Development of a New Robust Controller with Velocity Estimator for Docked Mobile Robots: Theory and Experiments

Negin Lashkari, Mohammad Biglarbegian, *Senior Member, IEEE,* Simon X. Yang, *Senior Member, IEEE*

***Abstract*—The tracking control problem of docked mobile robot systems is challenging due to their nonlinear and underactuated system dynamics as well as limited access to the required states of robots. The majority of the previously developed controllers in the literature are not robust to model uncertainties and are based on the assumption that full-states are accessible. In this paper, we develop a new robust tracking controller for a docked nonholonomic mobile robotic system with online velocity estimation. Our proposed controller, composed of sliding mode and robust saturation controllers, is developed to be robust to external disturbances, unmodeled dynamics and parameter uncertainties. To provide the required states for the controller, a model-aided particle filter estimator is developed to estimate the translational and rotational velocities. We perform several experiments to verify the effectiveness of our proposed control and estimation methodologies as well as the integrated system. We also compare our results with some conventional controllers developed in the literature, such as sliding mode control, and demonstrate its superior performance in terms of unmodeled dynamics and parametric uncertainties. This comparison indicated that the steady state tracking performance increases by up to 28.7% and 22.2% under parametric uncertainties and unmodeled dynamics, respectively, showing a significant improvement over the sliding mode control. Our proposed integrated (controller-estimator) method can be used in uncertain systems with good tracking performance where accessing velocity directly is not possible.**

***Index Terms*—Docked Mobile Robots, Robust Tracking Control, Velocity Estimator.**

I. Introduction

Multiple mobile robots have enhanced functionalities compared to a single robot while being more efficient and reliable. More specifically, when multiple robots are capable of being docked to each other, their range of operations will significantly increase, e.g., ability to transport heavier loads, transfer powers amongst robots (as charging units), mobility in rough terrains, robustness to hardware failure, and self-perception. Some important applications of docked mobile robots (DMR) are seen in agriculture [1], search and rescue in rough terrains [2], luggage carriers [3], and manipulation and transportation of wire-like objects [4].

Mechanical structures of docking mechanisms in DMR can be generally classified into two categories: on-axle and off-axle hitching. In DMR with on-axle hitching, the joint between the leader (front robot) and follower (rear robot) is located at the center of the rear axle of the leader robot. In DMR with off-axle hitching, the joint between the leader and follower robots is located at a point away from the rear axle of the leader robot. Although DMR with on-axle hitching has been widely used in the literature [5], [6], it confines the mobility and maneuverability of the system as reported in [7]. On the other hand, off-axle offers better tracking performance and it is capable of traveling in the narrower tracks due to requiring smaller width of path, yet, its kinematic structure is more complicated [8]. In this paper, we are concerned with an off-axle hitched DMR because it outperforms the trajectory tracking compared to on-axle hitched DMR.

In practical applications of robotic systems, robust control methods, such as sliding mode control [9], [10], H_∞ optimal control [11], [12], and adaptive robust control [13], [14], have been proven to outperform conventional controllers. To increase the functionality of DMR, thus, it is necessary to develop realistic robust motion controllers that can effectively navigate such robots. Several motion controllers for DMR have been developed, such as tracking control, stabilization, assistance, backward driving control, and formation control. Among the aforementioned control problems, tracking control is most important in applications where autonomous motion is desirable. Tracking control of DMR is complex due to being underactuated as well as having nonlinear dynamics and nonholonomic constraints. There are several attempts to tackle this problem which can be categorized into kinematics-based tracking control design [5], [15–19] and dynamics-based [6], [20], [21]. While most of the developed tracking controllers in the literature use only kinematics of DMR in their design, incorporating dynamics is crucial to improve the robustness of the DMR system especially when dimension of system is large, the weight of robots/vehicles is considerable, or when they carry or drop off heavy loads. Thus, a controller needs to be designed taking into account the dynamics of the system such that it is robust to unmodeled dynamics uncertainties. Only a few papers have considered dynamics of DMR in their control design [6], [21]. In [21], a linear quadratic regulator based on the linearized dynamic model was proposed for a tractor-trailer system. However, the robustness of the controller is questionable because of using linearization. In [6], a robust adaptive tracking controller was designed for a tractor-trailer robotic system including robots' dynamics. As was also mentioned in [6], the controller is not robust to unmodeled dynamics uncertainties, and it is based on the assumption that all the control states are available without providing details

N. Lashkari, M. Biglarbegian, and S. X. Yang are with the School of Engineering, University of Guelph, Canada, e-mail: nlashkar@uoguelph.ca, and mbiglarb@uoguelph.ca, syang@uoguelph.ca, Tel: (519) 824-4120 (ext. 56044).

on how to obtain them. In addition, tracking errors reported in [6] are rather large, and an on-axle trailer is considered which may limit maneuverability and adversely affect the tracking performance. In summary, the existing controllers in the literature for DMR are not robust to unmodeled dynamic uncertainties [6], [17], [18]. In this paper, we develop a new robust control method incorporating dynamics of DMR system, which is robust to model uncertainties (including unmodeled dynamics and parameter uncertainties) and without the assumption that full-state feedback is required.

The majority of the developed controllers for DMR in the literature require accessing velocity to achieve good performance [5], [6], [15], [22], and hence their performance will be deteriorated when the required control states cannot be measured accurately. In real applications, accessing the velocity is difficult due to limitations in measuring velocity directly using sensors; and, even if measured, its signal would be noisy. This failure in measuring robots' velocity leads to inefficient control performance. To make the controller more practical and applicable to a wider range of robotic tasks, it is necessary to develop an estimator to provide the necessary state(s) for the controller. Most of velocity estimators have been only developed for non-docked robots [23–26], and to the best of our knowledge, there is no work for DMR in the literature that has developed a methodology for estimating velocity and accordingly design a robust controller that can handle unmodeled uncertainties with good tracking error. Due to nonlinearities existing in the dynamics of DMR, a nonlinear estimator has to be developed that can accurately provide the required control states. In our paper, we develop a new estimator based on particle filter algorithm that fuses sensory data with the dynamics of DMR to provide a good estimate of translational and rotational velocities.

In this paper, we develop (1) a new robust tracking controller composed of two controllers: sliding mode and robust saturation controller. The sliding mode controller is designed for tracking under disturbances yet it does not guarantee robustness under model uncertainties. The robust saturation controller handles the model uncertainties to achieve good robust tracking, (2) an estimator to provide the required states (for control) that cannot be measured directly. Our proposed methodology is general and can be extended (and applied) to any underactuated robotic system with nonlinear dynamics and nonholonomic constraints for trajectory tracking purposes. In summary, the contributions of the paper are as follows: (i) development of a new robust controller that can handle model uncertainties such as unmodeled dynamics and parametric uncertainties; useful when robots/vehicles carry uncertain loads, (ii) development of a new particle filter-based estimator to estimate rotational and translational velocities profile given the availability of position and orientation sensory data and a dynamic model of the system, (iii) validation of the performance of our developed integrated controller and estimator experimentally and demonstrate its effectiveness in comparison to some well-known controllers.

The rest of the paper is organized as follows. Section II presents kinematics and dynamic models for a two-docked mobile robot system, Section III presents the control design, Section IV provides the development of the estimator, Section V presents the simulation and experimental results, and Section VI concludes the paper.

Table I: Parameters used to present DMR.

Symbol	Definition
O_n	geometric center of n^{th} robot
CO_n	center of mass of n^{th} robot and docking link
(x_n, y_n)	coordinates of n^{th} robot's center of mass
b	half of the distance between wheels of robot
r	radius of driving wheel
d_n	distance between n^{th} robot geometric center and center of mass
h_n	distance between the joint and center of mass of n^{th} robot
m_n	mass of n^{th} robot
m_{o_n}	mass of each docking link connected to n^{th} robot
θ_n	orientation of n^{th} robot
v_n	translational velocity of n^{th} robot
ω_n	rotational velocity of n^{th} robot
I_n	moment of inertia of n^{th} robot and docking link
τ_R	applied torque to the right wheel of leader robot
τ_L	applied torque to the left wheel of leader robot

* $n = 1, 2$ corresponds to the leader and follower robots, respectively.

II. Governing Dynamics

The DMR considered in this study is a two-docked mobile robotic system shown in Fig. 1. This system includes a mobile robot, defined as a leader (robot 1), docked to a follower robot (robot 2) via rigid links and a flexible joint. The links and joint allow the connected robots to have a relative rotational motion with respect to each other. The nature of hitching in the docking mechanism is considered to be off-axle. The two mobile robots are nonholonomic differential drive identical to each other but with different masses. The center of mass of each robot has an offset (d_1 and d_2) with the geometric center as shown in Fig. 1. The parameters of the considered DMR are listed in Table I. The configuration of the DMR is given by $\boldsymbol{q} = [x_2, y_2, \theta_2, \theta_1]^T$, where (x_2, y_2) is the coordinate of the follower's center of mass, and θ_1 and θ_2 are the orientations of the leader and follower robots, respectively. The reason that we can present the entire DMR system with only four generalized coordinates is because the leader position can be found using

$$x_1 = x_2 + h_2 \cos\theta_2 + h_1 \cos\theta_1 \quad (1)$$

$$y_1 = y_2 + h_2 \sin\theta_2 + h_1 \sin\theta_1. \quad (2)$$

Two types of kinematic constraints exist in the system: (i) docking constraints, and (ii) nonholonomic constraints. The docking constraints indicate the relation between the leader and follower robots' translational and rotational velocities, and are expressed as

$$\begin{aligned} v_2 &= v_1 \cos(\theta_1 - \theta_2) + \omega_1 (h_1 + d_1) \sin(\theta_1 - \theta_2) \\ \omega_2 &= \frac{1}{2d_2 + h_2}[v_1 \sin(\theta_1 - \theta_2) - \omega_1 (h_1 + d_1) \cos(\theta_1 - \theta_2)]. \end{aligned} \quad (3)$$

The nonholonomic constraints are due to no-slip conditions that do not allow robots to slide sideways and are given by

$$\begin{aligned} \sin\theta_1 \dot{x}_2 - \cos\theta_1 \dot{y}_2 - h_2 \cos(\theta_1 - \theta_2)\dot{\theta}_2 + (d_1 - h_1)\dot{\theta}_1 = 0 \\ \sin\theta_2 \dot{x}_2 - \cos\theta_2 \dot{y}_2 + d_2 \dot{\theta}_2 = 0. \end{aligned} \quad (4)$$

The dynamics of DMR with nonholonomic constraints is described by

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{\tau} - \boldsymbol{A}^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{\lambda} \tag{5}$$

in which $\boldsymbol{M}(\boldsymbol{q}) \in \Re^{4\times 4}$ is the positive definite inertia matrix, $\boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \Re^{4\times 4}$ represents centrifugal and Coriolis terms, $\boldsymbol{B}(\boldsymbol{q}) \in \Re^{4\times 2}$ is a transformation matrix, $\boldsymbol{A}(\boldsymbol{q}) \in \Re^{2\times 4}$ is related to nonholonomic constraints, $\boldsymbol{\lambda} \in \Re^{2\times 1}$ is the constraint force vector, $\boldsymbol{q}$ is the generalized coordinates introduced previously, and $\boldsymbol{\tau} \in \Re^{2\times 1}$ is the input vector indicating the applied torque required to drive the DMR. In this paper, because of kinematic constraints, we assumed that the follower robot is passive and cannot be actuated; thus, the input vector is $\boldsymbol{\tau} = [\tau_R, \tau_L]^{\mathrm{T}}$, where τ_R and τ_L are the applied torques of the right and left wheel of the leader robot, respectively.

Assuming d_1 and d_2 are negligible, the matrices $\boldsymbol{M}(\boldsymbol{q})$, $\boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})$, $\boldsymbol{B}(\boldsymbol{q})$ and $\boldsymbol{A}(\boldsymbol{q})$ are found as

$$\boldsymbol{M}(\boldsymbol{q}) = \begin{bmatrix} a_{11} & 0 & a_{13} & a_{14} \\ 0 & a_{11} & a_{23} & a_{24} \\ a_{13} & a_{23} & a_{33} & a_{34} \\ a_{14} & a_{24} & a_{34} & a_{44} \end{bmatrix}$$

$$\boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \begin{bmatrix} 0 & 0 & -h_2(m_1+m_{o_1})\mathrm{c}_2\dot{\theta}_2 & -h_1(m_1+m_{o_1})\mathrm{c}_1\dot{\theta}_1 \\ 0 & 0 & -h_2(m_1+m_{o_1})\mathrm{s}_2\dot{\theta}_2 & -h_1(m_1+m_{o_1})\mathrm{s}_1\dot{\theta}_1 \\ 0 & 0 & 0 & -h_1h_2(m_1+m_{o_1})\mathrm{s}_{12}\dot{\theta}_1 \\ 0 & 0 & h_1h_2(m_1+m_{o_1})\mathrm{s}_{12}\dot{\theta}_2 & 0 \end{bmatrix}$$

$$\boldsymbol{B}(\boldsymbol{q}) = \frac{1}{r}\begin{bmatrix} \mathrm{c}_1 & \mathrm{c}_1 \\ \mathrm{s}_1 & \mathrm{s}_1 \\ b & -b \\ h_2\mathrm{s}_{12} & h_2\mathrm{s}_{12} \end{bmatrix}, \quad \boldsymbol{A}(\boldsymbol{q}) = \begin{bmatrix} \mathrm{s}_1 & -\mathrm{c}_1 & -h_2\mathrm{c}_{12} & -h_1 \\ \mathrm{s}_2 & -\mathrm{c}_2 & 0 & 0 \end{bmatrix}$$

where

$$\begin{aligned} &a_{11} = m_1 + m_2 + m_{o_1} + m_{o_2}, && a_{13} = -h_2(m_1+m_{o_1})\mathrm{s}_2, \\ &a_{14} = -h_1(m_1+m_{o_1})\mathrm{s}_1, && a_{23} = h_2(m_1+m_{o_1})\mathrm{c}_2, \\ &a_{24} = h_1(m_1+m_{o_1})\mathrm{c}_1, && a_{33} = I_2 + h_2{}^2(m_1+m_{o_1}), \\ &a_{44} = I_1 + h_1h_2(m_1+m_{o_1}), && a_{34} = h_1h_2(m_1+m_{o_1})\mathrm{c}_{12}, \\ &\mathrm{c}_{12} = \cos(\theta_1-\theta_2), && \mathrm{s}_{12} = \sin(\theta_1-\theta_2), \\ &\mathrm{c}_1 = \cos\theta_1, && \mathrm{c}_2 = \cos\theta_2, \\ &\mathrm{s}_1 = \sin\theta_1, && \mathrm{s}_2 = \sin\theta_2. \end{aligned}$$

To include the nonholonomic constraints, we introduce

$$\dot{\boldsymbol{q}} = \boldsymbol{J}(\boldsymbol{q})\boldsymbol{z} \tag{6}$$

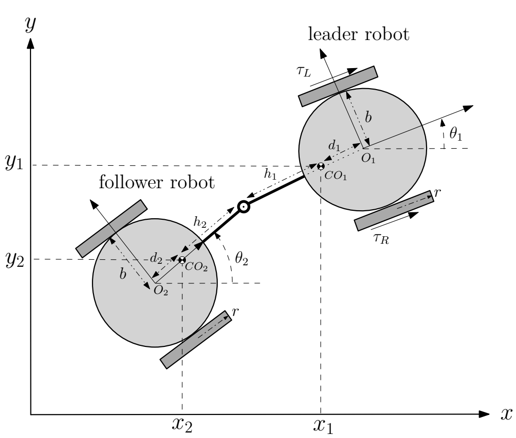

Figure 1: DMR with off-axle hitching.

where $\boldsymbol{z}$ is given by [6]

$$\boldsymbol{z} = \begin{bmatrix} v_1\cos(\theta_1-\theta_2) \\ \omega_1 \end{bmatrix}. \tag{7}$$

Using (3), we propose $\boldsymbol{J}(\boldsymbol{q})$ to be as follows

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} \cos\theta_2 & h_1\cos\theta_2\sin(\theta_1-\theta_2) \\ \sin\theta_2 & h_1\sin\theta_2\sin(\theta_1-\theta_2) \\ \dfrac{1}{h_2}\tan(\theta_1-\theta_2) & -\dfrac{h_1}{h_2}\cos(\theta_1-\theta_2) \\ 0 & 1 \end{bmatrix} \tag{8}$$

such that $\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{q}) = 0$. Therefore, by substituting (6) into (5), the system dynamics takes the following form:

$$\dot{\boldsymbol{z}} = \overline{\boldsymbol{M}}^{-1}(\boldsymbol{q})[-\overline{\boldsymbol{V}}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} + \overline{\boldsymbol{B}}(\boldsymbol{q})\boldsymbol{\tau}] \tag{9}$$

where $\overline{\boldsymbol{M}}(\boldsymbol{q})$=$\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})$, $\overline{\boldsymbol{B}}(\boldsymbol{q}) = \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{q})\boldsymbol{B}(\boldsymbol{q})$, and $\overline{\boldsymbol{V}}(\boldsymbol{q},\dot{\boldsymbol{q}})$=$\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{q})(\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{J}}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{J}(\boldsymbol{q}))$.

Note that when the passive follower robot in the DMR system skids during sharp turning, backward motion and fast paces [8], the leader and follower robots get perpendicular with respect to each other (jack-knife phenomenon) and consequently (9) reaches singular points. At this point, the determinant of $\overline{\boldsymbol{M}}(\boldsymbol{q})$ is zero and thus $\overline{\boldsymbol{M}}^{-1}(\boldsymbol{q})$ does not exist. The following assumption is thus made and will be used in the design of the controller:

Assumption 1. The robots trajectories do not contain sharp turnings and no backward movement and fast speed is allowed for the DMR. Thus, the inertia matrix $\overline{\boldsymbol{M}}(\boldsymbol{q})$ is invertible throughout the entire motion and the system is stabilizable.

In the following sections, our derived dynamics, (9), is used to design a robust controller and a velocity estimator.

III. Control Design

Assume the DMR system described by (9) has some external disturbances and uncertainties in the parameters and the dynamic model. Therefore, $\overline{\boldsymbol{M}}(\boldsymbol{q})$ and $\overline{\boldsymbol{V}}(\boldsymbol{q},\dot{\boldsymbol{q}})$ are as follows (note that $\overline{\boldsymbol{B}}(\boldsymbol{q})$ does not contain uncertain parameters):

$$\overline{\boldsymbol{M}}(\boldsymbol{q}) = \overline{\boldsymbol{M}}_0(\boldsymbol{q}) + \boldsymbol{\Delta M}(\boldsymbol{q}) \tag{10}$$

$$\overline{\boldsymbol{V}}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{\Delta V}(\boldsymbol{q},\dot{\boldsymbol{q}}) \tag{11}$$

where $\overline{\boldsymbol{M}}_0(\boldsymbol{q})$ and $\overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}})$ are nominal matrices, $\boldsymbol{\Delta M}(\boldsymbol{q})$ and $\boldsymbol{\Delta V}(\boldsymbol{q},\dot{\boldsymbol{q}})$ represent system uncertainty. A bounded input disturbance $\boldsymbol{\tau}_d$ is also considered for the DMR, where it is assumed to be upper bounded by a positive number. Therefore, by including uncertainties and disturbances, (9) can be expressed as follows:

$$\dot{\boldsymbol{z}} = \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q})\left[-\overline{\boldsymbol{V}}_{\boldsymbol{0}}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} + \overline{\boldsymbol{B}}(\boldsymbol{q})\boldsymbol{\tau} + \boldsymbol{h}(t)\right] \tag{12}$$

where

$$\boldsymbol{h}(t) = -\boldsymbol{\Delta M}(\boldsymbol{q})\dot{\boldsymbol{z}} - \boldsymbol{\Delta V}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} - \boldsymbol{\tau}_d. \tag{13}$$

We propose our robust control law to be as follows:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{SMC} + \boldsymbol{\Delta\tau} \tag{14}$$

where $\boldsymbol{\tau}_{SMC}$ is a sliding mode controller that controls DMR to track desired trajectories, yet, it does not guarantee the robustness to model uncertainties [27]. The term $\boldsymbol{\Delta\tau}$ is a robust saturation controller that handles model uncertainties, i.e., unmodeled dynamics and parameter uncertainties, when it is combined with the sliding mode controller. Later, in Section V, the effectiveness and robustness of our proposed approach is demonstrated experimentally.

In this section, first we design a sliding mode controller for the DMR system with no uncertainty and disturbance (we call it nominal system) to obtain $\boldsymbol{\tau}_{SMC}$. Next, we develop a robust saturation controller, $\boldsymbol{\Delta\tau}$, to achieve a robust tracking for the entire DMR system in the presence of model uncertainties and disturbances.

A. Sliding Mode Controller Design

Under the condition of no uncertainties and disturbances in the DMR, the following nominal system is obtained by letting $\boldsymbol{h}(t)$ to be zero in (12) and substituting $\boldsymbol{\tau}_{SMC}$ into $\boldsymbol{\tau}$:

$$\dot{\boldsymbol{z}} = \overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\left[-\overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} + \overline{\boldsymbol{B}}(\boldsymbol{q})\boldsymbol{\tau}_{SMC}\right]. \tag{15}$$

We design the controller in polar coordinates because it makes it easier to define the sliding surfaces and later prove the stability. Thus, we assume the desired trajectory of the DMR in polar coordinates is given by $\boldsymbol{q_d} = [\rho_{d_1}, \phi_{d_1}, \theta_{d_1}, \theta_{d_2}]^T$, where ρ_{d_1} and ϕ_{d_1} represent the desired radial and angular coordinates of the leader, and θ_{d_1} and θ_{d_2} denote desired orientations of the leader and follower robots, respectively. Similarly, the given general coordinates of the system, $\boldsymbol{q}$, can be obtained in polar coordinates as $\boldsymbol{q}_p = [\rho_1, \phi_1, \theta_1, \theta_2]^T$, by using the expressions given in (1) and (2), and then a transformation from Cartesian to polar coordinates. Also, the derivative of $\boldsymbol{q}_p$ is obtained as follows:

$$\dot{\boldsymbol{q}}_p = \begin{bmatrix}\dot{\rho}_1\\ \dot{\phi}_1\\ \dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix} = \begin{bmatrix} v_1\cos(\phi_1-\theta_1)\\ -\dfrac{v_1}{\rho_1}\sin(\phi_1-\theta_1)\\ \omega_1\\ \omega_2\end{bmatrix}. \tag{16}$$

Before proceeding to the design of sliding mode control, the following assumption on DMR trajectory is given:

Assumption 2. The difference between the leader robot orientation (θ_1) and its angular position (ϕ_1) should not be $\frac{\pi}{2}$. This implies that the leader robot should not have a posture whose orientation is tangential of any circle drawn around the origin of polar coordinates.

We define tracking errors in polar coordinates as $e_\rho = \rho_1 - \rho_{d_1}$, $e_\phi = \phi_1 - \phi_{d_1}$, $e_{\theta_1} = \theta_1 - \theta_{d_1}$, and $e_{\theta_2} = \theta_2 - \theta_{d_2}$. The sliding surface vector is defined as

$$\boldsymbol{S} = \begin{bmatrix}s_1\\ s_2\end{bmatrix} = \begin{bmatrix}\dot{e}_\rho + \lambda_1 e_\rho + \lambda_2\int e_\rho\\ \dot{e}_{\theta_1} + \lambda_3 e_{\theta_1} + \lambda_4\int e_{\theta_1} + \gamma\mathrm{sgn}(e_{\theta_1})|e_\phi|\end{bmatrix} \tag{17}$$

where $\lambda_1, ..., \lambda_4$ and γ are positive constants, and sgn(.) represents sign function. The control law should be chosen such that states reach the sliding surfaces. To do so, we design the controller with the reaching condition of $\dot{\boldsymbol{S}} = 0$, where $\dot{\boldsymbol{S}}$ is

$$\dot{\boldsymbol{S}} = \begin{bmatrix}\dot{s}_1\\ \dot{s}_2\end{bmatrix} = \begin{bmatrix}\ddot{e}_\rho + \lambda_1\dot{e}_\rho + \lambda_2 e_\rho\\ \ddot{e}_{\theta_1} + \lambda_3\dot{e}_{\theta 1} + \lambda_4 e_{\theta_1} + \gamma\mathrm{sgn}(e_{\theta_1})\mathrm{sgn}(e_\phi)\dot{e}_\phi\end{bmatrix}. \tag{18}$$

The control torque input using the computed torque method is expressed as

$$\boldsymbol{\tau}_{SMC} = \overline{\boldsymbol{B}}^{-1}(\boldsymbol{q})\left[\overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} + \overline{\boldsymbol{M}}_0(\boldsymbol{q})(\dot{\boldsymbol{z}}_d + \boldsymbol{u})\right] \tag{19}$$

where $\boldsymbol{u}$ is an auxiliary control variable, $\boldsymbol{u} \equiv [u_1, u_2]^{\mathrm{T}}$, and $z_d = [z_{d_1}, z_{d_2}]^{\mathrm{T}}$ is the desired value of z in (7) that is expressed as

$$\boldsymbol{z_d} = \begin{bmatrix} v_{d1}cos(\theta_{d_1} - \theta_{d_2})\\ \omega_{d_1}\end{bmatrix} \tag{20}$$

where v_{d_1} and ω_{d_1} are desired magnitudes of translational and rotational velocities of the leader robot, respectively. The components of the auxiliary control variable $\boldsymbol{u}$, i.e., u_1 and u_2, are given according to the following theorem:

Theorem 1. The control law given by the following expressions:

$$\begin{aligned} u_1 = \quad & -\frac{\cos(\theta_1-\theta_2)}{\cos(\phi_1-\theta_1)}\Big[\kappa_1 s_1 + \zeta_1\mathrm{sgn}(s_1) - \ddot{\rho}_{d_1} + \lambda_1\dot{e}_\rho \\ & +\lambda_2 e_\rho + \frac{v_1^2}{\rho_1}\sin^2(\phi_1-\theta_1) + v_1\omega_1\sin(\phi_1-\theta_1)\Big] \\ & -v_1(\omega_1-\omega_2)\sin(\theta_1-\theta_2) - \dot{z}_{d_1} \end{aligned} \tag{21}$$

$$\begin{aligned} u_2 = \quad & -\kappa_2 s_2 - \zeta_2\mathrm{sgn}(s_2) - \lambda_3\dot{e}_{\theta_1} - \lambda_4 e_{\theta_1} \\ & -\gamma\mathrm{sgn}(e_{\theta_1})\mathrm{sgn}(e_\phi)\dot{e}_\phi. \end{aligned} \tag{22}$$

stabilizes the sliding surface vector (17).

Proof: Consider the following Lyapunov function:

$$V_1 = \frac{1}{2}\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S}. \tag{23}$$

The sufficient condition for the asymptotic stability of the closed-loop system is

$$\dot{V}_1 = \boldsymbol{S}^{\mathrm{T}}\dot{\boldsymbol{S}} = s_1\dot{s_1} + s_2\dot{s_2} < 0. \tag{24}$$

Before proving the stability, we substitute (19) into (15) to get $\dot{\boldsymbol{z}} = \dot{\boldsymbol{z}}_d + \boldsymbol{u}$, which can be alternatively expressed as

$$\begin{bmatrix}u_1\\ u_2\end{bmatrix} = \begin{bmatrix}\dot{z}_1\\ \dot{z}_2\end{bmatrix} - \begin{bmatrix}\dot{z}_{d_1}\\ \dot{z}_{d_2}\end{bmatrix}. \tag{25}$$

We use (25) in the rest of the proof.

To obtain $\dot{s}_1$, we first find $\ddot{e}_\rho = \ddot{\rho}_1 - \ddot{\rho}_{d_1}$ by getting derivative of $\dot{\rho}_1$ given in (16) that yields

$$\begin{aligned} \ddot{e}_\rho \quad = \quad & \dot{v}_1\cos(\phi_1-\theta_1) + \frac{v_1^2}{\rho_1}\sin^2(\phi_1-\theta_1) \\ & +v_1\omega_1\sin(\phi_1-\theta_1) - \ddot{\rho}_{d_1}. \end{aligned} \tag{26}$$

Also, getting derivative of the first component of $\boldsymbol{z}$ expressed in (7), z_1, follows that

$$\dot{z}_1 = \dot{v}_1\cos(\theta_1-\theta_2) - v_1(\omega_1-\omega_2)\sin(\theta_1-\theta_2). \tag{27}$$

By substituting (26) and (27) into u_1 given in (21), it follows that

$$\begin{aligned} u_1 \quad = \quad & -\frac{\cos(\theta_1-\theta_2)}{\cos(\phi_1-\theta_1)}\left[\kappa_1 s_1 + \zeta_1\mathrm{sgn}(s_1) + \ddot{e}_\rho \right. \\ & \left. +\lambda_1\dot{e}_\rho + \lambda_2 e_\rho\right] + (\dot{z}_1 - \dot{z}_{d_1}). \end{aligned} \tag{28}$$

Alternatively, from (25) it yields that $u_1 = \dot{z}_1 - \dot{z}_{d_1}$. Thus, we further express (28) as $\ddot{e}_\rho = -\kappa_1 s_1 - \zeta_1 \mathrm{sgn}(s_1) - \lambda_1 \dot{e_\rho} - \lambda_2 e_\rho$. Consequently, by substituting $\ddot{e}_\rho$ into $\dot{s}_1$ given by (18), the final expression for $\dot{s}_1$ is obtained as follows:

$$\dot{s}_1 = -\kappa_1 s_1 - \zeta_1 \mathrm{sgn}(s_1). \tag{29}$$

Similarly, for obtaining $\dot{s}_2$, we use $u_2 = \dot{z}_2 - \dot{z}_{d_2}$ given in (25) which can be further written as $u_2 = \ddot{e}_{\theta_1}$ and substitute it into (22) to find $\ddot{e}_{\theta_1}$ as

$$\begin{aligned}\ddot{e}_{\theta_1} \quad = \quad & -\kappa_2 s_2 - \zeta_2 \mathrm{sgn}(s_2) - \lambda_3 \dot{e_{\theta 1}} \\ & -\lambda_4 e_{\theta_1} - \gamma \mathrm{sgn}(e_{\theta_1}) \mathrm{sgn}(e_\phi) \dot{e_\phi}.\end{aligned} \tag{30}$$

Using (30) and $\dot{s}_2$ in (18), $\dot{s}_2$ is alternatively expressed as

$$\dot{s}_2 = -\kappa_2 s_2 - \zeta_2 \mathrm{sgn}(s_2). \tag{31}$$

Finally, we substitute (29) and (31) into (24) to get

$$\dot{V}_1 = -\kappa_1 s_1^2 - \zeta_1 |s_1| - \kappa_2 s_2^2 - \zeta_2 |s_2|. \tag{32}$$

Therefore, for $\dot{V}_1$ to be negative definite, it is sufficient to have κ_1, κ_2, ζ_1, and ζ_2 to be positive. ■

The sliding mode controller designed in this section guarantees the trajectory tracking of the DMR under no system uncertainty and disturbance.

B. Robust Saturation Controller Design

We now design the robust saturation controller to handle external disturbances and model uncertainties. Before designing the controller, the following assumptions are made for proving uniformly ultimately boundedness of the system [28–31]:

Assumption 3. The norm of $\boldsymbol{J}(\boldsymbol{q})$ matrix is upper bounded such that $\|\boldsymbol{J}(\boldsymbol{q})\| \leq J_\alpha$, where $\|.\|$ denotes L_2 norm for vector and induced norm for matrix.

Assumption 4. The inertia matrix $\overline{\boldsymbol{M}}_{\boldsymbol{0}}(\boldsymbol{q})$ is upper bounded, i.e., $\|\overline{\boldsymbol{M}}_{\boldsymbol{0}}(\boldsymbol{q})\| \leq M_\alpha$.

Assumption 5. The upper bound of the norm of control inputs for most robotic systems that do not have acceleration term can be described by a positive function as follows [31]:

$$\|\boldsymbol{\tau}\| \leq \alpha_0 + \alpha_1 \|\boldsymbol{q}\| + \alpha_2 \|\dot{\boldsymbol{q}}\|^2 \tag{33}$$

where $\alpha_0, \alpha_1, \alpha_2$ are positive numbers that are only used in the proof of Lemma 1, and we show later that our methodology does not depend on them.

Lemma 1. The norm of the system uncertainty $\boldsymbol{h}(t)$ in (13) is upper bounded by a positive function described by

$$\|\boldsymbol{h}(t)\| \leq \beta_0 + \beta_1 \|\boldsymbol{q}\| + \beta_2 \|\boldsymbol{z}\| \|\dot{\boldsymbol{J}}\| + \beta_3 \|\boldsymbol{z}\|^2 \tag{34}$$

where positive numbers $\beta_0, \beta_1, \beta_2, \beta_3$ are the upper bound parameters of the system uncertainty which will be estimated later using adaptive laws.

Proof: The proof of Lemma 1 is given in the Appendix. ■

To design the robust saturation controller, let us define $\boldsymbol{\epsilon} \equiv \boldsymbol{z} - \boldsymbol{z}_d$ considering system uncertainties. To find the DMR error dynamics, first we find $\boldsymbol{\tau}$ by substituting (19) into (14), and then, we use the obtained $\boldsymbol{\tau}$ in expression (12) to get

$$\dot{\boldsymbol{z}} = \dot{\boldsymbol{z}}_d + \boldsymbol{u} + \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q}) \overline{\boldsymbol{B}}(\boldsymbol{q}) \boldsymbol{\Delta\tau} + \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q}) \boldsymbol{h}(t). \tag{35}$$

Afterwards, by getting derivative of the defined $\boldsymbol{\epsilon}$ and substituting it into (35), the error dynamics of the DMR can be obtained as follows:

$$\dot{\boldsymbol{\epsilon}} = \boldsymbol{u} + \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q}) \overline{\boldsymbol{B}}(\boldsymbol{q}) \boldsymbol{\Delta\tau} + \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q}) \boldsymbol{h}(t). \tag{36}$$

To design $\boldsymbol{\Delta\tau}$, we choose the Lyapunov function as

$$V_2 = \frac{1}{2} \boldsymbol{\epsilon}^{\mathrm{T}} \boldsymbol{L} \boldsymbol{\epsilon} \tag{37}$$

where $\boldsymbol{L}$ is a 2×2 diagonal matrix with the main diagonals of l_1 and l_2. Therefore, $\boldsymbol{\Delta\tau}$ is designed such that the tracking error, $\boldsymbol{\epsilon}$, converges to zero under large system uncertainties. The discontinuous control law is given as follows:

$$\boldsymbol{\Delta\tau} = \begin{cases} \dfrac{(\nabla V_2^{\mathrm{T}} \boldsymbol{L} \overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q}) \overline{\boldsymbol{B}}(\boldsymbol{q}))^{\mathrm{T}}}{\|\nabla V_2^{\mathrm{T}} \boldsymbol{L} \overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q}) \overline{\boldsymbol{B}}(\boldsymbol{q}))\|^2} \boldsymbol{\nu} & \|\nabla V_2\| \neq 0 \\ 0 & \|\nabla V_2\| = 0 \end{cases} \tag{38}$$

where $\boldsymbol{\nu} = -\nabla V_2^{\mathrm{T}} \boldsymbol{L} \boldsymbol{u} - \|\nabla V_2\| \|\boldsymbol{L} \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q})\| (\beta_0 + \beta_1 \|\boldsymbol{q}\| + \beta_2 \|\boldsymbol{z}\| \|\dot{\boldsymbol{J}}\| + \beta_3 \|\boldsymbol{z}\|^2)$.

The robust saturation control law designed in (38) requires a prior knowledge of the upper bounds of the system uncertainty $(\beta_0, \beta_1, \beta_2, \beta_3)$, which is difficult to access due to nonlinear nature of the DMR system. An adaptive estimation technique is then used to find these parameters. Assuming $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3$ are the estimates and using the following rules:

$$\begin{aligned} \dot{\hat{\beta}}_0 &= k_0 \|\nabla V_2\| \|\boldsymbol{L} \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q})\| && (39) \\ \dot{\hat{\beta}}_1 &= k_1 \|\nabla V_2\| \|\boldsymbol{L} \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q})\| \|\boldsymbol{q}\| && (40) \\ \dot{\hat{\beta}}_2 &= k_2 \|\nabla V_2\| \|\boldsymbol{L} \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q})\| \|\boldsymbol{z}\| \|\dot{\boldsymbol{J}}\| && (41) \\ \dot{\hat{\beta}}_3 &= k_3 \|\nabla V_2\| \|\boldsymbol{L} \overline{\boldsymbol{M}}_{\boldsymbol{0}}^{-1}(\boldsymbol{q})\| \|\boldsymbol{z}\|^2 && (42) \end{aligned}$$

the upper bound parameters of $\|\boldsymbol{h}(t)\|$ can be estimated. In expressions (39)-(42), the positive constants k_0, k_1, k_2, k_3 and initial values of $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3$ can be selected arbitrarily.

Theorem 2. Using the control law given by sliding mode controller in (19) and robust saturation controller in (38), the system (12) is uniformly ultimately bounded.

Proof: Consider the following function which is a part of the Lyapunov candidate as

$$V_3 = \frac{1}{2} \sum_{j=0}^{3} k_j^{-1} \left(\beta_j - \hat{\beta}_j \right)^2. \tag{43}$$

Therefore, using (23), (37), and (43), the complete Lyapunov function is

$$\begin{aligned} V &= V_1 + V_2 + V_3 \\ &= \frac{1}{2} \boldsymbol{S}^{\mathrm{T}} \boldsymbol{S} + \frac{1}{2} \boldsymbol{\epsilon}^{\mathrm{T}} \boldsymbol{L} \boldsymbol{\epsilon} + \frac{1}{2} \sum_{j=0}^{3} k_j^{-1} \left(\beta_j - \hat{\beta}_j \right)^2. \end{aligned} \tag{44}$$

Observe that $V > 0$, we now show $\dot{V} < 0$. In the proof of Theorem 1, it was shown that $\dot{V}_1 < 0$ along the system

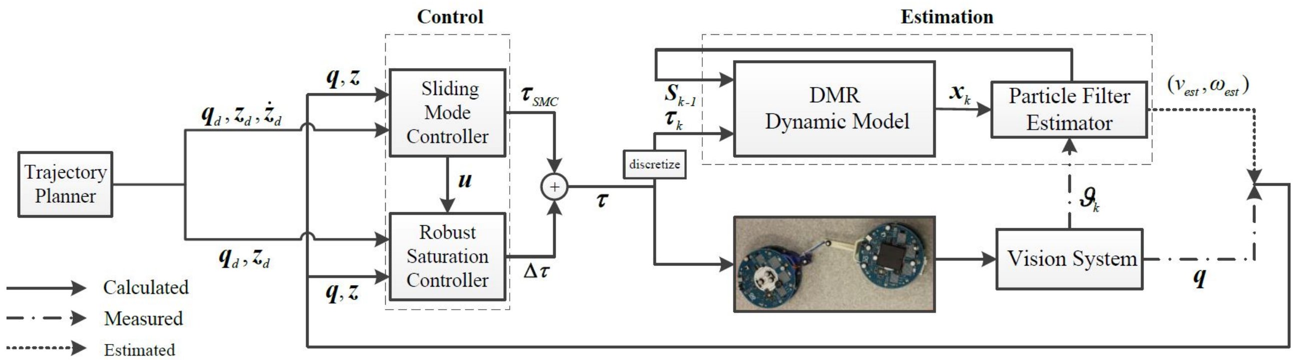

Figure 2: Schematic diagram of proposed methodology.

trajectories. Thus, it is sufficient to prove that $\dot{V}_2 + \dot{V}_3 < 0$. Consider the following two cases:

1) If $\|\nabla V_2\| \neq 0$, one has

$$\begin{aligned}
\dot{V}_2 &= \frac{1}{2}\dot{\boldsymbol{\epsilon}}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{\epsilon} + \frac{1}{2}\boldsymbol{\epsilon}^{\mathrm{T}}\boldsymbol{L}\dot{\boldsymbol{\epsilon}} \\
&= \nabla V_2^{\mathrm{T}}\boldsymbol{L}\dot{\boldsymbol{\epsilon}} \\
&= \nabla V_2^{\mathrm{T}}\boldsymbol{L}\left[\boldsymbol{u} + \overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\overline{\boldsymbol{B}}(\boldsymbol{q})\Delta\boldsymbol{\tau} + \overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\boldsymbol{h}(t)\right] \\
&= \nabla V_2^{\mathrm{T}}\boldsymbol{L}\boldsymbol{u} + \boldsymbol{\nu} + \nabla V_2^{\mathrm{T}}\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\boldsymbol{h}(t) \\
&= \nabla V_2^{\mathrm{T}}\boldsymbol{L}\boldsymbol{u} - \Big[\nabla V_2^{\mathrm{T}}\boldsymbol{L}\boldsymbol{u} + \|\nabla V_2\|\|\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\| \\
&\quad \times \left(\hat{\beta}_0 + \hat{\beta}_1\|\boldsymbol{q}\| + \hat{\beta}_2\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| + \hat{\beta}_3\|\boldsymbol{z}\|^2\right)\Big] \\
&\quad + \nabla V_2^{\mathrm{T}}\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\boldsymbol{h}(t) \\
&= \nabla V_2^{\mathrm{T}}\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\boldsymbol{h}(t) - \|\nabla V_2\|\|\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\| \\
&\quad \times \left(\hat{\beta}_0 + \hat{\beta}_1\|\boldsymbol{q}\| + \hat{\beta}_2\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| + \hat{\beta}_3\|\boldsymbol{z}\|^2\right) \qquad (45)
\end{aligned}$$

where $\times$ denotes multiplication. Also, $\dot{V}_3$ can be obtained as follows:

$$\begin{aligned}
\dot{V}_3 &= -\sum_{j=0}^{3} k_j^{-1}\dot{\hat{\beta}}_j\left(\beta_j - \hat{\beta}_j\right) \\
&= -\|\nabla V_2\|\|\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\| \\
&\quad \times\Big[(\beta_0 - \hat{\beta}_0) + (\beta_1 - \hat{\beta}_1)\|\boldsymbol{q}\| \\
&\quad + (\beta_2 - \hat{\beta}_2)\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| + (\beta_3 - \hat{\beta}_3)\|\boldsymbol{z}\|^2\Big] \qquad (46)
\end{aligned}$$

Subsequently, using (45) and (46), $\dot{V}_2 + \dot{V}_3$ is expressed as

$$\begin{aligned}
\dot{V}_2 + \dot{V}_3 &= \nabla V_2^{\mathrm{T}}\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\boldsymbol{h}(t) - \|\nabla V_2\|\|\boldsymbol{L}\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\| \\
&\quad \times\left(\beta_0 + \beta_1\|\boldsymbol{q}\| + \beta_2\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| + \beta_3\|\boldsymbol{z}\|^2\right) \\
&< 0
\end{aligned}$$

2) If $\|\nabla V_2\| = 0$, it is straightforward to show $\dot{V}_2 + \dot{V}_3 \leq 0$. Therefore, for both cases, $\dot{V} < 0$. ■

Note that the components of the auxiliary control variable $\boldsymbol{u}$ given in (21) and (22) are a function of the leader's translational and rotational velocities (v_1 and ω_1). However, in practice, measuring absolute velocity of the robots is not possible due to the limited sensing capability of available sensors. In the following section, we develop an estimator to estimate these velocities.

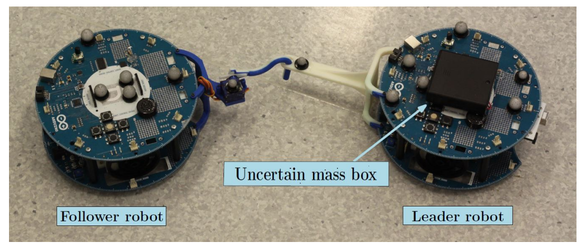

Figure 3: Experimental DMR.

IV. Estimator Design

In this section, we propose an estimator that fuses sensory data and dynamics of the DMR to estimate the required velocities. Considering the nonlinearities in the DMR system dynamics, a nonlinear estimator such as extended Kalman filter, unscented Kalman filter, or particle filter, has to be developed. We choose particle filter because of its capability in dealing with non-Gaussian and nonlinear systems [32].

To develop the particle filter, we discretize the system with constant velocity assumption and designate the discrete current state vector as $\boldsymbol{x}_k = [x_1, y_1, \theta_1, \theta_2, \dot{x}_1, \dot{y}_1, \dot{\theta}_1, \dot{\theta}_2]^{\mathrm{T}}$, that is obtained at each time step using the state information from the dynamic model (9), the transformation (6), and the expressions (1) and (2). To predict the velocities at each time step, we define the following measurement model:

$$\boldsymbol{y}_k = \begin{bmatrix} \sqrt{\dot{x}_1^2 + \dot{y}_1^2} \\ \dot{\theta}_1 \end{bmatrix}. \qquad (47)$$

In addition, DMR translational and rotational velocities are measured at time k by differentiating the docked robots' position and orientation feedback from a centralized vision system, i.e., v_m and ω_m, that might be noisy or subjected to error. To model the velocity measurements realistically, we assume that these measurements are also corrupted by an additive noise. Hence, the measured velocity vector is $\boldsymbol{\vartheta}_k = [v_m, \omega_m]^{\mathrm{T}} + \boldsymbol{\delta}_k$, where $\boldsymbol{\delta}_k$ can be a zero mean white Gaussian or non-Gaussian noise.

Our particle filter algorithm is developed to generate a particle set which can be the best representation of the true velocities. Each particle set is composed of current state sample and its weight. Our particle filter takes the current torque and prior particle sets as its input to estimate the true

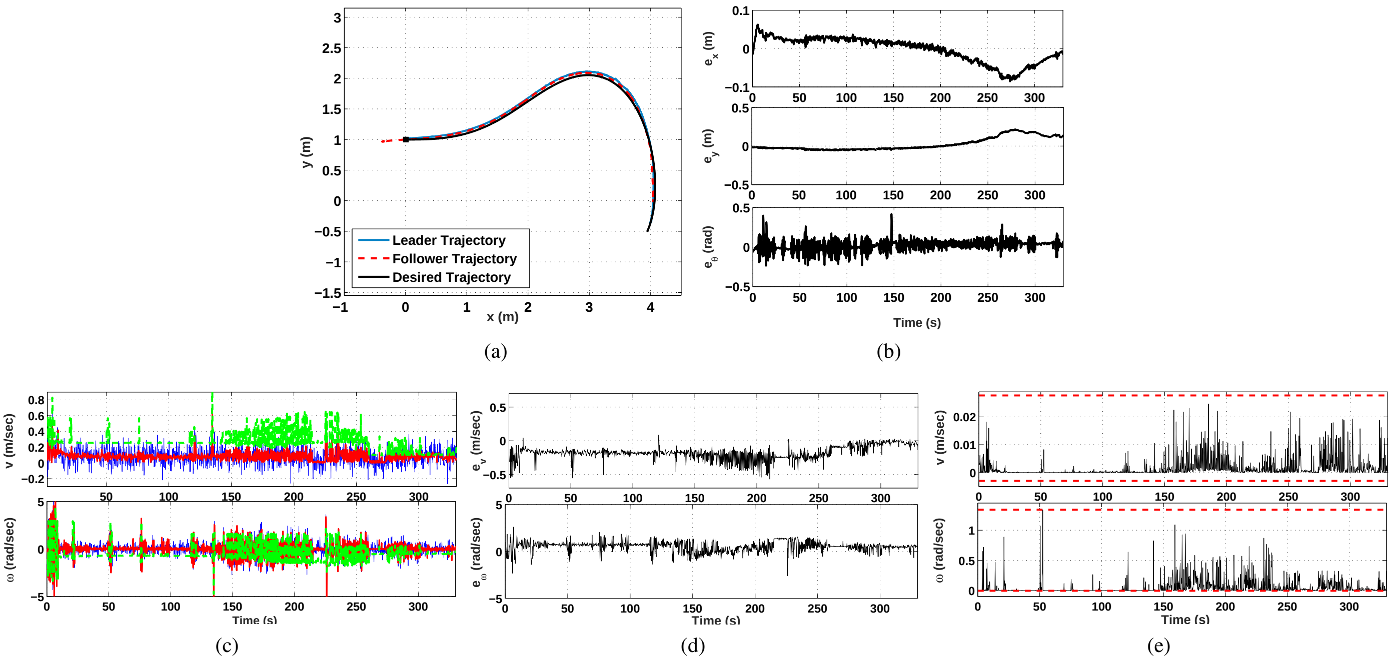

Figure 4: Experimental results of the first trajectory under measurement noise: (a) leader-follower trajectories; (b) tracking error; (c) estimated (green), measured (blue), and true (red) states; (d) estimation error; (e) velocity error covariance.

Table II: System and control design parameters.

Parameter	Value	Parameter	Value
b	0.0850 m	$\lambda_1, \lambda_2, \lambda_3, \lambda_4$	400
h_1, h_2	0.05 m	κ_1, κ_2	25, 15
m_1	0.65 Kg	ζ_1, ζ_2	25, 15
m_2	0.55 Kg	γ	400
m_o	0.050 Kg	l_1, l_2	10, 5
I_1, I_2	0.0026 Kg.m^2	k_0, k_1, k_2, k_3	12

belief of velocities, v_{est} and ω_{est}, using dynamic model (9) and measurement model (47). The dynamic model (9) is used to propagate samples forward in each time step based on the previous particles and the current inputs. The measurement model (47) is used to predict the velocities using samples of current state vector. To assign a weight to each state sample, we find the normal probability density function of measured velocity vector $\boldsymbol{\vartheta}_k$ using a normal distribution centered at each sample of predicted velocity vector. Afterwards, resampling is done with the purpose of transforming the predicted belief particle set to belief using importance sampling. If we consider the current input as $\boldsymbol{\tau}_k$ and prior state as $\boldsymbol{x}_{k-1}$, then we can define the measurement model in (47) as $p(\boldsymbol{y}_k|\boldsymbol{x}_k)$ which is the probability of measurement $\boldsymbol{y}$ occurring at time k given the state $\boldsymbol{x}$, and similarly, we can define the motion model given in (9) as $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\boldsymbol{\tau}_k)$. Assuming the total number of particles is I, the distribution of samples for predicted belief of current state and measurement vector as well as the belief of current state and measurement vector respectively are denoted by $\bar{\boldsymbol{x}}_k^{[i]}$, $\bar{\boldsymbol{y}}_k^{[i]}$, $\boldsymbol{x}_k^{[i]}$, and $\boldsymbol{y}_k^{[i]}$, for $i = 1, ..., I$. Each particle set, $\boldsymbol{s}_k^{[i]}$, is a combination of the sample ($\boldsymbol{x}_k^{[i]}$) and its weight ($\boldsymbol{w}_k^{[i]}$) that is shown with $\boldsymbol{s}_k^{[i]}$=$\{\boldsymbol{x}_k^{[i]}$,$\boldsymbol{w}_k^{[i]}\}$. We now present the details of our developed particle filter velocity estimator in Algorithm 1.

Algorithm 1 Particle Filter for Velocity Estimation

Inputs: Prior particle sets $\boldsymbol{S}_{k-1} = \{\boldsymbol{s}_{k-1}^{[1]}, ..., \boldsymbol{s}_{k-1}^{[I]}\}$, current input $\boldsymbol{\tau}_k$, and measured velocity $\boldsymbol{\vartheta}_k$
Outputs: True belief of velocities v_{est} and ω_{est}
for each particle in $\boldsymbol{S}_{k-1}$ **do**
Sampling: propagate sample forward using motion model (9): $\bar{\boldsymbol{x}}_k^{[i]} \sim p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{[i]}, \boldsymbol{\tau}_k)$
Prediction: predicting measurements using sample $\bar{\boldsymbol{x}}_k^{[i]}$ and measurement model (47): $\bar{\boldsymbol{y}}_k^{[i]} \sim p(\boldsymbol{y}_k|\bar{\boldsymbol{x}}_k^{[i]})$
Weighting: define weights from measured velocities $\boldsymbol{\vartheta}_k$, with normal distribution centered at predictions $\bar{\boldsymbol{y}}_k^{[i]}$: $\boldsymbol{w}_k^{[i]}$
Store in interim particle set: $\bar{\boldsymbol{S}}_k = \bar{\boldsymbol{S}}_k + \{\bar{\boldsymbol{s}}_k^{[i]}\}$, where $\bar{\boldsymbol{s}}_k^{[i]} = \{\bar{\boldsymbol{x}}_k^{[i]}, \boldsymbol{w}_k^{[i]}\}$.
end for
Normalize weights
for each particle in $\bar{\boldsymbol{S}}_k$ **do**
Resampling: draw particle $\bar{\boldsymbol{x}}_k^{[i]}$ with probability $\boldsymbol{w}_k^{[i]}$: $\boldsymbol{s}_k^{[i]} = \{\boldsymbol{x}_k^{[i]}, \boldsymbol{w}_k^{[i]}\}$
Add to final particle set: $\boldsymbol{S}_k = \boldsymbol{S}_k + \{\boldsymbol{s}_k^{[i]}\}$
Calculate true velocities through measurement model (47): $\boldsymbol{y}_k^{[i]}$=$[\boldsymbol{v}_k^{[i]}, \boldsymbol{\omega}_k^{[i]}]^{\mathrm{T}} = p(\boldsymbol{y}_k|\boldsymbol{x}_k^{[i]})$
end for
Calculate mean of $\boldsymbol{v}_k^{[i]}$ and mean of $\boldsymbol{\omega}_k^{[i]}$ over $i = 1, ..., I$ to get v_{est} and ω_{est}.

where I is the total number of particles, $\boldsymbol{s}_k^{[i]}$ is the combination of sample and weight of i^{th} particle, and $\bar{\boldsymbol{x}}_k^{[i]}$, $\bar{\boldsymbol{y}}_k^{[i]}$, $\boldsymbol{x}_k^{[i]}$, $\boldsymbol{y}_k^{[i]}$ are respectively the i^{th} predicted belief of states, predicted belief of measurements, current belief of states and current belief of measurements, all at time k for $i = 1, ..., I$.

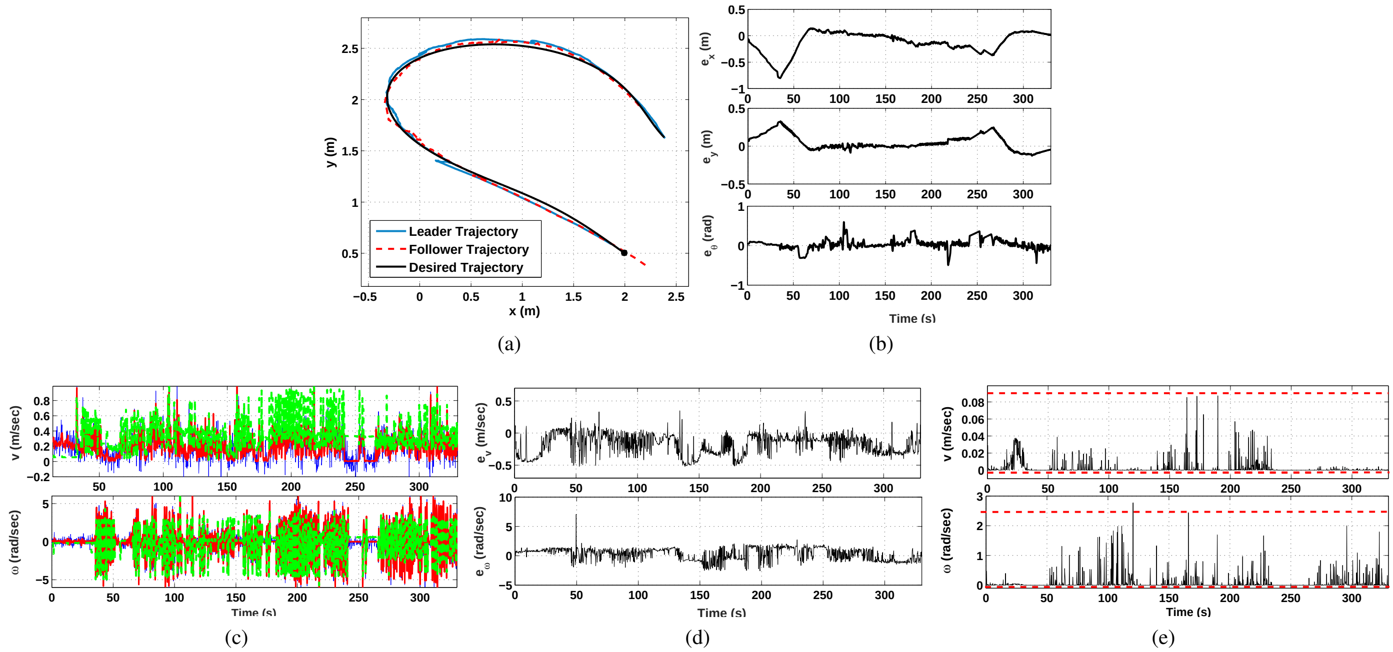

Figure 5: Experimental results of the second trajectory under unmodeled dynamics (a) leader-follower trajectories; (b) tracking error; (c) estimated (green), measured (blue), and true (red) states; (d) estimation error; (e) velocity error covariance.

In summary, Algorithm 1 inputs the prior particle sets, current control input and velocity measurements, and consequently it outputs the true belief of leader's translational and rotational velocities. To verify the developed controller and estimator algorithms, experiments were carried out and are presented in the next section.

V. Experimental Studies

The entire closed-loop system composed of a trajectory planner, particle filter estimator, and robust controller is shown in Fig. 2. The trajectory planner generates desired trajectories required for the controller to provide the input torque of the leader's wheels, i.e., $\boldsymbol{\tau}$. Our proposed controller also inputs the position, and translational and rotational velocities of the DMR. The position of the DMR, $\boldsymbol{q}$, can be measured accurately using Vicon system. By getting the derivatives of these terms, we can find the translational and rotational velocities. However, to model velocities realistically that can behave like real sensory data, we corrupt these signals by white Gaussian noise and then we deploy an estimator to obtain the velocity profile more accurately. The particle filter estimator fuses the measured velocity $\boldsymbol{\vartheta}_k$ obtained by Vicon, and current state $\boldsymbol{x}_k$ which is calculated using the DMR dynamic model. Afterwards, the estimated velocities (v_{est} and ω_{est}), measured states ($\boldsymbol{q}$), and desired states ($\boldsymbol{q}_d$, $\boldsymbol{z}_d$ and $\dot{\boldsymbol{z}}_d$) are used by the controller to generate the input torque required for control.

A. Experimental Setup

Two Arduino mobile robots docked via two rigid links connected through an off-axle hitch were used in experiments. The experimental setup is shown in Fig. 3. The robots were equipped with wireless Xbee modules for communication with the computer with the sampling frequency of 10 Hz.

For tracking the DMR position and orientation, we used the Vicon vision system. The translational and rotational velocities were given by the Vicon, and the known sampling rate of 0.1 sec. We later added noise to these. The control input of the leader robot is input voltage in the range of $[-5V, 5V]$.

The parameters of the DMR system and the controller used for experiments are given in Table II. To determine the control design parameters prior to experiments, we used Genetic Algorithm. For implementing the particle filter, it was assumed that the initial velocities are unknown and the particles are spread randomly. For the sake of accuracy and calculation time, the number of particles was assumed to be 200 that enables the estimator to perform online estimation. We assumed a zero mean white Gaussian noise $\boldsymbol{\delta_k}$ is added to the velocity profile given by Vicon, where standard deviations are 0.1 m/sec and 0.3 rad/sec for v_1 and ω_1, respectively.

B. Experimental Results

The performances of our methodology are demonstrated in five sections. In sections 1 to 3, we present the results under three case studies: (1) measurement noise, (2) unmodeled dynamics, and (3) parametric uncertainty. In section 4, we compare the performance of our proposed control strategy with the case that only sliding mode is deployed. In section 5, we compare our results with previously developed controllers in the literature.

1) Robustness to measurement noise: to investigate the robustness of our approach to measurement noise, we added a white Gaussian noise with the standard deviation of 0.4 m for both x_1 and y_1, and 0.05 rad for θ_1. Fig. 4 presents

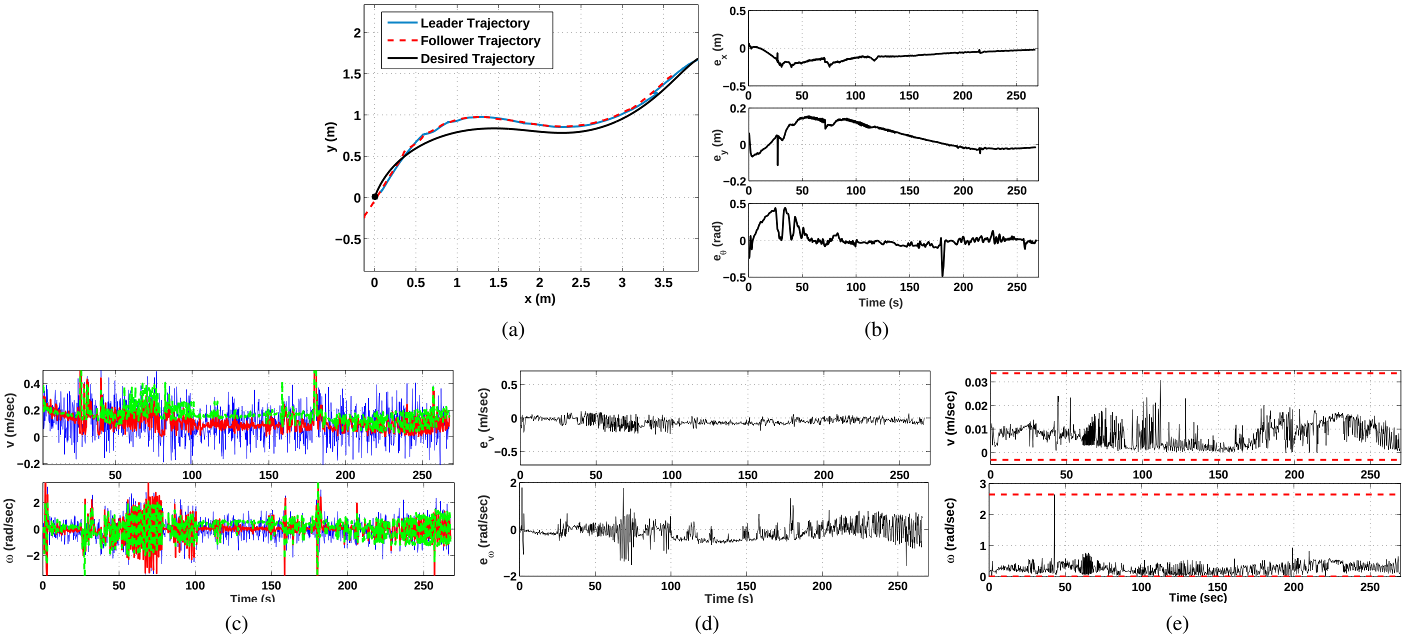

Figure 6: Experimental results of the third trajectory under parametric uncertainty: (a) leader-follower trajectories; (b) tracking error; (c) estimated (green), measured (blue), and true (red) states; (d) estimation error; (e) velocity error covariance.

the experimental results. The DMR trajectories are shown in Fig. 4a, where the black square denotes the starting point of the desired trajectory. To show how the developed particle filter estimates the velocities, Figs. 4c, 4d, and 4e are provided. In Figs. 4c and 4d convergence of the estimated velocities to the true values obtained by Vicon and the estimation errors to zero are illustrated, respectively. The convergence and boundedness of the estimation errors covariance are also shown in Fig. 4e. From these graphs, it is evident that the estimator is capable of estimating the required states with good accuracy. The tracking performance of the integrated estimator and controller system is shown to be good in Fig. 4b. As can be seen, the tracking error of the controller is small. The peak observed in tracking error is because of the orientation mismatch between the leader and follower robots arising from turning on the trajectory, thus making the robots to have some errors in following their paths. After passing the turning point, the tracking error gets smaller as it is demonstrated. The root mean square of errors ($\boldsymbol{e}_{rms}$) are 6 cm, 6.8 cm, 0.07 rad for x, y, θ, respectively.

2) Robustness to unmodeled dynamics: to implement unmodeled dynamics, we set $h_1 = 0$ (on-axle hitched DMR) in the dynamics (9) that is used in estimation and control while in the experiments $h_1 \neq 0$ (off-axle hitched DMR shown in Fig. 3). Fig. 5 displays the estimation and control results under unmodeled dynamics effects. The estimation results provided in Figs. 5c, 5d, and 5e show that the outputs have converged to true velocities obtained by Vicon. It is shown in Figs. 5a and 5b that DMR successfully tracks the trajectory with small errors, where $\boldsymbol{e}_{rms}$ is 19.7 cm, 6.8 cm, 0.09 rad respectively for x, y, θ.

3) Robustness to parametric uncertainty: we assess the performance of the proposed approach in the presence of parametric uncertainty, which is important in loading and unloading applications. To do so, we added 15% mass uncertainty to the leader robot using the uncertain mass box shown in Fig. 3. The results are shown in Fig. 6. From Figs. 6c, 6d and 6e it is evident that the designed particle filter estimator converges to true DMR velocities. Tracking results are demonstrated in Figs. 6a and 6b which confirms the robust tracking performance of our approach. The $\boldsymbol{e}_{rms}$ was found to be 17.2 cm, 9.5 cm, 0.15 rad for x, y, θ, respectively.

4) Comparison with sliding mode controller: in this part, we compare the performance of our proposed controller in (14) with the case when only sliding mode is deployed. Both controllers were integrated with the developed estimator and tested under the same conditions, i.e., measurement noise, unmodeled dynamics and parametric uncertainties To further investigate the robustness of our proposed methodology, we tested each condition for three different trajectories (three trials). In Table III, the tracking performance of controllers are compared in terms of $\boldsymbol{e}_{rms}$ and the percentage maximum tracking error during steady state phase (e_{ss}). This table shows that our proposed robust controller outputs lower tracking error and improves robustness under measurement noise and model uncertainties in comparison to the case when only sliding mode controller is used. In average, the improvement for measurement noise, unmodeled dynamics, and parametric uncertainties for three trials, are respectively 5.1%, 22.2%, and 28.7% during the steady state[1]. It has not escaped our notice that horizontal steady state tracking error of the pure sliding mode controller seems smaller than our proposed controller; however, by taking into account both horizontal and vertical tracking errors in steady state, we can conclude that the robust controller still outperforms the sliding mode controller. As also

[1] Average of steady state error for each trial: $\dfrac{e_{x_{ss}} + e_{y_{ss}} + e_{\theta_{ss}}}{3}$

Table III: Comparing tracking performance of sliding mode controller ($\boldsymbol{\tau}_{SMC}$) and robust controller ($\boldsymbol{\tau}_{SMC} + \boldsymbol{\Delta\tau}$).

Case	Control law	$(e_{x_{rms}}, e_{y_{rms}}, e_{\theta_{rms}})$ (cm, cm, rad)			$(e_{x_{ss}}, e_{y_{ss}}, e_{\theta_{ss}})\%$		
		Trial 1	Trial 2	Trial 3	Trial 1	Trial 2	Trial 3
Measurement noise	$\boldsymbol{\tau}_{SMC}$	(5.5, 10.1, 0.15)	(6.0, 8.2, 0.08)	(19.7, 19.5, 0.28)	(0.05, 27.7, 12.1)	(1.5, 5.9, 34.8)	(6.6, 7.8, 29.2)
	$\boldsymbol{\tau}_{SMC} + \boldsymbol{\Delta\tau}$	(6, 6.8, 0.07)	(5.1, 6, 0.06)	(17.3, 11.7, 0.26)	(4.7, 12, 9.4)	(0.02, 5.5, 18.5)	(5.1, 2.5, 20.5)
Unmodeled dynamics	$\boldsymbol{\tau}_{SMC}$	(24.1, 17, 0.16)	(35.0, 27, 0.32)	(17.5, 35.4, 0.44)	(10.3, 11.3, 31.2)	(4.4, 13.1, 75.2)	(9.5, 28.4, > 100)
	$\boldsymbol{\tau}_{SMC} + \boldsymbol{\Delta\tau}$	(19.7, 6.8, 0.09)	(5.6, 4.8, 0.08)	(6.3, 14.7, 0.24)	(4.4, 4.9, 24.1)	(0.1, 7.6, 18.9)	(1.08, 3.5, 18.4)
Parametric uncertainties	$\boldsymbol{\tau}_{SMC}$	(21, 15.9, 0.19)	(21.5, 34.8, 0.4)	(21.0, 17.3, 0.33)	(3.9, 2.1, > 100)	(2.8, 24.9, 77.9)	(5.3, 10.9, > 100)
	$\boldsymbol{\tau}_{SMC} + \boldsymbol{\Delta\tau}$	(17.2, 9.5 , 0.15)	(8, 4.1, 0.07)	(12.8, 13.2, 0.28)	(3.7, 5.0, 17.8)	(0.01, 1.7, 21.8)	(1.1, 2.2, 15.4)

Table IV: Performance of different controllers under parametric uncertainties (Trial 1) and unmodeled dynamics (Trial 2 and 3).

Control method	$(e_{x_{rms}}, e_{y_{rms}}, e_{\theta_{rms}})$ (cm, cm, rad)			$(e_{x_{ss}}, e_{y_{ss}}, e_{\theta_{ss}})\%$			$(\sigma_{u_r}, \sigma_{u_l})$ (V)		
	Trial 1	Trial 2	Trial 3	Trial 1	Trial 2	Trial 3	Trial 1	Trial 2	Trial 3
Our method	(17.2, 9.5, 0.15)	(5.6, 4.8, 0.08)	(6.3, 14.7, 0.24)	(3.7, 5.0, 17.8)	(0.1, 7.6, 18.9)	(1.08, 3.5, 18.4)	(2.9, 2.5)	(3.2, 3.1)	(3.5, 3.2)
Lyapunov [33]	(16.2, 28.3, 0.24)	(12.6, 13.2, 0.09)	(22.9, 23.1, 0.36)	(4.0, 12.7, 100)	(0.8, 16.8, 80.6)	(10.3, 5.7, > 100)	(2.2, 1.7)	(2.9, 2.8)	(2.4, 1.9)
PD	(46.4, 28.8 , 0.3)	(37.5, 29.2, 0.34)	(28.1, 19.5, 0.31)	(12.3, 24.7 , > 100)	(5.2, 17.7, 87.5)	(16.6, 7.5, > 100)	(0.8, 0.9)	(2.5, 1.7)	(2.6, 2.4)
PID	(9.5, 12.7, 0.16)	(14.4, 10.6, 0.12)	(29.1, 20.8, 0.37)	(4.0, 1.0, > 100)	(4.6, 16.5, 80.9)	(13.4, 4.7, > 100)	(1.7, 1.9)	(2.1, 2.0)	(3.1, 2.9)

stated previously, the pure sliding mode controller is capable of handling some disturbances and measurement noises, yet it does not guarantee robustness under model uncertainties. Therefore, it is reasonable that the tracking performance improvement of our controller over the sliding mode controller is not much distinctive under measurement noise.

5) Comparison with other controllers: we also compared the performance of our controller with previously developed controllers in the literature. The experiments were conducted with three various trajectories under parametric uncertainties (Trial 1) and unmodeled dynamics (Trials 2 and 3) for different controllers, i.e., Lyapunov-based [33], PD, and PID controllers. Results are summarized in Table IV based on the tracking error and control effort. In this table, the control effort is represented by the standard deviation (σ) of input voltage range for each wheel of the leader robot (right and left). The results verify that our controller outperforms other controllers in terms of tracking and robustness, although it requires a bit more control effort. It is worth noting that among the presented controllers, the tracking performance of the PID controller is comparable with our methodology under parametric uncertainties (trial 1); however, it is obvious in Table IV that the steady state error for orientation obtained by implementing our proposed approach is significantly smaller than the PID control. Moreover, applying unmodeled dynamic uncertainties (Trial 2 and 3), the advantage of our robust controller can be observed more apparent over standard PID control. In average, the improvement that our proposed controller makes over PID controller in tracking three reference trajectories are respectively 26.1%, 25.1% and 31.6% during the steady state. In tuning all of these controllers, we noticed that our controller is less sensitive to the design parameters while others require careful tuning of control parameters and are significantly sensitive to the parameters change.

VI. Conclusion

In this paper, we developed a new integrated system composed of a robust tracking controller and an estimator for a nonholonomic docked mobile robotic system without having access to translational and rotational velocities directly. Our extensive experimental results showed that, first, the developed estimator is able to estimate the velocities with good accuracy required for control, and second, the integrated estimator and controller have very good tracking performance under measurement noise, unmodeled dynamics, and parameter uncertainties. It was concluded that if only the sliding mode control method (integrated with the estimation) is implemented on the robots, tracking performance is deteriorated in the presence of uncertainties. Thus, we showed that the robust saturation control has to be combined with the sliding mode control to achieve robust tracking, especially with parametric uncertainties that can decrease the steady state tracking error by up to 28.7%. Experimental results were also compared to previously developed control methods in the literature and it was demonstrated that the performance of our proposed methodology is superior to other developed controllers in the literature in terms of tracking accuracy and robustness to model uncertainties. Moreover, our controller requires less tuning effort compared to other methods due to being less sensitive to parameter changes. Therefore, the proposed methodology can be used for tracking applications without relying on velocity measurements and hence making it a viable approach for effective navigation of DMR systems.

Appendix

Proof of Lemma 1: substituting (12) into (13) gives

$$\begin{aligned} \boldsymbol{h}(t) \;=\; & \Big(\boldsymbol{I} + \boldsymbol{\Delta M}(\boldsymbol{q})\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\Big)^{-1} \\ & \times \Big[\boldsymbol{\Delta M}(\boldsymbol{q})\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\left(\overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} - \overline{\boldsymbol{B}}(\boldsymbol{q})\boldsymbol{\tau}\right) \\ & -\boldsymbol{\Delta V}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z} - \boldsymbol{\tau}_d\Big]. \end{aligned} \tag{48}$$

Using (48), it follows that

$$\begin{aligned} \|\boldsymbol{h}(t)\| \;\leq\; & \|\Big(\boldsymbol{I} + \Delta\boldsymbol{M}(\boldsymbol{q})\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\Big)^{-1}\| \\ & \times \Big[\|\Delta\boldsymbol{M}(\boldsymbol{q})\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\| \times \big(\|\overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z}\| \\ & +\|\overline{\boldsymbol{B}}(\boldsymbol{q})\|\|\boldsymbol{\tau}\|\big) + \|\Delta\boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z}\| + \|\boldsymbol{\tau}_d\|\Big] \end{aligned} \tag{49}$$

From the Assumptions 1 to 4, each term of (49) is bounded from above as follows:

$$\|(\boldsymbol{I}+\Delta\boldsymbol{M}(\boldsymbol{q})\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q}))^{-1}\| < a_1 \quad (50)$$

$$\|\Delta\boldsymbol{M}(\boldsymbol{q})\overline{\boldsymbol{M}}_0^{-1}(\boldsymbol{q})\| < a_2 \quad (51)$$

$$\|\overline{\boldsymbol{V}}_0(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z}\| < a_3 + a_4\|\boldsymbol{q}\| + a_5\|\dot{\boldsymbol{q}}\|^2 + a_6\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| \quad (52)$$

$$\|\Delta\boldsymbol{V}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{z}\| < a_7 + a_8\|\boldsymbol{q}\| + a_9\|\dot{\boldsymbol{q}}\|^2 + a_{10}\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| \quad (53)$$

$$\|\overline{\boldsymbol{B}}(\boldsymbol{q})\| < a_{11} \quad (54)$$

$$\|\boldsymbol{\tau}_d\| < d_1 \quad (55)$$

where $a_1, ..., a_{11}$ are positive. Using (50)-(55) and Assumption 5, upper bound of $\|\boldsymbol{h}(t)\|$ can be obtained as follows:

$$\begin{aligned}\|\boldsymbol{h}(t)\| \leq\; & a_1(a_2a_3 + a_7 + a_2a_{11}\alpha_0 + d_1)\\ & + a_1(a_2a_4 + a_8 + a_2a_{11}\alpha_1)\|\boldsymbol{q}\|\\ & + a_1(a_2a_6 + a_{10})\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\|\\ & + a_1(a_2a_5 + a_2a_3a_{11}\alpha_2 + a_9)\|\dot{\boldsymbol{q}}\|^2\end{aligned}$$

Using (6) and Assumption 3, it yields that $\|\dot{\boldsymbol{q}}\| \leq J_\alpha\|\boldsymbol{z}\|$. Thus, upper bound of $\|\boldsymbol{h}(t)\|$ can be alternatively expressed as

$$\|\boldsymbol{h}(t)\| \leq \beta_0 + \beta_1\|\boldsymbol{q}\| + \beta_2\|\boldsymbol{z}\|\|\dot{\boldsymbol{J}}\| + \beta_3\|\boldsymbol{z}\|^2 \quad (56)$$

where

$$\beta_0 = a_1(a_2a_3 + a_7 + a_2a_{11}\alpha_0 + d_1) \quad (57)$$

$$\beta_1 = a_1(a_2a_4 + a_8 + a_2a_{11}\alpha_1) \quad (58)$$

$$\beta_2 = a_1(a_2a_6 + a_{10}) \quad (59)$$

$$\beta_3 = a_1J_\alpha(a_2a_5 + a_2a_3a_{11}\alpha_2 + a_9). \quad (60)$$

Therefore, $\boldsymbol{h}(t)$ is upper bounded as shown in (56).

REFERENCES

[1] J. Backman, T. Oksanen, and A. Visala, "Navigation system for agricultural machines: nonlinear model predictive path tracking," *Computers and Electronics in Agriculture*, vol. 82, pp. 32–43, 2012.

[2] J. Morales, J. L. Martinez, A. Mandow, J. Seron, and A. J. Garcia-Cerezo, "Static tip-over stability analysis for a robotic vehicle with a single-axle trailer on slopes based on altered supporting polygons," *IEEE/ASME Transactions on Mechatronics*, vol. 18, pp. 697–705, April 2013.

[3] J. P. Laumond, "Controllability of a multibody mobile robot," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 6, pp. 755–763, 1993.

[4] Z. Echegoyen, I. Villaverde, R. Moreno, M. Graña, and A. d'Anjou, "Linked multi-component mobile robots: modeling, simulation and control," *Robotics and Autonomous Systems*, vol. 58, no. 12, pp. 1292–1305, 2010.

[5] J. Yuan, F. Sun, and Y. Huang, "Trajectory generation and tracking control for double-steering tractor-trailer mobile robots with on-axle hitching," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7665–7677, 2015.

[6] A. K. Khalaji and S. A. A. Moosavian, "Robust adaptive controller for a tractor-trailer mobile robot," *IEEE/ASME Transactions on Mechatronics*, vol. 19, pp. 943–953, June 2014.

[7] J.-H. Lee, W. Chung, M. Kim, and J.-B. Song, "A passive multiple trailer system with off-axle hitching," *International Journal of Control Automation and Systems*, vol. 2, pp. 289–297, 2004.

[8] J. David and P. Manivannan, "Control of truck-trailer mobile robots: a survey," *Intelligent Service Robotics*, vol. 7, no. 4, pp. 245–258, 2014.

[9] M. Defoort, T. Floquet, A. Kokosy, and W. Perruquetti, "Sliding-mode formation control for cooperative autonomous mobile robots," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 11, pp. 3944–3953, 2008.

[10] V. Gazi, "Swarm aggregations using artificial potentials and sliding-mode control," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1208–1214, 2005.

[11] B.-S. Chen, T.-S. Lee, and J.-H. Feng, "A nonlinear H_∞ control design in robotic systems under parameter perturbation and external disturbance," *International Journal of Control*, vol. 59, no. 2, pp. 439–461, 1994.

[12] H. Liu, X. Tian, G. Wang, and T. Zhang, "Finite-time H_∞ control for high-precision tracking in robotic manipulators using backstepping control," *IEEE Transactions on Industrial Electronics*, vol. 63, pp. 5501–5513, Sept 2016.

[13] Z. Chen, B. Yao, and Q. Wang, "μ-synthesis-based adaptive robust control of linear motor driven stages with high-frequency dynamics: A case study," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 3, pp. 1482–1490, 2015.

[14] Z. Chen, B. Yao, and Q. Wang, "Accurate motion control of linear motors with adaptive robust compensation of nonlinear electromagnetic field effect," *IEEE/ASME Transactions on Mechatronics*, vol. 18, pp. 1122–1129, June 2013.

[15] E. Kayacan, H. Ramon, and W. Saeys, "Robust trajectory tracking error model-based predictive control for unmanned ground vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2, pp. 806–814, 2016.

[16] A. W. Divelbiss and J. T. Wen, "Trajectory tracking control of a car-trailer system," *IEEE Transactions on Control Systems Technology*, vol. 5, no. 3, pp. 269–278, 1997.

[17] A. Astolfi, P. Bolzern, and A. Locatelli, "Path-tracking of a tractor-trailer vehicle along rectilinear and circular paths: a Lyapunov-based approach," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 154–160, 2004.

[18] E. Kayacan, E. Kayacan, H. Ramon, and W. Saeys, "Learning in centralized nonlinear model predictive control: Application to an autonomous tractor-trailer system," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 197–205, 2015.

[19] E. Kayacan, E. Kayacan, H. Ramon, and W. Saeys, "Robust tube-based decentralized nonlinear model predictive control of an autonomous tractor-trailer system," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 1, pp. 447–456, 2015.

[20] M. Sampei, T. Tamura, T. Itoh, and M. Nakamichi, "Path tracking control of trailer-like mobile robot," in *IEEE/RSJ International Workshop on Intelligent Robots and Systems (IROS).*, pp. 193–198 vol.1, 1991.

[21] M. Karkee and B. L. Steward, "Study of the open and closed loop characteristics of a tractor and a single axle towed implement system," *Journal of Terramechanics*, vol. 47, no. 6, pp. 379 – 393, 2010.

[22] M. M. Michalek, "Cascade-like modular tracking controller for non-standard n-trailers," *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–9, 2016.

[23] H. Yang, X. Fan, P. Shi, and C. Hua, "Nonlinear control for tracking and obstacle avoidance of a wheeled mobile robot with nonholonomic constraint," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 2, pp. 741–746, 2016.

[24] G. Antonelli, F. Arrichiello, F. Caccavale, and A. Marino, "Decentralized time-varying formation control for multi-robot systems," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1029–1043, 2014.

[25] E. Kayacan, E. Kayacan, H. Ramon, O. Kaynak, and W. Saeys, "Towards agrobots: Trajectory control of an autonomous tractor using type-2 fuzzy logic controllers," *IEEE/ASME Transactions on Mechatronics*, vol. 20, pp. 287–298, Feb 2015.

[26] X. Song, L. D. Seneviratne, and K. Althoefer, "A kalman filter-integrated optical flow method for velocity sensing of mobile robots," *IEEE/ASME Transactions on Mechatronics*, vol. 16, pp. 551–563, June 2011.

[27] J.-M. Yang and J.-H. Kim, "Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 578–587, 1999.

[28] C. Abdallah, D. M. Dawson, P. Dorato, and M. Jamshidi, "Survey of robust control for rigid robots," *IEEE Control Systems*, vol. 11, pp. 24–30, Feb 1991.

[29] W. M. Grimm, "Robot non-linearity bounds evaluation techniques for robust control," *International Journal of Adaptive Control and Signal Processing*, vol. 4, no. 6, pp. 501–522, 1990.

[30] M. Zhihong and M. Palaniswami, "Robust tracking control for rigid robotic manipulators," *IEEE Transactions on Automatic Control*, vol. 39, pp. 154–159, Jan 1994.

[31] M. Zhihong and X. Yu, "Adaptive terminal sliding mode tracking control for rigid robotic manipulators with uncertain dynamics.," *JSME International Journal Series C*, vol. 40, no. 3, pp. 493–502, 1997.

[32] S. Yin and X. Zhu, "Intelligent particle filter and its application to fault detection of nonlinear system," *IEEE Transactions on Industrial Electronics*, vol. 62, pp. 3852–3861, June 2015.

[33] A. Luca, G. Oriolo, and M. Vendittelli, *Ramsete: Articulated and Mobile Robotics for Services and Technologies*, ch. Control of Wheeled Mobile Robots: An Experimental Overview, pp. 181–226. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001.

List of Figures

List of Tables