H_∞-Optimal Tracking Controller for Three-Wheeled Omnidirectional Mobile Robots with Uncertain Dynamics

Amir Salimi Lafmejani, Hamed Farivarnejad, and Spring Berman

***Abstract*— In this paper, we present an optimal control approach using Linear Matrix Inequalities (LMIs) for trajectory tracking control of a three-wheeled omnidirectional mobile robot in the presence of external disturbances on the robot's actuators and noise in the robot's sensor measurements. First, a state-space representation of the omnidirectional robot dynamics is derived using a point-mass dynamic model. Then, we propose an LMI-based full-state feedback H_∞-optimal controller for the tracking problem. The robot's tracking performance with the H_∞-optimal controller is compared to its performance with a classical full-state feedback tracking controller in simulations with circular and bowtie-shaped reference trajectories. In order to evaluate our proposed controller in practice, we also implement the H_∞-optimal and classical controllers for these reference trajectories on a three-wheeled omnidirectional robot. The H_∞-optimal controller guarantees stabilization of the robot motion and attenuates the effects of frictional disturbances and measurement noise on the robot's tracking performance. Using the H_∞-optimal controller, the robot is able to track the reference trajectories with up to a 47.8% and 45.8% decrease in the maximum pose and twist errors, respectively, over a full cycle of the trajectory compared to the classical controller. The simulation and experimental results show that our LMI-based H_∞-optimal controller is robust to undesired effects of disturbances and noise on the dynamic behavior of the robot during trajectory tracking and can outperform the classical controller in attenuating their effects.**

I. Introduction

Mobile robots are robotic systems that are capable of moving in their environment, such as Unmanned Aerial Vehicles (UAVs) [1], Autonomous Underwater Vehicles (AUVs) [2], legged mobile robots [3], and Wheeled Mobile Robots (WMRs) [4]. WMRs in particular are widely used in different applications, such as autonomous driving [5], mining and inspection [6], and space exploration. Depending on the application, WMRs may be designed with different types of wheeled drive systems such as two-wheeled differential-drive, car-like, and omni-directional. The kinematics of two-wheeled and car-like robots are described by the unicycle model [4]. In this model, there exists a constraint along the direction of wheels' axis, which is called a nonholonomic constraint and does not allow the robot to produce controlled motions along this direction. Due to this nonholonomic constraint, WMRs whose model conforms with the unicycle model are not linearly controllable and cannot be stabilized to the origin by a continuous smooth time-invariant feedback control law [7]. Given the complexity of designing controllers for nonholonomic WMRs, there has been considerable attention to the use of holonomic WMRs such as three- or four-wheeled omnidirectional mobile robots. The design of omnidirectional robots' wheels (omni and Mecanum wheels) provide these type of WMRs with the capability of moving from any arbitrary configuration to any other arbitrary configuration [8]. This in turn enables the design of linear time-invariant control laws for omnidirectional WMRs.

A linear time-invariant state-feedback control law could be a promising control strategy for an omnidirectional WMR only when an exact dynamical model of the robot is available and no disturbances or sensor noise are present. However, disturbances and noise, known as undesired exogenous inputs, can significantly affect the controlled performance of the robot in practice. These exogenous inputs can originate from different sources, such as the friction force between the robot's wheels and the ground, actuator inaccuracy, and noise in sensor measurements. Therefore, a proper control design for real-world applications should be robust enough to reject or at least attenuate the undesired effects of exogenous inputs. Optimal control approaches, such as H_2 and H_∞ controllers, employ properties of Linear Matrix Inequalities (LMIs) in order to produce control laws with these characteristics. Moreover, these optimal controllers guarantee asymptotic stability of the closed-loop system [9], with a robustness margin that depends on the type and magnitude of parametric uncertainties. By minimizing the H_2-norm and/or H_∞-norm of the closed-loop system, these controllers reduce and potentially minimize the effects of disturbances and noise on the robot's performance [10], [11]. Several control approaches that employ the properties of LMIs have been recently developed for the control of WMRs, e.g. [12], [13].

In order to achieve a desired closed-loop behavior, an exact dynamic model of the omnidirectional WMR and the environment in which it is moving is of significant importance [14], [15]. Although it is possible to derive an inexact dynamic model of the robot and environment, the resulting controller often fails to produce the desired performance in real-world implementations due to uncertainties in the model. These uncertainties include inaccuracies in identification of the aforementioned friction forces, parameters of the driving motors, the robot's inertia matrix, and the position of the robot's center of mass, all of which are necessary to account for in dynamic modeling of the robot [16], [17].

This work was supported by the Arizona State University Global Security Initiative.

A. Salimi Lafmejani is with the School of Electrical, Computer and Energy Engineering, Arizona State University (ASU), Tempe, AZ, 85287 asalimil@asu.edu.

H. Farivarnejad and S. Berman are with School for Engineering of Matter, Transport and Energy, ASU, Tempe, AZ 85287 {hamed.farivarnejad, spring.berman}@asu.edu.

In this paper, we propose an LMI-based control approach, in the form of a full-state feedback H_∞-optimal controller, for trajectory tracking control of a three-wheeled omnidirectional WMR that is subject to frictional disturbances and sensor noise. In order for the control approach to be implementable on any three-wheeled omnidirectional WMR, we design the controller using a point-mass model of the robot dynamics. Our proposed control approach **(1)** does not require a rigid-body dynamical model of the omnidirectional robot, and **(2)** is robust to uncertainties originating from undesired exogenous inputs on the robot. In Section II, we first derive the state-space representation of the dynamic model based on a point-mass model of the robot. Then, we convert this model into the standard LMI framework required for designing a full-state feedback H_∞-optimal controller. In Section III, we describe both a classical full-state feedback controller and a full-state feedback H_∞-optimal controller for the trajectory tracking problem. In Section IV, we present results from simulations of a three-wheeled omnidirectional robot that tracks circular and bowtie-shaped reference trajectories using both the classical and H_∞ controllers while experiencing the effects of disturbances and measurement noise. We also implement the classical and H_∞ controllers on a real three-wheeled omnidirectional robot in order to validate the effectiveness of our proposed controller in practice. In Section V, we describe the odometry, twist, and torque calculations required for implementation of the controller and provide experimental results on the robot's performance at tracking the circular and bowtie trajectories.

II. State-Space Model of Robot Dynamics

Figure 1 shows a schematic of a three-wheeled omnidirectional robot equipped with omni wheels, which allow the robot move at any direction at each time instant. This holonomic WMR can be modeled as a point-mass with double-integrator dynamics as follows, without including disturbances and noise:

$$\mathbf{M}\ddot{\mathbf{x}} = \mathbf{u}, \tag{1}$$

where $\mathbf{M}_{3\times3} = \text{diag}(m, m, I)$ is a diagonal mass matrix in which m and I are defined as the robot's mass and moment of inertia about the axis perpendicular to the ground, respectively. We assume that m is a known constant and I can be calculated as the moment of inertia of a solid cylinder with radius d as $I = \frac{1}{2}md^2$. As shown in Fig. 1, the variables ${}^G x$ and ${}^G y$ are the position coordinates of the robot's center of mass along the $\bar{x}$-axis and $\bar{y}$-axis, and φ is the robot's heading angle. The vector $\mathbf{x} = [{}^G x, \quad {}^G y, \quad \varphi]^{\mathrm{T}} \in \mathbb{R}^3$, which is defined in the global coordinate frame $\bar{x} - \bar{y}$, indicates the pose of the robot in a planar configuration. Moreover, $\mathbf{u} = [{}^G f_x, \quad {}^G f_y, \quad {}^G \tau_z]^{\mathrm{T}} \in \mathbb{R}^3$ is the control input vector, which is interpreted as the wrench (forces and torque) applied to the robot's body. We define the state variables as follows:

$$\mathbf{x}_1 = [{}^G x \quad {}^G y \quad \varphi]^{\mathrm{T}}, \quad \mathbf{x}_2 = [{}^G \dot{x} \quad {}^G \dot{y} \quad \dot{\varphi}]^{\mathrm{T}}. \tag{2}$$

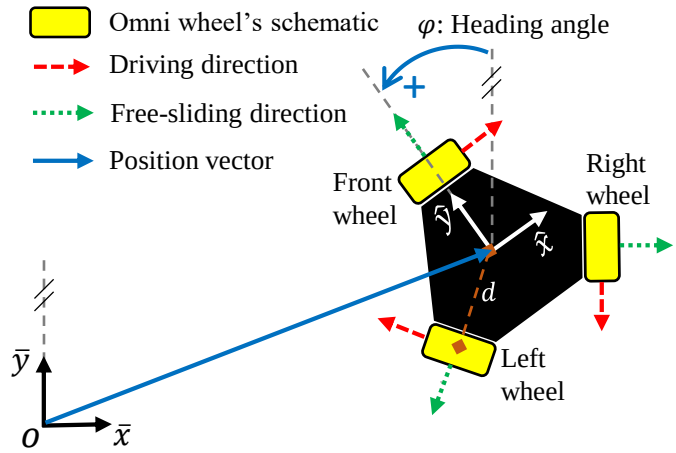


Fig. 1: Schematic of a three-wheeled omnidirectional WMR, illustrating the driving and free-sliding directions of the robot's wheels.

Then, one can readily write the state-space representation of the robot dynamics in Eq. (1) as:

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2, \quad \dot{\mathbf{x}}_2 = \mathbf{M}^{-1}\mathbf{u}. \tag{3}$$

The vectors $\mathbf{X}_{6\times1} := [\mathbf{x}_1^{\mathrm{T}} \quad \mathbf{x}_2^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^6$ and $\dot{\mathbf{X}}_{6\times1} := [\dot{\mathbf{x}}_1^{\mathrm{T}} \quad \dot{\mathbf{x}}_2^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^6$ are defined to characterize the state-space representation of the robot's dynamics as:

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{A}_p\mathbf{X} + \mathbf{B}_p\mathbf{u} \\ \mathbf{y}_p &= \mathbf{C}_p\mathbf{X} + \mathbf{D}_p\mathbf{u}, \end{aligned} \tag{4}$$

where $\mathbf{y}_p$ denotes the output and the four matrices $\mathbf{A}_p$, $\mathbf{B}_p$, $\mathbf{C}_p$, and $\mathbf{D}_p$ for the trajectory tracking control problem are given by:

$$\begin{aligned} &\mathbf{A}_p = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}, \quad \mathbf{B}_p = \mathbf{M}^{-1}\begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} \end{bmatrix} \\ &\mathbf{C}_p = \mathbf{I}_{6\times6}, \qquad \mathbf{D}_p = \mathbf{0}_{6\times3}. \end{aligned} \tag{5}$$

Since we are designing a full-state feedback controller, the output $\mathbf{y}_p$ is defined as the state vector $\mathbf{X}$. Note that $\mathbf{I}$ denotes the identity matrix and $\mathbf{0}$ denotes a matrix of zeros.

Figure 2 shows the block diagram of the proposed tracking controller when actuation disturbances and output measurement noise are included in the robot model [18]. The plant in Fig. 2 consists of the state-space model of the robot's dynamics without undesired exogenous inputs, described in Eq. (4). We can expand the state-space model to include the undesired effects of disturbances on the robot, arising from the friction force between the omni wheels and the ground, and noise in the sensor measurements. To do so, we formulate the model as a tracking control problem in the LMI framework, using the notation described in [11]. We start by defining the reference trajectory, $\mathbf{r}_{6\times1}$, the disturbance input, $\mathbf{d}_{3\times1}$, and the output measurement noise, $\mathbf{n}_{6\times1}$. The error, $\mathbf{e}_{6\times1}$, between the tracked trajectory and the reference trajectory is defined as $\mathbf{z}_1$. In addition, $\mathbf{y}_1$ and $\mathbf{y}_2$ are the reference trajectory and the noisy output, respectively. Thus,

we have that:

$$\begin{aligned} &\mathbf{w} := [\mathbf{r}^{\mathrm{T}} \quad \mathbf{d}^{\mathrm{T}} \quad \mathbf{n}^{\mathrm{T}}]^{\mathrm{T}}, \quad \mathbf{z}_1 := \mathbf{e} = \mathbf{r} - \mathbf{y}_p, \quad \mathbf{z}_2 := \mathbf{u} \\ &\mathbf{y}_1 := \mathbf{r}, \quad \mathbf{y}_2 := \mathbf{n} + \mathbf{y}_p. \end{aligned} \tag{6}$$

As a result, the state-space representation of the robot dynamics for the trajectory tracking problem, including disturbances and sensor noise, can be written as:

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{A}_p\mathbf{X} + [\mathbf{0} \quad \mathbf{B}_p \quad \mathbf{0}]\mathbf{w} + \mathbf{B}_p\mathbf{u} \\ \mathbf{z}_1 &= -\mathbf{C}_p\mathbf{X} + [\mathbf{I} \quad -\mathbf{D}_p \quad \mathbf{0}]\mathbf{w} - \mathbf{D}_p\mathbf{u} \\ \mathbf{z}_2 &= \mathbf{0}\mathbf{X} + [\mathbf{0} \quad \mathbf{0} \quad \mathbf{0}]\mathbf{w} + \mathbf{I}\mathbf{u} \\ \mathbf{y}_1 &= \mathbf{0}\mathbf{X} + [\mathbf{I} \quad \mathbf{0} \quad \mathbf{0}]\mathbf{w} + \mathbf{0}\mathbf{u} \\ \mathbf{y}_2 &= \mathbf{C}_p\mathbf{X} + [\mathbf{0} \quad \mathbf{D}_p \quad -\mathbf{I}]\mathbf{w} + \mathbf{D}_p\mathbf{u}. \end{aligned} \tag{7}$$

The realization in Eq. (7) results in the following 9-matrix representation for the LMI-based tracking controller:

$$\begin{aligned} &\mathbf{A} := \mathbf{A}_p, \quad \mathbf{B}_1 := \begin{bmatrix}\mathbf{0} & \mathbf{B}_p & \mathbf{0}\end{bmatrix}, \quad \mathbf{B}_2 := \mathbf{B}_p \\ &\mathbf{C}_1 := \begin{bmatrix}-\mathbf{C}_p & \mathbf{0}\end{bmatrix}^{\mathrm{T}} \quad \mathbf{C}_2 := \begin{bmatrix}\mathbf{0} & \mathbf{C}_p\end{bmatrix}^{\mathrm{T}} \\ &\mathbf{D}_{11} := \begin{bmatrix}\mathbf{I} & -\mathbf{D}_p & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix} \quad \mathbf{D}_{12} := \begin{bmatrix}-\mathbf{D}_p & \mathbf{I}\end{bmatrix}^{\mathrm{T}} \\ &\mathbf{D}_{21} := \begin{bmatrix}\mathbf{I} & -\mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_p & -\mathbf{I}\end{bmatrix} \quad \mathbf{D}_{22} := \begin{bmatrix}\mathbf{0} & \mathbf{D}_p\end{bmatrix}^{\mathrm{T}}. \end{aligned} \tag{8}$$

III. Trajectory Tracking Controller Design

In the trajectory tracking control problem, the three-wheeled omnidirectional robot should follow predefined reference trajectories that are described by desired poses (position and orientation) and twists (linear and angular velocities) of the robot with respect to time. In this section, we first describe a classical full-state feedback controller for this problem, and then propose a full-state feedback H_∞-optimal controller.

A. Classical Full-State Feedback Controller

Given the point-mass dynamic model for the omnidirectional robot in Eq. (1), define $\mathbf{K} \in \mathbb{R}^{6\times 6}$ as a positive definite gain matrix and $\ddot{\mathbf{x}}_r$ as the desired acceleration of the reference trajectory, including linear and angular accelerations. Referring to [19], the control law for the classical controller is described by:

$$\mathbf{u} = -\mathbf{S} + \ddot{\mathbf{x}}_r, \tag{9}$$

where $\mathbf{S} \in \mathbb{R}^{6\times 1}$ is defined as follows:

$$\mathbf{S} = \mathbf{K}\dot{\mathbf{e}} + \mathbf{\Lambda}\mathbf{e}, \tag{10}$$

in which $\mathbf{\Lambda} \in \mathbb{R}^{6\times 6}$ is a positive definite gain matrix. Here, the vector $\mathbf{e} \in \mathbb{R}^{6\times 1}$ denotes the error between the measured pose of the robot, $\mathbf{x}$, and the reference pose, $\mathbf{x}_r$. In addition, the vector $\dot{\mathbf{e}} \in \mathbb{R}^{6\times 1}$ denotes the error between the measured twist of the robot, $\dot{\mathbf{x}}$, and the reference twist, $\dot{\mathbf{x}}_r$. Accordingly, we have that:

$$\mathbf{e} = \mathbf{x}_1 - \mathbf{x}_r, \quad \dot{\mathbf{e}} = \mathbf{x}_2 - \dot{\mathbf{x}}_r. \tag{11}$$

Substituting Eq. (10) into Eq. (9) and using the fact that $\ddot{\mathbf{e}} = \ddot{\mathbf{x}} - \ddot{\mathbf{x}}_r$, we can obtain the equation of the error dynamics

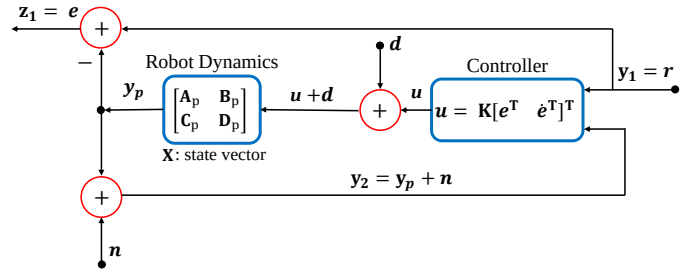


Fig. 2: Block diagram of the tracking controller, including disturbances and sensor noise as input signals.

for the closed-loop system as a second-order differential equation with positive definite matrix coefficients:

$$\ddot{\mathbf{e}} + \mathbf{K}\dot{\mathbf{e}} + \mathbf{\Lambda}\mathbf{e} = \mathbf{0}. \tag{12}$$

In [20], it was proved that a differential equation of the form Eq. (12) is stable if both matrix coefficients are positive definite. This proves that the pose and twist errors will converge to zero.

B. Full-State Feedback H_∞-Optimal Controller

The input-output equations in the state-space model of the robot dynamics in Eq. (7) can be written in an abbreviated form as follows [9], where $\mathbf{z} := [\mathbf{z}_1^{\mathrm{T}} \quad \mathbf{z}_2^{\mathrm{T}}]^{\mathrm{T}}$:

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{A}\mathbf{X} + \mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{u} \\ \mathbf{z} &= \mathbf{C}_1\mathbf{x} + \mathbf{D}_{11}\mathbf{w} + \mathbf{D}_{12}\mathbf{u}. \end{aligned} \tag{13}$$

We propose the following control law for the H_∞-optimal controller:

$$\mathbf{u} = -\mathbf{K}_{H_\infty}[\mathbf{e}^{\mathrm{T}} \quad \dot{\mathbf{e}}^{\mathrm{T}}]^{\mathrm{T}} + \ddot{\mathbf{x}}_r \tag{14}$$

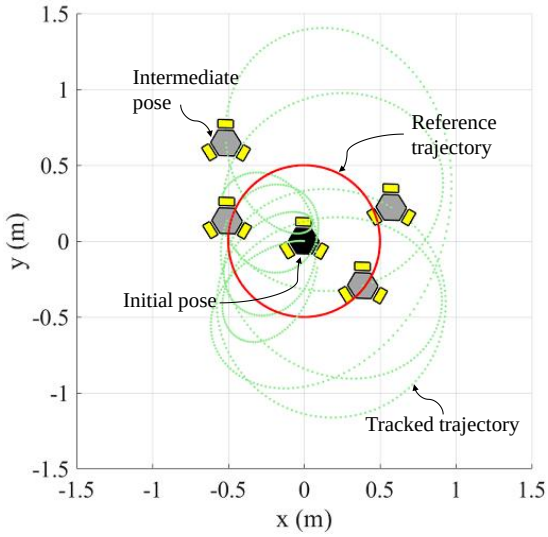
where the controller gain matrix, $\mathbf{K}_{H_\infty}$, is determined such that the H_∞-norm of the closed-loop system is minimized when the exogenous inputs are applied to the robot. The closed-loop system represented in state-space form and its transfer function, $\mathbf{T}(s) = \mathbf{z}(s)/\mathbf{w}(s)$, are described by:

$$\begin{aligned} \dot{\mathbf{X}} &= (\mathbf{A} + \mathbf{B}_2\mathbf{K}_{H_\infty})\mathbf{X} + \mathbf{B}_1\mathbf{w} \\ \mathbf{z} &= (\mathbf{C}_1 + \mathbf{D}_{12}\mathbf{K}_{H_\infty})\mathbf{X} + \mathbf{D}_{11}\mathbf{w} \\ \mathbf{T}(s) &= (\mathbf{C}_1 + \mathbf{D}_{12}\mathbf{K}_{H_\infty})(s\mathbf{I} - (\mathbf{A} + \mathbf{B}_2\mathbf{K}_{H_\infty}))^{-1}\mathbf{B}_1 \\ &\quad + \mathbf{D}_{11} \end{aligned} \tag{15}$$

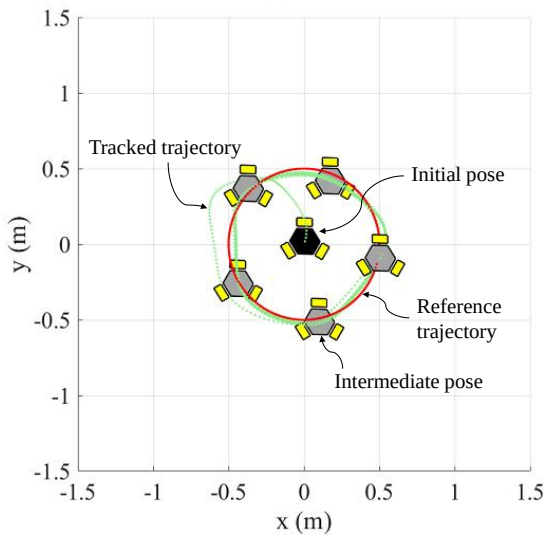
Using the Bounded Real Lemma [9], [11], it can be shown that γ is the minimum value of the H_∞-norm of the closed-loop system if there exist a positive definite matrix $\mathbf{P}$ and a matrix $\mathbf{F}$ which satisfy the LMI in the following optimization problem (here, "$*$" represents a symmetric element of a matrix):

$$\begin{aligned} &\min \quad \gamma \\ &\mathbf{P} > \mathbf{0} \\ &\begin{bmatrix} \mathbf{A}\mathbf{P} + \mathbf{B}_2\mathbf{F} + (\mathbf{A}\mathbf{P} + \mathbf{B}_2\mathbf{F})^{\mathrm{T}} & \mathbf{B}_1 & (\mathbf{C}_1\mathbf{P} + \mathbf{D}_{12}\mathbf{F})^{\mathrm{T}} \\ *^{\mathrm{T}} & -\gamma\mathbf{I} & \mathbf{D}_{11}^{\mathrm{T}} \\ *^{\mathrm{T}} & *^{\mathrm{T}} & -\gamma\mathbf{I} \end{bmatrix} < \mathbf{0} \end{aligned} \tag{16}$$

Given such matrices $\mathbf{P}$ and $\mathbf{F}$, we can calculate the H_∞-optimal controller gain matrix as $\mathbf{K}_{H_\infty} = \mathbf{F}\mathbf{P}^{-1}$.



(a)



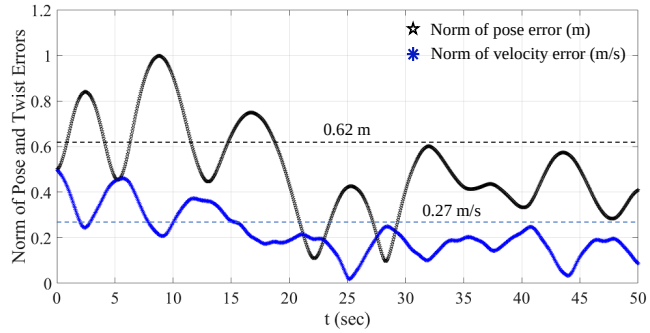
(b)

Fig. 3: Trajectory of a simulated three-wheeled omnidirectional robot that tracks the circular reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

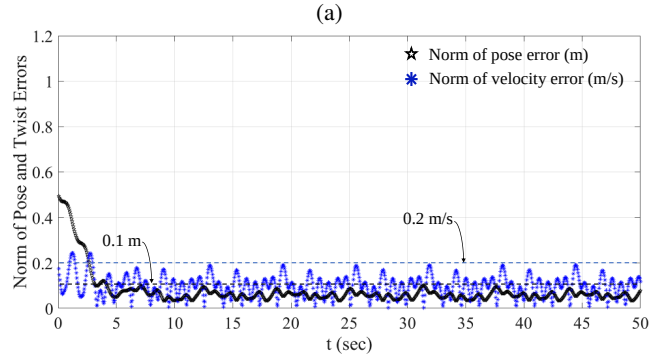
IV. Simulation Results

In this section, we implement the controllers described in Section III on a simulated three-wheeled omnidirectional robot in MATLAB®. We compare the performance of the H_∞-optimal controller and the classical controller for two reference trajectories: a circular reference trajectory described by:

$$\begin{aligned}\mathbf{x}_{r,c} &= [2+\cos(t) \quad 2+\sin(t) \quad 0]^{\mathrm{T}}\\ \dot{\mathbf{x}}_{r,c} &= [-\sin(t) \quad \cos(t) \quad 0]^{\mathrm{T}}\\ \ddot{\mathbf{x}}_{r,c} &= [-\cos(t) \quad -\sin(t) \quad 0]^{\mathrm{T}},\end{aligned} \tag{17}$$



(a)



(b)

Fig. 4: Norm of the pose and twist errors over 50 s of the simulations in which the robot tracks the circular reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

and a bowtie-shaped reference trajectory described by:

$$\begin{aligned}\mathbf{x}_{r,b} &= [0.5\cos(t) \quad 0.5\sin(t)\cos(t) \quad 0]^{\mathrm{T}}\\ \dot{\mathbf{x}}_{r,b} &= [-0.5\sin(t) \quad 0.5\cos(2t) \quad 0]^{\mathrm{T}}\\ \ddot{\mathbf{x}}_{r,b} &= [-0.5\cos(t) \quad -\sin(2t) \quad 0]^{\mathrm{T}}.\end{aligned} \tag{18}$$

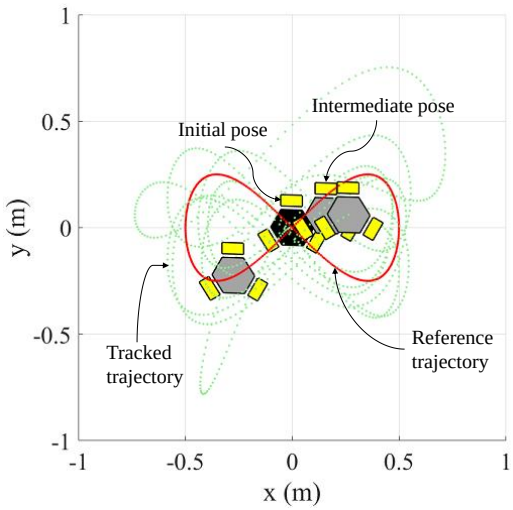
In addition, we simulate the disturbance input $\mathbf{d}$ and output measurement noise $\mathbf{n}$ as the following sinusoidal functions:

$$\begin{aligned}\mathbf{d} =&[0.05\sin(t/2) \quad 0.15\sin(t/4) \quad 0]^{\mathrm{T}}\\ \mathbf{n} =&[0.15\sin(2t) \quad 0.1\sin(3t) \quad 0 \quad ...\\ &0.25\sin(4t) \quad 0.25\sin(2t) \quad 0]^{\mathrm{T}}.\end{aligned} \tag{19}$$

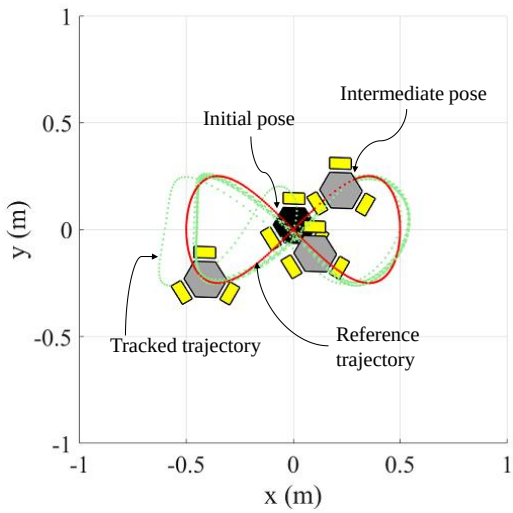
Note that the first three elements of the noise signal are applied to the measurement of the robot's pose, $[{}^G x \quad {}^G y \quad \varphi]^{\mathrm{T}}$, and the last three elements affect the twist measurements, $[{}^G\dot{x} \quad {}^G\dot{y} \quad \dot{\varphi}]^{\mathrm{T}}$.

For the classical controller, we tuned the gain matrices to $\mathbf{K} = 0.25\mathbf{I}_{3\times3}$ and $\mathbf{\Lambda} = 0.5\mathbf{I}_{3\times3}$, which produce the best tracking performance in terms of the lowest values of $||\mathbf{e}(t)||_2$ and $||\dot{\mathbf{e}}(t)||_2$, the 2-norms of the pose error and twist error. In order to find the controller gain for the H_∞-optimal controller, we solve the LMI in Eq. (16) using YALMIP [21], which employs SeDuMi [22] for this purpose. The optimal controller gain matrix of the H_∞ controller is obtained as:

$$\mathbf{K}_{H_\infty} = \begin{bmatrix} 5.01 & 0 & 0 & 12.33 & 0 & 0\\ 0 & 5.01 & 0 & 0 & 12.33 & 0\\ 0 & 0 & 35.31 & 0 & 0 & 39.54\end{bmatrix} \tag{20}$$



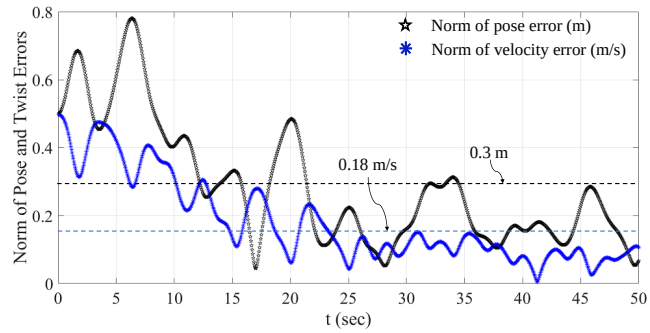
(a)



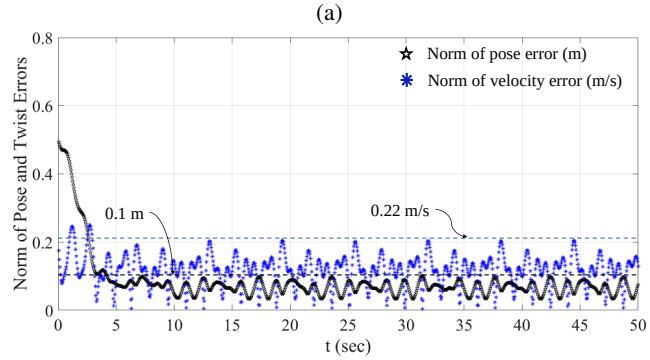
(b)

Fig. 5: Trajectory of a simulated three-wheeled omnidirectional robot that tracks the bowtie reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

Figures 3a and 3b show the trajectory of the simulated omnidirectional robot over 50 s as it tracks the circular reference trajectory using the classical and H_∞-optimal controllers, respectively, in the presence of the disturbances and noise defined in Eq. (19). For these simulations, Figs. 4a and 4b each plot the time evolution of $||\mathbf{e}(t)||_2$ and $||\dot{\mathbf{e}}(t)||_2$. Similarly, Figs. 5a and 5b plot the trajectory of the simulated robot over 50 s as it tracks the bowtie reference trajectory using both controllers, and Figs. 6a and 6b plot the corresponding time evolution of $||\mathbf{e}(t)||_2$ and $||\dot{\mathbf{e}}(t)||_2$. In the plots of the error norms over time, the black and blue dashed horizontal lines indicate the maximum values of $||\mathbf{e}(t)||_2$ and $||\dot{\mathbf{e}}(t)||_2$, respectively, over one complete cycle of the robot around the reference trajectory after the robot's initial transient behavior as it approached this trajectory. We will refer to these maximum values as $||\mathbf{e}||_2^{\max}$ and



(a)



(b)

Fig. 6: Norm of the pose and twist errors over 50 s of the simulations in which the robot tracks the bowtie reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

$||\dot{\mathbf{e}}||_2^{\max}$. Figs. 4a and 4b show that for the circular reference trajectory, the H_∞ controller reduces $||\mathbf{e}||_2^{\max}$ from 0.62 m for the classical controller to 0.1 m, and reduces $||\dot{\mathbf{e}}||_2^{\max}$ from 0.27 m/s for the classical controller to 0.2 m/s. Figs. 5a and 5b show that for the bowtie reference trajectory, the H_∞ controller reduces $||\mathbf{e}||_2^{\max}$ from 0.3 m to 0.1 m, and slightly raises $||\dot{\mathbf{e}}||_2^{\max}$ from 0.18 m/s to 0.22 m/s.

These results demonstrate that the LMI-based full-state feedback H_∞-optimal controller can significantly improve the tracking performance of the robot in simulation compared to the classical full-state feedback controller. This is because the H_∞-optimal controller attenuates the undesired effects of the exogenous inputs, $\mathbf{d}$ and $\mathbf{n}$, whereas the classical controller cannot mitigate the effects of these undesired inputs.

V. Experimental Implementation and Results

In this section, we implement the classical and H_∞ trajectory tracking controllers described in Section III on the three-wheeled omni-directional robot shown in Fig. 7. This robot has three omni wheels connected to Dynamixel DC motors that are spaced 120° apart. The rollers around the rims of the omni wheels allow the robot to move freely to any arbitrary configuration. The Dynamixel motors can measure the omni wheels' rotation angles, θ_i $(i = 1, 2, 3)$, and their angular velocities, $\dot{\theta}_i$ $(i = 1, 2, 3)$, using embedded encoders. The parameters r and d denote the radius of the omni wheels

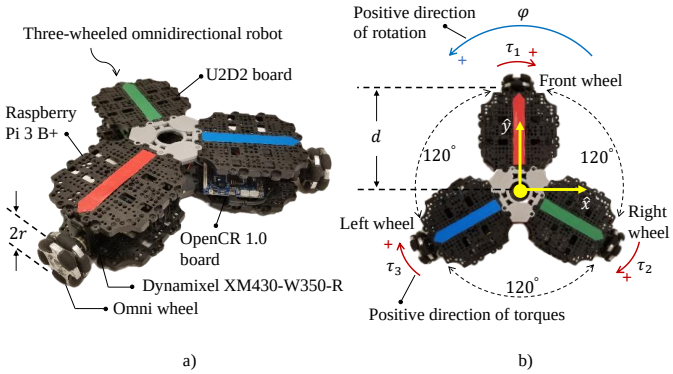


Fig. 7: (a) Isometric and (b) overhead views of the three-wheeled omnidirectional WMR used in the experiments. The global coordinate frame is defined as in Fig. 1.

and the distance between the center of each wheel and the origin of the body-fixed coordinate frame, defined as $\hat{x}-\hat{y}$.

A. Odometry, Twist, and Torque Calculations

The twist of the omnidirectional robot, $\dot{\mathbf{x}} = [{}^G\dot{x} \quad {}^G\dot{y} \quad \dot{\varphi}]^{\mathrm{T}}$, represented in the global coordinate frame $\bar{x}-\bar{y}$, can be expressed in terms of the angular velocities of the omni wheels. We first define the rotation matrix, $\mathbf{R}_{-\varphi}(z)$, that describes rotation of the body-fixed coordinate frame, $\hat{x}-\hat{y}$, by an angle $-\varphi$ about the $\bar{z}$-axis. To calculate the robot's twist in the global coordinate frame, we pre-multiply the robot's twist in the body-fixed frame, ${}^b\dot{\mathbf{x}} = [{}^b\dot{x}, \quad {}^b\dot{y}, \quad \dot{\varphi}]^{\mathrm{T}}$, by this rotation matrix as follows [4]:

$$\begin{bmatrix} {}^G\dot{x} \\ {}^G\dot{y} \\ \dot{\varphi} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{R}_{-\varphi}(z)} \underbrace{\begin{bmatrix} \frac{2r}{3} & \frac{-r}{3} & \frac{r}{3} \\ 0 & \frac{-\sqrt{3}r}{3} & \frac{\sqrt{3}r}{3} \\ \frac{-r}{3d} & \frac{-r}{3d} & \frac{-r}{3d} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}}_{{}^b\dot{\mathbf{x}}} \tag{21}$$

This equation can be used to compute the robot's twist error, $\dot{\mathbf{e}}$, in Eq. (11). Odometry to estimate the robot's pose, $[{}^Gx \quad {}^Gy \quad \varphi]^{\mathrm{T}}$, is possible using the rotational feedback provided by the Dynamixel motors. This can be done by assuming that the wheels' angular velocities remain constant during the time step Δt between readings, such that $\dot{\theta}_i = \Delta\theta_i/\Delta t$. Regardless of the units used to measure time, we can set $\Delta t = 1$ without loss of generality, which results in $\dot{\theta}_i = \Delta\theta_i$ [4]. Thus, the robot's velocity in the global frame $\bar{x}-\bar{y}$ can be computed in terms of the rotation angles of the omni wheels measured by the encoders. In turn, we can calculate the robot's pose error, $\mathbf{e}$, in Eq. (11) given the odometry information of the robot.

We calculate the control input vector, $\mathbf{u}$, by substituting the robot's pose and twist errors into the control laws of the classical and H_∞ controllers, defined in Eqs. (9) and (14), respectively. We set the classical controller gains to the values used in the simulations, and the gains of the H_∞ controller were set according to Eq. (20). The control inputs are defined as the desired wrench of the robot, $\mathbf{F} = [{}^Gf_x \quad {}^Gf_x \quad {}^G\tau_z]^{\mathrm{T}}$, represented in the global coordinate frame $\bar{x}-\bar{y}$. This wrench should be produced by the Dynamixel motors. We first transform this desired wrench from the global frame to the body-fixed coordinate frame of the robot by pre-multiplying the wrench in the global frame by the rotation matrix $\mathbf{R}_\varphi(z)$. Then, the wrench in the body-fixed frame can be converted to the desired torques that should be produced by the motors on the omni wheels, $\mathbf{T} = [\tau_1 \quad \tau_2 \quad \tau_3]^{\mathrm{T}}$, as follows:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} \frac{2r}{3} & 0 & \frac{-r}{3d} \\ \frac{-r}{3} & \frac{-\sqrt{3}r}{3} & \frac{-r}{3d} \\ \frac{-r}{3} & \frac{\sqrt{3}r}{3} & \frac{-r}{3d} \end{bmatrix} \begin{bmatrix} \cos(\varphi) & \sin(\varphi) & 0 \\ -\sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^Gf_x \\ {}^Gf_y \\ {}^G\tau_z \end{bmatrix} \tag{22}$$

The torques applied by the Dynamixel motors are controlled by the Raspberry Pi computer through the U2D2 board, which are both powered by the OpenCR board. We programmed both controllers on the robot in the C++ language based on the Robot Operating System (ROS) platform.

B. Experimental Results

In our experiments, the three-wheeled omnidirectional robot attempted to track the circular and bowtie reference trajectories defined in Eq. (17) and (18) in the presence of unknown disturbances arising from the friction force between the omni wheels and the ground, as well as noise in the output measurements. We compared the robot's tracking performance with both the classical and H_∞-optimal controllers. Snapshots of the robot's trajectory over one full cycle of the circular and bowtie reference trajectories for both controllers are plotted using in MATLAB® in Figs. 8a, 8b and in Figs. 10a, 10b. As in the simulations, the H_∞ controller is able to attenuate the undesired effects of exogenous inputs on the tracking performance of the robot, yielding lower tracking errors than the classical controller. Figs. 9a and 9b show that for the circular reference trajectory, the H_∞ controller reduces $||\mathbf{e}||_2^{\max}$ from 0.4 m for the classical controller to 0.11 m, and reduces $||\dot{\mathbf{e}}||_2^{\max}$ from 0.24 m/s for the classical controller to 0.13 m/s. Figs. 10a and 10b show that for the bowtie reference trajectory, the H_∞ controller reduces $||\mathbf{e}||_2^{\max}$ from 0.23 m to 0.12 m, and $||\dot{\mathbf{e}}||_2^{\max}$ from 0.12 m/s to 0.07 m/s. These experimental results validate the effectiveness of our proposed control approach. Video recordings of the experiments are online at [23].

VI. Conclusion

In this paper, we proposed a full-state feedback H_∞-optimal controller based on the LMI framework for trajectory tracking control by a three-wheeled omnidirectional robot. First, a point-mass dynamic model of the robot represented in state-space form was derived, and then the effects of disturbances on its actuators and noise in its output measurements were incorporated into the model. Along with the H_∞-optimal controller, we also described a classical full-state feedback controller for this control problem. We simulated a three-wheeled omnidirectional robot and implemented both controllers in simulation for tracking of circular and bowtie reference trajectories. Furthermore, we experimentally tested

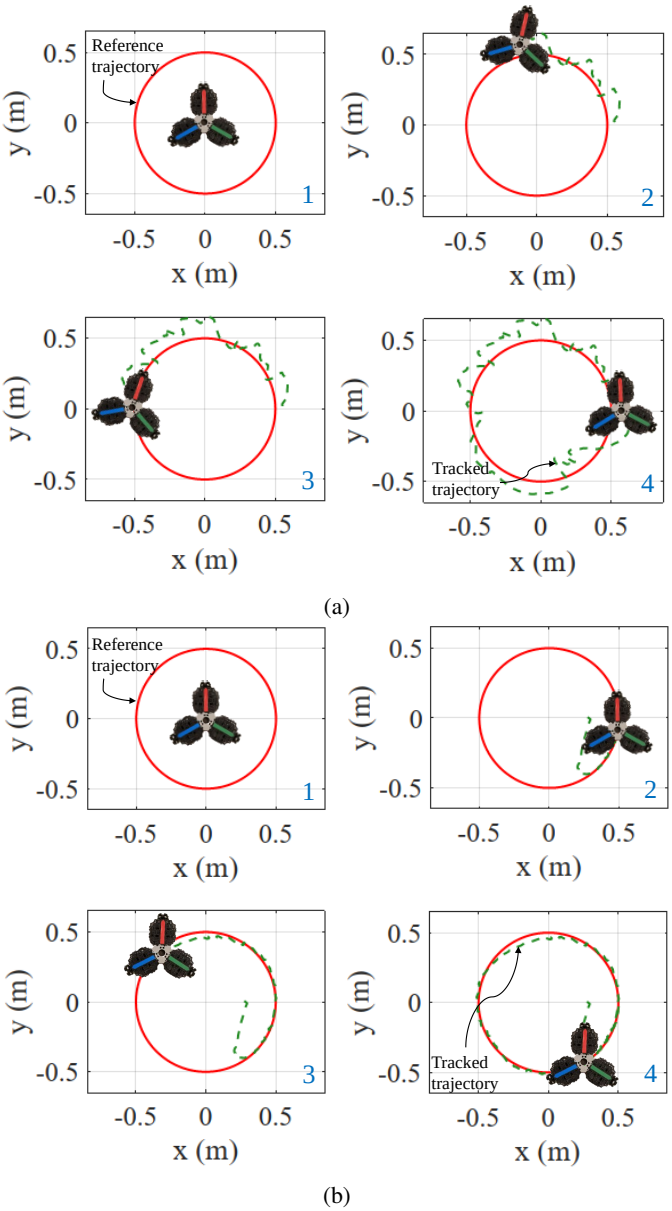


Fig. 8: Snapshots of the trajectory of a three-wheeled omni-directional robot during an experiment in which it tracks the circular reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

the classical and H_∞ controllers on a real three-wheeled omnidirectional robot for the same reference trajectories. We compared the tracking performance of the robot with both controllers in terms of the norms of the robot's pose and twist errors. The simulation and experimental results demonstrate that our proposed full-state feedback H_∞-optimal controller can significantly improve the robot's tracking performance compared to the classical controller by attenuating the effect of undesired exogenous inputs on the robot. Future work includes the extension of this control approach to multiple WMRs that must perform tasks such as flocking and target

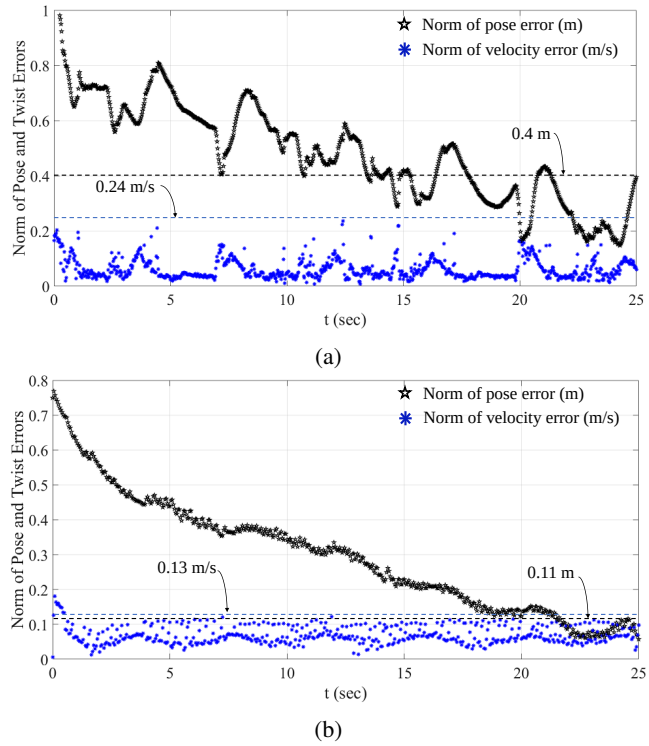


Fig. 9: Norm of the pose and twist errors over 25 s of an experiment in which the robot tracks the circular reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

tracking while avoiding collisions with each other and with objects in the environment. Moreover, the state-space model of the closed-loop system can be modified to include weights on the reference, output, disturbance, and noise signals, which can be tuned to improve the robot's trajectory tracking performance.

References

[1] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[2] O. Khatib, X. Yeh, G. Brantner, B. Soe, B. Kim, S. Ganguly, H. Stuart, S. Wang, M. Cutkosky, A. Edsinger *et al.*, "Ocean one: A robotic avatar for oceanic discovery," *IEEE Robotics & Automation Magazine*, vol. 23, no. 4, pp. 20–29, 2016.

[3] N. A. Radford, P. Strawser, K. Hambuchen, J. S. Mehling, W. K. Verdeyen, A. S. Donnan, J. Holley, J. Sanchez, V. Nguyen, L. Bridgwater *et al.*, "Valkyrie: NASA's first bipedal humanoid robot," *Journal of Field Robotics*, vol. 32, no. 3, pp. 397–419, 2015.

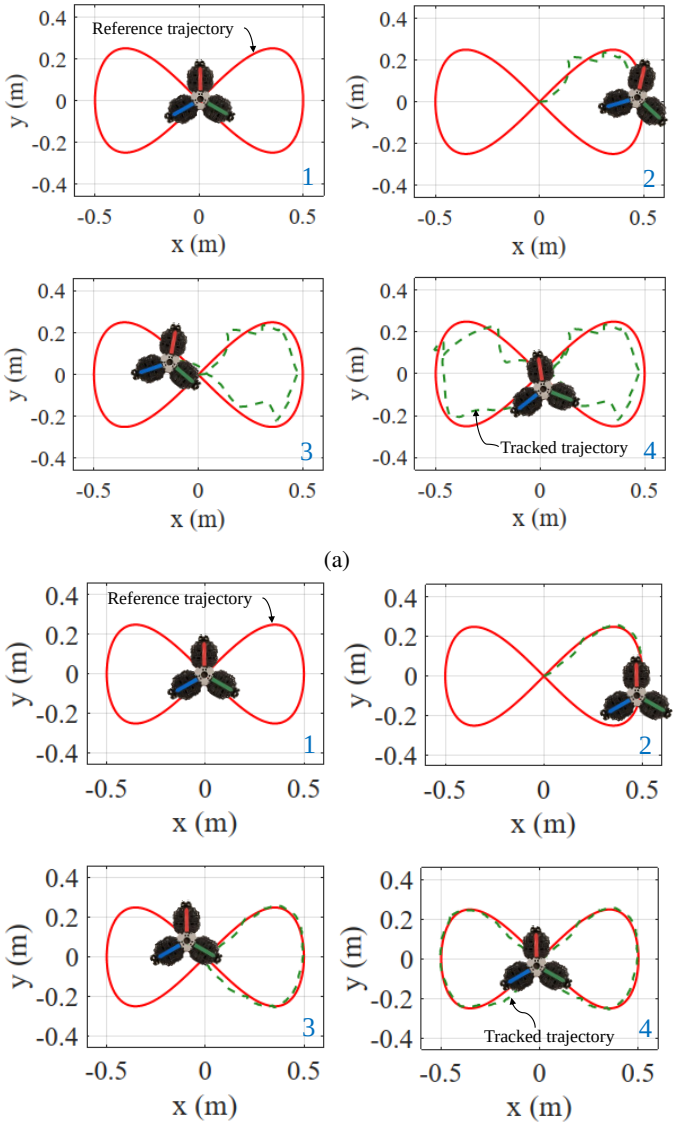
[4] K. M. Lynch and F. C. Park, *Modern Robotics*. Cambridge University Press, 2017.

[5] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135–1145, 2015.

[6] A. H. Reddy, B. Kalyan, and C. S. Murthy, "Mine rescue robot system–a review," *Procedia Earth and Planetary Science*, vol. 11, pp. 457–462, 2015.

[7] R. W. Brockett *et al.*, "Asymptotic stability and feedback stabilization," *Differential Geometric Control Theory*, vol. 27, no. 1, pp. 181–191, 1983.

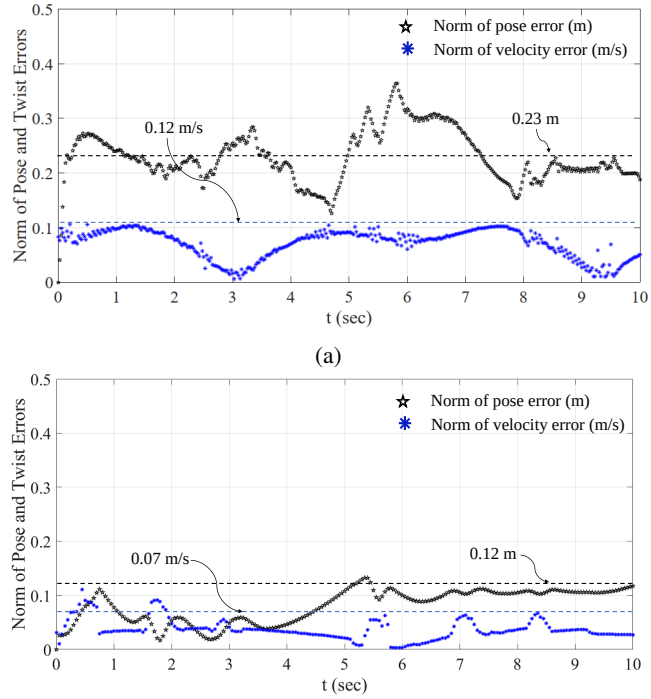
[8] T. Bräunl, "Omni-directional robots," *Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems*, pp. 147–156, 2008.



(a)

(b)

Fig. 10: Snapshots of the robot's trajectory during an experiment in which it tracks the bowtie reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.



(a)

(b)

Fig. 11: Norm of the pose and twist errors over 10 s of an experiment in which the robot tracks the bowtie reference trajectory using (a) the classical full-state feedback controller; (b) the full-state feedback H_∞-optimal controller.

[9] G.-R. Duan and H.-H. Yu, *LMIs in control systems: analysis, design and applications*. CRC Press, 2013.

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

[11] R. J. Caverly and J. R. Forbes, "LMI properties and applications in systems, stability, and control theory," *arXiv preprint arXiv:1903.08599*, 2019.

[12] H. X. Araújo, A. G. Conceição, G. H. Oliveira, and J. Pitanga, "Model predictive control based on LMIs applied to an omni-directional mobile robot," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 8171–8176, 2011.

[13] G. Rigatos and P. Siano, "An H-infinity feedback control approach to autonomous robot navigation," in *IECON 2014-40th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2014, pp. 2689–2694.

[14] J. Vázquez and M. Velasco-Villa, "Path-tracking dynamic model based control of an omnidirectional mobile robot," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 5365–5370, 2008.

[15] C. Ren, Y. Ding, X. Li, X. Zhu, and S. Ma, "Extended state observer based robust friction compensation for tracking control of an omnidirectional mobile robot," *Journal of Dynamic Systems, Measurement, and Control*, vol. 141, no. 10, 2019.

[16] C. Ren and S. Ma, "Trajectory tracking control of an omnidirectional mobile robot with friction compensation," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 5361–5366.

[17] R. L. Williams, B. E. Carter, P. Gallina, and G. Rosati, "Dynamic model with slip for wheeled omnidirectional robots," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 3, pp. 285–293, 2002.

[18] C. Scherer, "Theory of robust control," *Delft University of Technology*, pp. 1–160, 2001.

[19] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice Hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.

[20] N. Sreedhar and S. Rao, "Stability of a system of linear differential equations," *IEEE Transactions on Automatic Control*, vol. 13, no. 3, pp. 307–308, 1968.

[21] J. Lofberg, "YALMIP: a toolbox for modeling and optimization in MATLAB," in *2004 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2004, pp. 284–289.

[22] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11, no. 1-4, pp. 625–653, 1999.

[23] "H-infinity optimal tracking controller for three-wheeled omnidirectional mobile robots with uncertain dynamics," Autonomous Collective Systems Laboratory Youtube channel, https://www.youtube.com/watch?v=vgaU4Yk6X5s.