: A new approach to laser scan matching," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, vol. 3. IEEE, 2003, pp. 2743–2748.

[17] P. Sodhi, S. Choudhury, J. G. Mangelson, and M. Kaess, "Ics: Incremental constrained smoothing for state estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 279–285.

[18] H. Hur and H.-S. Ahn, "Discrete-time H_∞ filtering for mobile robot localization using wireless sensor network," *IEEE Sensors Journal*, vol. 13, no. 1, pp. 245–252, 2012.

[19] M. C. De Oliveira, J. C. Geromel, and J. Bernussou, "Extended H_2 and H_∞ norm characterizations and controller parametrizations for discrete-time systems," *International Journal of Control*, vol. 75, no. 9, pp. 666–679, 2002.

[20] J. C. Geromel, J. Bernussou, G. Garcia, and M. C. de Oliveira, "H_2 and H_∞ robust filtering for discrete-time linear systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 5, pp. 1353–1368, 2000.

[21] S. Boyd, V. Balakrishnan, E. Feron, and L. ElGhaoui, "Control system analysis and synthesis via linear matrix inequalities," in *1993 American Control Conference*. IEEE, 1993, pp. 2147–2154.

[22] U. Shaked and Y. Theodor, "H_∞-optimal estimation: a tutorial," in *[1992] Proceedings of the 31st IEEE Conference on Decision and Control*. IEEE, 1992, pp. 2278–2286.

[23] K. M. Grigoriadis and J. T. Watson, "Reduced-Order H_∞ and $L_2 - L_\infty$ filtering via linear matrix inequalities," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1326–1338, 1997.

[24] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*. MIT press, 2011.

[25] A. Nemirovskii and P. Gahinet, "The projective method for solving linear matrix inequalities," in *Proceedings of 1994 American Control Conference-ACC'94*, vol. 1. IEEE, 1994, pp. 840–844.

[26] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. Siam, 1994, vol. 15.

[27] E. Rohmer, S. P. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1321–1326.

[28] M. ApS, "The MOSEK optimization toolbox for MATLAB," *Users Guide and Reference Manual, version*, vol. 4, 2019.

[29] E. D. Andersen and K. D. Andersen, "The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm," in *High performance optimization*. Springer, 2000, pp. 197–232.

[30] J. Lofberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *2004 IEEE international conference on robotics and automation (IEEE Cat. No. 04CH37508)*. IEEE, 2004, pp. 284–289.

[31] P. Gahinet, A. Nemirovskii, A. J. Laub, and M. Chilali, "The LMI control toolbox," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, vol. 3. IEEE, 1994, pp. 2038–2041.

[32] L. Vandenberghe, V. R. Balakrishnan, R. Wallin, A. Hansson, and T. Roh, "Interior-point algorithms for semidefinite programming problems derived from the KYP lemma," in *Positive polynomials in control*. Springer, 2005, pp. 195–238.