

Research Article

Observer-Based Sliding Mode Control for Path Tracking of a Spherical Robot

Majid Taheri Andani,¹ Zahra Ramezani ,² Saeed Moazami,¹ Jinde Cao ,^{3,4}
Mohammad Mehdi Arefi ,⁵ and Hassan Zargarzadeh ¹

¹Phillip M. Drayer Electrical Engineering Department, Lamar University, Beaumont, TX, USA

²Electrical Engineering Department, Iran University of Science and Technology, Tehran, Iran

³School of Mathematics, Southeast University, Nanjing 210096, China

⁴Research Centre for Complex Systems and Network Sciences, Southeast University, Nanjing 210096, China

⁵Department of Power and Control Engineering, School of Electrical and Computer Engineering, Shiraz University, Shiraz 71348-51154, Iran

Correspondence should be addressed to Hassan Zargarzadeh; hzargarzadeh@lamar.edu

Received 12 June 2018; Accepted 13 August 2018; Published 16 October 2018

Academic Editor: Avimanyu Sahoo

Copyright © 2018 Majid Taheri Andani et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Due to their complicated dynamics and underactuated nature, spherical robots require advanced control methods to reveal all their manoeuvrability features. This paper considers the path tracking control problem of a spherical robot equipped with a 2-DOF pendulum. The pendulum has two input torques that allow it to take angles about the robot's transverse and longitudinal axes. Due to mechanical technicalities, it is assumed that these angles are immeasurable. First, a neural network observer is designed to estimate the pendulum angles. Then a modified sliding mode controller is proposed for the robot's tracking control in the presence of uncertainties. Next, the Lyapunov theorem is utilized to analyse the overall stability of the proposed scheme, including the convergence of the observer estimation and the trajectory tracking errors. Finally, simulation results are provided to indicate the effectiveness of the proposed method in comparison with the other available control approaches.

1. Introduction

Spherical robots are a class of rolling robots, which can be used in many applications: surveillance [1], environmental monitoring [2], exploration operations in unknown environments [3], rehabilitation [4], hobbies [5], and so on [6]. Due to the ball-shaped shell, the enclosed components in a spherical robot are protected from collision and environmental states such as dust, humidity, and radiations. Spherical robots have highly manoeuvrable dynamics that enjoy the advantage of never tipping over [7] and the minimum skidding and resistance with the surface [8]. Nonetheless, the underactuated nature of these robots calls for advanced control schemes to succeed with the path tracking problem [9, 10].

Over the last few years, numerous studies have been conducted on modelling, point stabilization [11], and tracking

control of the spherical systems. The motion of the sphere under rolling constraints is studied in [12]. In [13], two computationally path-planning algorithms are presented for the rolling sphere. A first-order model for a spherical robot is derived in [14] from noninclined limitation and conservation of the angular momentum, as well as a path tracking method with minimum energy and time. In [15], a dynamic equation model is obtained for a spherical robot's motion with a mobile mass, which is placed inside the shell and it proposes its path planning on a surface with the presence of obstacles. Euler parameters are utilized in [16] to establish the kinematic equations for a spherical robot and represent the trajectory planning problems. The dynamics model and control of a spherical robot with a driving wheel on the bottom of the sphere are investigated in [8] where the platform is not plain. The dynamics model of a spherical robot with a rolling

mechanism consisting of a 2-DOF is studied in [17], and the rotation matrices are derived to analyse the kinematics of the system. Then the Euler-Lagrange formalism is applied to develop the decoupled dynamic modelling for the motion of the robot in transverse and longitudinal axes. Also, fuzzy and feedback linearization controllers are employed for the trajectory tracking problem. In [18], utilizing the Lagrange method, the dynamics model and the motion control of a pendulum-based spherical robot on an inclined plane were investigated using a PID controller.

In the literature, a variety of control schemes is introduced for the tracking control of spherical robots. By considering the spherical robot's model, a backstepping feedback controller for the kinematic equations is developed in [3]. In [6], a real-time fuzzy guidance approach for the trajectory tracking of a 2-DOF pendulum spherical robot is introduced, with a focusing on simultaneous convergence of attitude and position of the robot. The work in [19] proposes a neural network (NN) PID controller that has the advantages of simple forward and dynamic self-processing structure for an amphibious spherical robot. Modelling errors and uncertainties have motivated many researchers to utilize sliding mode control (SMC), as a kind of nonlinear robust controller [20, 21], for the spherical robot path tracking problem. For example, in [22], a 1-DOF spherical robot is modelled where a SMC online learning method with fuzzy neural network controller is used for velocity control of the robot. In [23], a hierarchical sliding mode is introduced to control the spherical robot in the presence of unwanted disturbances and uncertainties in real time.

The aforementioned control approaches are designed with the assumption that all the states of the system are measurable. Nonetheless, it is sometimes technically costly or mechanically impossible to measure all the states of the physical systems. To address this issue, linear and nonlinear observers are introduced [24–26]. There have been numerous efforts conducted on estimating unknown parameters and immeasurable states in spherical robots. For instance, in [27], disturbance in an omnidirectional spherical robot is estimated, and then it is delivered to an observer-based generalized proportional integral (GPI) for cancellation of the disturbance. In [28], an extended observer-based adaptive hierarchical sliding mode for longitudinal motion of the spherical robot is designed, in which the uncertain rolling resistance can be obtained without using traction sensors.

In this paper, the trajectory tracking problem of a popular spherical robot is studied for a robot that uses a 2-DOF pendulum for driving the motion by moving the sphere's mass centre, which creates gravitational torque [17]. While all the robot's states need to be measurable for the controller design, the mechanical structure makes it a challenge to acquire the pendulum angles using sensors. Even if sensors are installed, the acquired values may not be precise due to the noise and measurement delays. Besides, uncertainties and nonlinear dynamics are intrinsic to spherical robots. To overcome these problems, in this paper, a sliding mode controller is introduced [29] that estimates the immeasurable states via a NN-based nonlinear observer [30].

NNs are widely utilized as adaptive observers to estimate the system nonlinearities and the immeasurable states of the nonlinear systems [31]. Moreover, NNs are extensively used in path tracking by mobile robots [27]. In [32], an adaptive NN with a robust controller is applied to overcome the challenge of environmental border trajectory tracking of under-actuated mobile robots with uncertain dynamics as well as the presence of disturbances. In [33], the authors employ an adaptive path tracking control with an interneural computing approach for a mobile robot to follow a moving target. This controller is a mathematical scheme that employs neural pruning to the functions of rewards and punishment.

Due to the complicated nonlinear dynamics, in this paper, the spherical robot is modelled in a way that the dynamics are decoupled in two separated subsystems about transverse and longitudinal axes [17]. It is assumed that the pendulum's angles around these axes are immeasurable. To measure the unknown states, a NN-based observer is designed that is adaptive against the robot's nonlinearities and unknown parameters. Then an extended SMC is developed that uses the estimated states in the presence of the model structural uncertainties. The SMC provides the torques for the spherical robot around transverse and longitudinal axes to track the desired trajectory. The proposed adaptive observer-based SMC not only rejects the uncertainties but can also estimate immeasurable states while the robot dynamics nonlinearities are assumed to be unknown. The Lyapunov theorem is utilized to guarantee that the overall closed-loop system is semiglobally uniformly ultimately bounded (SGUUB). The performance of the proposed scheme is verified with application to the spherical robot for tracking a desired trajectory on the horizontal plane. To the best knowledge of the authors, this is the first time that an adaptive observer-based SMC that estimates immeasurable states has been introduced to a nonlinear system in the presence of uncertainties and unknown nonlinearities.

The main contributions of this paper are as follows: (1) designing an observer to estimate immeasurable states, which are the angles of the pendulum about transverse and longitudinal angles, (2) designing a radial basis function (RBF) neural network to approximate the uncertainties and nonlinearities of the robot model, and (3) designing a modified SMC that guarantees the SGUUB of the overall closed-loop scheme in the presence of the state estimation errors, uncertainties, and unknown nonlinearities.

The rest of the paper is organized into six sections: the model of the robot is introduced in the second section; in the third section, kinematics and dynamics of the system are introduced; in section four, an observer for the robot is designed; and section five presents an NN and SMC. The simulation results and comparison with relative approaches in the literature are discussed in section six, followed by the conclusions in the last section.

2. Description of the Robot Model

In this paper, a 2-DOF pendulum-driven spherical robot is studied. The pendulum rolls the robot in two directions by shifting the sphere's centre of gravity to different points.

The schematic of this robot is shown in Figure 1. In the sphere, there are two aligned diagonal holes through which a shaft is fixed to the sphere. A 2-DOF pendulum is hanged from the centre point of the shaft. As indicated in Figure 1, two motors are employed to provide torque, so the pendulum takes the required direction. Motor 1 rotates the pendulum about the sphere's shaft that is considered as the transverse shaft. Motor 2 rotates the shaft about the longitudinal axis about the horizon. Rotation of the pendulum about the transverse axis moves the gravity centre towards the longitudinal axis, which causes the sphere to roll about that axis. Rotating the pendulum about the longitudinal axis makes the sphere roll about the longitudinal axis. Therefore, by using two motors, the motion of the robot can be controlled over a flat surface.

For deriving the dynamics model of the robot, it is assumed that two degrees of freedom do not have interaction on each other [17]; therefore, the equations of motion consist of two decoupled subsystems. With this assumption, the motions of a sphere on a flat surface (rolling without slipping) formulation of the kinematic and dynamic equations are derived in [17].

There exist three frames in Figure 2— xyz represents the reference frames that is fixed to the ground, $o_1x_1y_1z_1$ is the moving reference on the spherical centre that can translate with respect to xyz , and $o_2x_2y_2z_2$ is a moving reference to the spherical centre which rotates with the sphere with respect to $o_1x_1y_1z_1$. The relative rotation angle of $o_2x_2y_2z_2$ around o_1x_1 is θ , and the relative rotation angle of $o_2x_2y_2z_2$ around o_1y_1 is ϕ . Figure 3 depicts the sphere's turning motion about its transverse axis. In Figures 2 and 3, θ , ϕ , and ψ are the rotational angles of the robot around the $o-x$, $o-y$, and $o-z$ axes.

3. Kinematics and Dynamics of the Spherical Robots

In this section, the kinematics and dynamics of the presented spherical robot are expressed. Linear and turning motions of the robot are represented in Figures 2 and 3, respectively.

3.1. Kinematics Equations. For deriving the kinematics model, we use two rotations around the transverse and longitudinal axes. $(i, j, k)^T$, $(i_1, j_1, k_1)^T$, and $(i_2, j_2, k_2)^T$ denote unit vectors which are attached to the xyz (C_{fix}), $o_1x_1y_1z_1$ (C_1), and $o_2x_2y_2z_2$ (C_2) coordinates, respectively. The first rotation is written by the following transformation matrix.

$$R_1 = \text{Rot}(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix},$$

$$\begin{bmatrix} i_1 \\ j_1 \\ k_1 \end{bmatrix} = \text{Rot}(x, \theta) \begin{bmatrix} i_2 \\ j_2 \\ k_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} i_2 \\ j_2 \\ k_2 \end{bmatrix}. \quad (1)$$

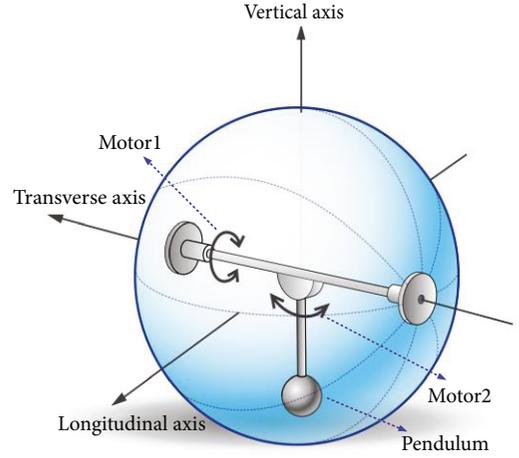


FIGURE 1: Mechanical schematic of a 2-DOF pendulum-based spherical robot.

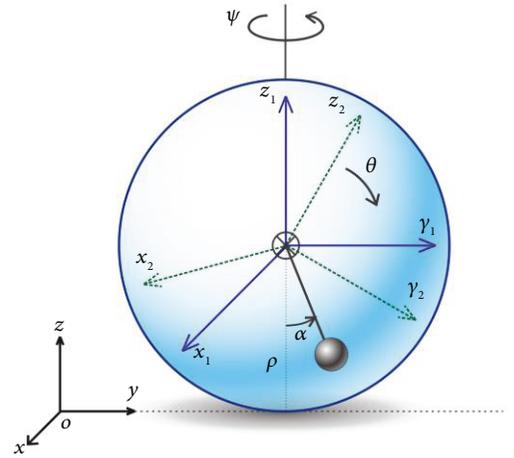


FIGURE 2: Reference frames and kinematic schemes of rotation around the transverse axis resulting in longitudinal motion.

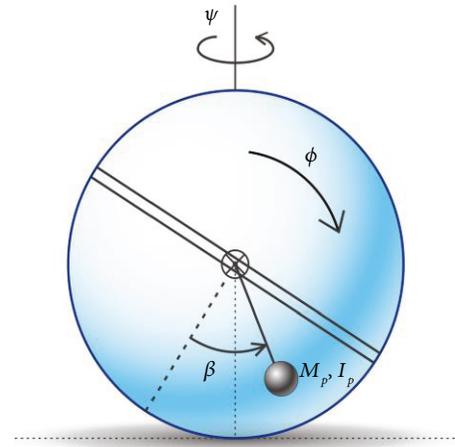


FIGURE 3: Reference frames and kinematic schemes of rotation around the longitudinal axis resulting in transverse motion of the sphere.

The second transformation matrix for the second rotation is represented as follows:

$$R_2 = \text{Rot}(y, \phi) = \begin{bmatrix} \cos \phi & 0 & -\sin \phi \\ 0 & 1 & 0 \\ \sin \phi & 0 & \cos \phi \end{bmatrix},$$

$$\begin{bmatrix} i_1 \\ j_1 \\ k_1 \end{bmatrix} = \text{Rot}(y, \phi) \begin{bmatrix} i_2 \\ j_2 \\ k_2 \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 & -\sin \phi \\ 0 & 1 & 0 \\ \sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} i_2 \\ j_2 \\ k_2 \end{bmatrix}. \quad (2)$$

According to Figure 2, the angular and linear velocities of the sphere are written as follows:

$$\begin{aligned} \omega_s(C_{\text{fix}}) &= -\dot{\theta}i, \\ \omega_s(C_{\text{fix}}) &= -\dot{\phi}j, \\ V_x &= -\rho\dot{\theta}i, \\ V_y &= \rho\dot{\phi}j, \end{aligned} \quad (3)$$

where ρ is the radius of the sphere. The position of the centre of mass of the pendulum can be written as follows:

$$r_x(C_2) = h \sin \alpha j_2 - h \cos \alpha k_2, \quad (4)$$

$$r_y(C_2) = -h \sin \beta i_2 - h \cos \beta k_2, \quad (5)$$

in (4) and (5), α and β denote the angle of the pendulum about transverse and longitudinal axes, respectively, and h is the length of the shaft which connects the pendulum to the center of the sphere. Using transformation matrices, the position of the pendulum in the $o_1x_1y_1z_1$ coordinate is written as follows:

$$\begin{aligned} r_x(C_1) &= \text{Rot}(x, \theta)r_x(C_2) = h \sin(\alpha - \theta)j_1 - h \cos(\alpha - \theta)k_1, \\ r_y(C_1) &= \text{Rot}(y, \phi)r_y(C_2) = -h \sin(\beta - \phi)j_1 - h \cos(\beta - \phi)k_1, \end{aligned} \quad (6)$$

where the position of the centre of mass of the pendulum in fixed coordinate (C_{fix}) is equal to $r_x(C_1)$ and $r_y(C_1)$. The angular velocities of the pendulum in $o_2x_2y_2z_2$ and $o_1x_1y_1z_1$ coordinates are written as

$$\begin{aligned} \omega_x^p(C_2) &= \dot{\alpha}i_2, \\ \omega_x^p(C_1) &= \omega_s(C_1) + \text{Rot}(x, \theta)\omega_x^p(C_2) = (\dot{\alpha} - \dot{\theta})i_1, \\ \omega_y^p(C_2) &= \dot{\beta}j_2, \\ \omega_y^p(C_1) &= \omega_s(C_1) + \text{Rot}(y, \phi)\omega_y^p(C_2) = (\dot{\beta} - \dot{\phi})j_1, \end{aligned} \quad (7)$$

where the angular velocities in the fixed coordinates are equal to $\omega_x^p(C_1)$ and $\omega_y^p(C_2)$. The linear velocities of the

centre of mass of the pendulum in two subsystems are as follows:

$$\begin{aligned} V_x^p &= V_x + \omega_x^p \times r_x \left[-\rho\dot{\theta} + h \cos(\alpha - \theta)(\dot{\alpha} - \dot{\theta}) \right] j \\ &\quad + \left[h \sin(\alpha - \theta)(\dot{\alpha} - \dot{\theta}) \right] k, \\ V_y^p &= V_y + \omega_y^p \times r_y \left[-\rho\dot{\phi} - h \cos(\beta - \phi)(\dot{\beta} - \dot{\phi}) \right] j \\ &\quad + \left[h \sin(\beta - \phi)(\dot{\beta} - \dot{\phi}) \right] k. \end{aligned} \quad (8)$$

3.2. Dynamics Equations. As mentioned in Section 2, for deriving the dynamics equations of the robot, it is considered that the degrees of freedom do not have interaction on the dynamics of each other, so the dynamic equations consist of two decoupled subsystems, in which the following assumptions are made:

- (1) Rotations about transverse and longitudinal axes do not have any interaction with each other
- (2) Around the vertical axis, there is no rotational motion

Using the Lagrangian equations, dynamic equations of the systems can be derived. The Lagrangian formula for a system of particles can be defined by

$$L = T - V, \quad (9)$$

where T is the total kinetic energy of the system and V is the potential energy of the dynamic system. Since the kinetic energy of the system is equal to the kinetic energies of the whole particles of the systems, the kinetic energy of the system is presented as

$$T = T_s + T_p, \quad (10)$$

where T_s and T_p , are equal to the kinetic energy of the sphere and pendulum, respectively.

$$\begin{aligned} T_s &= \frac{1}{2} M_s \|V_s\|^2 + \frac{1}{2} I_s \|\omega_s\|^2, \\ T_p &= \frac{1}{2} M_p \|V_p\|^2 + \frac{1}{2} I_p \|\omega_p\|^2. \end{aligned} \quad (11)$$

In (11), the moments of inertia of the sphere and pendulum are

$$\begin{aligned} I_s &= \frac{2}{3} M_s \rho^2, \\ I_p &= \frac{1}{12} M_p h^2 + M_p \left(\frac{h}{2} \right)^2, \end{aligned} \quad (12)$$

where M_s and M_p are masses; the weight of the sphere and pendulum, ω_s , ω_p , V_s and V_p , represents angular velocities of sphere and pendulum and linear velocities of them, respectively. As the centre of mass of the sphere remains in constant height, the potential energy of the spherical robot depends on

the pendulum's mass and its vertical distance from the centre of the sphere.

$$V = M_p g h. \quad (13)$$

Let us decouple (8) into (11) and decouple the Lagrangian equation into two parts based on rotation in the transverse axis (L_x) and vertical axis (L_y).

$$\begin{aligned} L_x &= \frac{1}{2} M_s \dot{X}^2 + \frac{1}{2} I_s (\omega_x)^2 + \frac{1}{2} M_p \left[\left(\dot{X} + h \cos(\alpha - \theta) (\dot{\alpha} - \dot{\theta}) \right)^2 \right. \\ &\quad \left. + \left(h \sin(\alpha - \theta) (\dot{\alpha} - \dot{\theta}) \right)^2 \right] + \frac{1}{2} I_p (\dot{\alpha} - \dot{\theta})^2 \\ &\quad + M_p g h \cos(\alpha - \theta), \\ L_y &= \frac{1}{2} M_s \dot{Y}^2 + \frac{1}{2} I_s (\omega_y)^2 + \frac{1}{2} M_p \left[\left(\dot{Y} - h \cos(\beta - \phi) (\dot{\beta} - \dot{\phi}) \right)^2 \right. \\ &\quad \left. + \left(h \sin(\beta - \phi) (\dot{\beta} - \dot{\phi}) \right)^2 \right] + \frac{1}{2} I_p (\dot{\beta} - \dot{\phi})^2 \\ &\quad + M_p g h \cos(\beta - \phi). \end{aligned} \quad (14)$$

Euler's equations for systems can be presented as follows:

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L_x}{\partial \dot{q}_i} \right) - \frac{\partial L_x}{\partial q_i} &= Q_i, \\ \frac{d}{dt} \left(\frac{\partial L_y}{\partial \dot{q}_j} \right) - \frac{\partial L_y}{\partial q_j} &= Q_j, \end{aligned} \quad (15)$$

where q_i expresses θ and α ($q_1 = \theta, q_2 = \alpha$) and represents generalized coordinates. Besides, Q_i is considered as input torque in the transverse axis ($Q_{1,2} = \tau_x$). Also, in the above equations, q_j is equal to ϕ and β ($q_3 = \phi, q_4 = \beta$) and they are generalized coordinates. Q_j represents input torque in the longitudinal axis ($Q_{3,4} = \tau_y$).

Let us plug (14) into (15) and rearrange the derived equations in a united state space and write the equation of motion in the matrix form [17].

$$M(q(t)) \ddot{q}(t) + Z(q(t), \dot{q}(t)) = \tau, \quad (16)$$

where $\tau = [\tau_x, \tau_x, \tau_y, \tau_y]^T$,

$$\begin{aligned} M_{11} &= 2M_p \rho h \cos(q_2 - q_1) + M_p h^2 + M_p \rho^2 + I_p + M_s \rho^2 + I_s, \\ M_{12} &= -M_p \rho h \cos(q_2 - q_1) - M_p h^2 - I_p, \\ M_{13} &= 0, \\ M_{14} &= 0, \\ M_{21} &= -M_p \rho h \cos(q_2 - q_1) - M_p h^2 - I_p, \\ M_{22} &= M_p h^2 + I_p, \end{aligned}$$

$$M_{23} = M_{24} = M_{31} = M_{32} = 0,$$

$$M_{33} = 2M_p \rho h \cos(q_3 - q_4) + M_p h^2 + M_p \rho^2 + I_p + M_s \rho^2 + I_s,$$

$$M_{34} = -M_p \rho h \cos(q_4 - q_3) - M_p h^2 - I_p,$$

$$M_{41} = M_{42} = 0,$$

$$M_{43} = -M_p \rho h \cos(q_4 - q_3) - M_p h^2 - I_p,$$

$$M_{44} = M_p h^2 + I_p,$$

$$\begin{aligned} Z_{11} &= -M_p g h \sin(q_2 - q_1) + M_p \rho h \sin(q_2 - q_1) \dot{q}_2^2 \\ &\quad + M_p \rho h \sin(q_2 - q_1) \dot{q}_1^2 - 2M_p \rho h \sin(q_2 - q_1) \dot{q}_1 \dot{q}_2, \end{aligned}$$

$$Z_{21} = M_p g h \sin(q_2 - q_1),$$

$$\begin{aligned} Z_{31} &= -M_p g h \sin(q_4 - q_3) + M_p \rho h \sin(q_4 - q_3) \dot{q}_4^2 \\ &\quad + M_p \rho h \sin(q_4 - q_3) \dot{q}_3^2 - 2M_p \rho h \sin(q_4 - q_3) \dot{q}_3 \dot{q}_4, \end{aligned}$$

$$Z_{41} = M_p g h \sin(q_4 - q_3).$$

(17)

The objective of this paper is to design a path tracking control scheme for this robot, in the following sections: first, an observer will be designed to estimate the immeasurable states; then an adaptive neural network observer-based sliding mode controller will be proposed.

4. Observer Design

To the best knowledge of the authors, in the works that have been done on the spherical robots, it is assumed that α and β are available [17, 18, 23]. However, since all the parts of the spherical robot, installed in the shell, rotate about some axis, measurement of α and β with sensors is costly or impractical due to mechanical inaccessibility. To overcome this problem, in this paper, an observer is designed to estimate the unavailable system states which are the degrees of freedom of the pendulum with respect to the transverse and longitudinal axes.

Considering the dynamics of system (16), two outputs of systems are to be controlled via the two inputs. In this study, X and Y are the outputs and the torques are control inputs of the system. Let us transform the dynamic system into the following form:

$$\ddot{q}(t) = -M^{-1} Z(q(t), \dot{q}(t)) + M^{-1} \tau, \quad (18)$$

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \\ \ddot{q}_4 \end{bmatrix} = -M^{-1} \begin{bmatrix} Z_{11} \\ Z_{21} \\ Z_{31} \\ Z_{41} \end{bmatrix} + M^{-1} \begin{bmatrix} \tau_x \\ \tau_x \\ \tau_y \\ \tau_y \end{bmatrix}. \quad (19)$$

In order to design an observer, (19) can be rewritten in the following state space form:

$$\begin{aligned}
\dot{x}_1 &= x_2, \\
\dot{x}_2 &= f_2(x, u), \\
\dot{x}_3 &= x_4, \\
\dot{x}_4 &= f_4(x, u), \\
\dot{x}_5 &= x_6, \\
\dot{x}_6 &= f_6(x, u), \\
\dot{x}_7 &= x_8, \\
\dot{x}_8 &= f_8(x, u),
\end{aligned} \tag{20}$$

where $x \in \mathbb{R}^{8 \times 1}$ are states of the system in which $x_1 = \theta$, $x_3 = \alpha$, $x_5 = \phi$, $x_7 = \beta$, $u = \tau \in \mathbb{R}^{4 \times 1}$, and $f_i(x, u) \in \mathbb{R}$ with $i \in \{2, 4, 6, 8\}$ and $f_2(x, u) = \dot{q}_1$, $f_4(x, u) = \dot{q}_2$, $f_6(x, u) = \dot{q}_3$, $f_8(x, u) = \dot{q}_4$.

Moreover, (20) can be rewritten in the following form:

$$\begin{aligned}
\dot{x} &= Ax + B(F(x, u)), \\
y &= C^T x,
\end{aligned} \tag{21}$$

where $F(x, u) = [f_2(x, u) f_4(x, u) f_6(x, u) f_8(x, u)]^T$ and

$$\begin{aligned}
A &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \\
B &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T, \\
C^T &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix},
\end{aligned} \tag{22}$$

since θ , $\dot{\theta}$, ϕ , and $\dot{\phi}$ are available and measurable states, the output matrix is considered to be C^T .

$F(x, u)$ can be taken as

$$F(x, u) = F(x, u) - \Omega + \Omega = h(\bar{x}) + \Omega, \tag{23}$$

where $h(\bar{x}) \in \mathbb{R}^{4 \times 1}$ and $\Omega \in \mathbb{R}^{4 \times 1}$ are a neural network-based affinning term that will be defined in (32). So, (21) can be

represented in the following form:

$$\dot{x} = Ax + B(h(\bar{x}) + \Omega), \tag{24}$$

where $\bar{x} = [x, u, \Omega]^T$. Consider an observer represented as follows:

$$\dot{\hat{x}} = A\hat{x} + k_0(y - \hat{y}), \tag{25}$$

where \hat{x} is the estimation of the vectors x and $\hat{y} = C^T \hat{x}$. $k_0 \in \mathbb{R}^{8 \times 4}$ is a vector which is called observer gain, and it is taken to make the characteristic polynomial of $A_0 = A - k_0 C^T$ Hurwitz matrix. Defining the estimation error as $\tilde{x} = x - \hat{x}$, we have

$$\dot{\tilde{x}} = A_0 \tilde{x} + B(h(\bar{x}) + \Omega). \tag{26}$$

Since A_0 is a Hurwitz matrix, there exists $P \in \mathbb{R}^{8 \times 8}$

$$A_0^T P + P A_0 = -Q, \tag{27}$$

and $Q \in \mathbb{R}^{8 \times 8}$ is an arbitrary positive definite matrix.

Since $C^T \tilde{x}$ relies on available states, the strictly positive real (SPR) Lyapunov design approach is used to guarantee the stability of the closed-loop system [34]. Considering (26) as a linear system with $h(\bar{x}) + \Omega$ as its input, we define $H(s) = C^T (sI - A_0)^{-1} B$. So, we have $C^T \tilde{x} = H(s)[h(\bar{x}) + \Omega]$. The transfer function $H(s)$ is known as a stable transfer function. In order to utilize the SPR-Lyapunov design approach, $C^T \tilde{x} = H(s)[h(\bar{x}) + \Omega]$ can be taken as

$$C^T \tilde{x} = H(s)L(s)[L^{-1}(s)h(\bar{x}) + L^{-1}\Omega], \tag{28}$$

with

$$\begin{aligned}
L_i(s) &= s^{m_i} + b_{1i}s^{m_i-1} + \dots + b_{m_i} \quad (m_i \leq 8), \quad i = 1, 2, \dots, 4, \\
L(s) &= \text{diag} [L_1(s), \dots, L_4(s)].
\end{aligned} \tag{29}$$

$L(s)$ is chosen, so that $L^{-1}(s)$ is a proper stable transfer function and $H(s)L(s)$ is a proper SPR transfer function. Now, system (28) can be realized in state space as follows:

$$\dot{\tilde{x}} = A_0 \tilde{x} + B_c L^{-1}(s)[h(\bar{x}) + \Omega], \tag{30}$$

where B_c is chosen such that $PB_c = C$.

Moving on, $h(\bar{x})$ needs to be approximated by utilizing a RBF neural network [35]. RBF is a neural network that is used in the practical applications for its simple structure and its capabilities of nonlinear function approximation [35, 36]. Afterwards, $h(\bar{x})$ is used to develop an adaptive controller with its adaptation law to meet the control objective and guarantee boundedness of all signals in the closed-loop system. $h(\bar{x})$ is approximated ($z \in \Omega_{\text{NN}} \subset \mathbb{R}^n$ is the input

vector) by using RBF NN, as follows:

$$\hat{h}(\bar{x}) = \hat{\psi}\xi(\bar{x}), \quad (31)$$

where $\hat{\psi} = [\hat{\psi}_{11}, \hat{\psi}_{12}, \dots, \hat{\psi}_{1l}]$ is the weight vector of the neural network system function vector with l nodes and $\hat{\psi}_{1i} \in \mathbb{R}^{4 \times l}$, $i = 1, 2, \dots, l$. And $\xi(\bar{x}) = [\xi_1(\bar{x}), \xi_2(\bar{x}), \dots, \xi_l(\bar{x})]^T$ is a basis function with Gaussian function $\xi_i(\bar{x}) = \exp(-\|\bar{x} - \mu_i\|^2/\eta_i^2)$, with centre $\mu_i = [\mu_{i1}, \mu_{i2}, \dots, \mu_{il}]^T$ and width η_i , and $\delta(x, \hat{x}) = h(\bar{x}) - \hat{h}(\hat{x} | \hat{\psi})$ is the neural network approximation error. The neural network controller is designed.

$$\Omega = -\hat{\psi}\xi(\hat{x}). \quad (32)$$

Hence, $\tilde{\psi} = \hat{\psi} - \psi$ where $\hat{\psi}$ is an estimate of ψ . The adaptive law is

$$\dot{\hat{\psi}} = \gamma B_c^T P \tilde{x} L^{-1}(s) \xi^T(\hat{x}). \quad (33)$$

Assumption 1. $\delta_m = L^{-1}(s)(\delta(x, \hat{x}))$ is bounded, i.e., $\|\delta_m\| < \bar{\delta}$.

Consider the following Lyapunov function for systems (30) and (33):

$$V_1 = \frac{1}{2} \tilde{x}^T P \tilde{x} + \frac{1}{2\gamma} \tilde{\psi} \tilde{\psi}^T. \quad (34)$$

By calculating the time derivative of Lyapunov function and using (27), we obtain

$$\dot{V}_1 = \frac{1}{2} \tilde{x}^T (A_0^T P + P A_0) \tilde{x} + \tilde{x}^T P B_c L^{-1}(s) [h(\bar{x}) + \Omega] + \frac{1}{\gamma} \tilde{\psi} \dot{\tilde{\psi}}^T. \quad (35)$$

In (35), $h(\bar{x}) = \psi\xi(\bar{x}) + \delta(x, \hat{x}) = \hat{\psi}\xi(\hat{x}) - \tilde{\psi}\xi(\hat{x}) + \delta(x, \hat{x})$ can be approximated based on (31). Then by replacing neural network controller (32) in (35), we have

$$\begin{aligned} \dot{V}_1 = & \frac{1}{2} \tilde{x}^T (A_0^T P + P A_0) \tilde{x} + \tilde{x}^T P B_c L^{-1}(s) [\hat{\psi}\xi(\hat{x}) - \tilde{\psi}\xi(\hat{x}) \\ & + \delta(x, \hat{x}) - \hat{\psi}\xi(\hat{x})] + \frac{1}{\gamma} \tilde{\psi} \dot{\tilde{\psi}}^T. \end{aligned} \quad (36)$$

By replacing (27) in (36), we have:

$$\begin{aligned} \dot{V}_1 = & -\frac{1}{2} \tilde{x}^T Q \tilde{x} + \tilde{x}^T P B_c L^{-1}(s) [\hat{\psi}\xi(\hat{x}) - \tilde{\psi}\xi(\hat{x}) + \delta(x, \hat{x}) \\ & - \hat{\psi}\xi(\hat{x})] + \frac{1}{\gamma} \tilde{\psi} \dot{\tilde{\psi}}^T. \end{aligned} \quad (37)$$

Now, $\dot{\tilde{\psi}} = \dot{\hat{\psi}} - \dot{\psi}$ and then by using (33) in (37), we have

$$\begin{aligned} \dot{V}_1 = & -\frac{1}{2} \tilde{x}^T Q \tilde{x} + \tilde{x}^T P B_c L^{-1}(s) [\hat{\psi}\xi(\hat{x}) - \tilde{\psi}\xi(\hat{x}) - \hat{\psi}\xi(\hat{x}) \\ & + \delta(x, \hat{x})] + \frac{1}{\gamma} \tilde{\psi} (\gamma \xi(\hat{x}) \tilde{x}^T P B_c L^{-1}(s)) \\ = & -\frac{1}{2} \tilde{x}^T Q \tilde{x} + \tilde{x}^T P B_c L^{-1}(s) \delta(x, \hat{x}). \end{aligned} \quad (38)$$

So, based on Assumption (1), the above equation can be written in the following equation:

$$\dot{V}_1 \leq -\frac{1}{2} \tilde{x}^T Q \tilde{x} + |\tilde{x}^T P B_c| \bar{\delta}. \quad (39)$$

In the above equation, based on Young's inequality, we have

$$|\tilde{x}^T P B_c| \bar{\delta} = |\tilde{x}^T C| \bar{\delta} \leq \frac{1}{2} \|\tilde{x}\|^2 + \frac{1}{2} \bar{\delta}^2, \quad (40)$$

and then we have

$$\dot{V}_1 \leq -\frac{1}{2} \tilde{x}^T Q \tilde{x} + \frac{1}{2} \|\tilde{x}\|^2 + \frac{1}{2} \bar{\delta}^2 \leq -\frac{1}{2} q_0 \|\tilde{x}\|^2 + \frac{1}{2} \bar{\delta}^2, \quad (41)$$

where $q_0 = \lambda_{\min}(Q) - 1$ and $\lambda_{\min}(Q)$ is the minimum characteristic value of Q . So, the observer is provided; in the next section, the SMC is given to control the spherical robot.

5. Sliding Mode Control

The aim of this paper is designing a robust controller, so the spherical robot tracks a desired trajectory, where the torques of the spherical robot about the transverse and longitudinal axes are the system inputs. To that end, the controller for the kinematics part of the robot will be designed. The kinematic models of the robot are written as follows.

$$\begin{aligned} \dot{X} &= -\rho \dot{\theta}, \\ \dot{X} &= -\rho \dot{\theta}, \\ \dot{Y} &= \rho \dot{\phi}, \\ \dot{Y} &= \rho \dot{\phi}, \end{aligned} \quad (42)$$

where X and Y are the position of the spherical robot. By substituting $\dot{\theta}$ and $\dot{\phi}$ into (42) from (19), second derivatives of X, Y are calculated as follows:

$$\ddot{G} = D(x) + J(x)U, \quad (43)$$

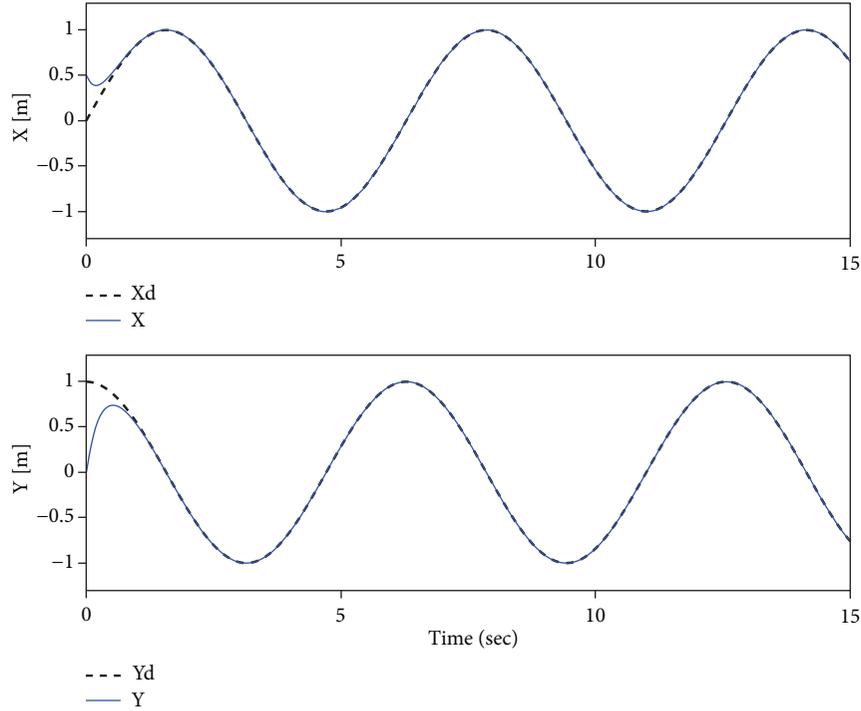


FIGURE 4: Robot trajectory around X and Y directions.

with

$$\begin{aligned} \ddot{G} &= \begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix}, \\ U &= \begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix}, \\ J(x) &= \begin{bmatrix} J_{1,1}(x) & J_{1,2}(x) \\ J_{2,1}(x) & J_{2,2}(x) \end{bmatrix} = \begin{bmatrix} J_{1,1}(x) & 0 \\ 0 & J_{2,2}(x) \end{bmatrix}, \end{aligned} \quad (44)$$

where the first row of $D(x) \in \mathbb{R}^{2 \times 1}$ is derived from multiplying $-\rho$ and the first row of $-M^{-1}G$ and the second row of $D(x)$ is derived from multiplying ρ and the third row of $-M^{-1}G$. And, in $J(x) \in \mathbb{R}^{2 \times 2}$, $J_{1,1}(x)$ and $J_{1,2}(x)$ are derived from multiplying $-\rho$ and the first row of M^{-1} , and $J_{2,1}(x)$ and $J_{2,2}(x)$ are derived from multiplying ρ and the third row of M^{-1} . One should note that from (17), the matrix $M \in \mathbb{R}^{4 \times 4}$ can be represented in a subblock-wise format as follows:

$$\begin{aligned} M &= \begin{bmatrix} M_A & 0 \\ 0 & M_D \end{bmatrix}, \\ M^{-1} &= \begin{bmatrix} M_A^{-1} & 0 \\ 0 & M_D^{-1} \end{bmatrix}, \end{aligned} \quad (45)$$

where $M_A, M_D \in \mathbb{R}^{2 \times 2}$. Therefore, in (43), $J_{1,2}(x)$ and $J_{2,1}(x)$ are zero. Since all the states are not available for

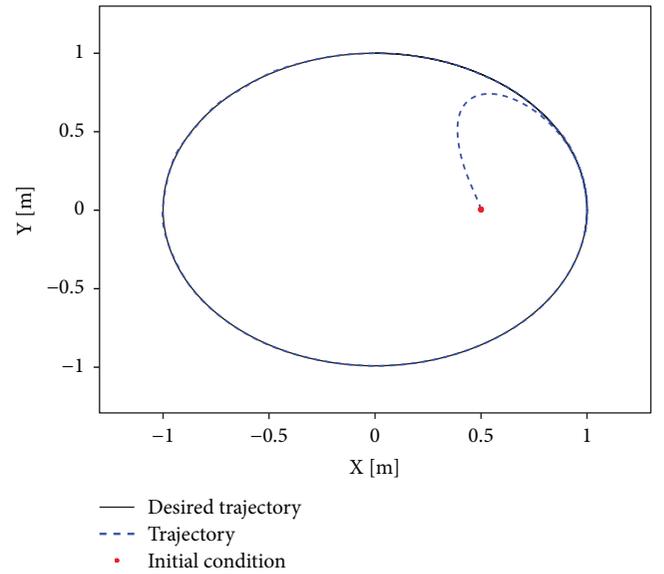


FIGURE 5: Robot trajectory around the X - Y plane.

measurement, the observer state estimation from the previous section will be used, i.e., the estimated states (\hat{x}) are used instead of the immeasurable states (x).

Taking the sliding surface,

$$\begin{bmatrix} S_x \\ S_y \end{bmatrix} = \begin{bmatrix} \dot{X} - \dot{X}_d \\ \dot{Y} - \dot{Y}_d \end{bmatrix} + \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} X - X_d \\ Y - Y_d \end{bmatrix} = \dot{e} + \lambda e, \quad (46)$$

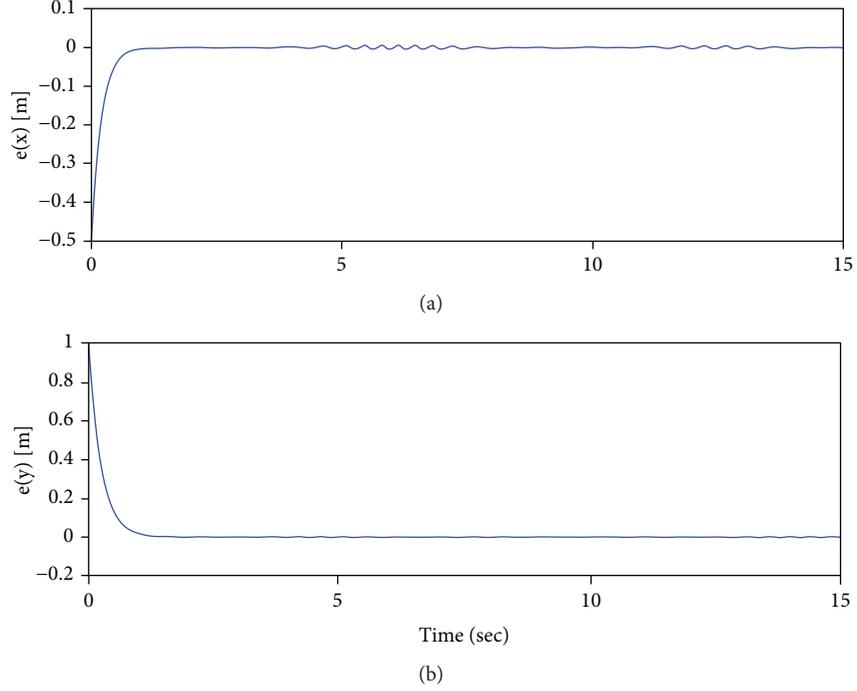


FIGURE 6: Trajectory error around X and Y directions.

where X_d , Y_d , and their derivatives are the desired position of the robot's centre and its linear velocity. λ_1 and λ_2 are strictly positive and constant values. The derivative of the sliding surface is calculated as

$$\dot{S}(t) = \begin{bmatrix} \dot{S}_x \\ \dot{S}_y \end{bmatrix} = \begin{bmatrix} \dot{X} - \dot{X}_d \\ \dot{Y} - \dot{Y}_d \end{bmatrix} + \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} \dot{X} - \dot{X}_d \\ \dot{Y} - \dot{Y}_d \end{bmatrix}. \quad (47)$$

To reach trajectory tracking, $\dot{S}(t)$ should remain zero as long as the robot is moving. So, the following equivalent control law (u_{eq}) is taken:

$$\begin{aligned} u_{eq} &= J^{-1}(\hat{x}) \left[-D(\hat{x}) + \ddot{G}_d - \gamma \zeta^{-1} S \right], \\ \dot{G}_d &= \begin{bmatrix} \dot{X}_d \\ \dot{Y}_d \end{bmatrix}, \\ \ddot{G}_d &= \begin{bmatrix} \ddot{X}_d \\ \ddot{Y}_d \end{bmatrix}, \\ \zeta &= \begin{bmatrix} 1 & \frac{1}{2} \\ -1 & \frac{1}{2} \end{bmatrix}, \end{aligned} \quad (48)$$

where γ is a constant value. One should note that J is invertible due to the invertibility of M . As mentioned above, trajectories should reach a sliding manifold in a limited bound and remain being bound in the presence of

uncertainty and unmodelled dynamics. Hence, the control law (U) is taken as

$$U = u_{eq} + u_s, \quad (49)$$

where u_s is taken for robust stability of the spherical robot in facing uncertainties and unmodelled dynamics.

$$\begin{aligned} u_s &= \text{inv}(J(\hat{x})) \left(-k \zeta^{-1} \text{sgn}(S) - \lambda (\dot{G} - \dot{G}_d) \right), \\ S &= \begin{bmatrix} S_x \\ S_y \end{bmatrix}, \\ k &= \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix}, \end{aligned} \quad (50)$$

where K_1, K_2 are positive design parameters. Also, we take the Lyapunov stability theorem for this system as follows:

$$V_2 = \frac{1}{2} S^T \zeta S. \quad (51)$$

By considering the sliding surface in (46), the control signal is

$$\begin{aligned} U &= J^{-1}(\hat{x}) \left[-D(\hat{x}) + \ddot{G}_d - \gamma \zeta^{-1} S \right. \\ &\quad \left. - k \zeta^{-1} \text{sgn}(S) - \lambda (\dot{G} - \dot{G}_d) \right]. \end{aligned} \quad (52)$$

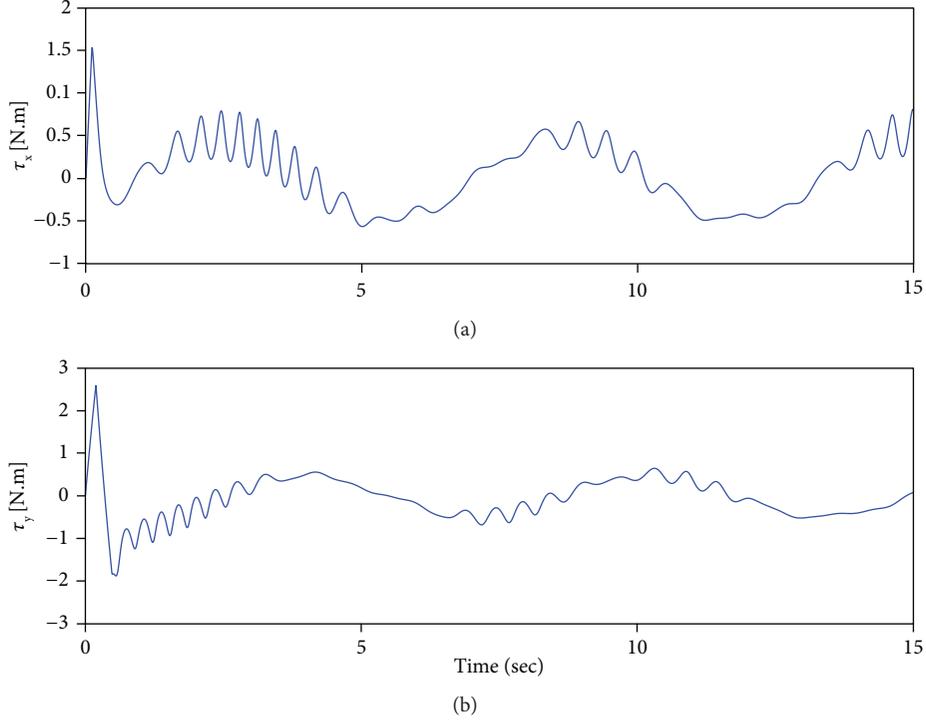


FIGURE 7: The input torques applied to the robot.

The derivative of Lyapunov function with respect to time is

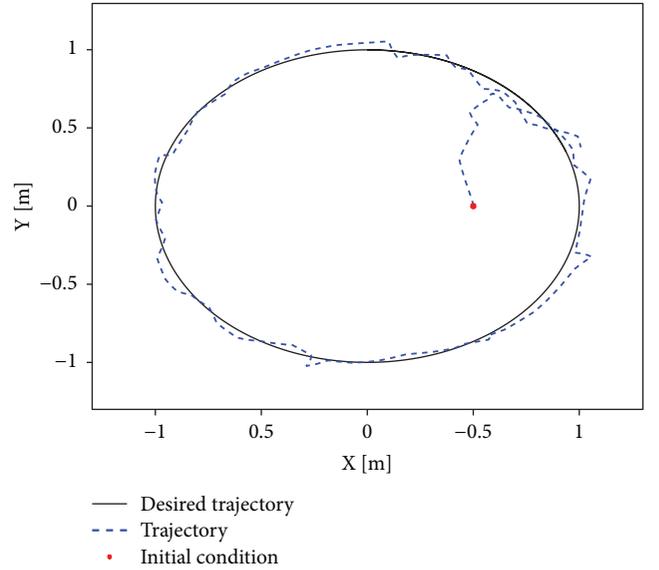
$$\begin{aligned}\dot{V}_2(t) &= S^T \zeta \left(\ddot{G} - \ddot{G}_d + \lambda (\dot{G} - \dot{G}_d) \right) \\ &= S^T \zeta \left(D(\hat{x}) + J(\hat{x})U - \ddot{G}_d + \lambda (\dot{G} - \dot{G}_d) \right).\end{aligned}\quad (53)$$

By replacing U in (53), we have

$$\begin{aligned}\dot{V}_2(t) &= S^T \zeta \left(D(\hat{x}) + J(\hat{x})J^{-1}(\hat{x}) \left[-D(\hat{x}) + \ddot{G}_d - \gamma \zeta^{-1} S \right. \right. \\ &\quad \left. \left. - k \zeta^{-1} \text{sgn}(S) - \lambda (\dot{G} - \dot{G}_d) \right] - \ddot{G}_d + \lambda (\dot{G} - \dot{G}_d) \right) \\ &= S^T (-\gamma S - k \text{sgn}(S)) \leq -\gamma \|S\|^2 - k \|S\|^2 \leq -(\gamma + k) \|S\|^2.\end{aligned}\quad (54)$$

Now, the design of the observer-based SMC is complete and the steps of the design can be summarized as the following steps: first, designing a NN-based observer whose stability is studied in (34) and developing a SMC for tracking control of the spherical robot using the Lyapunov function in (51) with the stability analysis in (53). Now, we are ready to prove the overall stability of the proposed controller in the following theorem.

$$V = V_1 + V_2. \quad (55)$$

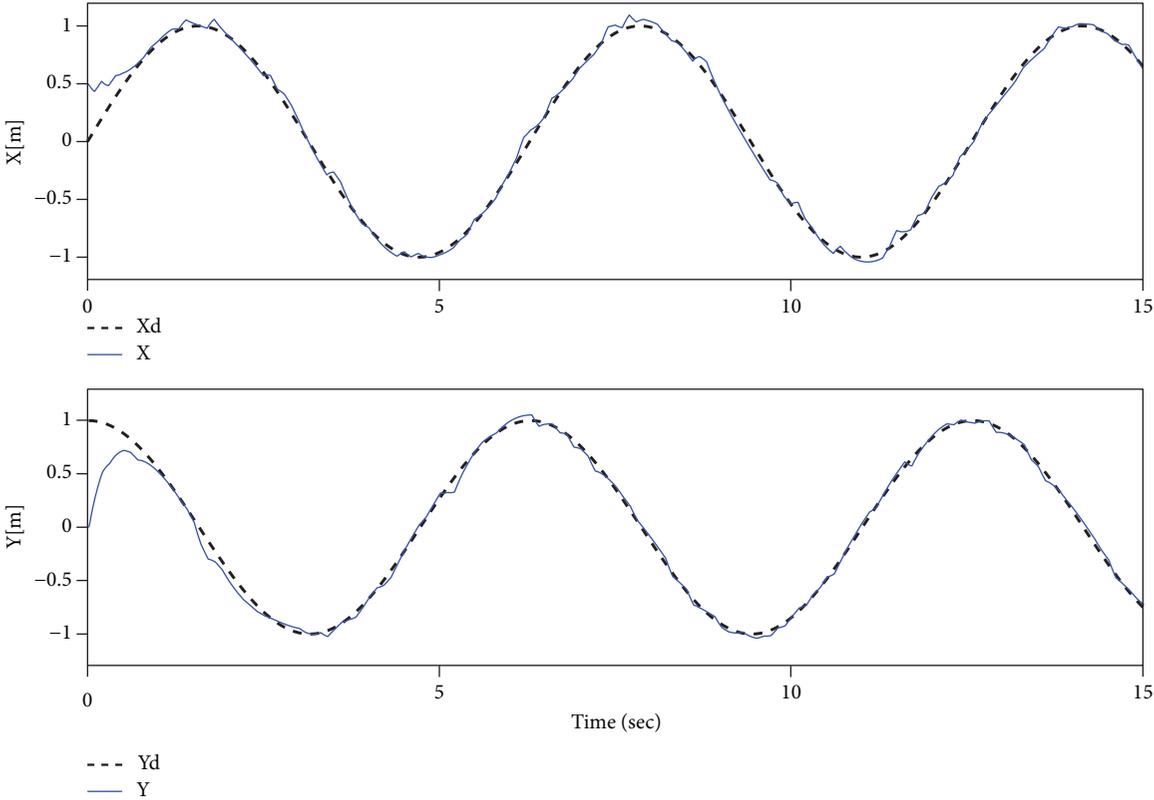
FIGURE 8: Robot trajectory in the $X - Y$ plane.

By replacing (41) and (54) in the time derivative of V , it takes the following form:

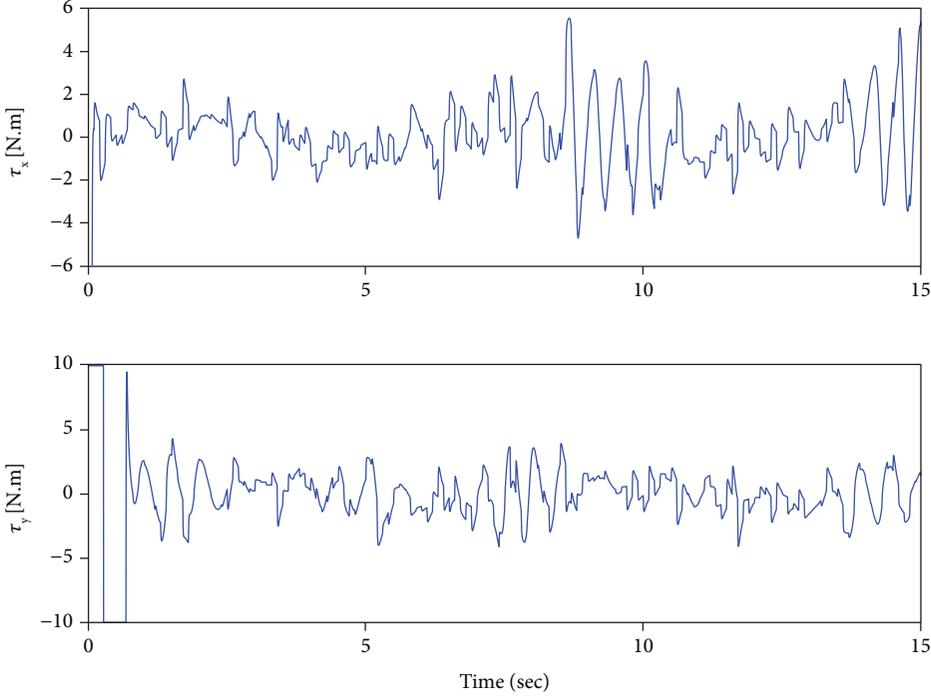
$$\dot{V} = \dot{V}_1 + \dot{V}_2 \leq -\frac{1}{2} q_0 \|\tilde{x}\|^2 + \frac{1}{2} \bar{\delta}^2 - (\gamma - k) \|S\|^2. \quad (56)$$

By defining $\zeta = \gamma - k$ and by choosing $k \geq \gamma$, we have

$$\dot{V} \leq -\frac{1}{2} q_0 \|\tilde{x}\|^2 + \frac{1}{2} \bar{\delta}^2 - \zeta \|S\|^2 \quad (57)$$



(a)



(b)

FIGURE 9: (a) Robot trajectory tracking performance around the X and Y directions in the presence of measurement noise. (b) Input torques applied to the robot in the presence of the measurement noise.

that indicates $\dot{V} \leq 0$ as long as

$$\|\tilde{x}\| \geq \sqrt{\frac{q_0}{\delta^2}}, \quad (58)$$

or

$$\|S\| \geq \sqrt{\frac{\delta^2}{2\zeta}}. \quad (59)$$

Since $S = \dot{e} + \text{diag}(\lambda_1, \lambda_2)e$, (59) implies that [37]

$$\|e\| \geq \frac{1}{\lambda_{\min}} \sqrt{\frac{\delta^2}{2\zeta}}, \quad (60)$$

where $\lambda_{\min} = \min(\lambda_1, \lambda_2)$. Now, from (57), (59), and (60) one can conclude that $(\tilde{x}, \tilde{\psi}, e)$ are semi globally uniformly ultimately bounded with the bounds of $\sqrt{q_0/\delta^2}$ and $1/\lambda_{\min} \sqrt{\delta^2/2\zeta}$ for \tilde{x} and e , respectively.

The next section is dedicated to the numerical evaluation of the proposed controller and comparing it with other controllers in the literature.

6. Simulation Results

6.1. Path Tracking. This section provides the simulation results through MATLAB, to verify the efficiency of the designed trajectory tracking controller observer-based SMC represented. The parameters of the spherical robot are $M_s = 3$ (kg), $M_p = 2$ (kg), $\rho = 0.2$ (m), $l = 0.075$ (m), and $g = 9.81$ (m/s²). Simulation is carried out with initial conditions as follows:

$$G_d = \begin{bmatrix} X_d \\ Y_d \end{bmatrix} = \begin{bmatrix} \sin(t) \\ \cos(t) \end{bmatrix}, \quad (61)$$

$$G(t_0) = \begin{bmatrix} X(0) \\ Y(0) \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}.$$

In the path tracking scheme, it is assumed that the centre of the spherical robot shell should move about the reference trajectory. The aim of the control is that the output tracks the reference path. The design parameters of the presented controller are chosen by trial and error as $\lambda_1 = 50$, $\lambda_2 = 50$, $\gamma_1 = 40$, $\gamma_2 = 4$, $k_1 = 0.1$, $k_2 = 0.1$. The observer gain is given by

$$K_0 = \begin{bmatrix} 140 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 140 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 140 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 140 & 2 \end{bmatrix}^T. \quad (62)$$

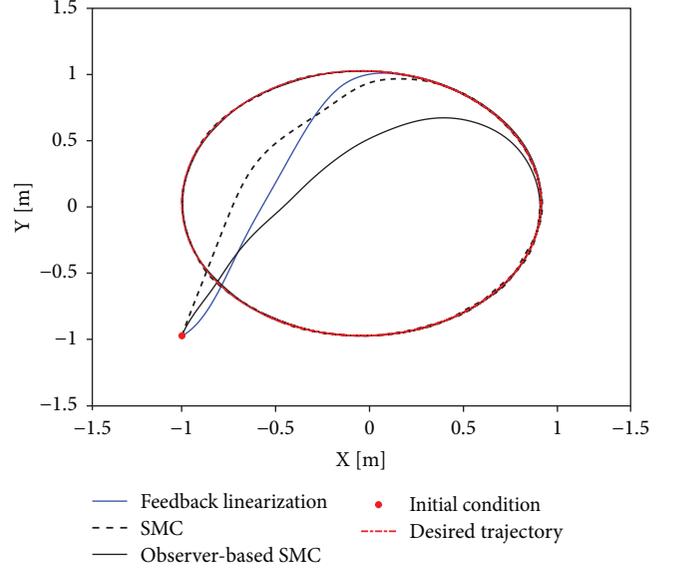


FIGURE 10: Comparing the observer-based SMC performance with the SMC and feedback linearization controllers in the $X - Y$ plane.

The positive definite matrix $Q = 10I_{8 \times 8}$ is chosen where $I_{8 \times 8}$ is an identity matrix. Besides, A_0 is computed as follows:

$$A_0 = \begin{bmatrix} -140 & -139 & 0 & 0 & -140 & -140 & 0 & 0 \\ -2 & -2 & 0 & 0 & -2 & -2 & 0 & 0 \\ -140 & -140 & 0 & 1 & -140 & -140 & 0 & 0 \\ -2 & -2 & 0 & 0 & -2 & -2 & 0 & 0 \\ -140 & -140 & 0 & 0 & -140 & -139 & 0 & 0 \\ -2 & -2 & 0 & 0 & -2 & -2 & 0 & 0 \\ -140 & -140 & 0 & 0 & -140 & -140 & 0 & 1 \\ -2 & -2 & 0 & 0 & -2 & -2 & 0 & 0 \end{bmatrix}. \quad (63)$$

And the filter is chosen as

$$L_i^{-1} = s + 2. \quad (64)$$

The parameters of the proposed NN are selected by trial and error. In the simulations, for making the NN, 18 nodes are used. Moreover, to avoid the chattering caused by the SMC, the discontinuous sign function in the control law is substituted with the continuous tanh function [37]. The equations of the system are sampled with a sampling period of 1.0 ms for the trajectories. The simulation results are obtained as demonstrated in Figures 4–7.

Figures 4–7 show that X and Y are able to track the desired path successfully where X and Y are illustrated against the time and in an $X - Y$ plane, respectively. It can be observed that the tracking error is bounded and the

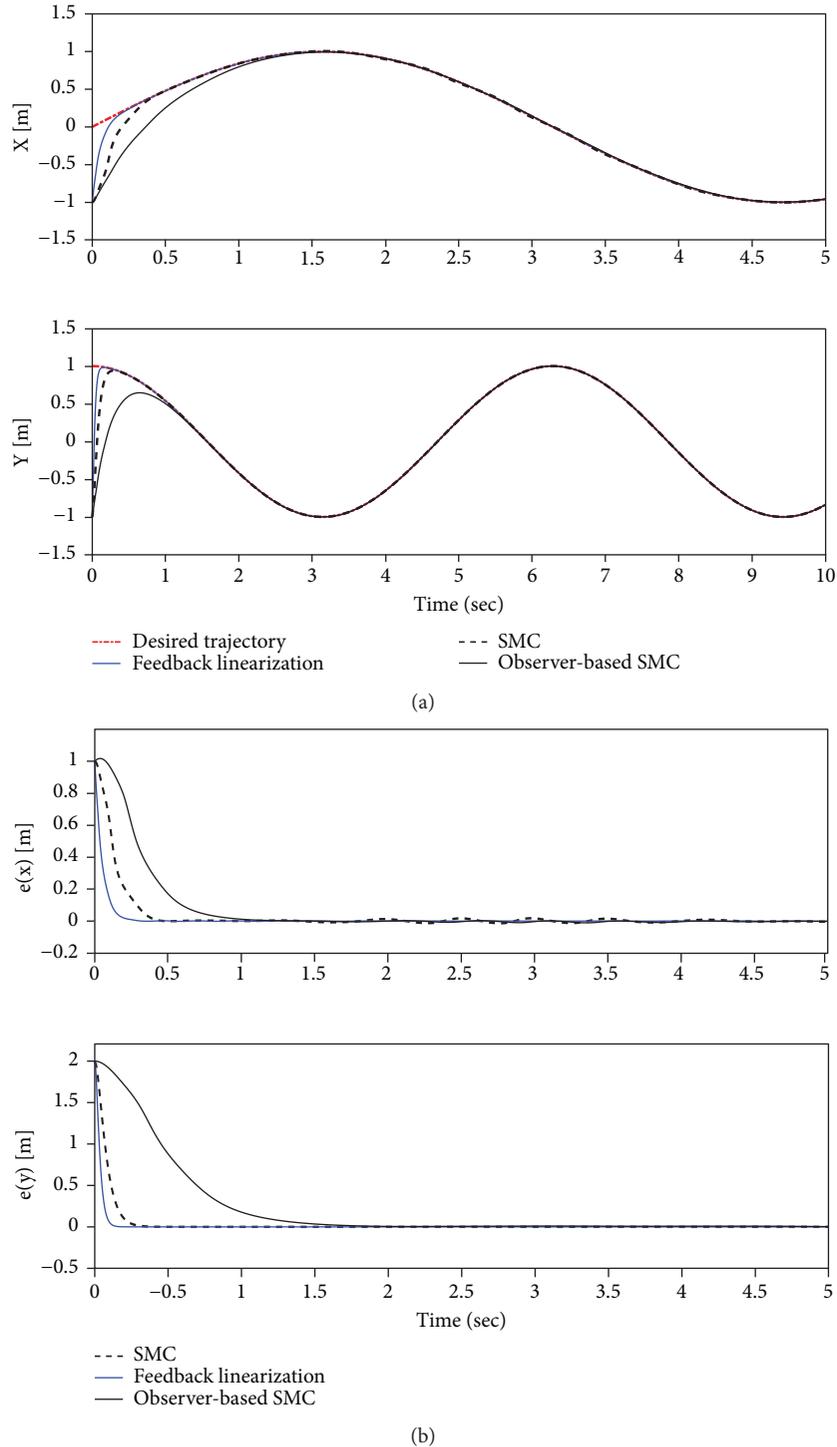


FIGURE 11: Comparing the observer-based SMC performance with the SMC and feedback linearization controllers: (a) trajectory tracking performance around X and Y directions and (b) trajectory tracking error convergence.

designed observer-based SMC can effectively reach the desired trajectory of the plant. In Figure 7, the input torques, τ_x and τ_y , about transverse and longitudinal axes for the desired trajectory are shown.

Simulation results are presented in Figures 4–7. Figures 4 and 5 show that X and Y are able to track the desired path

successfully. Moreover, in Figure 6, it is crystal clear that the system errors are converging to a small neighbourhood around the origin. In order to examine the robustness of the proposed controller, another simulation was carried out in the presence of measurement noise. A white noise with a power of 0.01 and sampling time of 0.1 s that passed through a filter

with the transfer function of $1/(2s + 1)$ was applied to the position sensor output signals.

While disturbed by noise, Figure 8 shows that the robot is successfully able to track the desired trajectory with a bounded tracking error. Figure 9(a) shows the position of the robot with respect to time. Therefore, from the above results, it can be concluded that the spherical robot with the observer-based SMC-designed controller can robustly track a desired circular path in the presence of the measurement noise. The applied input torques in the presence of noise are illustrated in Figure 9(b). To prevent impractical large input torques, they are bounded by applying saturation functions with ranges of 10 N·m.

6.2. The Controller Design Comparison with SMC and Feedback Linearization. In this subsection, the performance of the proposed approach is compared with the two other controllers in the literature. To this end, two of the most recent and well-known control methods, namely, the SMC [29] and feedback linearization [17], are selected. Figures 10 and 11 depict the performance of the above controllers against the desired trajectory in (61) with $G(0) = [-1 \quad -1]^T$. As it is expected, under relatively similar conditions, the feedback linearization has the convergence time due to the assumption that the dynamics nonlinearities are known. However, while SMC and observer-based SMC are slower, they have more robustness to the uncertainties and oscillations. In addition, despite the SMC and feedback linearization that enjoys the measurability of all the states, the observer-based SMC must estimate the immeasurable states (α, β) . This fact has made the proposed controller the slowest one in terms of convergence. Overall, in comparison with other methods, the proposed controller demonstrates the same accuracy with a slightly slower response in the tracking performance.

6.3. Conclusions. In this work, we introduced an extended sliding mode control scheme for trajectory tracking control of a spherical robot that uses a pendulum with 2-DOFs to provide the rolling motion around the transverse and longitudinal axes. The angles of the pendulum with respect to these axes are assumed to be immeasurable due to mechanical technicalities. To estimate the immeasurable states, an adaptive neural network-based observer was proposed. The observer adaptively estimates the nonlinearities and rejects the uncertainties of the robot while estimating the sphere states using an output feedback. The SMC uses the estimated states instead of the actual ones to stabilize the robot's path tracking error. Finally, using the Lyapunov theorem, we showed that the proposed scheme is able to SGUUB stabilize the overall closed-loop system. Simulation results verify the proposed control scheme performance in comparison with other available approaches in the literature.

The following improvements are considered in future work: (1) manufacturing the spherical robot and practical implementation of the proposed observer-based SMC on the spherical robot and (2) modelling the spherical robot by taking into account the friction between the shell and the ground and without decoupling the two actuation mechanisms.

Data Availability

All data generated or analysed during this study are included in this article. Any further data are available from the corresponding author on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M. Seeman, M. Broxvall, A. Saffiotti, and P. Wide, "An autonomous spherical robot for security tasks," in *2006 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety*, pp. 51–55, Alexandria, VA, USA, October 2006.
- [2] M. Li, S. Guo, H. Hirata, and H. Ishihara, "Design and performance evaluation of an amphibious spherical robot," *Robotics and Autonomous Systems*, vol. 64, pp. 21–34, 2015.
- [3] Z. Qiang, L. Zengbo, and C. Yao, "A back-stepping based trajectory tracking controller for a non-chained nonholonomic spherical robot," *Chinese Journal of Aeronautics*, vol. 21, no. 5, pp. 472–480, 2008.
- [4] F. Michaud and S. Caron, "An autonomous toy-rolling robot," in *Proceedings of the 2000 PERCARN-IRIS International Symposium on Robotics (ISR)*, pp. 114–119, Montreal Quebec, 2000.
- [5] F. Michaud and S. Caron, "Roball, the rolling robot," *Autonomous Robots*, vol. 12, no. 2, pp. 211–222, 2002.
- [6] Y. Cai, Q. Zhan, and X. Xi, "Path tracking control of a spherical mobile robot," *Mechanism and Machine Theory*, vol. 51, pp. 58–73, 2012.
- [7] J. Alves and J. Dias, "Design and control of a spherical mobile robot," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 217, no. 6, pp. 457–467, 2003.
- [8] A. Halme, T. Schonberg, and Y. Wang, "Motion control of a spherical mobile robot," in *Proceedings of 4th IEEE International Workshop on Advanced Motion Control - AMC '96 - MIE*, pp. 259–264, Mie, Japan, March 1996.
- [9] I. Duleba, W. Khefffi, and I. Karcz-Duleba, "Layer, Lie algebraic method of motion planning for nonholonomic systems," *Journal of the Franklin Institute*, vol. 349, no. 1, pp. 201–215, 2012.
- [10] A. D. Schutte, "Permissible control of general constrained mechanical systems," *Journal of the Franklin Institute*, vol. 347, no. 1, pp. 208–227, 2010.
- [11] E.-H. Zheng, J.-J. Xiong, and J.-L. Luo, "Second order sliding mode control for a quadrotor UAV," *ISA Transactions*, vol. 53, no. 4, pp. 1350–1356, 2014.
- [12] Z. Li and J. Canny, "Motion of two rigid bodies with rolling constraint," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 1, pp. 62–72, 1990.
- [13] R. Mukherjee, M. A. Minor, and J. T. Pukrushpan, "Motion planning for a spherical mobile robot: revisiting the classical ball-plate problem," *Journal of Dynamic Systems, Measurement, and Control*, vol. 124, no. 4, pp. 502–511, 2002.
- [14] S. Bhattacharya and S. K. Agrawal, "Spherical rolling robot: a design and motion planning studies," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 835–839, 2000.

- [15] A. Bicchi, A. Balluchi, D. Prattichizzo, and A. Gorelli, "Introducing the "SPHERICLE": an experimental testbed for research and teaching in nonholonomy," in *Proceedings of International Conference on Robotics and Automation*, pp. 2620–2625, Albuquerque, NM, USA, April 1997.
- [16] V. A. Joshi and R. N. Banavar, "Motion analysis of a spherical mobile robot," *Robotica*, vol. 27, no. 3, pp. 343–353, 2009.
- [17] E. Kayacan, Z. Y. Bayraktaroglu, and W. Saeys, "Modeling and control of a spherical rolling robot: a decoupled dynamics approach," *Robotica*, vol. 30, no. 4, pp. 671–680, 2012.
- [18] M. Roozegar and M. J. Mahjoob, "Modelling and control of a non-holonomic pendulum-driven spherical robot moving on an inclined plane: simulation and experimental results," *IET Control Theory & Applications*, vol. 11, no. 4, pp. 541–549, 2017.
- [19] Z. Wang, S. Guo, L. Shi, S. Pan, and Y. He, "The application of PID control in motion control of the spherical amphibious robot," in *2014 IEEE International Conference on Mechatronics and Automation*, pp. 1901–1906, Tianjin, China, August 2014.
- [20] M. Chadli, S. Aouaouda, H.-R. Karimi, and P. Shi, "Robust fault tolerant tracking controller design for a VTOL aircraft," *Journal of the Franklin Institute*, vol. 350, no. 9, pp. 2627–2645, 2013.
- [21] X. Yang, X. Li, and J. Cao, "Robust finite-time stability of singular nonlinear systems with interval time-varying delay," *Journal of the Franklin Institute*, vol. 355, no. 3, pp. 1241–1258, 2018.
- [22] E. Kayacan, E. Kayacan, H. Ramon, and W. Saeys, "Adaptive neuro-fuzzy control of a spherical rolling robot using sliding-mode-control-theory-based online learning algorithm," *IEEE transactions on cybernetics*, vol. 43, no. 1, pp. 170–179, 2013.
- [23] B. Liu, M. Yue, and R. Liu, "Motion control of an underactuated spherical robot: a hierarchical sliding-mode approach with disturbance estimation," in *2012 IEEE International Conference on Mechatronics and Automation*, pp. 1804–1809, Chengdu, China, August 2012.
- [24] X. Liu, D. W. Ho, J. Cao, and W. Xu, "Discontinuous observers design for finite-time consensus of multiagent systems with external disturbances," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 11, pp. 2826–2830, 2017.
- [25] T. Youssef, M. Chadli, H. Karimi, and R. Wang, "Actuator and sensor faults estimation based on proportional integral observer for TS fuzzy model," *Journal of the Franklin Institute*, vol. 354, no. 6, pp. 2524–2542, 2017.
- [26] H. K. Khalil, *Nonlinear Systems*, Prentice-Hall, Upper Saddle River, NJ, USA, 1996.
- [27] H. Sira-Ramírez, C. López-Urbe, and M. Velasco-Villa, "Linear observer-based active disturbance rejection control of the omnidirectional mobile robot," *Asian Journal of Control*, vol. 15, no. 1, pp. 51–63, 2013.
- [28] M. Yue, B. Liu, C. An, and X. Sun, "Extended state observer-based adaptive hierarchical sliding mode control for longitudinal movement of a spherical robot," *Nonlinear Dynamics*, vol. 78, no. 2, pp. 1233–1244, 2014.
- [29] M. R. Azizi and J. Keighobadi, "Robust sliding mode trajectory tracking controller for a nonholonomic spherical mobile robot," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 4541–4546, 2014.
- [30] B. Chen, H. Zhang, and C. Lin, "Observer-based adaptive neural network control for nonlinear systems in nonstrict-feedback form," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 1, pp. 89–98, 2016.
- [31] D. Nodland, H. Zargarzadeh, and S. Jagannathan, "Neural network-based optimal adaptive output feedback control of a helicopter UAV," *IEEE Transactions on Neural Networks and Learning systems*, vol. 24, no. 7, pp. 1061–1073, 2013.
- [32] T. Sun, H. Pei, Y. Pan, and C. Zhang, "Robust adaptive neural network control for environmental boundary tracking by mobile robots," *International Journal of Robust and Nonlinear Control*, vol. 23, no. 2, pp. 123–136, 2013.
- [33] F.-S. Tsai, S.-Y. Hsu, and M.-H. Shih, "Adaptive tracking control for robots with an interneural computing scheme," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 832–844, 2018.
- [34] M. M. Arefi and M. R. Jahed-Motlagh, "Observer-based adaptive neural control of uncertain MIMO nonlinear systems with unknown control direction," *International Journal of Adaptive Control and Signal Processing*, vol. 27, no. 9, pp. 741–754, 2013.
- [35] M. M. Arefi, Z. Ramezani, and M. R. Jahed-Motlagh, "Observer-based adaptive robust control of nonlinear nonaffine systems with unknown gain sign," *Nonlinear Dynamics*, vol. 78, no. 3, pp. 2185–2194, 2014.
- [36] Z. Ramezani, M. M. Arefi, H. Zargarzadeh, and M. R. Jahed-Motlagh, "Neuro-adaptive backstepping control of SISO non-affine systems with unknown gain sign," *ISA Transactions*, vol. 65, pp. 199–209, 2016.
- [37] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood cliffs, NJ, USA, 1991.

