RFID-Based Mobile Robot Trajectory Tracking and Point Stabilization Through On-line Neighboring Optimal Control

M. Suruz Miah · Wail Gueaieb

Abstract In this manuscript, we propose an on-line trajectory-tracking algorithm for nonholonomic Differential-Drive Mobile Robots (DDMRs) in the presence of possibly large parametric and measurement uncertainties. Most mobile robot tracking techniques that depend on reference RF beacons rely on approximating line-of-sight (LOS) distances between these beacons and the robot. The approximation of LOS is mostly performed using Received Signal Strength (RSS) measurements of signals propagating between the robot and RF beacons. However, an accurate mapping between RSS measurements and LOS distance remains a significant challenge and is almost impossible to achieve in an indoor reverberant environment. This paper contributes to the development of a neighboring optimal control strategy where the two major control tasks, trajectory tracking and point stabilization, are solved and treated as a unified manner using RSS measurements emitted from Radio Frequency IDentification (RFID) tags. The proposed control scheme is divided into two cascaded phases. The first phase provides the robot's nominal control inputs (speeds) and its trajectory using full-state feedback. In the second phase, we design the neighboring optimal controller, where RSS measurements are used to better estimate the robot's pose by employing an optimal filter. Simulation and experimental results are presented to demonstrate the performance of the proposed optimal feedback controller for solving the stabilization and trajectory tracking problems using a DDMR.

Frequently Used Symbols

$\mathbf{K}(t)$	Feedback control gain at time t
$\mathscr{H}_{\mathbf{K}}$	Hamiltonian's gradient with respect to $\mathbf{K}$
s	Number of RFID tags in the environment
ψ	Costate variable (Lagrange multiplier)
$\mathbf{q}(t)$, $\mathbf{q}^d(t)$	Robot's actual and desired pose at time t
$\mathbf{q}_t^j \in \mathbb{R}^3$	jth tag position in 3D space
t_0, t_f	Initial and final time instants
$\mathscr{I} \equiv [t_0, t_f]$	Time interval
$Tr[\cdot]$	Trace of matrix $[\cdot]$

M. S. Miah (✉) · W. Gueaieb
Machine Intelligence, Robotics, and Mechatronics
(MIRaM) Lab, School of Electrical Engineering
and Computer Science, University of Ottawa,
Ottawa, ON K1N 6N5, Canada
e-mail: suruz.miah@uottawa.ca

W. Gueaieb
e-mail: wgueaieb@eecs.uOttawa.ca

$\mathbf{u}(t)$	Robot's control input vector at time t
$\boldsymbol{\xi}(t)$	Robot's actuator noise at time t
$\boldsymbol{\zeta}(t)$	Measurement noise vector at time t
$(\cdot)^o$, $(\cdot)^\varepsilon$, $(\cdot)_{ad}$	Optimal, perturb, admissible value of $(\cdot)$
$\mathscr{L}$	Lebesgue measurable function space
$\boldsymbol{\nu}^T dJ(\cdot)$	Gateaux (directional) derivative of J in direction $\boldsymbol{\nu}$

1 Introduction

Feedback control design problem for tracking a pre-defined trajectory or stabilizing to a fixed point using a nonholonomic mobile robot are quite challenging tasks. In particular, Brockett's theorem [6] proves the nonexistence of smooth state-feedbacks for the asymptotic stabilization of fixed configurations. As such, practical alternative control solutions that guarantee acceptable tracking and stabilization performance for systems with non-integrable kinematic constraints are well motivated. In this manuscript, the proposed control strategy treats both tracking and point stabilization problems as a unified manner, where solving the point stabilization problem becomes an special case of the trajectory tracking problem. Hence, we focus on the design of neighboring optimal control scheme for solving trajectory tracking problem using a nonholonomic differential drive mobile robot.

Tracking problems have been addressed in a variety of robotic platforms [4, 13, 38, 39, 54, 56] using intelligent control laws coupled with adaptation. For wheeled mobile robots, conventional control laws have been applied for solving tracking problems [1, 20, 27, 29, 40, 46, 55] and stabilization problems [3, 7, 15, 48, 51]. For example, see [11, 12, 25, 26, 36, 45] for backstepping methods, [10, 21, 50] for sliding mode control, [8, 16, 31] for moving horizon $\mathscr{H}_\infty$ tracking control coupled with disturbance effect, and [44] for transverse function approach. A vector-field orientation feedback control method for a differentially driven wheeled vehicle has been demonstrated in [43]. This technique solves both the trajectory tracking and the point stabilization problems as in our current work. The dynamic effects of the vehicle and the noisy

feedback signal may affect the vehicle to stabilize on a fixed configuration. Several contributions have been made to the design of non-conventional control laws (fuzzy logic control, for instance) for mobile robot with a particular focus on trajectory tracking, see [24, 34] and some references therein, for example. In [37], a fuzzy logic control law is designed for a car-like mobile robot for autonomous garage-parking and parallel-parking capability by using real-time image processing technique. In the literature, less attention is paid towards solving tracking problems since it is simpler than point stabilization problem for nonholonomic systems [40].

RFID technology drew the attention of a large body of research on mobile robot localization owing to its wide availability, contactless recognition ability, and affordability [9, 18, 33, 47]. In most cases, RFID systems are deployed for solving localization problem (not stabilization or tracking problems) of mobile robots in a particular environment [19, 28, 32, 41]. A sliding mode controller in cooperation with RFID system is proposed in [35] to track a desired trajectory, where RFID tags are placed on the floor in a triangular pattern to estimate the position of the mobile robot. This technique, however, is not suitable if the operating environment is dynamically changed. Besides trajectory tracking, some researchers contributed to develop pose estimation techniques using vision technology [22, 23, 52]. Yet, they are based on known noise statistics and require complex image processing techniques. In 2008, Gueaieb and Miah pioneered a navigation algorithm, where the phase difference of RFID signals is exploited to navigate a mobile robot in an indoor environment [17]. The navigation system is, however, based on a customized RFID reader (not RFID tag) architecture and the navigation performance is evaluated using computer simulations. Moreover, the robot's trajectory tracking and stabilization problems were not explicitly solved in our previous work.

Despite aforementioned contributions on mobile robotics, the stabilization and tracking problems in a dynamic environment still face some significant technical challenges that must be overcome. Hence, our effort is devoted to solve these two main control tasks of a DDMR in two phases. In the first phase,

a nominal full-state feedback controller is designed to provide nominal speeds and its corresponding trajectory, where the process and measurement noise (external disturbances) are not considered. In the second phase, the nominal control and trajectory are employed to design the on-line neighboring optimal control inputs which are applied to the robot's actuators. Note that the robot's actuators' noise and RSS measurement noise are taken into consideration in this phase. RSS measurements emitted from RFID tags are used to better estimate the robot's pose by incorporating an optimal filter. It is important to mention, however, that the proposed optimal feedback controller is different from many alike controllers suggested in the literature in that we optimize a general feedback control gain which eventually provides optimal control inputs to the robot's actuator. Unlike many other controllers, the proposed control method does not require the linearization of the robot model. Hence, this novel work of optimizing the feedback control gain opens the door for solving problems of a general class of nonlinear dynamical systems. The work described herein is pioneered by using a DDMR operating in an indoor office environment where RFID tags are placed at 3-D positions. It is worth mentioning that the controller proposed in this manuscript does not represent an all-case alternative to vision-based navigation systems. Rather, it can serve as a substitute for such systems in environments of variable or limited lighting conditions.

The rest of the paper is outlined as follows. Some mathematical preliminaries are given in Section 2. Section 3 illustrates the high level architecture of the proposed mobile robot trajectory tracking system along with the robot's kinematic model and its feedback system using RSS measurements. The nominal optimal control and its corresponding nominal optimal trajectory are computed using the smooth state feedback control as detailed in Section 4. Section 5 describes the robot's on-line neighboring optimal control strategy, where RSS measurements from RFID system are used for its optimal pose estimation. A thorough evaluation of the current work with some numerical simulation results is presented in Section 6 followed by experimental results in Section 7. Finally, conclusions are drawn in Section 8.

2 Preliminaries

In the rest of the paper, small and capital bold letters will be used to denote vectors and matrices, respectively. Scalars will be denoted by non-bold letters. The 2-norm and scalar product are defined by

$$\|\mathbf{x}\| \equiv \left[\sum_{i=1}^{n} |x_i|^2 \right]^{1/2} \quad \text{and} \quad (\mathbf{x} \cdot \mathbf{y}) \equiv \mathbf{x}^T \mathbf{y} \equiv \sum_{i=1}^{n} x_i y_i,$$

respectively, for vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and positive n. For matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{m \times n}$, these quantities are given by

$$\|\mathbf{X}\| \equiv \left[\sum_{i=1}^{m} \sum_{j=1}^{n} |x_{i,j}|^2 \right]^{1/2} \quad \text{and}$$

$$(\mathbf{X} \cdot \mathbf{Y}) \equiv Tr\left[\mathbf{X}^T \mathbf{Y} \right] \equiv Tr\left[\mathbf{X}\mathbf{Y}^T \right],$$

respectively, where $Tr[\cdot]$ denotes the trace of matrix $[\cdot]$. Clearly, $Tr\left[\mathbf{X}^T \mathbf{X} \right] = \|\mathbf{X}\|^2$.

If the function $J : \mathbb{R}^n \to \mathbb{R}$ is differentiable at $\mathbf{x} \in \mathbb{R}^n$, then for any $\boldsymbol{v} \in \mathbb{R}^n$, $\boldsymbol{v}^T dJ(\mathbf{x})$ denotes the Gateaux (directional) derivative in the direction of $\boldsymbol{v}$, which is given by

$$\boldsymbol{v}^T dJ(\mathbf{x}) = \lim_{\varepsilon \to 0} \frac{J(\mathbf{x} + \varepsilon \boldsymbol{v}) - J(\mathbf{x})}{\varepsilon}.$$

However, if $J : \mathbb{R}^{m \times n} \to \mathbb{R}$, then for any $\mathbf{X}, \mathbf{V} \in \mathbb{R}^{m \times n}$, the directional derivative in the direction of $\mathbf{V}$ is defined by

$$Tr[\mathbf{V}^T dJ(\mathbf{X})] = \lim_{\varepsilon \to 0} \frac{J(\mathbf{X} + \varepsilon \mathbf{V}) - J(\mathbf{X})}{\varepsilon}.$$

For any bounded interval $\mathscr{I} \equiv [t_0, t_f]$, $C(\mathscr{I}, \mathbb{R}^n)$ denotes the class of all continuous functions on $\mathscr{I}$ taking values in $\mathbb{R}^n$. Let $p \in [1, \infty)$ and any finite time interval $\mathscr{I}$, we use $\mathscr{L}_p(\mathscr{I}, \mathbb{R}^n)$ to denote the set of Lebesgue measurable functions $\{\mathbf{f}\}$ defined on the measurable set $\mathscr{I}$ and taking values in $\mathbb{R}^n$ whose norms are p-th power integrable [30, 49] i.e.,

$$L_p(\mathbf{f}) = \left(\int_{t_0}^{t_f} \|\mathbf{f}\|^p dt \right)^{1/p} < \infty,$$

where $L_p(\mathbf{f})$ denotes the p-th norm of the function $\mathbf{f}$. For $p = \infty$, $\mathscr{L}_\infty(\mathscr{I}, R^n)$ denotes the space of Lebesgue measurable functions $\{\mathbf{f}\}$ defined on $\mathscr{I}$ and taking values in $\mathbb{R}^n$ satisfying ess-sup$\{\|\mathbf{f}(t)\|, t \in \mathscr{I}\} < \infty$.
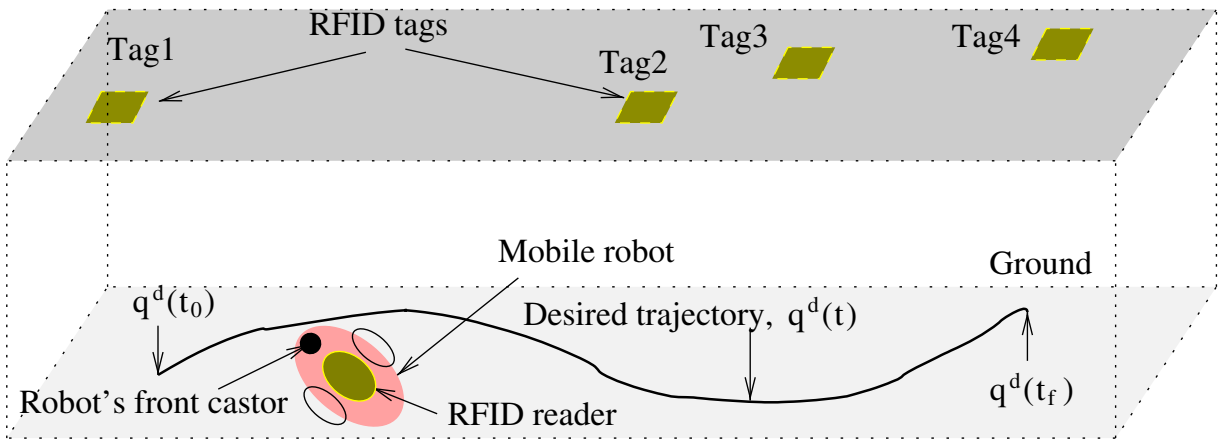
Fig. 1 High level system architecture of the proposed tracking system

3 System Overview

3.1 System Architecture

A high level setup of the proposed tracking system with four RFID tags, Tag1, Tag2, Tag3, and Tag4, attached to the ceiling of an indoor space (office, for instance), is depicted in Fig. 1. In this configuration, the robot's desired trajectory on the ground is defined in continuous time, $\mathbf{q}^d(t)$ with $\mathbf{q}^d(t_0)$ and $\mathbf{q}^d(t_f)$ being the initial and final poses, respectively. For instance, if a mobile robot is provided with the list of 16-bit tag IDs, 0xFFF9, 0xFFF2, 0xFFF5, and 0xFFF4, then it is supposed to continuously read these tag IDs and their corresponding RSS values through an RFID reader mounted on it [42]. Based on the tags' RSS measurements, optimal control actions are then generated for its actuators to track the desired trajectory, $\mathbf{q}^d(t)$. In the following, we provide a detailed description of how these optimal control actions are generated for the robot to track its desired trajectory.

3.2 Robot's Feedback Model Using RSS

Let (x, y) and θ be the position and the heading angle of a robot with respect to a ground-fixed inertial reference frame X-Y. The rotational velocities of the robot's left and right wheels are characterized by the sole (scalar) axis angular velocities u_L and u_R, respectively. The robot's position is the midpoint of the wheelbase of length l connecting the two lateral

wheels along their axis. The mobile robot[1] used in this study is pictured in Fig. 2a and its corresponding kinematic model is shown in Fig. 2b.

Consider t_0 and t_f be the initial and final time to complete the robot's mission, respectively, and $\mathscr{I} \equiv [t_0, t_f]$ denotes the time interval. At any time $t \in \mathscr{I}$, the robot kinematic model is given by

$$\dot{\mathbf{q}}(t) = \mathbf{f}[\mathbf{q}(t), \mathbf{u}(t)] = \frac{r}{2}\mathbf{B}[\mathbf{q}(t)]\mathbf{u}(t) , \qquad (1)$$

where the robot's configuration $\mathbf{q}(t) \equiv [x(t) \ y(t) \ \theta(t)]^T \in \mathscr{Q} \subset \mathbb{R}^2 \times \mathbb{S}^1$, its control input vector $\mathbf{u}(t) \equiv [u_R(t) \ u_L(t)]^T \in \mathscr{U} \subset \mathbb{R}^2$, $\theta(t) \in [-\pi, +\pi)$,

$$\mathbf{B}[\mathbf{q}(t)] = \begin{bmatrix} \cos\theta(t) & \cos\theta(t) \\ \sin\theta(t) & \sin\theta(t) \\ \frac{2}{l} & -\frac{2}{l} \end{bmatrix},$$

and r is the radius of each wheel. Since the robot itself is subject to the noisy speed, the model (1) can be rewritten as

$$\dot{\mathbf{q}}(t) = \mathbf{f}[\mathbf{q}(t), \mathbf{u}(t), \boldsymbol{\xi}(t)], \qquad (2)$$

where $\boldsymbol{\xi}(t)$ is the noise associated with control input $\mathbf{u}(t)$. For simplicity, assume that $\boldsymbol{\xi} : [0, \infty) \to \mathbb{R}^2$, is any measurable stochastic process taking values from the closed (Euclidean) ball $\mathscr{B}_u\left(\bar{\boldsymbol{\xi}}, r_1'\right)$ defined by

$$\mathscr{B}_u\left(\bar{\boldsymbol{\xi}}, r_1'\right) = \left\{ \boldsymbol{\xi}(t) \in \mathbb{R}^2 \ : \ \|\boldsymbol{\xi}(t) - \bar{\boldsymbol{\xi}}\| \le r_1' \right\},$$

---

[1]The photo of this mobile robot is taken from www.drrobot.com
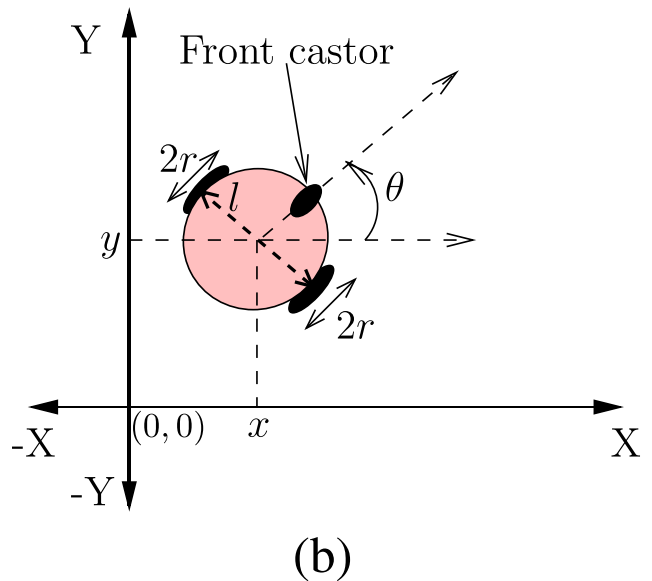
Fig. 2 a Sputnik mobile robot and b its kinematic model

where $r_1' > 0$ is the radius of the noise associated with the robot's speed and $\bar{\boldsymbol{\xi}}$ is the mean of $\boldsymbol{\xi}(t)$, for $t \in \mathscr{I}$.

However, due to the speed limits of the wheels, the inputs are constrained as

$$|u_L(t)| \le u_L^{max} \text{ and } |u_R(t)| \le u_R^{max}, \ t \in \mathscr{I}. \quad (3)$$

In other words, $\mathbf{u}(t)$ must be chosen from a set of admissible speeds, $\mathscr{U}_{ad}$, i.e., $\mathbf{u}(t) \in \mathscr{U}_{ad}$. Note that a DDMR is a nonholonomic system with the nonholonomic constraint given by

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0, \quad (4)$$

which ensures the wheel's non-slip movement in the lateral direction.

Let us model the robot's RSS measurements as

$$\mathbf{z}(t) = \hat{\mathbf{h}}[\mathbf{q}(t)] + \boldsymbol{\zeta}(t), \quad (5)$$

where $\mathbf{z}(t) \in \mathbb{R}^s$ is the RSS measurement vector (in dBm) from s RFID tags in the environment and the noise $\boldsymbol{\zeta} : [0, \infty) \to \mathbb{R}^s$, is defined such that

$$\mathscr{B}_m\left(\bar{\boldsymbol{\zeta}}, r_2'\right) = \left\{\boldsymbol{\zeta}(t) \in \mathbb{R}^s \ : \ \|\boldsymbol{\zeta}(t) - \bar{\boldsymbol{\zeta}}\| \le r_2'\right\},$$

where $r_2' > 0$ is the radius of the noise associated with the RSS measurements and $\bar{\boldsymbol{\zeta}}$ is the mean of $\boldsymbol{\zeta}(t)$, for

$t \in \mathscr{I}$. The nonlinear measurement function

$$\hat{\mathbf{h}}[\mathbf{q}(t)] = \left[\hat{h}_1[\mathbf{q}(t)] \ \dots \ \hat{h}_s[\mathbf{q}(t)]\right]^T$$

of Eq. (5) is given by $\mathbf{h} : \mathbb{R}^2 \times \mathbb{S}^1 \to \mathbb{R}^s$ with

$$\hat{h}_j[\mathbf{q}(t)] = \alpha e^{\beta \hat{d}_j}, \text{ for } j = 1, \dots, s, \quad (6)$$

where α and β are the parameters which are obviously dependent on the operating environment. Hence, these parameters can be optimized using the nonlinear least square method. $\hat{d}_j$ of Eq. (6) is simply the Euclidean distance between the robot's current position (x, y) and j-th RFID tag position $\mathbf{q}_t^j$. If $\mathbf{q}_t^j = \left[x_t^j \ y_t^j \ z_t^j\right]^T$ represents the 3D position of the j-th tag, then

$$\hat{d}_j = \sqrt{\left(x - x_t^j\right)^2 + \left(y - y_t^j\right)^2 + \left(z_t^j\right)^2}, \text{ for } j = 1, \dots, s.$$

We now define the robot's control input $\mathbf{u}(t)$ as the feedback model defined as

$$\mathbf{u}(t) = \boldsymbol{\chi}[\mathbf{z}(t)] \quad (7)$$

subject to Eq. (3), where $\boldsymbol{\chi}[\cdot]$, is a function that takes RSS measurements as the feedback information. Clearly, the model (2) is underactuated (i.e., two inputs but three state-variables to control). Hence, the

challenge is to design the feedback function $\chi[\cdot]$ of the model (7), which we tackle using the feedback gain (see Section 4) coupled with RSS measurements from RFID tags (see Section 5, for details).

Substituting Eq. (7) in Eq. (1), we can formulate the robot's measurement-feedback system as

$$\dot{\mathbf{q}}(t) = \frac{r}{2}\mathbf{B}[\mathbf{q}(t)]\boldsymbol{\chi}[\mathbf{z}(t)]. \qquad (8)$$

3.3 Problem Formulation

Let $\mathbf{q}^d(t) = [x^d(t)\ y^d(t)\ \theta^d(t)]^T$ be the desired trajectory that the robot is supposed to track and

$$e(t) = \sqrt{[x^d(t) - x(t)]^2 + [y^d(t) - y(t)]^2}$$

denote its position tracking error, for $t \in \mathscr{I}$. The objective is to find the optimal control input $\mathbf{u}(t) \in \mathscr{U}_{ad}$ that generates the trajectory $\mathbf{q}(t) \in \mathscr{Q}$ while minimizing the total position tracking error, $\mathscr{E}$, given by

$$\mathscr{E} = \int_{t_0}^{t_f} e(t)dt. \qquad (9)$$

Given the robot's kinematic model (1), its non-holonomic constraint (4), and for any $\boldsymbol{\xi}(t) \in \mathscr{B}_u\left(\bar{\boldsymbol{\xi}}, r_1'\right)$, $\boldsymbol{\zeta}(t) \in \mathscr{B}_m\left(\bar{\boldsymbol{\zeta}}, r_2'\right)$, the problem can be stated as follows:

$$\inf_{\{\mathbf{q}\in\mathscr{Q},\ \mathbf{u}\in\mathscr{U}_{ad}\}} [\mathscr{E}]. \qquad (10)$$

Although not explicitly stated, this goal implicitly imposes the optimization of the robot orientation $\theta(t)$ since it is coupled with the robot position (see model (1)). This point will be clearer in the next section.

4 Nominal Pose and Control Generation

In order to determine the robot's nominal optimal pose and its corresponding control input, assume that the process noise $\boldsymbol{\xi}(t) = \mathbf{0}$ and also define the robot's feedback control model (7) as the full-state feedback control:

$$\mathbf{u}(t) = \mathbf{K}(t)\mathbf{q}(t), \qquad (11)$$

subject to Eq. (3), where $\mathbf{K}(t) \neq \mathbf{0}$ is the feedback control gain for the robot model (1). Since the sets $\mathscr{U}$ and $\mathscr{Q}$ are convex, $\mathbf{K}(t)$ must be chosen from a convex

set $\mathscr{K} \subset \mathbb{R}^{2\times3}$. Furthermore, due to the constraint on the wheel speeds, $\mathbf{K}(t)$ has to be chosen from the admissible matrix space $\mathscr{K}_{ad} \subset \mathscr{K}$.

Substituting Eq. (11) in Eq. (1), yields the following full-state feedback system:

$$\dot{\mathbf{q}}(t) = \frac{r}{2}\mathbf{B}[\mathbf{q}(t)]\mathbf{K}(t)\mathbf{q}(t) = \hat{\mathbf{f}}[\mathbf{q}(t), \mathbf{K}(t)], \ \mathbf{q}(t_0) = \mathbf{q}_0, \qquad (12)$$

where the robot's initial pose $\mathbf{q}_0 \neq \mathbf{0}$ since model (12) is a nonlinear homogeneous equation (drift-free system). The actual trajectory of the robot can be described by

$$\mathbf{q}(t) = \mathbf{q}(t_0) + \frac{r}{2}\int_{t_0}^{t} \{B[\mathbf{q}(\tau)]\mathbf{K}(\tau)\mathbf{q}(\tau)\}\,d\tau, \qquad (13)$$

for $t \in \mathscr{I}$. Since feedback control gain $\mathbf{K}(t)$ in Eq. (11) needs to be optimized in order to determine nominal optimal control $\mathbf{u}^o(t)$, the full-state feedback control problem boils down to the optimization problem.

For the robot to find the nominal optimal trajectory $\mathbf{q}^o(t)$, define the cost functional as

$$J(\mathbf{K}) = \phi[t_f, \mathbf{q}(t_f)] + \int_{t_0}^{t_f} \ell[t, \mathbf{q}(t)]dt, \qquad (14)$$

with

$$\phi[t_f, \mathbf{q}(t_f)] = \frac{1}{2}[\mathbf{q}(t_f) - \mathbf{q}^d(t_f)]^T\mathbf{P}(t_f)[\mathbf{q}(t_f) - \mathbf{q}^d(t_f)]$$

$$\ell[t, \mathbf{q}(t)] = \frac{1}{2}[\mathbf{q}(t) - \mathbf{q}^d(t)]^T\mathbf{Q}(t)[\mathbf{q}(t) - \mathbf{q}^d(t)],$$

where $\mathbf{P}(t_f), \mathbf{Q}(t) \in \mathbb{R}^{3\times3}$ are symmetric positive definite matrices that indicate the relative importance of the error components along $\mathbb{R}^2 \times \mathbb{S}^1$. If the robot's purpose is to stabilize on a fixed point in its environment, then the weight matrix $\mathbf{P}(t_f)$ must be higher than $\mathbf{Q}(t)$. However, the opposite is true for the robot to track a desired trajectory. The performance index $J(\mathbf{K})$ in Eq. (14) depends on the feedback control gain matrix $\mathbf{K}(t)$ through the state variable $\mathbf{q}(t)$ as it is clear from the feedback system (12). The nominal trajectory and control $(\mathbf{q}^o(t), \mathbf{u}^o(t))$ can be obtained by minimizing $J(\mathbf{K})$ subject to Eqs. (1), (3), and (4).

Assume that the feedback gain $\mathscr{K} \subset \mathbb{R}^{2\times3}$ is a closed bounded convex set and

$$\mathscr{K}_{ad} \equiv \left\{ \mathbf{K}(t) \in \mathscr{L}_\infty^{\ell oc}([0, \infty), \mathbb{R}^{2\times3}) : \mathbf{K}(t) \in \mathscr{K} \right\}$$

where $\mathscr{L}_p^{\ell oc}([0,\infty),\mathbb{R}^{2\times 3})$ are locally convex topological function spaces of p-th power locally integrable functions containing the spaces $\mathscr{L}_p(\mathscr{I},\mathbb{R}^{2\times 3})$.

To solve for the nominal optimal trajectory using the feedback system (12) that minimizes the objective functional (14), we need to derive the necessary conditions of optimality. These necessary conditions are most readily found if the integrand of the cost functional (14) is recast in terms of Hamiltonian

$$\mathscr{H} : \mathscr{I} \times \mathbb{R}^2 \times \mathbb{S}^1 \times \mathbb{R}^3 \times \mathbb{R}^{2\times 3} \longmapsto \mathbb{R},$$

which is given by

$$\mathscr{H}[t,\mathbf{q}(t),\boldsymbol{\psi}(t),\mathbf{K}(t)] = \boldsymbol{\psi}^T\hat{\mathbf{f}}[\mathbf{q}(t),\mathbf{K}(t)] + \ell[t,\mathbf{q}(t)], \tag{15}$$

where $\boldsymbol{\psi}(t) \in \mathbb{R}^3$, $t \in \mathscr{I}$, is a vector of Lagrange multipliers whose elements are the costates. We now derive the necessary conditions of optimality for the feedback model (12).

Theorem 1 (**Necessary Conditions of Optimality**) *The optimal trajectory* $\mathbf{q}^o(t)$ *for the feedback model* (12) *can be obtained if there exists an optimal feedback control gain* $\mathbf{K}^o(t) \in \mathscr{K}_{ad}$ *and an optimal multiplier* $\boldsymbol{\psi}^o(t) \in C(\mathscr{I},\mathbb{R}^3)$ *such that the triple* $\{\mathbf{q}^o,\boldsymbol{\psi}^o,\mathbf{K}^o\}$ *satisfies the following necessary conditions:*

$$\mathscr{H}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}(t)] \geq \mathscr{H}[t,\mathbf{q^o}(t),\boldsymbol{\psi}^o(t),\mathbf{K^o}(t)],$$
$$\mathbf{K}(t) \in \mathscr{K}, t \in \mathscr{I}, \tag{16}$$

$$\dot{\mathbf{q}}^o = \frac{\partial\mathscr{H}}{\partial\boldsymbol{\psi}}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K^o}(t)],\ \mathbf{q}^o(t_0) = \mathbf{q_0},\ t \in \mathscr{I}, \tag{17}$$

$$\dot{\boldsymbol{\psi}}^o = -\frac{\partial\mathscr{H}}{\partial\mathbf{q}}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}^o(t),\ \boldsymbol{\psi}^o(t_f)$$
$$= \frac{\partial\phi}{\partial\mathbf{q}}[t_f,\mathbf{q}(t_f)]. \tag{18}$$

Theorem 1 states that there exists a feedback control gain $\mathbf{K}^o(t) \in \mathscr{K}_{ad}$ for the robot to determine nominal optimal control inputs for its actuator. Its proof is given in Appendix A.1. In order to solve for

$\mathbf{K}^o(t)$, we determine the gradient of the Hamiltonian defined in Eq. (15) and set it to zero,

$$\mathscr{H}_{\mathbf{K}} \equiv \frac{\partial\mathscr{H}}{\partial\mathbf{K}} = \frac{r}{2}\mathbf{B}^T[\mathbf{q}(t)]\boldsymbol{\psi}(t)\mathbf{q}^T(t) = \mathbf{0}. \tag{19}$$

Note that the expression in Eq. (19) is independent of the gain matrix $\mathbf{K}(t)$. Hence, the problem boils down to finding $\mathbf{K}(t)\ t \in \mathscr{I}$, such that the robot's actual trajectory (13) and the costate trajectory from Eq. (18) satisfy Eq. (19). The optimal feedback control gain $\mathbf{K}^o(t)$ can be determined by satisfying the Hamiltonian inequality (16). In other words, $\mathbf{K}(t)$ is to be adaptively tuned to minimize the robot's tracking error.

Corollary 1 (**Adapting the gain K**) *Consider the robot's feedback system* (12) *defined over the time horizon* $\mathscr{I}$. *Adapting the gain* **K** *according to the following offline update rule*

$$\mathbf{K}^{new} = \mathbf{K}^{old} - \varepsilon\mathscr{H}_{\mathbf{K}}, \text{ for } 0 < \varepsilon < 1 \tag{20}$$

satisfies the Hamiltonian inequality (16) *and, hence, guarantees the converge of the robot's trajectory towards its target.*

In the following, we numerically solve for the gain **K** such that Eq. (19) is satisfied, aggregating the components described earlier.

Let $\mathbf{K}_i \equiv \mathbf{K}_i(t)$, $t \in \mathscr{I}$, be the gain at the i-th iteration of the optimization procedure.

Step 0 (initialization): Subdivide the time interval $\mathscr{I} \equiv [t_0,t_f]$ into N subintervals. Assume a piecewise-constant $\mathbf{K_i}(t) = \mathbf{K_i}(t_k)$, $t \in [t_k,t_{k+1}]$, for $k = 0,\ldots,N-1$.
Find the optimal gain $\mathbf{K}^o$ by repeating Steps 1–5 until the stopping criterion in Step 5 is met.
Step 1: Integrate the robot's feedback system (12) as in Eq. (13) with $\mathbf{K} \equiv \mathbf{K_i}(t)$, $t \in \mathscr{I}$.
Step 2: Solve the costate equation (18) backward for $\boldsymbol{\psi}_i$.
Step 3: Define the Hamiltonian $\mathscr{H}(t,\mathbf{q}_i,\boldsymbol{\psi}_i,\mathbf{K}_i)$ as in Eq. (15).
Step 4: Compute the cost function $J(\mathbf{K}_i)$ using Eq. (14), the gradients of the Hamiltonian $\mathscr{H}_{\mathbf{K}}$ in Eq. (19), and its corresponding intergrated norm $\int_{t_0}^{t_f}\|\mathscr{H}_{\mathbf{K}}\|^2 dt$.
Step 5: If $J(\mathbf{K}_i) \leq \delta_1$ or $\int_{t_0}^{t_f}\|\mathscr{H}_{\mathbf{K}}\|^2 dt \leq \delta_2$, for pre-defined small positive tolerance constants δ_1 and δ_2, then $\mathbf{K_i}$ is regarded close enough to

its nominal optimal value, and so the algorithm is halted.

Otherwise, use the following update rule to adjust the piecewise-constant feedback control gain:

$$\mathbf{K}_{i+1}(t_k) = \mathbf{K}_i(t_k) - \varepsilon \, \mathscr{H}_{\mathbf{K}(t_k)} + \lambda \, \Delta \mathbf{K}_i(t_k)$$
$$\Delta \mathbf{K}_i(t_k) = \mathbf{K}_i(t_k) - \mathbf{K}_{i-1}(t_k)$$

for $k = 0, \ldots, N-1$, where ε and λ are the step size and the momentum constant (for faster convergence), respectively.

We now have the optimal feedback gain, $\mathbf{K}^o$. Using the robot's initial pose $\mathbf{q}^o(t) = \mathbf{q}(t_0) = \mathbf{q}_0$, it's nominal-optimal control can thus be computed by

$$\mathbf{u}^o(t) = \mathbf{K}^o(t)\mathbf{q}^o(t), \qquad (21)$$

with the corresponding nominal-optimal state model

$$\dot{\mathbf{q}}^o(t) = \mathbf{f}[\mathbf{q}^o(t), \mathbf{u}^o(t)]. \qquad (22)$$

Models (21) and (22) will be employed to solve for the robot's actual optimal control inputs and its corresponding state trajectory which are illustrated in the next section.

5 Robot's Optimal Trajectory

For the robot to operate in real-time, it is conceivable that the exact optimal control could be updated continuously to provide the instantaneous control input to the robot's actuator. A practical method of doing so is to partition the robot's actual trajectory and control into: a) nominal and b) neighboring parts, where the former represents the off-line deterministic solution (nominal) which is illustrated in Section 4 (no external disturbances were considered) and the later represents the on-line (real-time) solution [53]. Note that the robot's actuator noise and RSS measurement noise from RFID tags are taken into account for determining its neighboring optimal control inputs.

5.1 Neighboring Optimal Control

In order to compute the robot's neighboring optimal control input, let us rewrite the cost functional (14) as

$$J(\mathbf{q}) = \phi[t_f, \mathbf{q}(t_f)] + \int_{t_0}^{t_f} \ell[t, \mathbf{q}(t), \mathbf{u}(t)]dt, \qquad (23)$$

where

$$\phi[t_f, \mathbf{q}(t_f)] = \frac{1}{2}[\mathbf{q}(t_f) - \mathbf{q}^o(t_f)]^T \mathbf{P}(t_f)[\mathbf{q}(t_f) - \mathbf{q}^o(t_f)]$$

$$\ell[t, \mathbf{q}(t), \mathbf{u}(t)] = \frac{1}{2}[\mathbf{q}(t) - \mathbf{q}^o(t)]^T \mathbf{Q}(t)[\mathbf{q}(t) - \mathbf{q}^o(t)]$$
$$+ \frac{1}{2}[\mathbf{u}(t) - \mathbf{u}^o(t)]^T \hat{\mathbf{R}}(t)[\mathbf{u}(t) - \mathbf{u}^o(t)]$$

with $\hat{\mathbf{R}}(t) \in \mathbb{R}^{2 \times 2}$ being a symmetric positive definite matrix (i.e., $\hat{\mathbf{R}}(t) \neq \mathbf{0}$). Defining the perturbations from the nominal optimal solutions as

$$\Delta \mathbf{q}(t) = \mathbf{q}(t) - \mathbf{q}^o(t), \quad \Delta \mathbf{u}(t) = \mathbf{u}(t) - \mathbf{u}^o(t), \ t \in \mathscr{I},$$
$$(24)$$

the robot's model (1) can be expanded as the Taylor series

$$\dot{\mathbf{q}}^o(t) + \Delta \dot{\mathbf{q}}(t) = \mathbf{f}\left[\mathbf{q}^o(t), \mathbf{u}^o(t)\right] + \frac{\partial \mathbf{f}}{\partial \mathbf{q}}\left[\mathbf{q}^o(t), \mathbf{u}^o(t)\right] \Delta \mathbf{q}(t)$$
$$+ \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\left[\mathbf{q}^o(t), \mathbf{u}^o(t)\right] \Delta \mathbf{u}(t) + \mathscr{O}[\Delta \mathbf{q}, \Delta \mathbf{u}],$$

where $\mathscr{O}[\Delta \mathbf{q}, \Delta \mathbf{u}]$ is the higher order terms of $\Delta \mathbf{q}(t)$ and $\Delta \mathbf{u}(t)$. Using the robot's nominal state model (22) and assuming the perturbation variables to be "small", the above expression can be truncated to the first degree, yielding the robot's linear kinematic constraint

$$\Delta \dot{\mathbf{q}}(t) = \mathbf{F}(t)\Delta \mathbf{q}(t) + \mathbf{G}(t)\Delta \mathbf{u}(t), \quad \Delta \mathbf{q}(t_0) = \Delta \mathbf{q}_0,$$
$$(25)$$

where

$$\mathbf{F}(t) = \frac{\partial \mathbf{f}}{\partial \mathbf{q}}\left[\mathbf{q}^o(t), \mathbf{u}^o(t)\right], \ \ \mathbf{G}(t) = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\left[\mathbf{q}^o(t), \mathbf{u}^o(t)\right].$$

The cost function (23) can be expanded as

$$J[\mathbf{q}^o + \Delta \mathbf{q}] \cong J[\mathbf{q}^o] + \Delta J[\Delta \mathbf{q}] + \Delta^2 J[\Delta \mathbf{q}].$$

However, the optimality guarantees that the first variation of $J[\cdot]$ (i.e., $\Delta J[\Delta \mathbf{q}(t)]$) is zero [53], yielding the above expression as

$$J[\mathbf{q}^o + \Delta \mathbf{q}] \cong J[\mathbf{q}^o] + \Delta^2 J[\Delta \mathbf{q}],$$

where the second variation of $J[\cdot]$ can be expressed as

$$\Delta^2 J[\Delta \mathbf{q}] = \frac{1}{2}\Delta \mathbf{q}^T(t_f)\phi_{qq}(t_f)\Delta \mathbf{q}(t_f) +$$
$$\frac{1}{2}\int_{t_0}^{t_f} \left\{ \begin{bmatrix} \Delta \mathbf{q}^T(t) \ \Delta \mathbf{u}^T(t) \end{bmatrix} \begin{bmatrix} \ell_{\mathbf{qq}} & \ell_{\mathbf{qu}} \\ \ell_{\mathbf{uq}} & \ell_{\mathbf{uu}} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{q}(t) \\ \Delta \mathbf{u}(t) \end{bmatrix} \right\} dt,$$
$$(26)$$

subject to Eq. (25).

Let us rewrite the expression (26) as:

$$\Delta^2 J[\Delta \mathbf{q}] \triangleq J = \frac{1}{2} \Delta \mathbf{q}^T(t_f) \mathbf{P}(t_f) \Delta \mathbf{q}(t_f) + \frac{1}{2}$$
$$\int_{t_0}^{t_f} \left\{ \begin{bmatrix} \Delta \mathbf{q}^T(t) & \Delta \mathbf{u}^T(t) \end{bmatrix} \begin{bmatrix} \mathbf{Q}(t) & \mathbf{M}(t) \\ \mathbf{M}^T(t) & \mathbf{R}(t) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{q}(t) \\ \Delta \mathbf{u}(t) \end{bmatrix} \right\} dt,$$
(27)

where

$$\mathbf{P}(t_f) \equiv \phi_{qq}(t_f) \equiv \frac{\partial^2 \phi}{\partial \mathbf{q}^2} \left[t_f, \mathbf{q}^o(t_f) \right]$$

$$\mathbf{Q}(t) \equiv \ell_{\mathbf{qq}} \equiv \frac{\partial^2 \ell}{\partial \mathbf{q}^2} \left[t, \mathbf{q}^o(t), \mathbf{u}^o(t) \right],$$

$$\mathbf{M}(t) \equiv \ell_{\mathbf{qu}} \equiv \frac{\partial^2 \ell}{\partial \mathbf{q} \partial \mathbf{u}} \left[t, \mathbf{q}^o(t), \mathbf{u}^o(t) \right], \text{ and}$$

$$\mathbf{R}(t) \equiv \ell_{\mathbf{uu}} \equiv \frac{\partial^2 \ell}{\partial \mathbf{u}^2} \left[t, \mathbf{q}^o(t), \mathbf{u}^o(t) \right].$$

Since $\mathbf{M}(t) = \mathbf{0}$, it follows from Eq. (27) that

$$J = \frac{1}{2} \Delta \mathbf{q}^T(t_f) \mathbf{P}(t_f) \Delta \mathbf{q}(t_f) + \frac{1}{2}$$
$$\int_{t_0}^{t_f} \left\{ \begin{bmatrix} \Delta \mathbf{q}^T(t) & \Delta \mathbf{u}^T(t) \end{bmatrix} \begin{bmatrix} \mathbf{Q}(t) & \mathbf{0} \\ \mathbf{0} & \mathbf{R}(t) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{q}(t) \\ \Delta \mathbf{u}(t) \end{bmatrix} \right\} dt,$$
(28)

which is a quadratic cost functional.

Theorem 2 (Adapted from [53]) *Consider the robot's linear kinematic model (25) and its quadratic cost functional given by* Eq. (28). *The optimal linear-quadratic state feedback control law is given by*

$$\Delta \mathbf{u}^o(t) = -\mathbf{R}^{-1}(t) \mathbf{G}^T(t) \mathbf{P}(t) \Delta \mathbf{q}(t) = -\mathbf{C}(t) \Delta \mathbf{q}(t),$$
(29)

where $\mathbf{C}(t)$ *is the* (2×3) *neighboring-optimal control gain matrix given by*

$$\mathbf{C}(t) = \mathbf{R}^{-1}(t) \mathbf{G}^T(t) \mathbf{P}(t)$$
(30)

and $\mathbf{P}(t)$ *is the solution of the differential matrix Riccati equation*

$$\dot{\mathbf{P}} = -\mathbf{F}^T(t) \mathbf{P}(t) - \mathbf{Q}(t) - \mathbf{P}(t) \mathbf{F}(t) +$$
$$\mathbf{P}(t) \mathbf{G}(t) \mathbf{R}^{-1}(t) \mathbf{G}^T(t) \mathbf{P}(t), \quad \mathbf{P}(t_f) = \mathbf{P}_f.$$
(31)

It is interesting to note that the solution for $\mathbf{P}(t)$ and, therefore, for $\mathbf{C}(t)$ is independent of $\Delta \mathbf{q}(t)$. Variations in $\Delta \mathbf{q}(t_0)$ or $\Delta \mathbf{q}(t_f)$ have no effect on $\mathbf{C}(t)$, although the linear-optimal control history obviously is affected by state perturbations [53].

It is clear from Theorem 2 that once the solution of the differential matrix Riccati equation (31) is available, the feedback control law given by Eq. (29) can be formally constructed. From the perturbation (24), the total control is formed as the sum of the nominal and the perturbation optimal controls as stated in the chapter introduction:

$$\mathbf{u}(t) = \mathbf{u}^o(t) + \Delta \mathbf{u}^o(t) = \mathbf{u}^o(t) - \mathbf{C}(t)[\hat{\mathbf{q}}(t) - \mathbf{q}^o(t)],$$
(32)

where $\hat{\mathbf{q}}(t)$ is the robot's estimated pose which will be determined in Section 5.2.

Substituting perturbed optimal control (29) in Eq. (25) yields the perturbed stated feedback system

$$\Delta \dot{\mathbf{q}}(t) = \left[\mathbf{F}(t) - \mathbf{G}(t) \mathbf{R}^{-1}(t) \mathbf{G}^T(t) \mathbf{P}(t) \right] \Delta \mathbf{q}(t),$$
$$\equiv \mathbf{A}(t) \Delta \mathbf{q}(t), \ \Delta \mathbf{q}(t_0) = \Delta \mathbf{q}_0 \neq \mathbf{0},$$
(33)

with $\mathbf{A}(t) \equiv \left[\mathbf{F}(t) - \mathbf{G}(t) \mathbf{R}^{-1}(t) \mathbf{G}^T(t) \mathbf{P}(t) \right]$ and the corresponding state trajectory can then be described by

$$\Delta \mathbf{q}(t) = \boldsymbol{\Phi}(t, t_0) \Delta \mathbf{q}(t_0),$$

where $\boldsymbol{\Phi}(t, t_0) = e^{t A(t)}$ is the state transition matrix.

The feedback model (33) with the quadratic cost functional (28) is similar to the optimal linear quadratic regulator problem, which is stable in the Lyapunov sense [2]. In other words, the optimality leads to stability.

Theorem 3 (**Optimality to stability**) *The feedback system given by* Eq. (33) *is*

(i) *stable if* $\mathbf{Q}(t)$ *is a real, symmetric, positive semi-definite matrix and*

(ii) *asymptotically stable if* $\mathbf{Q}(t)$ *is a real, symmetric, positive definite matrix.*

The proof of this Theorem is given in Appendix A.2. We now focus on estimating the robot's pose based on partial noisy RSS measurements from RFID tags placed in its operating environment.

5.2 Optimal Pose Estimation

The DDMR employed in the current work is subjected to actuator noise. As can be seen form Fig. 1, the robot receives RSS measurements from RFID tags. In an

indoor reverberant environment, these RSS measurements can be highly contaminated by ambient noise. In this section, we take into account the robot's actuator noise and RSS measurement noise from RFID tags. Noisy RSS measurements are used as feedback information for estimating the robot's state, which is eventually form the measurement feedback model (8). The estimated state is then used to find the total optimal control input as in Eq. (32). In the following, an optimal filter is presented to "filter out" the noisy RSS measurements as well as actuator signals for estimating the robot's pose.

The Taylor series expansion of Eq. (2) neglecting the higher order terms yields

$$\Delta \dot{\mathbf{q}}(t) = \mathbf{F}(t)\Delta \mathbf{q}(t) + \mathbf{G}(t)\Delta \mathbf{u}(t) + \mathbf{L}(t)\Delta \boldsymbol{\xi}(t), \text{ where}$$

$$\mathbf{L}(t) = \frac{\partial \mathbf{f}}{\partial \boldsymbol{\xi}}\left[\mathbf{q}^o(t), \mathbf{u}^o(t), \boldsymbol{\xi}^o(t)\right], \text{ and } \Delta \boldsymbol{\xi}(t) = \boldsymbol{\xi}(t) - \boldsymbol{\xi}^o(t),$$

with $\Delta \mathbf{q}(t_0) = \Delta \mathbf{q}_0$. Note that $\boldsymbol{\xi}^o(t) = \mathbf{0}$ because the deterministic solution of Eq. (22) has no process noise. The expected values of the initial state and covariance are

$$\mathbb{E}[\mathbf{q}(t_0)] = \hat{\mathbf{q}}_0, \quad \mathbb{E}\left\{[\mathbf{q}(t_0) - \hat{\mathbf{q}}_0][\mathbf{q}(t_0) - \hat{\mathbf{q}}_0]^T\right\} = \mathbf{S}_0. \tag{34}$$

For simplicity, assume that the robot's input and measurement noise are a white, zero-mean Gaussian random process. If $\mathbf{W}_C$ and $\mathbf{N}_C$ are spectral density matrices of the robot's input and measurement noise, respectively, the following expression holds:

$$\mathbb{E}\left[\boldsymbol{\xi}^T(t) \ \boldsymbol{\zeta}^T(t)\right] = \left[\bar{\boldsymbol{\xi}}^T \ \bar{\boldsymbol{\zeta}}^T\right]\mathbb{E}\left\{\begin{bmatrix}\boldsymbol{\xi}(\tau) \\ \boldsymbol{\zeta}(\tau)\end{bmatrix}\left[\boldsymbol{\xi}^T(t) \ \boldsymbol{\zeta}^T(t)\right]\right\}$$
$$= \begin{bmatrix}\mathbf{W}_C(t & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_C(t)\end{bmatrix}\delta(t - \tau), \tag{35}$$

where $\delta(\cdot)$ is the dirac delta function. The robot's a priori state estimate is described by

$$\hat{\mathbf{q}}(t) = \hat{\mathbf{q}}_0 + \int_{t_0}^{t}\mathbf{f}[\hat{\mathbf{q}}(t), \mathbf{u}(t)]dt. \tag{36}$$

From measurement model (5), the matrix $\mathbf{H}(t)$ is determined along the a priori estimate $\hat{\mathbf{q}}(t)$ found in Eq. (36) as

$$\mathbf{H}(t) = \frac{\partial \mathbf{h}}{\partial \mathbf{q}}[\hat{\mathbf{q}}(t)].$$

The optimal filter gain can then be computed as

$$\mathbb{K}_C = \mathbf{S}(t)\mathbf{H}^T(t)\mathbf{N}_C^{-1}(t), \tag{37}$$

where the state covariance matrix $\mathbf{S}(t)$ is the solution of the differential matrix Riccati equation

$$\dot{\mathbf{S}}(t) = \mathbf{F}(t)\mathbf{S}(t) + \mathbf{S}(t)\mathbf{F}^T(t) + \mathbf{L}(t)\mathbf{W}_C(t)\mathbf{L}^T(t)$$
$$- \mathbf{S}(t)\mathbf{H}^T(t)\mathbf{N}_C^{-1}(t)\mathbf{H}(t)\mathbf{S}(t), \quad \mathbf{S}(t_0) = \mathbf{S}_0. \tag{38}$$

Using the current RSS measurement, $\mathbf{z}(t)$ given in Eq. (5), the robot's a posteriori state estimate is determined by solving the following state model:

$$\dot{\hat{\mathbf{q}}}(t) = \mathbf{f}[\hat{\mathbf{q}}(t), \mathbf{u}(t)] + \mathbb{K}_C\left\{\mathbf{z}(t) - \mathbf{h}[\hat{\mathbf{q}}(t)]\right\}, \quad \hat{\mathbf{q}}(t_0) = \hat{\mathbf{q}}_0. \tag{39}$$

Hence, finding the robot's optimal trajectory contrains four parts:

1. compute robot's nominal-optimal control and trajectory as given in Eqs. (21) and (22),
2. computation of neighboring-optimal control gain matrix:

    (a) specify the cost function as given in Eq. (28) subject to the robot's linear kinematic constraint (25).
    (b) define the Hamiltonian for the neighboring-optimal trajectory and control.
    (c) solve the differential matrix Riccati equation (31) that results from minimizing the Hamiltonian to obtain the adjoint covariance matrix, $\mathbf{P}(t)$, from t_f to t_0.
    (d) compute the neighboring-optimal gain matrix, $\mathbf{C}(t)$ as given in Eq. (30).

3. optimal estimation of the robot's pose:

    (a) initialize estimated pose and state error covariance matrix as in Eq. (34).
    (b) use Eq. (35) to compute error covariance matrices, $\mathbf{W}_C$ and $\mathbf{N}_C$.
    (c) integrate the differential matrix Riccati equation (38).
    (d) compute the filter gain, $\mathbb{K}_C$, as in Eq. (37).
    (e) optimal pose is estimated using Eq. (39).

4. actual optimal control and trajectory generation:

    (a) compute the robot's actual optimal control by Eq. (32).
    (b) robot's actual trajectory is then the solution of Eq. (39).

The above steps can be used in conjunction with simulation or real-time control to generate robot's actual optimal trajectories corresponding to its cost function (28), kinematic contraint (1), and measurements (5). Figure 3 shows the schematic diagram of the keys steps of the proposed trajectory tracking system for a DDMR. Having computed the filter gain $\mathbb{K}_C$ and neighboring optimal control gain matrix $\mathbf{C}(t)$, the linear stochastic controller is seen to be driven by the nonlinear RSS measurements, $\mathbf{z}(t)$. Its output is summed with the nominal-optimal control, $\mathbf{u}^o(t)$; the nominal-optimal state, $\mathbf{q}^o(t)$, is used to derive the total pose estimate $\hat{\mathbf{q}}(t)$.

6 Simulation Results

We now illustrate the performance of the proposed neighboring optimal controller using the continuous-time model of a mobile robot, which is expected to follow a reference trajectory over the time horizon of $\mathscr{I} \equiv [0, 60]$ s. The robot employed in this work is a circular shaped differential drive virtual mobile robot with the wheel base of $l = 30$ cm and the radius of each wheel $r = 8.25$ cm. Its wheel speeds are constrained as

$$|u_R(t)| \leq u_R^{max} = 10 \, \text{rad} \cdot \text{s}^{-1},$$
$$|u_L(t)| \leq u_L^{max} = 10 \, \text{rad} \cdot \text{s}^{-1}.$$

The performance metrics adopted in the current work are the robot's pose (position and orientation) tracking error given by

$$\mathbf{q}(t) - \mathbf{q}^d(t) = \mathbf{q}_e(t) = \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} = \begin{bmatrix} x(t) - x^d(t) \\ y(t) - y^d(t) \\ \theta(t) - \theta^d(t) \end{bmatrix}$$

and the average cumulative position error defined in Eq. (9) over the time interval of $\mathscr{I} \equiv [0, 60]$ s, which allow us to make quantitative assessment of the proposed neighboring optimal controller. The dimension of the virtual test area is about $16 \times 16 \times 3 \, \text{m}^3$, where 25 RFID tags ($s = 25$) are uniformly placed on the ceiling of the workspace (denoted by x's in the Figs. 6a and 7a). In order to find the nominal solution as described in Section 4, the initial the feedback control gain $\mathbf{K}(t)$ is chosen as

$$\mathbf{K}(t) = 10^{-6} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix},$$

which is then optimized to find the nominal control $\mathbf{u}^o(t)$ and its corresponding nominal trajectory $\mathbf{q}^o(t)$, $t \in \mathscr{I}$, using Eqs. (21) and (22). The sampling time period for the simulation is set to 0.1 s. The weighting matrices of the cost function (14) are chosen as $\mathbf{P}(t_f) = diag(0.5, 0.5, 1)$ and $\mathbf{Q}(t) = diag(1, 1, 2)$, $\forall \, t \in \mathscr{I}$. Hence, trajectory tracking is regarded twice as important as just stabilizing on a fixed configuration. The mean and standard deviation of the robot's actuator noise, $\boldsymbol{\xi}(t)$, are chosen to be 0 and $0.8 \, \text{rad} \cdot \text{s}^{-1}$, respectively.
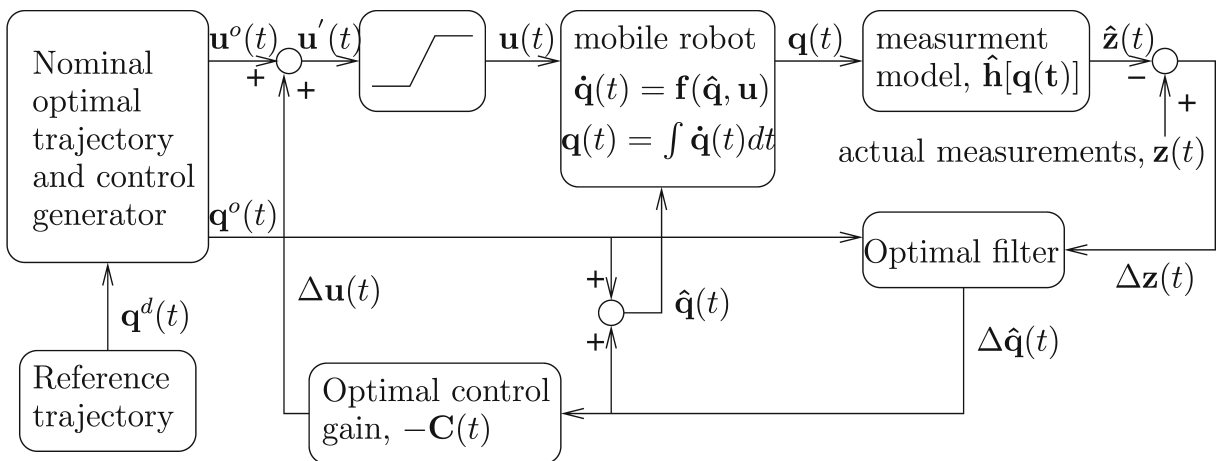

Fig. 3 Schematic of the robot's stochastic neighboring-optimal control law in continuous time

6.1 Modelling RSS Measurements and Noise

To make the controller's simulation as realistic as possible, the RSS signals were experimentally measured by emulating the RFID system using an XBee Pro RF module and a MakeController board (Fig. 4) [42]. The experiment was conducted by placing the XBee Pro RF module (emulating the tag) at (9, 8, 3) m and mounting the MakeController board (emulating the reader) on top of the robot which was initially placed at the origin with an orientation of 45°. The robot was programmed to travel along a straight line for 60 s during which the XBee Pro module's RSS values were measured and logged by the MakeController board at a sampling period of 0.6 s. Since the experiment was conducted in an open space with no obstacles, reverberations and noise were neglected and the collected data was used to model the ideal (clean) RSS values in terms of distance, i.e., model (6). This yielded the

parameter values $\alpha = -35.5$ and $\beta = 0.1071$. To articulate the performance of the proposed controller in a highly reverberant environment, an exaggerated noise model is adopted by adding a noise $\zeta(t)$ with a mean $\bar{\zeta} = -30$ dBm and a standard deviation of 50 dBm. This yielded a signal-to-noise ratio of -179.45 dBm. Figure 5 shows the resultant "noise-free" and noisy RSS signals obtained. The parameter values of the noisy RSS signal were found to be $\alpha = -60$ and $\beta = 0.2$. This noisy signal was used in the following controller's simulations.

6.2 Robot Stabilization on a Fixed Configuration

In this section, we present the robot's ability to stabilize on a fixed configuration regardless of its initial position and orientation. The stabilization performance of the proposed control scheme is evaluated by choosing the weight matrices as $\mathbf{P}(t_f) = diag(2, 2, 2)$
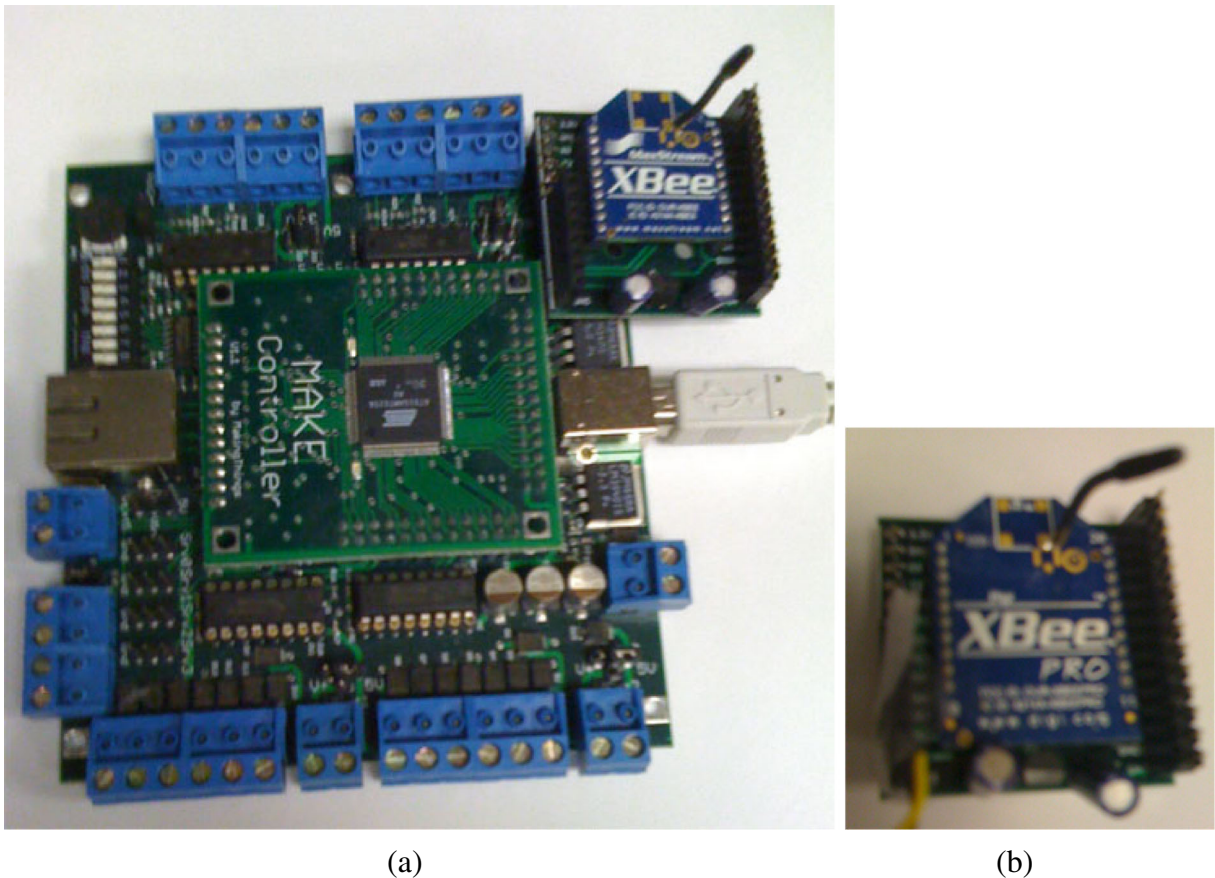

(a)   (b)

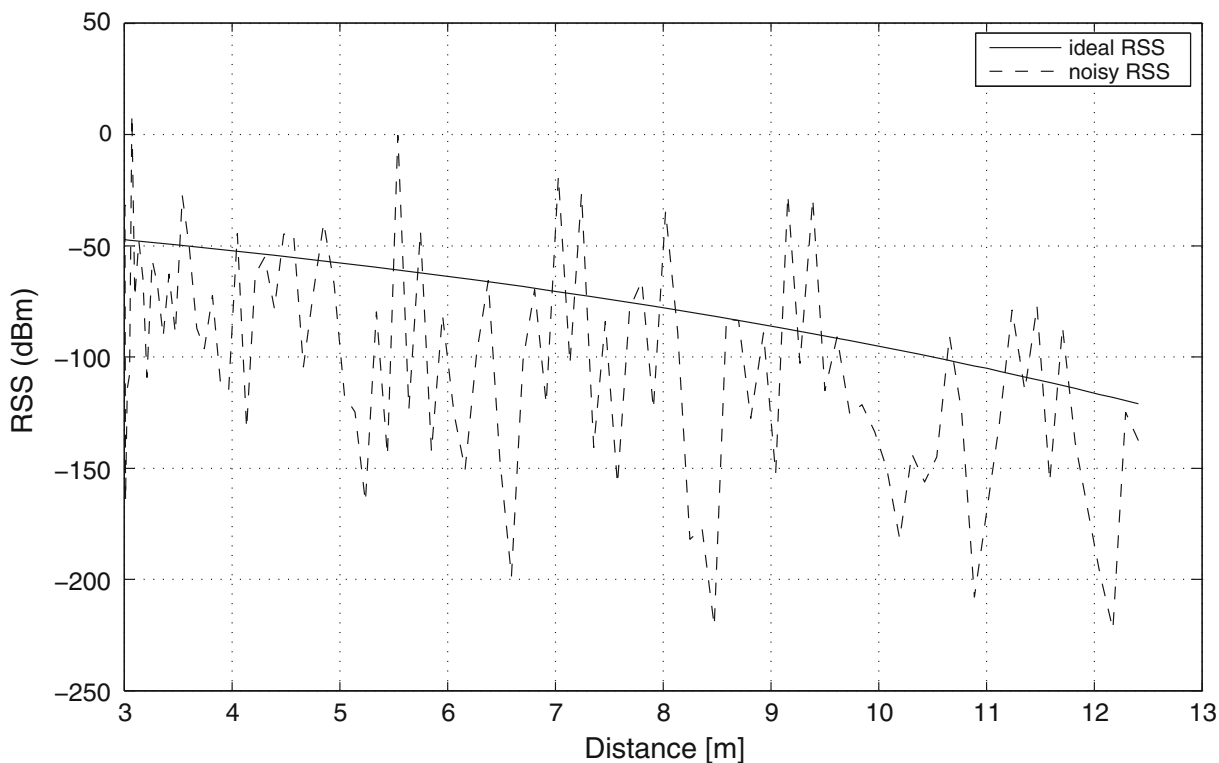**Fig. 4** **a** MakeController board, and **b** XBee Pro RF module

Fig. 5 Noise model considered for the simulation

and $\mathbf{Q}(t) = diag(0.01, 0.01, 0.01)$, $\forall\, t \in \mathscr{I}$. Hence, the stabilization at the target point is regarded 20 times more important than guiding the robot towards the target. The robot's goal is to stabilize itself at $(x, y) = (3, 8)$ m with an orientation of 90°. The initial position and orientation of the robot are set to $(0, 0)$ m and 28.6°, respectively. Figure 6a shows the simulation results, where the hollow and solid arrows represent the initial and final poses, respectively. The dashed path represents the robot's actual trajectory while the x's depict the 2-D projections of the RFID tags mounted on the ceiling. The distance between the robot and its target is shown in Fig. 6b. It reveals how fast the robot is approaching towards the target. The robot reached its target in about 20 s. Then, after some zigzagging, it could stabilize itself eventually with a position error of practically nil. This was achieved despite the excessively noisy RSS signals transmitted by the tags and the noisy actuator signals of the robot. The zigzagging behavior is expected due to the complexity of this task, especially for nonholonomic robots. Figure 6c reveals the corresponding control inputs, $\mathbf{u}(t)$, computed by the model (32). Since the

robot has to stabilize itself in 60 s, the zigzagging behavior on the corresponding control inputs was expected as it is clear from Fig. 6c, but eventually the robot stopped (speed is zero) at $t_f = 60$ s.

6.3 Tracking a Curvilinear Trajectory

The purpose of this test is to study the robot's tracking ability along a complex trajectory. To do that, we define a desired trajectory as $x^d(t) = 3\sin(\pi t/30)$, $y^d(t) = 3\sin(\pi t/15)$, and $\theta^d(t) = \tan^{-1}\left(\dot{y}^d/\dot{x}^d\right)$, for $t \in \mathscr{I}$. The robot is initially placed at $(0.5, 0)$ m with an initial orientation of 0°. Figure 7 summarizes the performance of the proposed neighboring optimal controller. The tracking error is plotted in Fig. 7b. Starting off its desired path, the robot converges in less than 3 s while keeping the left and right wheel rotational speeds within their limits (maximum of 10 rad $\cdot$ s^{-1}). See Fig. 7c for wheel speeds at time instant $t \in [0, 60]$ s. It is noticed from Fig. 7a that the robot looses track of its trajectory momentarily at a few sharp turns before converging back to it. The average tracking error throughout the whole
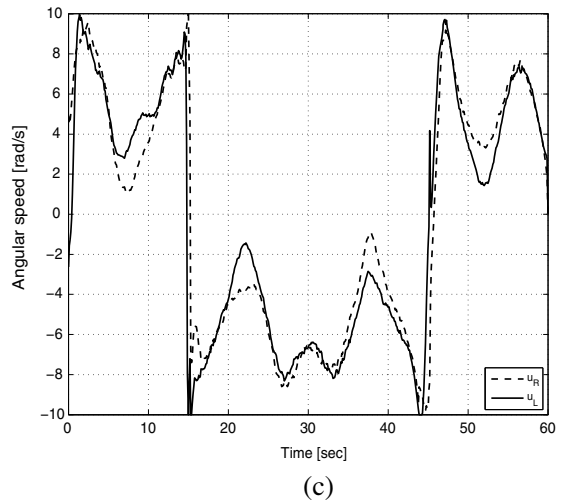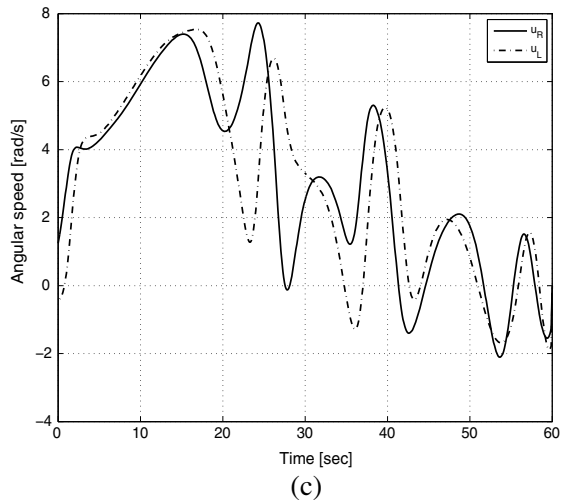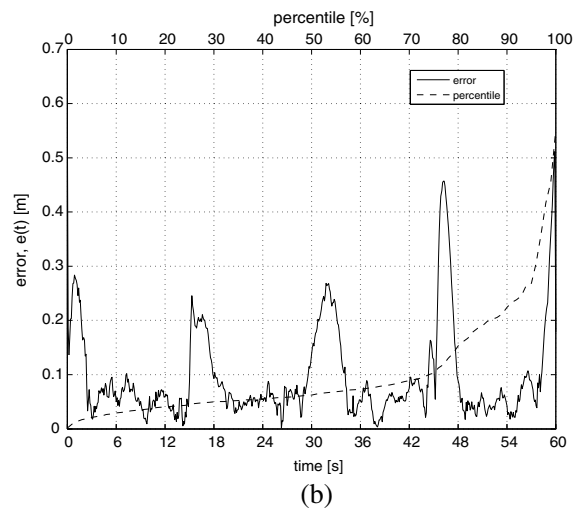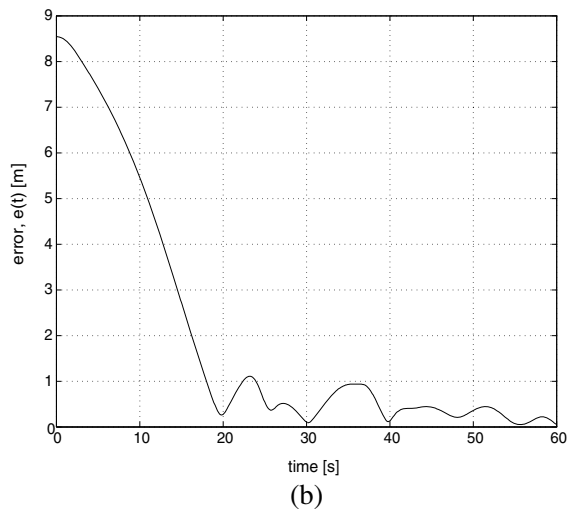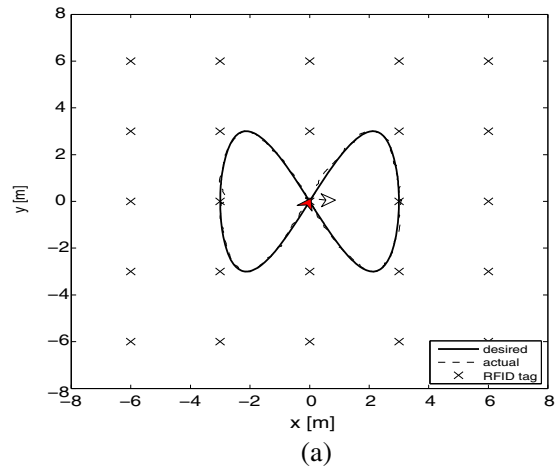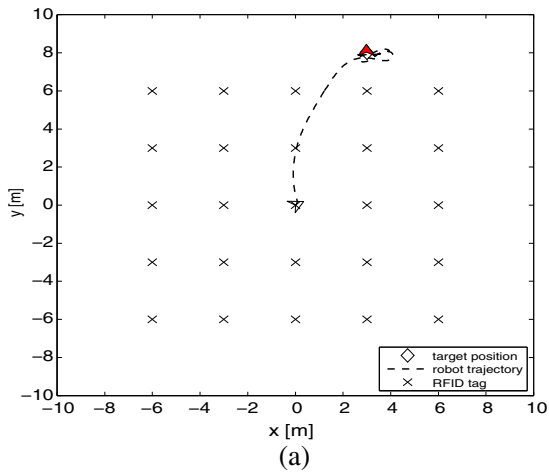
Fig. 6 Controller's performance in stabilizing on a fixed configuration **a** optimal trajectory, **b** error, and **c** control inputs (wheel speeds)

Fig. 7 Controller's performance in following a curvilinear trajectory **a** optimal trajectory, **b** error, and **c** control inputs (wheel speeds)

trajectory ($\mathscr{E}/60$ of Eq. 9) is 0.1 m. This is a very small error taking into account the total traveled distance of 30.8 m, the wheel speed constraints, and the excessive noise associated with the RF signals transmitted by the RFID tags. It is important to articulate that this error is non-cumulative. It is rather affected by the signal-to-noise ratio of the RF signals, but not by the traveled distance or navigation time. On the same figure, the percentile plot (whose x-axis is on top of the figure) shows that the tracking error is kept less than the average value (0.1 m) for about 75 % of the time, and less than 0.22 m for 90 % of the time. Once again, taking into account the aforementioned constraints, which are quite typical in any real-world robotic system, these values are considered very satisfactory.

The differential drive mobile robot's real-time motion and control performance are illustrated through a supplementary multimedia material enclosed with this paper. The material includes three short Matlab videos (1 min. each) showing the robot's capability in stabilizing itself on a fixed configuration (DDMR-stabilization.avi), in tracking an eight-shaped trajectory (DDMR-eight-tracking.avi), and in tracking a straight line segment (DDMR-line-tracking.avi).

7 Experimental Results

This section presents the results demonstrating the real-time performance of the proposed neighboring optimal controller. For that, the kinematic model (1) is realized by the Sputnik robot platform as pictured in Fig. 2a. The Sputnik is a two-wheel differential drive mobile robot whose kinematic model is derived by the conventional geometric model of a unicycle robot. The details of the low-level components, such as the torque-speed characteristics and interfacing mechanisms, were not disclosed by the robot manufacturer and have to be dealt with appropriately by the controller. Some of the robot's nominal relevant parameters as documented by its manufaturer are as follows: weight = 6.1 kg, diamater $\equiv l = 26$ cm, height = 47 cm, maximum linear speed = 1 m · s^{-1}, radius of each driving wheel $\equiv r = 8.25$ cm, maximum motor torque = 22kg · cm.

The MakeController board (emulating the RFID reader) mounted on the robot is connected to a laptop computer using an USB cable which allows

the robot to receive the RSS measurements coming from the XBee modules (emulating RFID tags). Note that the laptop computer and the robot is wirelessly connected through an wireless router in the robot's workspace. The experiment is conducted in the MIRaM laboratory of dimension about $10 \times 9 \times 3$ m at the University of Ottawa. The top-view of the workspace floor plan is depicted in Fig. 8.

The robot is supposed to follow the U-shaped rectilinear trajectory of length 16.5 m which is divided into three unequal segments A, B, and C. The four XBee modules in this case are located at positions $(1.7, 5.0, 0.7)$ m, $(-1.5, 4.9, 0.7)$ m, $(-1.5, 2.1, 0.7)$ m, and $(1.0, 2.6, 0.7)$ m and their corresponding 16-bit IDs are 0x5001, 0x5002, 0x5003, and 0x5004, respectively. The robot's real-time trajectory tracking performance is revealed in Fig. 9a, where the hollow and solid arrows indicate the robot's initial and final poses, respectively. The corresponding position tracking error, $e(t)$, is reported in Fig. 9b. The initial error spike is due to the uncertainty associated with the robot's initial pose but it rapidly went back to track the trajectory. Since the robot receives the RSS measurements from four XBee modules in the segment A, it's navigation performance is quite satisfactory in the sense that the tracking errors are less than 10 cm in almost everywhere in A. In addition, the error spikes at times $t \approx 25$ s and at $t \approx 39$ s are due to the turning points at the end of segments A and B of the trajectory. It is quite interesting to see that, unlike conventional odometric tracking algorithms, tracking errors are not cumulative as the robot travels over longer trajectory as it is clear from segments B and C, where the tracking errors are also less than 10 cm almost everywhere in these segments. The robot's tracking performance over the whole trajectory is quite satisfactory in the sense that 90 % of the time the error is less than 10 cm, as shown in the percentile plot of Fig. 9b.

The snapshots of this experiment while the robot is navigating along the U-shaped trajectory are summarized in Fig. 10. The robot is initially placed at the beginning of the segment A (see Fig. 10a). Figure 10a–g reveal the navigation performance of the segment A. After that, the robot had to turn at the first sharp corner to follow the segment B (see Fig. 10h). Note that the robot is no longer in line-of-sight with the XBee modules which shows the
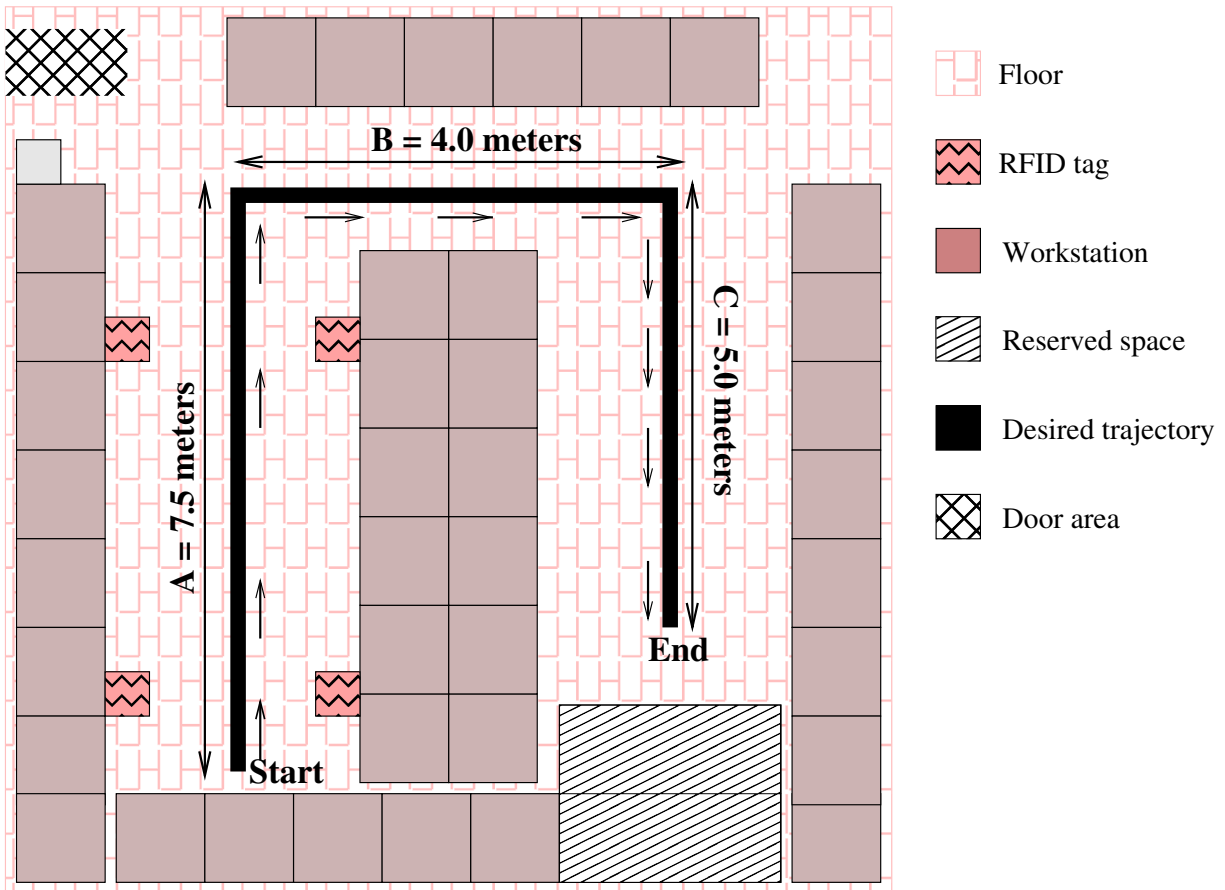
**Fig. 8** Floor plan of the robot's workspace at the MIRaM laboratory

power of RFID systems in navigating a mobile robot in indoor environment. When the robot is in segment C, it is completely out of line-of-sight from the XBee modules since the workstations are placed in between the segments B and C. As can be seen from Figs. 10m–p, the robot is still able to track this line segment with the tracking error of about 8 cm. It is important to articulate the fact that the main purpose of this experiment was to track the desired trajectory rather than stabilizing on a fixed point. Hence, the robot has stopped at about 5 cm away from the desired end point of segment C (see Fig. 10p).

Note that this controller is also able to stabilize the robot at a fixed configuration by simply tuning the weight matrices ($\mathbf{P}(t_f)$ and $\mathbf{Q}(t)$) of the cost function (14). This is in contrast to many recent RFID-based techniques which usually tackle the localization problem only [5, 14, 47]. The localization accuracy

reported there in is in the range of 0.1–0.5 m, despite neglecting the effect of reverberations and low signal-to-noise ratios. Moreover, the RFID-based robot navigation techniques presented in those papers are mostly based on simulations, see [14], for example, and some references therein.

8 Conclusion

In this paper, a neighboring optimal control strategy for solving trajectory tracking and point stabilization problems is proposed. It relies on dividing the whole control process into two sub-processes: finding nominal and neighboring optimal control inputs. The nominal trajectory is computed off-line which is deterministic. The neighboring trajectory is computed on-line. It requires an optimal filter for estimating the robots pose taking into account the noise associated
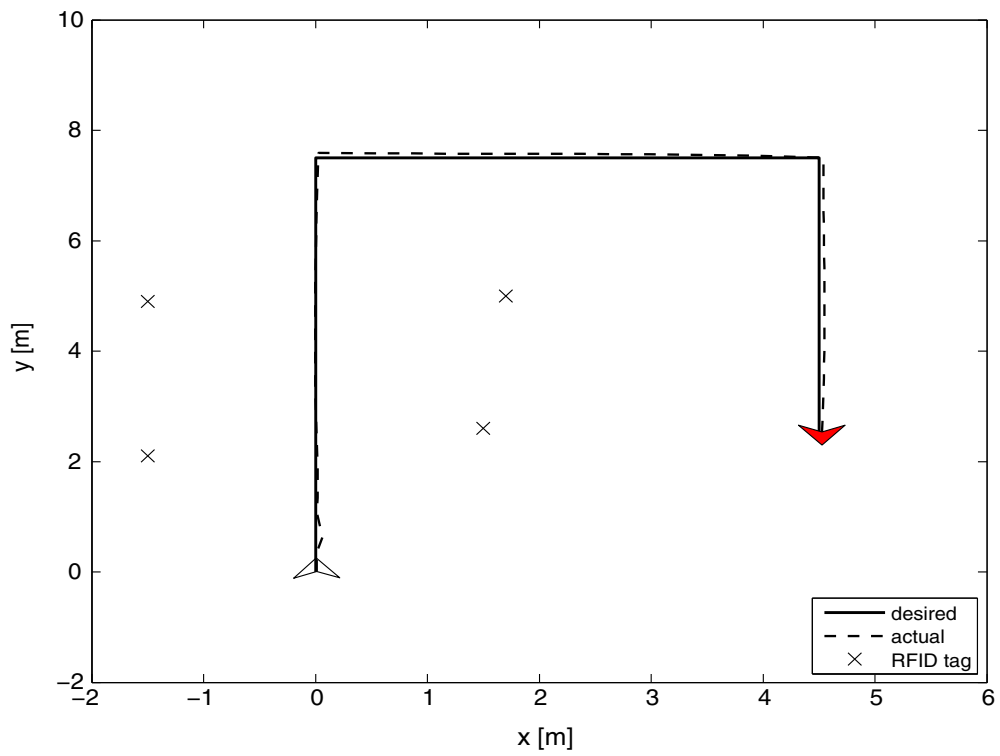
(a)
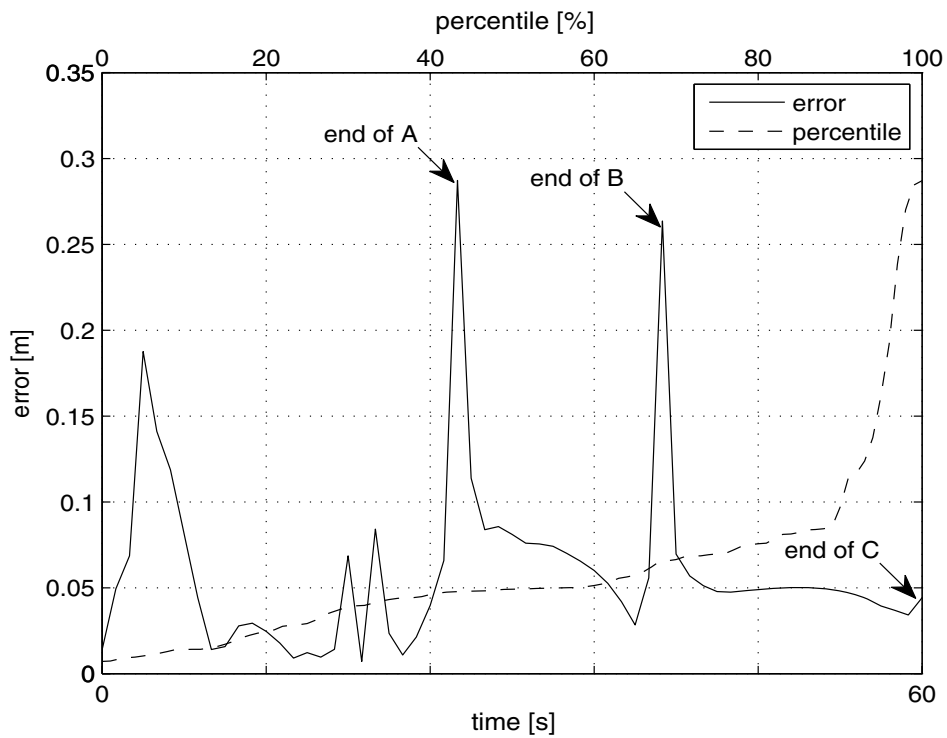
(b)

Fig. 9 Robot's real-time performance: **a** actual vs. desired trajectory and **b** tracking error
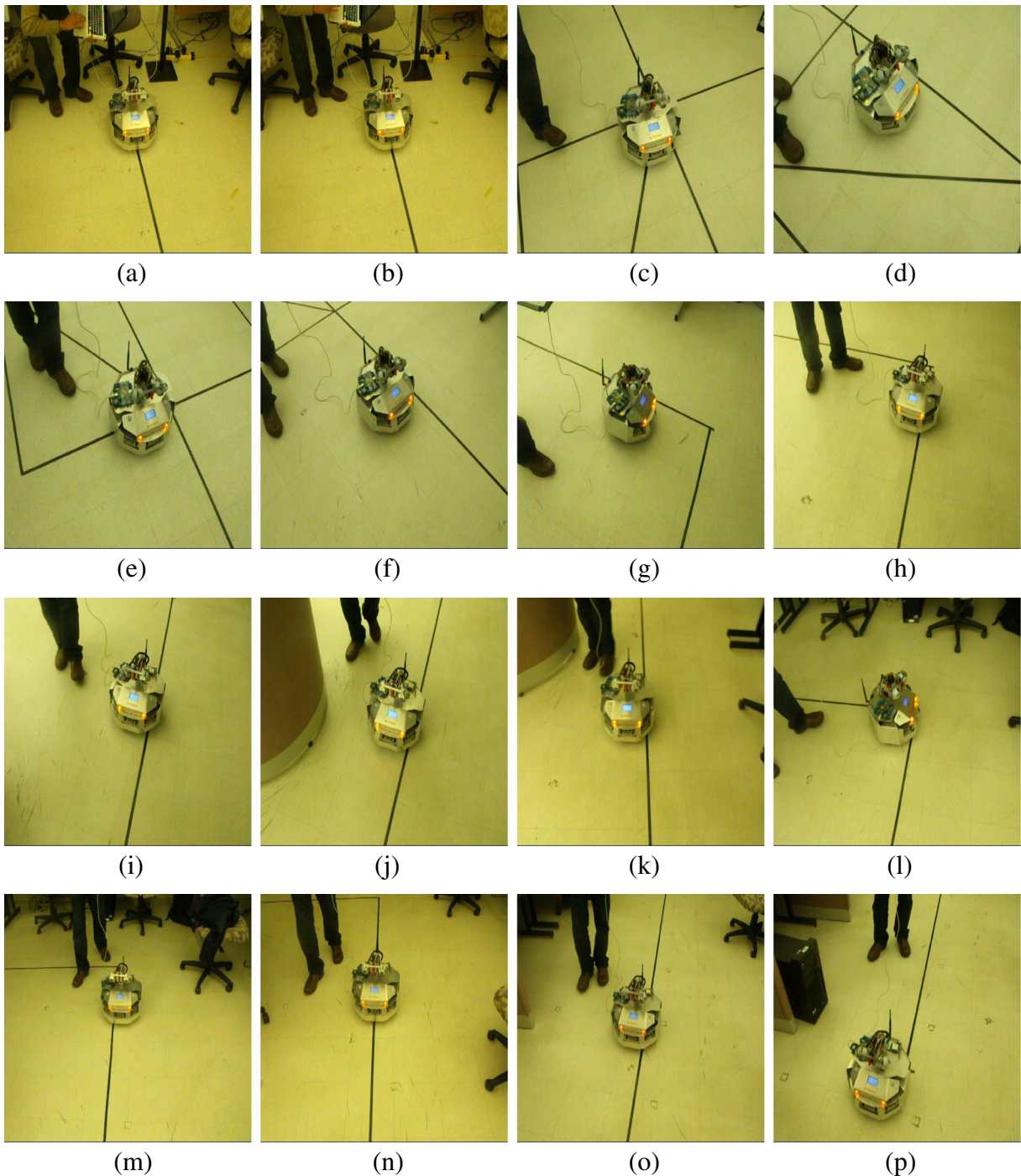
Fig. 10 Robot's real-time performance for tracking a rectilinear trajectory using neighboring optimal control

with the RSS measurements from the RFID tags and the robot's wheel speeds. The actual control actions are then computed by the sum of the nominal and the neighboring control inputs which lead the robot to track its pre-defined desired trajectory. Numerical results demonstrate the robot's ability to stabilize on a fixed configuration and to guide itself along a pre-defined trajectory with a satisfactory tracking error.

Moreover, the proposed control technique is modular in the sense that it is applicable to a broader class of nonlinear dynamic systems with process and measurement uncertainties.

Appendix

A.1 Proof of Theorem 1

Let $\mathbf{q}(t) \equiv \mathbf{q}[t, \mathbf{K}(t)]$ be the solution of the feedback system (12), with the cost functional (14) for any choice of $\mathbf{K}(t) \in \mathscr{K}_{ad}$. Since $\mathbf{K}^o(t)$ is optimal with the associated trajectory $\mathbf{q}^o(t)$, it is clear that $J(\mathbf{K}^o \leq J(\mathbf{K}), \ \forall \mathbf{K} \in \mathscr{K}_{ad}$. Suppressing the variable t for clarity and for any $\varepsilon \in [0, 1]$, we define $\mathbf{K}^\varepsilon = \mathbf{K}^o + \varepsilon(\mathbf{K} - \mathbf{K}^\mathbf{o})$. Since $\mathscr{K}$ is a closed convex set, $\mathscr{K}_{ad}$ is also a closed convex subset of $\mathscr{L}_\infty(\mathscr{I}, \mathbb{R}^{2\times3})$ and therefore $\mathbf{K}^\varepsilon \in \mathscr{K}_{ad}$. Thus $J(\mathbf{K}^\mathbf{o}) \leq J(\mathbf{K}^\varepsilon)$, which follows that

$$Tr\left[\left(\mathbf{K} - \mathbf{K}^o\right)^T dJ(\mathbf{K}^o)\right] \geq 0, \tag{40}$$

where $dJ(\mathbf{K}^o)$ denotes the Gateaux (directional) derivative of J evaluated at $\mathbf{K} = \mathbf{K}^o$ in the direction of $(\mathbf{K} - \mathbf{K}^o)$.

Let $\mathbf{q}^\varepsilon$ be the solution of the feedback system (12) corresponding to the gain $\mathbf{K}^\varepsilon$ with the same initial state $\mathbf{q}^\varepsilon(t_0) = \mathbf{q}_0$. It is easy to very that

$$\lim_{\varepsilon \to 0} \mathbf{K}^\varepsilon(t) = \mathbf{K}^o(t), \quad \text{and} \quad \lim_{\varepsilon \to 0} \mathbf{q}^\varepsilon(t) = \mathbf{q}^o(t).$$

Using $\dot{\mathbf{q}}^\varepsilon = \hat{\mathbf{f}}(\mathbf{q}^\varepsilon, \mathbf{K}^\varepsilon)$ and $\dot{\mathbf{q}}^\mathbf{o} = \hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K}^o)$ with $\mathbf{q}^\varepsilon(t_0) = \mathbf{q}^o(t_0) = \mathbf{q}_0$ which, yield the following equation

$$\dot{\mathbf{q}}^\varepsilon - \dot{\mathbf{q}}^o = \hat{\mathbf{f}}(\mathbf{q}^\varepsilon, \mathbf{K}^o) - \hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K}^o) + \varepsilon\hat{\mathbf{f}}(\mathbf{q}^\varepsilon, \mathbf{K} - \mathbf{K}^o). \tag{41}$$

Dividing by ε and denoting

$$\boldsymbol{\eta}(t) \equiv \lim_{\varepsilon \to 0} \left(\frac{\mathbf{q}^\varepsilon(t) - \mathbf{q}^o(t)}{\varepsilon}\right)$$

it follows from the expression (41) that $\boldsymbol{\eta}(t)$ must satisfy the following initial value problem

$$\dot{\boldsymbol{\eta}} = \frac{\partial \hat{\mathbf{f}}}{\partial \mathbf{q}}(\mathbf{q}^o, \mathbf{K}^o)\boldsymbol{\eta} + \hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o), \ \boldsymbol{\eta}(t_0) = 0. \tag{42}$$

Equation (42) is a linear non-homogeneous equation with $\hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o)$ being the driving force. As a result, it has a continuous solution $\boldsymbol{\eta}(t) \in C(\mathscr{I}, \mathbb{R}^3)$, which is continuously dependent on $\hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o)$.

By definition of Gateaux (directional) derivative we can derive the following expression

$$Tr\left[\left(\mathbf{K} - \mathbf{K}^o\right)^T dJ(\mathbf{K}^o)\right] = \lim_{\varepsilon \to 0} \frac{J[\mathbf{K}^o + \varepsilon(\mathbf{K} - \mathbf{K}^o)] - J(\mathbf{K}^o)}{\varepsilon}$$

$$= \boldsymbol{\eta}^T(t)\frac{\partial \phi}{\partial \mathbf{q}}[t_f, \mathbf{q}(t_f)] + \int_{t_0}^{t_f} \boldsymbol{\eta}^T(t)\frac{\partial \ell}{\partial \mathbf{q}}[t, \mathbf{q}(t)]dt.$$

Hence, inequality (40) yields

$$\boldsymbol{\eta}^T(t)\frac{\partial \phi}{\partial \mathbf{q}}[t_f, \mathbf{q}(t_f)] + \int_{t_0}^{t_f} \boldsymbol{\eta}^T(t)\frac{\partial \ell}{\partial \mathbf{q}}[t, \mathbf{q}(t)]dt \geq 0. \tag{43}$$

Since $\boldsymbol{\eta}(t)$ of the variational equation (42) is continuously dependent on $\hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o)$, the map

$$\hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o) \longmapsto \boldsymbol{\eta}(t), \ \ t \in \mathscr{I}$$

is continuous from $\mathscr{L}_1(\mathscr{I}, \mathbb{R}^3)$ to $C(\mathscr{I}, \mathbb{R}^3)$ [2, p. 260]. Hence, the map

$$\boldsymbol{\eta}(t) \longmapsto \boldsymbol{\eta}^T(t)\frac{\partial \phi}{\partial \mathbf{q}}[t_f, \mathbf{q}(t_f)] + \int_{t_0}^{t_f} \boldsymbol{\eta}^T(t)\frac{\partial \ell}{\partial \mathbf{q}}[t, \mathbf{q}(t)]dt$$

is a continuous linear functional on $C(\mathscr{I}, \mathbb{R}^3)$. Thus, the composition map

$$\hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o) \longmapsto \boldsymbol{\eta}^T(t)\frac{\partial \phi}{\partial \mathbf{q}}[t_f, \mathbf{q}(t_f)]$$
$$+ \int_{t_0}^{t_f} \boldsymbol{\eta}^T(t)\frac{\partial \ell}{\partial \mathbf{q}}[t, \mathbf{q}(t)]dt$$

is a continuous linear functional on $\mathscr{L}_1(\mathscr{I}, \mathbb{R}^3)$, where $\hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o) \in \mathscr{L}_1(\mathscr{I}, \mathbb{R}^3)$. Therefore, by the Riesz representation theorem or by the duality between $\mathscr{L}_1(\mathscr{I}, \mathbb{R}^3)$ and $\mathscr{L}_\infty(\mathscr{I}, \mathbb{R}^3)$, we may conclude that there exists an element $\boldsymbol{\psi}^o \in \mathscr{L}_\infty(\mathscr{I}, \mathbb{R}^3)$ such that

$$Tr\left[\left(\mathbf{K} - \mathbf{K}^o\right)^T dJ(\mathbf{K}^o)\right] = \boldsymbol{\eta}^T(t)\frac{\partial \phi}{\partial \mathbf{q}}[t_f, \mathbf{q}(t_f) +$$

$$\int_{t_0}^{t_f} \boldsymbol{\eta}^T(t)\frac{\partial \ell}{\partial \mathbf{q}}[t, \mathbf{q}(t)]\,dt = \int_{t_0}^{t_f} \left(\boldsymbol{\psi}^o\right)^T \hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o)dt. \tag{44}$$

It follows from inequality (43) that

$$\int_{t_0}^{t_f} \left(\boldsymbol{\psi}^o\right)^T \hat{\mathbf{f}}(\mathbf{q}^o, \mathbf{K} - \mathbf{K}^o)dt \geq 0, \ \ \forall \mathbf{K} \in \mathscr{K}_{ad}. \tag{45}$$

Using the variational equation (42), it follows from the second identity of Eq. (44) that

$$
\boldsymbol{\eta}^T(t)\frac{\partial\phi}{\partial\mathbf{q}}[t_f,\mathbf{q}(t_f)] + \int_{t_0}^{t_f}\boldsymbol{\eta}^T(t)\frac{\partial\ell}{\partial\mathbf{q}}[t,\mathbf{q}(t)]dt =
$$

$$
\int_{t_0}^{t_f}\left\{\left(\boldsymbol{\psi}^o\right)^T\left[\dot{\boldsymbol{\eta}} - \frac{\partial\hat{\mathbf{f}}}{\partial\mathbf{q}}(\mathbf{q}^o,\mathbf{K}^o)\right]\boldsymbol{\eta}(t)\right\}dt. \tag{46}
$$

Integrating by parts and since $\boldsymbol{\eta}(t_0)=0$,

$$
\int_{t_0}^{t_f}\left(\boldsymbol{\psi}^o\right)^T\left[\dot{\boldsymbol{\eta}}(t) - \frac{\partial\hat{\mathbf{f}}}{\partial\mathbf{q}}(\mathbf{q}^o,\mathbf{K}^o)\right]\boldsymbol{\eta}(t)dt =
$$

$$
\boldsymbol{\eta}^T(t_f)\boldsymbol{\psi}^o(t_f) + \int_{t_0}^{t_f}\boldsymbol{\eta}^T(t)\left(-\dot{\boldsymbol{\psi}} - \left[\frac{\partial\hat{\mathbf{f}}}{\partial\mathbf{q}}(\mathbf{q}^o,\mathbf{K}^o)\right]^T\boldsymbol{\psi}^o\right)dt.
$$

Expression (46) can now be written as

$$
\boldsymbol{\eta}^T(t)\frac{\partial\phi}{\partial\mathbf{q}}[t_f,\mathbf{q}(t_f)] + \int_{t_0}^{t_f}\boldsymbol{\eta}^T(t)\frac{\partial\ell}{\partial\mathbf{q}}[t,\mathbf{q}(t)]dt =
$$

$$
\boldsymbol{\eta}^T(t_f)\boldsymbol{\psi}^o(t_f) + \int_{t_0}^{t_f}\boldsymbol{\eta}^T\left\{-\dot{\boldsymbol{\psi}}^o - \left[\frac{\partial\hat{\mathbf{f}}}{\partial\mathbf{q}}(\mathbf{q}^o,\mathbf{K}^o)\right]^T\boldsymbol{\psi}^o\right\}dt \tag{47}
$$

It is clear from Eq. (47) that

$$
\dot{\boldsymbol{\psi}}^o = -\left[\frac{\partial\hat{\mathbf{f}}}{\partial\mathbf{q}}(\mathbf{q}^o,\mathbf{K}^o)\right]^T\boldsymbol{\psi}^o - \frac{\partial\ell}{\partial\mathbf{q}}[t,\mathbf{q}^o(t)] \text{ and}
$$

$$
\boldsymbol{\psi}^o(t_f) = \frac{\partial\phi}{\partial\mathbf{q}}[t_f,\mathbf{q}(t_f)]. \tag{48}
$$

The costate dynamics (48) is linear along the optimal trajectories. Thus, the necessary conditions of optimality is given by the integral inequality (45), the costate dynamics (48), and the state equation (12). In other words, the choice of $\mathbf{K}\in\mathcal{K}_{ad}$ determines the optimality conditions (45), (48), and (12).

Consider the optimality condition (45) and rewriting it as follows

$$
\int_{t_0}^{t_f}\left(\boldsymbol{\psi}^o\right)^T\hat{\mathbf{f}}(\mathbf{q}^o,\mathbf{K})dt \geq \int_{t_0}^{t_f}\left(\boldsymbol{\psi}^o\right)^T\hat{\mathbf{f}}(\mathbf{q}^o,\mathbf{K}^o)dt,
$$

$$
\forall\,\mathbf{K}\in\mathcal{K}_{ad}. \tag{49}
$$

Using the integral inequality (49), it is easy to derive the point-wise inequality [2]

$$
\left(\boldsymbol{\psi}^o\right)^T\hat{\mathbf{f}}(\mathbf{q}^o,\mathbf{K}) \geq \left(\boldsymbol{\psi}^o\right)^T\hat{\mathbf{f}}(\mathbf{q}^o,\mathbf{K}^o), \quad \forall\,\mathbf{K}\in\mathcal{K}_{ad}. \tag{50}
$$

Now adding the term $\ell[t,\mathbf{q}^o(t)]$ in both sides of Eq. (50) yields

$$
\left(\boldsymbol{\psi}^o\right)^T\hat{\mathbf{f}}(\mathbf{q}^o,\mathbf{K}) + \ell[t,\mathbf{q}^o(t)] \geq \left(\boldsymbol{\psi}^o\right)^T\hat{\mathbf{f}}(\mathbf{q}^o,\mathbf{K}^o) + \ell[t,\mathbf{q}^o(t)],
$$

which gives the Hamiltonian inequality

$$
\mathscr{H}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}(t)] \geq \mathscr{H}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}^o(t)].
$$

This is the same as inequality (16) stated in the theorem. Differentiating $\mathscr{H}$ with respect to the costate variable $\boldsymbol{\psi}$, we get

$$
\frac{\partial\mathscr{H}}{\partial\boldsymbol{\psi}}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}^o(t)] = \hat{\mathbf{f}}[\mathbf{q}^o(t),\mathbf{K}^o(t)],
$$

which leads to the state equation

$$
\dot{\mathbf{q}}^o = \frac{\partial\mathscr{H}^o}{\partial\boldsymbol{\psi}}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}^o(t)], \quad \mathbf{q}^o(t_0) = \mathbf{q}_0,
$$

as defined in Eq. (17).

Differentiating $\mathscr{H}$ with respect to the state variable $\mathbf{q}$ yields

$$
\frac{\partial\mathscr{H}}{\partial\mathbf{q}}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}^o(t)] = \left[\frac{\partial\hat{\mathbf{f}}}{\partial\mathbf{q}}(q^o,K^o)\right]^T\boldsymbol{\psi}^o
$$

$$
+ \frac{\partial\ell}{\partial\mathbf{q}}[t,\mathbf{q}^o(t)] = -\dot{\boldsymbol{\psi}}^o,
$$

Hence, the costate dynamics (48) can be expressed in terms of Hamiltonian as

$$
\dot{\boldsymbol{\psi}}^o = -\frac{\partial\mathscr{H}^o}{\partial\mathbf{q}}[t,\mathbf{q}^o(t),\boldsymbol{\psi}^o(t),\mathbf{K}^o(t)], \text{ and}
$$

$$
\boldsymbol{\psi}^o(t_f) = \frac{\partial\phi}{\partial\mathbf{q}}[t_f,\mathbf{q}(t_f)],
$$

which is the condition (18).

A.2 Proof of Theorem 3

Consider $V : [0,\infty)\times\mathbb{R}^3 \rightarrow \mathbb{R}$ is a Lyapunov-candidate-function and is given by

$$
V[t,\Delta\mathbf{q}(t)] = \frac{1}{2}\Delta\mathbf{q}^T(t)\mathbf{P}(t)\Delta\mathbf{q}(t) \tag{51}
$$

and $\mathbf{P}(t)$, which is the solution of Eq. (31), is a real, symmetric, positive semi-definite matrix. We

first proof that Eq. (51) satisfies the Lyapunov basic properties.

Clearly, from Eq. (51), $V[t, \mathbf{0}] = 0$, and $\frac{\partial V}{\partial \Delta \mathbf{q}}$ $[t, \Delta \mathbf{q}(t)] = \mathbf{P}(t)\Delta \mathbf{q}(t) \in C^1(\mathscr{I}, \mathbb{R}^3)$. Following [53, Ch. 3], let us define

$$\Delta \phi(t) = \Delta \mathbf{q}^T(t)\mathbf{P}(t)\Delta \mathbf{q}(t) = \Delta \mathbf{q}^T \Delta \boldsymbol{\psi}(t), \quad t \in \mathscr{I}. \tag{52}$$

It is known that

$$\Delta \phi(t_f) = \Delta \mathbf{q}^T(t_f)\Delta \boldsymbol{\psi}(t_f) = \Delta \mathbf{q}^T(t_f)\mathbf{P}(t_f)\Delta \mathbf{q}(t_f) \geq 0.$$

Differentiating (52) with respect to t and dropping the variable (t) for clarity, we get

$$\Delta \dot{\phi} = -\Delta \mathbf{q}^T \mathbf{Q}\Delta \mathbf{q} - \Delta \mathbf{u}^T \mathbf{R}\Delta \mathbf{u} \leq 0. \tag{53}$$

Now integrating and using the above expression, we find that

$$\Delta \phi(t) = \Delta \mathbf{q}^T(t_f)\mathbf{P}(t_f)\Delta \mathbf{q}(t_f) + \int_t^{t_f} [\Delta \mathbf{q}^T(\tau)\mathbf{Q}(\tau)\Delta \mathbf{q}(\tau) +$$
$$\Delta \mathbf{u}^T(\tau)\mathbf{R}(\tau)\Delta \mathbf{u}(\tau)]d\tau \geq 0, \quad \forall t \in \mathscr{I},$$

since $\mathbf{P}(t) \geq \mathbf{0}$, $\mathbf{Q}(t) \geq \mathbf{0}$, $\mathbf{R}(t) > \mathbf{0}$, and all are symmetric matrices. Hence, the solution of the differential matrix Riccati equation (31) has to be real, symmetric, and at least positive semi-definite matrix for $V[t, \Delta \mathbf{q}(t)] = (1/2)\Delta \phi(t) \geq 0$. Thus, $V[t, \Delta \mathbf{q}(t)]$ in Eq. (51) satisfies the Lyapunov basic properties.

By taking the time-derivative of the Lyapunov function (51) and using the expression (53), it follows that

$$\dot{V} = \frac{1}{2}\Delta \dot{\phi} = \frac{1}{2}\left\{-\Delta \mathbf{q}^T \mathbf{Q}\Delta \mathbf{q} - \Delta \mathbf{u}^T \mathbf{R}\Delta \mathbf{u}\right\} \leq 0. \tag{54}$$

(i) It is certain from Eq. (54) that if $\mathbf{Q}(t)$ is a real, symmetric, positive semi-definite matrix, then $\dot{V}[t, \Delta \mathbf{q}(t)] = (1/2)\Delta \dot{\phi} \leq 0$. Hence, the feedback system (33) is stable in the Lyapunov sense with respect to the Lyapunov function (51).

(ii) Since $\mathbf{R}$ is positive definite and if $\mathbf{Q}(t)$ is also positive definite, the expression (54) yields $\dot{V} < 0$. Therefore, the feedback system (33) is asymptotically stable.

References

1. Aguiar, A.P., Hespanha, J.P.: Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. IEEE Trans. Autom. Control **52**(8), 1362–1379 (2007)

2. Ahmed, N.U.: Dynamic Systems and Control with Applications. World Scientific, New Jersey (2006)

3. Aicardi, M., Casalino, G., Bicchi, A., Balestrino, A.: Closed loop steering of unicycle like vehicles via lyapunov techniques. Robot. Autom. Mag. IEEE **2**(1), 27–35 (1995)

4. Alanis, A.Y., Lopez-Franco, M., Arana-Daniel, N., Lopez-Franco, C.: Discrete-time neural control for electrically driven nonholonomic mobile robots. J. Adapt. Control Sig. Process. **26**(7), 630–44 (2012)

5. Bekkali, A., Matsumoto, M.: RFID indoor tracking system based on inter-tags distance measurements. In: Wireless Technology, Lecture Notes in Electrical Engineering, vol. 44, pp. 41–62. Springer, US (2009)

6. Brockett, R.W.: Asymptotic stability and feedback stabilization. Differ. Geom. Control Theory 181–191 (1983)

7. Cao, Z., Zhao, Y., Wang, S.: Trajectory tracking and point stabilization of noholonomic mobile robot. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1328–1333 (2010)

8. Chen, H., Ma, M.M., Wang, H., Liu, Z.Y., Cai, Z.X.: Moving horizon $\mathscr{H}_\infty$ tracking control of wheeled mobile robots with actuator saturation. IEEE Trans. Control Syst. Technol. **17**(2), 449–57 (2009)

9. Choi, B.S., Lee, J.W., Lee, J.J., Park, K.T.: A hierarchical algorithm for indoor mobile robot localization using RFID sensor fusion. IEEE Trans. Ind. Electron. **58**(6), 2226–2235 (2011)

10. Chwa, D.: Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates. IEEE Trans. Control Syst. Technol. **12**(4), 637–644 (2004)

11. Chwa, D.: Tracking control of differential-drive wheeled mobile robots using a backstepping-like feedback linearization. IEEE Trans. Syst. Man Cybern. A Syst. Hum. **40**(6), 1285–1295 (2010)

12. Cui, Q., Li, X., Wang, X., Zhang, M.: Backstepping control design on the dynamics of the omni-directional mobile robot. Appl. Mech. Mater. **203**, 51–6 (2012)

13. Cui, R., Ge, S.S., How, B.V.E., Choo, Y.S.: Leader–follower formation control of underactuated autonomous underwater vehicles. Ocean Eng. **37**(1), 1491–1502 (2010)

14. Di Giampaolo, E., Martinelli, F.: A passive UHF-RFID system for the localization of an indoor autonomous vehicle. IEEE Trans. Ind. Electron. **59**(10), 3961–3970 (2012)

15. Fierro, R., Lewis, F.L.: Robust practical point stabilization of a nonholonomic mobile robot using neural networks. J. Intell. Robot. Syst. **20**, 295–317 (1997)

16. Gonzalez, R., Fiacchini, M., Alamo, T., Guzman, J., Rodriguez, F.: Online robust tube-based mpc for time-varying systems: a practical approach. Int. J. Control **84**(6), 1157–1170 (2011)

17. Gueaieb, W., Miah, M.S.: An intelligent mobile robot navigation technique using RFID technology. IEEE Trans. Instrum. Meas. **57**(9), 1908–1917 (2008)

18. Hahnel, D., Burgard, W., Fox, D., Fishkin, K., Philipose, M.: Mapping and localization with RFID technology. In: IEEE International Conference on Robotics and Automation, pp. 1015–1020. New Orleans (2004)

19. Han, S., Lim, H., Lee, J.: An efficient localization scheme for a differential-driving mobile robot based on rfid system. IEEE Trans. Ind. Electron. **54**(6), 3362–3369 (2007)

20. Huang, H.C., Tsai, C.C.: Adaptive robust control of an omnidirectional mobile platform for autonomous service robots in polar coordinates. J. Intell. Robot. Syst. **51**, 439–460 (2008)

21. Hwang, C.L., Chang, N.W.: Fuzzy decentralized sliding-mode control of a car-like mobile robot in distributed sensor-network spaces. IEEE Trans. Fuzzy Syst. **16**(1), 97–109 (2008)

22. Hwang, S.Y., Song, J.B.: Monocular vision-based SLAM in indoor environment using corner, lamp, and door features from upward-looking camera. IEEE Trans. Ind. Electron. (2011). doi:10.1109/TIE.2011.2109333

23. Janabi-Sharifi, F., Marey, M.: A kalman-filter-based method for pose estimation in visual servoing. IEEE Trans. Robot. **26**(5), 939–947 (2010)

24. Jang, J.O.: Adaptive neuro-fuzzy network control for a mobile robot. J. Intell. Robot. Syst. **62**, 567–586 (2011)

25. Jiang, Z.P., Lefeber, E., Nijmeijer, H.: Saturated stabilization and tracking of a nonholonomic mobile robot. Syst. Control Lett. **42**(5), 327–32 (2001)

26. Jiang, Z.P., Nijmeijer, H.: Tracking control of mobile robots: a case study in backstepping. Automatica **33**(7), 1393–9 (1997)

27. Jiang, Z.P., Nijmeijer, H.: A recursive technique for tracking control of nonholonomic systems in chained form. IEEE Trans. Autom. Control **44**(2), 265–279 (1999)

28. Jing, L., Yang, P.: A localization algorithm for mobile robots in rfid system. In: International Conference on Wireless Communications, Networking and Mobile Computing, pp. 2109–2112 (2007)

29. Kanayama, Y., Kimura, Y., Miyazaki, F., Noguchi, T.: A stable tracking control method for an autonomous mobile robot. In: Proceedings-IEEE International Conference on Robotics and Automation, pp. 384–389. Cincinnati (1990)

30. Khalil, H.K.: Nonlinear Systems, 3rd edn. Addison-Wesley, Pearson Education Limited, Upper Saddle River (2002)

31. Kim, B., Necsulescu, D., Sastadek, J.: Autonomous mobile robot model predictive control. Int. J. Control. **77**(16), 1438–1445 (2004)

32. Kodaka, K., Niwa, H., Sakamoto, Y., Otake, M., Kanemori, Y., Sugano, S.: Pose estimation of a mobile robot on a lattice of rfid tags. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1385–1390 (2008)

33. Kulyukin, V., Gharpure, C., Nicholson, J., Pavithran, S.: RFID in robot-assisted indoor navigation for the visually impaired. In: 2004 IEEE/RSJ IROS, pp. 1979–1984. Sendai (2004)

34. Lacevic, B., Velagic, J.: Evolutionary design of fuzzy logic based position controller for mobile robot. J. Intell. Robot. Syst. **63**, 595–614 (2011)

35. Lee, J.H., Lin, C., Lim, H., Lee, J.M.: Sliding mode control for trajectory tracking of mobile robot in the RFID sensor space. Int. J. Control. Autom. Syst. **7**(3), 429–435 (2009)

36. Lee, T.C., Song, K.T., Lee, C.H., Teng, C.C.: Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. IEEE Trans. Control Syst. Technol. **9**(2), 305–18 (2001)

37. Li, T.H.S., Chang, S.J.: Autonomous fuzzy parking control of a car-like mobile robot. IEEE Trans. Syst. Man Cybern. A **33**(4), 451–65 (2003)

38. Li, Z., Yang, C.: Neural-adaptive output feedback control of a class of transportation vehicles based on wheeled inverted pendulum models. IEEE Trans. Control Syst. Technol. **20**(6), 1583–1591 (2012)

39. Li, Z., Yang, C., Tang, Y.: Decentralised adaptive fuzzy control of coordinated multiple mobile manipulators interacting with non-rigid environments. IET Control Theory Appl. **7**(3), 397–410 (2013)

40. de Luca, A., Oriolo, G., Samson, C.: Feedback control of a nonholonomic car-like robot. In: Laumond J.P. (ed.) Robot Motion Planning and Control, Lecture Notes in Control and Information Sciences, vol. 229, chap. 4, pp. 170–253. Springer (2000)

41. Luo, R., Chuang, C.T., Huang, S.S.: RFID-based indoor antenna localization system using passive tag and variable RF-attenuation. In: IEEE International Conference on Industrial Electronics Society, pp. 2254–2259 (2007)

42. Miah, M.S., Gueaieb, W.: Towards a computationally efficient relative positioning system for indoor environments: an RFID approach. In: ICINCO. Milan (2009)

43. Michalek, M., Kozowski, K.: Vector-field-orientation feedback control method for a differentially driven vehicle. IEEE Trans. Control Syst. Technol. **18**(1), 45–65 (2010)

44. Morin, P., Samson, C.: Control of nonholonomic mobile robots based on the transverse function approach. IEEE Trans. Robot. **25**(5), 1058–1073 (2009)

45. Nganga-Kouya, D., Okou, F.: Adaptive backstepping control of a wheeled mobile robot. In: 17th Mediterranean Conference on Control and Automation, pp. 85–91. Piscataway (2009)

46. Park, B.S., Yoo, S.J., Park, J.B., Choi, Y.H.: A simple adaptive control approach for trajectory tracking of electrically driven nonholonomic mobile robots. IEEE Trans. Control Syst. Technol. **18**(5), 1199–1206 (2010)

47. Park, S., Hashimoto, S.: Autonomous mobile robot navigation using passive RFID in indoor environment. IEEE Trans. Ind. Electron. **56**(7), 2366–2373 (2009)

48. Pomet, J.B.: Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift. Syst. Control Lett. **18**(2), 147–158 (1992)

49. Royden, H.L., Fitzpatrick, P.M.: Real Analysis, 4th edn. Pearson Education Inc., Boston (2010)

50. Rubagotti, M., Della Vedova, M., Ferrara, A.: Time-optimal sliding-mode control of a mobile robot in a dynamic envi

ronment. IET Control Theory Appl. **5**(16), 1916–1924 (2011)

51. Sheng, L., Guoliang, M., Weili, H.: Stabilization and optimal control of nonholonomic mobile robot. In: Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th, vol. 2, pp. 1427–1430 (2004)

52. Siagian, C., Itti, L.: Biologically inspired mobile robot vision localization. IEEE Trans. Robot. **25**(4), 861–873 (2009)

53. Stengel, R.F.: Optimal Control and Estimation. Dover Publications Inc (1994)

54. Sun, D., Wang, C., Shang, W., Feng, G.: A synchronization approach to trajectory tracking of multiple mobile robots while maintaining time-varying formations. IEEE Trans. Robot. **25**(5), 1074–1086 (2009)

55. de Wit, C.C., Samson, C., Sørdalen, O.J., Zheng, Y.F. (eds.): Nonlinear Control Design for Mobile Robots, vol. 11. World Scientific, Singapore (1993)

56. Yang, C., Li, Z., Li, J.: Trajectory planning and optimized adaptive control for a class of wheeled inverted pendulum vehicle models. IEEE Trans. Cybern. **43**(1), 24–36 (2013)