J. Appl. Comput. Mech., xx(x) (2026) 1-20
DOI: 10.22055/jacm.2026.49111.5743

ISSN: 2383-4536
jacm.scu.ac.ir

Shahid Chamran University of Ahvaz

Journal of Applied and Computational Mechanics

Research Paper

Barrier Function-Based Adaptive Sliding Mode Control for Disturbed Two-Wheeled Robot Obstacle Avoidance

Nasser Firouzi[1], Haider Khazal[2], Younis Khalid Khdir[3], Przemysław Podulka[4], Abdullah A. Faqihi[5], Haitham M. Hadidi[6]

[1] Institute of Structural Mechanics, Bauhaus-University Weimar, Weimar 99423, Germany, Email: Nasser.firouzi@uni-weimar.de

[2] Mechanical Engineering Department, College of Engineering, University of Basrah, Basrah, Iraq, Email: haider.mehbes@uobasrah.edu.iq

[3] Mechanical and Energy Engineering Department / Technical Engineering College / Erbil Polytechnic University, Erbil 44001, Iraq, Email: younis.khdir@epu.edu.iq

[4] Department of Manufacturing Processes and Production Engineering, Rzeszow University of Technology, Rzeszow, Poland, Email: przpod@prz.edu.pl

[5] Department of Industrial Engineering, College of Engineering and Computer Sciences, Jazan University, Jazan, Saudi Arabia, Email: afaqihi@jazanu.edu.sa

[6] Department of Mechanical Engineering, College of Engineering and Computer Science, Jazan University, Jazan 45142, Saudi Arabia, Email: haitham.hadidi1@gmail.com

Received January 31 2026; Revised August 07 2026; Accepted for publication September 23 2026.
✉ Corresponding author: N. Firouzi (Nasser.firouzi@uni-weimar.de)

Abstract. In this article, a novel optimal adaptive sliding mode control integrated with a control barrier function is designed for the safe navigation of a two-wheeled mobile robot subjected to unknown external disturbances. The nominal reference tracking is achieved using the sliding mode approach in which the switching gains capable of disturbance and uncertainty rejection are updated by developed adaptive laws, eliminating the need for prior knowledge of the disturbance upper bound. To further improve the closed-loop performance, additional control parameters are optimized using a particle swarm optimization algorithm, which reduces steady-state tracking errors and mitigates high-value overshoots. Integrating the control barrier function in the proposed control framework guarantees obstacle avoidance by minimizing deviations between the safe control signals and those of the nominal control inputs. Simulation results illustrate the effectiveness of the proposed method in achieving accurate trajectory tracking, ensuring collision-free navigation, and maintaining control efficiency under uncertain conditions. Moreover, comparative simulation studies with a PD controller confirm that the proposed control structure has significantly better performance in disturbance rejection, minimizing control efforts, and path following.

Keywords: Adaptive sliding mode control, Obstacle avoidance, Mobile robot, Control barrier function, Particle swarm optimization.

1. Introduction

Mobile robots are autonomous car-like vehicles capable of locomotion in various environments, distinguishing them from stationary robotic systems. These robots have evolved remarkably since their conceptualization during World War II, integrating advancements in computer science, electronics, and artificial intelligence (AI). Initially developed for military applications such as teleoperations, mobile robots now encompass wheeled, legged, aerial, and underwater variants, enabling independent movement and task execution [1, 2].

Control algorithms are crucial for enabling mobile robots to achieve precise navigation, trajectory tracking, and obstacle avoidance. These approaches address the nonlinear and nonholonomic nature of robot kinematics and dynamics, often incorporating feedback mechanisms to handle external disturbances and uncertainties. Classical linear control methods include Proportional-Integral-Derivative (PID) [3] and Linear Quadratic Regulator (LQR) [4] which are simple and effective for basic trajectory tracking, but they struggle with nonlinearities in system dynamics. Model predictive Control (MPC) [5] evolves the states of the system dynamics with prediction and control horizons by solving constrained optimization problems; yet it is not suitable for nonlinear systems. Nonlinear controllers such as Back-stepping [6] which decomposes the dynamics into subsystems, ensuring stability via Lyapunov criteria, and Sliding Mode Control (SMC) [7], using sliding manifolds as stable surfaces, on the other hand, are of interest for mobile robots due to their nonlinear coupling nature present between states and control signals.

SMC is the most famous and simplest control approach among nonlinear control methods for nonlinear systems, and it is widely applied to mobile robots [8, 9]. Amarasiri et al. [10] applied an SMC scheme that is robust to uncertainties. A super-twisting SMC with a finite-time disturbance observer for a wheeled mobile robot was designed by Li and Zhai [11]. The closed-loop stability was ensured via the Lyapunov theorem and all tracking errors converged to zero asymptotically. A Fuzzy logic base is combined with SMC in [12] to tune the controller parameters adaptively and to improve the response time. Comparative simulation results verify the proposed control effectiveness in faster trajectory tracking. Still, experimental tests have not been performed.

Optimization algorithms combined with linear and nonlinear controllers have also been investigated for wheeled mobile robots. A Genetic Algorithm (GA) based PID control was designed by Ha et al. [13] for a two-wheel robot for path following a circular trajectory. The GA is responsible for adjusting the PID gains offline. Despite the optimal parameter tuning, the control signals may saturate because they are not used in the performance index defined for the GA, but the trajectory tracking results are of acceptable accuracy. A fuzzy logic controller was developed by Pérez-Juárez et al. [14] for a mobile robot to guarantee safe navigation even in the presence of unknown disturbances. The real-time simulations in a virtual platform validate the robustness of the suggested control scheme in the reduction of settling time and steady-state errors to a great extent. Nasir et al. [15] proposed a Particle Swarm Optimization (PSO) integrated with a fractional-order SMC for finite-time trajectory following of a differential drive robot. Since PSO is capable of handling a large parameter space, it is much faster than GA and it is used for offline adjustment of control parameters. Simulation studies indicate the superiority of the proposed controller over a single SMC approach. A novel neural network-based backstepping control was developed by Beharu Arega et al. [16] for a three-wheeled mobile robot. A nonlinear PID control is utilized to train the network and the stability of the backstepping controller is proved via Lyapunov's criteria. Several comparative simulations with a GA based PID, a GA-based back-stepping, and a neural network-based nonlinear PID were conducted to verify the effectiveness of the proposed controller.

Despite the control algorithms implemented on mobile robots for trajectory tracking objectives, path planning and safe navigation algorithms have also been utilized for obstacle avoidance [17-19]. Zheng [20] implemented an Artificial Potential Field (APF) based obstacle avoidance for a mobile robot to track an obstacle-free path. A dual-loop control structure has been designed for the position and attitude of the robot. The experimental tests of the proposed closed-loop algorithm indicate that the computational costs have been decreased compared to other control methods. A dynamic window algorithm is proposed by Cao and Mohamad Nor [21] for dynamic obstacle avoidance. The obstacle velocity is evaluated such that the robot changes its speed and angular velocity to avoid colliding. Compared with traditional algorithms, the recommended approach has shortened the avoidance time by 37% while keeping the safe distance. A Control Barrier Function-based Prescribed Time Control (CBF-PTC) is introduced by Nasab et al. [22] for trajectory and path following objectives. The use of CBF generates safe control signals via a second order optimization so that the vehicle can avoid collision with static obstacles. Simulation results indicate the superiority of the controller over a nonlinear MPC.

Despite the effectiveness of SMC and CBF individually, their integration into a unified framework for mobile robots remains underexplored. Recent works have begun combining SMC with CBF to achieve both robustness and safety. For instance, authors in Xu et al. [23] proposed an SMC-CBF approach for uncertain systems, but their method requires a priori knowledge of disturbance bounds, limiting practical applicability. Li et al. [24] designed a CBF-based sliding mode controller for obstacle avoidance; however, the switching gains are fixed, leading to conservatism. Chen et al. [25] introduced adaptive gains for SMC-CBF, but their adaptation law depends on the CBF constraint, causing coupling issues.

To further focus on the contributions within the SMC-CBF literature, it is important to observe the design choices that characterize existing methodologies. The recent work on SMC-CBF integration can be extensively categorized into two major structural units: coupled single-layer architectures in which SMC schemes are embedded directly within the CBF quadratic programming (CBF-QP), and two-layer structures in which SMC serves as a base controller for implementation and the CBF plays a crucial role as a safety filter, minimally altering the control input when constraints are to be violated by the system. In the design of single-layer control, Chinelato et al. [26] proposed a Sliding Mode Control Barrier Function (SMCBF) where the SMC reaching law is integrated into the CBF formulation, achieving accurate tracking and safe navigation via a simple optimization problem. Whereas this combination of SMC with CBF guarantees simultaneous tracking and safety, it introduces feasibility opposition between SMC reaching law and the CBF constraint, particularly when the sliding mode command and the safety condition are contradictory. Nguyen and Vu [27] extended this idea by incorporating hierarchical SMC with CBF for a ball-balancing robot, still the same feasible vulnerability exists in the coupled QP formulation. For the two-layer decoupled unit, Syntakas et al. [28] developed an architecture for marine vessels, employing SMC as the nominal controller and a higher-order CBF (HOCBF) with fast half-space projection as the safety filter, replacing the computationally expensive QP with a projection-based method. Sawarkar et al. [29] designed a similar decoupled structure using SMC integrated with a Collision Cone CBF (C3BF) filter for ground robots, with hardware validation on differential-drive platforms. However, in these decoupled architectures, the CBF safety filter may override the SMC signal during safety-critical events, pushing the system off the sliding surface and degrading tracking performance, and the transient behavior when the filter activates and deactivates can introduce high-frequency chattering. Additionally, the SMC parameters in both coupled and decoupled architectures are commonly constant and manually tuned, and no existing decoupled architecture in the literature incorporates adaptive switching gains that compensate for unknown disturbance bounds and uncertainties without coupling the adaptive policies into the CBF constraint.

A distinction must be examined carefully between two different uses of the term "barrier function" in the adaptive SMC literature. A substantial body of work employs logarithmic barrier functions solely to constrain the adaptive growth of SMC, rather than to enforce safety constraints via CBFs. Obeidet al. [30] pioneered this methodology by using BFs to bound the adaptive parameter of first-order SMC, avoiding gain overestimation and reducing chattering phenomenon. The authors extended this closed-loop framework to higher-order SMC [31] for systems with input saturation [32]. Subsequent works have applied similar barrier-function-based adaptation to adaptive fuzzy super-twisting SMC [33], state-constrained SMC [34], and fractional-order SMC [35]. More importantly, in all of these articles, the barrier function constrains only the adaptation rule of the SMC gains and is totally independent of any CBF constraint. Conversely, when CBFs are incorporated with SMC for safety purposes, the adaptiveness is applied to the CBF parameters rather than to the SMC gains. For instance, Taylor and Ames [36] proposed adaptive CBFs for time-varying control limits within the CBF-CLF-QP programming, but neither utilized adaptive switching gains. This reveals a significant research gap: no existing work designs an adaptive SMC whose gain adaptation law operates independently of the CBF safety constraint while both components are implemented in the closed-loop system. The approach proposed in this article fills this gap by employing a decoupled two-layer architecture in which the inner-loop adaptive SMC adjusts its switching gains via Lyapunov-based adaptation laws that are entirely independent of the outer-loop CBF optimization, hereby accomplishing the goal of feasibility and modularity.

Optimizing controller parameters within the SMC-CBF framework represents another underexplored work. PSO and other metaheuristic algorithms have been extensively employed for tuning SMC parameters alone. Nasir et al. [15] used PSO for offline adjustment of fractional-order SMC gains for mobile robots, and similar PSO-based SMC tuning has been reported for chattering reduction [37] and online gain adaptation [38]. On the CBF side, on the other hand, parameter adaptation has been approached through learning-based methods, gradient-based optimization, and parametric CBF formulations [39, 40]. However, a careful examination of the literature divulges that no work has applied PSO or any other metaheuristic optimization algorithm to jointly optimize both SMC and CBF parameters, altogether. This gap is highly crucial since the SMC gains and the CBF class-κ parameters are dependent in practice: increasing the switching gains of SMC enhances disturbance rejection but necessitate larger CBF safety margins to accommodate more aggressive control signals. On the other hand, tightening the CBF constraint for safety requires smaller SMC gains to maintain QP feasibility but may not be able to compensate for external disturbances. Existing fixed-gain SMC-CBF designs [23, 24] ignore this interdependency, leading to either conservative safety margins or degraded tracking performance. This current article addresses fully this gap by formulating a PSO-based optimization problem that jointly adjusts the SMC parameters within a unified performance index that penalizes control efforts and safety margin violation.

According to the aforementioned articles investigated in the literature, in the present research, we propose an optimal adaptive sliding mode control combined with a control barrier function to address the problem of safe navigation of a wheeled mobile robot. We develop an optimal adaptive nonlinear SMC serving as the nominal controller for trajectory tracking purposes. The switching gains showcasing the robustness of the SMC against external disturbances are adjusted via adaptive laws, compensating for the lack of knowledge about the unknown upper bound of disturbances. Other SMC parameters, on the other hand, have been tuned via a PSO algorithm to alleviate tracking errors, sudden overshoots, and to prevent control saturation. Whilst many control algorithms have been designed for the aim of prescribed trajectory tracking [41-43], the CBF controller designed in this paper ensures obstacle avoidance via minimizing the deviation from the nominal SMC. The combination of optimal adaptive SMC with the CBF guarantees both trajectory tracking and obstacle avoidance. Despite traditional MPC and SMC algorithms in which the precise system dynamics along with the upper bound of external disturbances must be prioritized [44, 45], the proposed adaptive closed-loop control algorithm not only does not require the upper bound of external disturbances appeared in the system dynamics as lateral and longitudinal push and slip but it utilizes the kinematic model of the two-wheeled robot instead of the dynamic equations which greatly reduces the computational costs. Our proposed method decouples the adaptive SMC design from the CBF optimization via a two-layer structure: the inner loop adaptively rejects disturbances without requiring their upper bounds, while the outer loop ensures safety via minimal intervention. Furthermore, PSO-based tuning of CBF parameters has never been reported in the SMC-CBF literature. Existing optimization studies focus on SMC gains alone [46], ignoring the safety-performance trade-off introduced by CBF. Our work fills this gap by jointly optimizing SMC and CBF parameters. Finally, most SMC-CBF methods rely on full dynamic models [47], increasing computational burden. Our kinematic-based approach with adaptive disturbance compensation offers a computationally lighter alternative suitable for real-time embedded implementation.

Thus, the main contributions of this paper are classified as follows:

- Deriving the non-holonomic two-wheeled mobile robot kinematics as the base model in the global reference frame. Existing safety-critical controllers for mobile robots rely on complex dynamic models, increasing computational load. Our approach uses the kinematic model with adaptive disturbance compensation, reducing computational cost while maintaining robustness, a key advantage for real-time embedded systems.
- Designing an adaptive SMC for predefined path following of the mobile robot and to guarantee the stability of the system kinematics in the presence of bounded external disturbances. This extends applicability to environments with unknown or time-varying disturbances
- While PSO has been frequently implemented for SMC tuning, this paper presents the first integration of PSO with CBF constraints to jointly optimize tracking accuracy, control effort, and safety margin. This resolves the inherent trade-off between performance and safety that plagues fixed-gain CBF-SMC designs.
- Introducing the CBF to model the geometry of the obstacle and solving the optimization control problem to obtain the safe control inputs for the horizontal and angular velocity of the robot.
- Unlike coupled designs where the adaptive law interacts with CBF constraints, leading to potential infeasibility, we propose a decoupled two-layer architecture for both disturbance rejection and safe motion. This guarantees feasibility and simplifies implementations.

The rest of this paper is classified as follows: Part 2 formulates the mobile robot kinematics, Section 3 denotes the design of the nominal SMC approach, Section 4 briefly explains the CBF algorithm, Part 5 demonstrates the simulation results of the proposed closed-loop approach, and finally, the paper is finished in Section 6.

2. Mobile Robot Motion Equations

The kinematic equations of a mobile robot are well known and discussed comprehensively in many articles [48, 49]. We have used the simple model demonstrated in Fig. 1. Two reference frames are needed to described the motion of the vehicle, one attached to the center of mass of the robot for local position, and the other one called the Earth frame describing the global positioning of the robot, respectively. The relative vector $[x, y]^T$ represents the position of the robot in the body frame while the global position vector $[X, Y]^T$ denotes the robot's position in the Earth frame, respectively. Two wheels called *left* and *right* rotate with the angular velocities $\dot{\phi}_l$ and $\dot{\phi}_r$, related to the voltages of the motors. The attitude of the robot can be described by the angle θ expressed in the body frame and the control input vector can be defined as $u = [v, \omega]^T$. Moreover, the position vector $r(t)$ describes the location of the mobile robot in either global or body frame, in terms of the unit vectors [50].

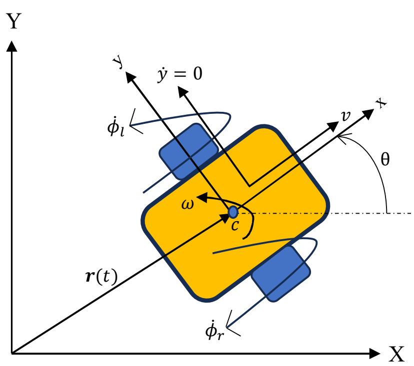

Fig. 1. Two-wheeled mobile robot schematic along with the coordinate frames and physical structure of the robot.

The equations of motion of the non-holonomic vehicle with the assumption of no lateral or sideway slip in the global frame can be described as follows:

$$\begin{gathered}\dot{X} = \upsilon\cos(\theta) \\ \dot{Y} = \upsilon\sin(\theta) \\ \dot{\theta} = \omega\end{gathered} \tag{1}$$

The rotation matrix R expressing the relation between two frames can be defined as:

$$\begin{bmatrix} x \\ y \end{bmatrix} = R\begin{bmatrix} X \\ Y \end{bmatrix} \tag{2}$$

$$R = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \tag{3}$$

The nonholonomic constraint could be expressed mathematically as $\dot{x}\sin(\theta) + \dot{y}\cos(\theta) = 0$ which must be satisfied during the motion time intervals. This nonlinear constraint states that the robot cannot slip in the direction perpendicular to its movement. The following virtual desired model can be defined for trajectory tracking and path following objectives:

$$\begin{gathered}\dot{X}_d = \upsilon_d\cos(\theta_d) \\ \dot{Y}_d = \upsilon_d\sin(\theta_d) \\ \dot{\theta}_d = \omega_d\end{gathered} \tag{4}$$

Such that the tracking errors can be defined in the global coordinate system:

$$\begin{gathered}e_X = X_d - X \\ e_Y = Y_d - Y \\ e_\theta = \theta_d - \theta\end{gathered} \tag{5}$$

Because the control signals are defined in the body frame, it is not possible to directly control the position and the orientation of the robot in the global Earth frame. Thus, it is common to convert the tracking errors from global reference frame to the body frame via the rotation matrix [51]:

$$\begin{bmatrix} e_x \\ e_y \end{bmatrix} = R\begin{bmatrix} e_X \\ e_Y \end{bmatrix} \tag{6}$$

$$e_x = (X_d - X)\cos(\theta) + (Y_d - Y)\sin(\theta) \tag{7}$$

$$e_y = -(X_d - X)\sin(\theta) + (Y_d - Y)\cos(\theta) \tag{8}$$

In relation to Eq. (4), the reference signals for linear and angular velocities and the heading angle can be easily obtained as follows:

$$\begin{gathered}\upsilon_d = \sqrt{\dot{X}_d^2 + \dot{Y}_d^2} \\ \omega_d = \frac{\dot{X}_d\ddot{Y}_d - \ddot{X}_d\dot{Y}_d}{\upsilon_d^2} \\ \theta_d = \tan^{-1}\left(\frac{\dot{Y}_d}{\dot{X}_d}\right)\end{gathered} \tag{9}$$

Then, taking time derivative from the above equations and substitution of the kinematic relations (1) to (4) yields:

$$\dot{e}_x = \upsilon_d\cos(e_\theta) - \upsilon + \omega e_y \tag{10}$$

$$\dot{e}_y = \upsilon_d\sin(e_\theta) - \omega e_x \tag{11}$$

$$\dot{e}_\theta = \omega_d - \omega \tag{12}$$

Assuming the presence of disturbances in the longitudinal and lateral position, and the attitude of the robot due to side slip and push, Eqs. (10) to (12) can be rewritten to consider external disturbances:

$$\dot{e}_x = \upsilon_d\cos(e_\theta) - \upsilon + \omega e_y + d_x \tag{13}$$

$$\dot{e}_y = v_d \sin(e_\theta) - \omega e_x + d_y \tag{14}$$

$$\dot{e}_\theta = \omega_d - \omega + d_\theta \tag{15}$$

Thus, the compact form of Eqs. (10) to (12) can be expressed in the state space form for the control purposes as follows:

$$\dot{e} = f(e) + g(e)u + d \tag{16}$$

where,

$$\begin{gathered} f(e) = \begin{bmatrix} v_d \cos(e_\theta) & v_d \sin(e_\theta) & \omega_d \end{bmatrix}^T \\ g(e)u = \begin{bmatrix} -1 & e_y \\ 0 & e_x \\ 0 & -1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \\ d = \begin{bmatrix} d_x & d_y & d_\theta \end{bmatrix}^T \end{gathered} \tag{17}$$

Equations (13) to (17) represent the complete error dynamics of the two-wheeled mobile robot and will be used to design the closed-loop control structure in the upcoming sections.

3. Optimal Adaptive Sliding Mode Control

3.1. Preliminaries and math definitions

A. Fundamental stability notions

We begin by recalling the core stability concepts for continuous-time dynamical systems evolving on an infinite horizon. Throughout this section we consider the non-autonomous system:

$$\begin{gathered} \dot{y}(t) = g(y,t), \\ g : \mathbb{R}^n \times [0,\infty) \to \mathbb{R}^n, \end{gathered} \tag{18}$$

where g is piecewise continuous and locally Lipschitz continuous in y. The rest state of interest is y = 0.

Statement 1. [Asymptotic and exponential stability] [52]. The equilibrium points y = 0 of system (18) is said to be:

1. **Stable** if there exists a class-$\mathcal{K}$ function φ such that:

$$\| y(t) \| \le \varphi \| y(0) \| \tag{19}$$

2. **Asymptotically stable** if it is stable and, additionally,

$$\| y(t) \| \to 0, \quad when \quad t \to \infty \tag{20}$$

3. **Exponentially stable** if there exist positive constants μ_1, μ_2 such that, for all sufficiently small $\| y(0) \|$,

$$\| y(t) \| \le \mu_1 \| y(0) \| e^{-\mu_2 t} \tag{21}$$

The notions of finite-time (FT) stability extend the classical framework by requiring the trajectory to reach the origin in a bounded amount of time.

Statement 2. [Finite-time stability] [53]. The equilibrium of system (18) is called FT stable if it is indeed stable and some settling function $\tau = \tau(y(0))$ exists so that:

$$\| y(t) \| = 0 \quad \forall t \ge \tau(y(0)) \tag{22}$$

A key distinction among these stability notions is how the terminal time relates to the initial conditions and design parameters. In FT stability, the settling time $\tau(y(0))$ is coupled to the initial state.

B. Lyapunov-Based Stability Propositions

Establishing FT stability in practice relies on constructing a Lyapunov function that satisfies an appropriate differential inequality. The propositions below collect representative results of this kind.

Proposition 1 [52, 53]. Consider system (18). Suppose there exists a continuously differentiable function $W : R^n \to [0,\infty)$ together with constants $\kappa > 0$ and $0 < \sigma < 1$ such that the time-derivative of W along the trajectories of Eq. (18) satisfies:

$$\dot{W}(y) \le -\kappa W^\sigma(y) \tag{23}$$

Then the closed-loop system is globally FT stable and the settling time obeys:

$$\tau := \frac{1}{\kappa(1-\sigma)} W^{1-\sigma}(y(0)) \tag{24}$$

Proposition 2 (Semi-global FT stability with a linear correction term) [52,54]. Suppose there exists a continuously differentiable function $W : R^n \to [0,\infty)$ and constants $\kappa_1, \kappa_2 > 0$, and $0 < \sigma < 1$ such that:

$$\dot{W}(y) \le -\kappa_1 W^{\sigma}(y) + \kappa_2 W(y) \tag{25}$$

Then the closed-loop system is semi-globally finite-time stable and the settling time satisfies:

$$\tau := \frac{1}{\kappa_2(1-\sigma)} \ln\left(1 - \frac{\kappa_1}{\kappa_2} W^{1-\sigma}(y(0))\right)^{-1} \tag{26}$$

3.2. Control scheme

In this section, the SMC scheme is proposed for trajectory tracking objectives. According to Eqs. (13) to (15), since we have two control inputs, two sliding manifolds, one for longitudinal motion and one for lateral heading coupling, are designed as follows:

$$s_1 = e_x \tag{27}$$

$$s_2 = e_y + c e_\theta \tag{28}$$

where $c > 0$.

Assumption 1. The external disturbances acting on the mobile robot are assumed to be upper bounded such that $\| d_x \| \le D_x$, $\| d_y \| \le D_y$, and $\| d_\theta \| \le D_\theta$, where D_x, D_y, and D_θ are unknown positive constants, respectively.

Lemma 1. [55] For any time-varying vector $\chi(t) \in \mathbb{R}^n$, if $0 < \nu \le 1$, then we have:

$$\left(\sum_{j=1}^{n} |\chi_j| \right)^{\nu} \le \sum_{j=1}^{n} |\chi_j|^{\nu} \tag{29}$$

Lemma 2. [56] Suppose that there exists a positive definite Lyapunov Function $\upsilon(x)$, for a nonlinear system $\dot{x}(t) = f(x(t),u(t),t)$, satisfying the following nonlinear differential inequality:

$$\dot{\upsilon}(x) \le -\zeta_1 \upsilon(x) - \zeta_2 \upsilon(x)^{\mu} \tag{30}$$

for positive parameters ζ_1, ζ_2, and $0 < \mu < 1$, respectively. Then the state variables $x(t)$ converge to zero in a finite time T bounded by:

$$T \le \frac{1}{\zeta_1(1-\mu)} \ln\left(\frac{\zeta_2 + \zeta_1 \upsilon(x)^{1-\mu}}{\zeta_2} \right) \tag{31}$$

The equivalent control laws may be derived by setting the derivatives of the sliding manifolds to zero:

$$\dot{s}_1 = \dot{e}_x = \upsilon_d \cos(e_\theta) - \upsilon + \omega e_y + d_x \tag{32}$$

$$\dot{s}_2 = \dot{e}_y + c\dot{e}_\theta = \upsilon_d \sin(e_\theta) - \omega e_x + d_y + c\omega_d - c\omega + c d_\theta \tag{33}$$

Since the external disturbances are unknown, the nominal equivalent controllers can be designed as follows:

$$\omega_{eq} = (c + e_x)^{-1} \left(\upsilon_d \sin(e_\theta) + c\omega_d \right) \tag{34}$$

$$\upsilon_{eq} = \upsilon_d \cos(e_\theta) + \omega e_y \tag{35}$$

To guarantee the robustness of the controller against external disturbances and uncertainties, the switching laws are designed as follows:

$$\omega_{sw} = \lambda_2 s_2 + \hat{\rho}_2 \operatorname{sign}(s_2) \tag{36}$$

$$\upsilon_{sw} = \lambda_1 s_1 + \hat{\rho}_1 \operatorname{sign}(s_1) \tag{37}$$

where $\lambda_1 > 0$, $\lambda_2 > 0$, and $\hat{\rho}_1$, $\hat{\rho}_2$ are the estimates of the switching gains ρ_1 and ρ_2 defined as:

$$\rho_1 = D_x + \epsilon_1 \tag{38}$$

$$\rho_2 = D_y + c D_\theta + \epsilon_2 \tag{39}$$

where $\epsilon_i > 0, (i = 1,2)$. The adaptive laws for tuning switching parameters may be calculated as:

$$\dot{\hat{\rho}}_1 = \gamma_1 |s_1| \tag{40}$$

$$\dot{\hat{\rho}}_2 = \gamma_2 |s_2| \tag{41}$$

where $\gamma_i, (i = 1,2)$ are positive constants. The adaptive control inputs will then be derived:

$$\omega = (c + e_x)^{-1} \left(\upsilon_d \sin(e_\theta) + c\omega_d + \lambda_2 s_2 + \hat{\rho}_2 \operatorname{sign}(s_2) \right) \tag{42}$$

$$v = v_d \cos(e_\theta) + \omega e_y + \lambda_1 s_1 + \hat{\rho}_1 \operatorname{sign}(s_1) \tag{43}$$

Theorem 1. If the control policies (42) and (43) are applied to error dynamics (13) to (15), the tracking errors will converge to the sliding surfaces in a finite time.

Proof. Consider the following candidate Lyapunov function:

$$V = \frac{1}{2}s_1^2 + \frac{1}{2}s_2^2 + \frac{1}{2\gamma_1}\tilde{\rho}_1^2 + \frac{1}{2\gamma_2}\tilde{\rho}_2^2 \tag{44}$$

in which $\tilde{\rho}_i = \rho_i - \hat{\rho}_i,(i=1,2)$ are the adaptive tracking errors, s_1,s_2 are the sliding variables, and $\gamma_i,(i=1,2)$ are positive constants to be computed for faster convergence of adaptive gains to their true values. Then first time-derivative of the Lyapunov function along trajectories (13) to (15) with the substitution of control laws (40) to (43) results in:

$$\begin{aligned}
\dot{V} &= s_1\dot{s}_1 + s_2\dot{s}_2 - \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\hat{\rho}}_1 - \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\hat{\rho}}_2 \\
&= s_1\left(v_d\cos(e_\theta) - v_d\cos(e_\theta) - \omega e_y - \lambda_1 s_1 - \hat{\rho}_1\operatorname{sign}(s_1) + \omega e_y + d_x\right) \\
&\quad + s_2\left(v_d\sin(e_\theta) + c\omega_d + d_y + cd_\theta - v_d\sin(e_\theta) - c\omega_d - \lambda_2 s_2 - \hat{\rho}_2\operatorname{sign}(s_2)\right) \\
&\quad + \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\rho}_1 - \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\hat{\rho}}_1 + \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\rho}_2 - \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\hat{\rho}}_2
\end{aligned}$$

Since ρ_1,ρ_2 are constants, then their time derivative will be zero, i.e., $\dot{\rho}_1 = \dot{\rho}_2 = 0$. Then, we have:

$$\dot{V} = -\lambda_1 s_1^2 - \hat{\rho}_1|s_1| + s_1 d_x - \lambda_2 s_2^2 - \hat{\rho}_2|s_2| + s_2 d_y + cs_2 d_\theta - \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\hat{\rho}}_1 - \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\hat{\rho}}_2$$

By using the relation $\hat{\rho}_i = \rho_i - \tilde{\rho}_i$, and that $s_i\operatorname{sign}(s_i) = |s_i|$, one has:

$$= -\lambda_1 s_1^2 - \rho_1|s_1| + \tilde{\rho}_1|s_1| + s_1 d_x - \lambda_2 s_2^2 - \rho_2|s_2| + \tilde{\rho}_2|s_2| + s_2 d_y + cs_2 d_\theta - \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\hat{\rho}}_1 - \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\hat{\rho}}_2$$

Using **Assumption 1** simultaneously with Eqs. (38), (39) for the positive gains ρ_1,ρ_2, we have:

$$\begin{aligned}
&\le -\lambda_1 s_1{}^2 - \lambda_2 s_2{}^2 - D_x|s_1| - \varepsilon_1|s_1| + \tilde{\rho}_1|s_1| + D_x|s_1| + (D_y + cD_\theta)|s_2| - (D_y + cD_\theta)|s_2| \\
&\quad - \varepsilon_2|s_2| + \tilde{\rho}_2|s_2| - \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\hat{\rho}}_1 - \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\hat{\rho}}_2 \\
&\le -\lambda_1 s_1{}^2 - \lambda_2 s_2{}^2 - \varepsilon_1|s_1| - \varepsilon_2|s_2| + \tilde{\rho}_1|s_1| + \tilde{\rho}_2|s_2| - \frac{1}{\gamma_1}\tilde{\rho}_1\dot{\hat{\rho}}_1 - \frac{1}{\gamma_2}\tilde{\rho}_2\dot{\hat{\rho}}_2 \\
&\le -\lambda_1 s_1{}^2 - \lambda_2 s_2{}^2 - \varepsilon_1|s_1| - \varepsilon_2|s_2| + \tilde{\rho}_1\left(|s_1| - \frac{1}{\gamma_1}\dot{\hat{\rho}}_1\right) + \tilde{\rho}_2\left(|s_2| - \frac{1}{\gamma_2}\dot{\hat{\rho}}_2\right)
\end{aligned}$$

Employing adaptive laws (40), (41), one will get:

$$\begin{aligned}
&\le -\lambda_1 s_1{}^2 - \lambda_2 s_2{}^2 - \varepsilon_1|s_1| - \varepsilon_2|s_2| \\
&\le 0
\end{aligned} \tag{45}$$

Since $\dot{V} < 0$, then the closed-loop system is asymptotically stable. To further verify the finite time convergence of error trajectories to the sliding manifolds s_1,s_2, we may rewrite Eqs. (45) as follows:

$$\dot{V} \le -\lambda_1 s_1{}^2 - \lambda_2 s_2{}^2 - \varepsilon_1|s_1| - \varepsilon_2|s_2|$$

Since $-\lambda_1 s_1{}^2 - \lambda_2 s_2{}^2 \le 0$, then one can conclude:

$$\begin{aligned}
\dot{V} &\le -\sqrt{2}\varepsilon_1\sqrt{\frac{1}{2}s_1^2} - \sqrt{2}\varepsilon_2\sqrt{\frac{1}{2}s_2^2} + \sqrt{\frac{1}{2\gamma_1}\tilde{\rho}_1^2} + \sqrt{\frac{1}{2\gamma_2}\tilde{\rho}_2^2} - \sqrt{\frac{1}{2\gamma_1}\tilde{\rho}_1^2} - \sqrt{\frac{1}{2\gamma_2}\tilde{\rho}_2^2} \\
&\le -\ell\left(\left(\frac{1}{2}s_1^2\right)^{\frac{1}{2}} + \left(\frac{1}{2}s_2^2\right)^{\frac{1}{2}} + \left(\frac{1}{2\gamma_1}\tilde{\rho}_1^2\right)^{\frac{1}{2}} + \left(\frac{1}{2\gamma_2}\tilde{\rho}_2^2\right)^{\frac{1}{2}}\right) + \xi
\end{aligned} \tag{46}$$

where $\ell = \min\{1, \sqrt{2}\varepsilon_1, \sqrt{2}\varepsilon_2\}$ and $\xi = \sqrt{\tilde{\rho}_1^2 / 2\gamma_1} + \sqrt{\tilde{\rho}_2^2 / 2\gamma_2} \ge 0$. Using **Lemma 1**, then:

$$\begin{aligned}
\dot{V} &\le -\ell\left(\left(\frac{1}{2}s_1^2\right)^{\frac{1}{2}} + \left(\frac{1}{2}s_2^2\right)^{\frac{1}{2}} + \left(\frac{1}{2\gamma_1}\tilde{\rho}_1^2\right)^{\frac{1}{2}} + \left(\frac{1}{2\gamma_2}\tilde{\rho}_2^2\right)^{\frac{1}{2}}\right) + \xi \\
&\le -\ell\left(\frac{1}{2}s_1^2 + \frac{1}{2}s_2^2 + \frac{1}{2\gamma_1}\tilde{\rho}_1^2 + \frac{1}{2\gamma_2}\tilde{\rho}_2^2\right)^{\frac{1}{2}} + \xi \\
&\le -\ell V^{\frac{1}{2}} + \xi
\end{aligned} \tag{47}$$

Then the integration yields, using **Lemma 2**:

$$\int_{V_0}^{0} \frac{dV}{\ell V^{\frac{1}{2}} + \xi} \le -\int_{0}^{t_r} dt \tag{48}$$

$$\Rightarrow \int_{0}^{V_0} \frac{dV}{\ell V^{\frac{1}{2}} + \xi} \ge t_r \tag{49}$$

By changing variable $\ell V^{1/2} + \xi = u \rightarrow \ell V^{-1/2} dV / 2 = du,$ we will have:

$$\frac{2}{\ell^2}\int\left(\frac{u-\xi}{u}\right)du = \frac{2}{\ell^2}\left(u - \xi \ln u\right) \ge t_r \tag{50}$$

$$\Rightarrow t_r \le \frac{2}{\ell^2}\left(\ell V^{\frac{1}{2}} + \xi - \xi \ln\left(\ell V^{\frac{1}{2}} + \xi\right)\right)\Bigg|_{0}^{V_0} \tag{51}$$

$$\le \frac{2}{\ell^2}\left(\ell V_0^{\frac{1}{2}} + \xi \ln\left(\frac{\xi}{\ell V_0^{\frac{1}{2}} + \xi}\right)\right) \tag{52}$$

which proves that the upper limit of reaching time is finite when $\xi \le \delta \ll 1$. For a special case $\xi = 0$ where the adaptive laws converge faster than the control unit, we will have:

$$t_r \le \frac{2}{\ell^2}\left(\ell V_0^{\frac{1}{2}} + L\right) \tag{53}$$

$$L = \lim_{\xi \to 0^+} \xi \ln\left(\frac{\xi}{\ell V_0^{\frac{1}{2}} + \xi}\right) \tag{54}$$

To find the limit, we can use L'Hôpital Rule:

$$L = \lim_{\xi \to 0^+} \frac{\ln\left(\frac{\xi}{\ell V_0^{\frac{1}{2}} + \xi}\right)}{\frac{1}{\xi}} = \lim_{\xi \to 0^+} \frac{\frac{d}{d\xi}\left(\ln\left(\frac{\xi}{\ell V_0^{\frac{1}{2}} + \xi}\right)\right)}{\frac{d}{d\xi}\left(\frac{1}{\xi}\right)} = \lim_{\xi \to 0^+} \frac{-\xi \ell V_0^{\frac{1}{2}}}{\ell V_0^{\frac{1}{2}} + \xi} = 0 \tag{55}$$

Finally, we will get:

$$t_r \le \frac{2}{\ell} V_0^{\frac{1}{2}} \tag{56}$$

And this completes the finite-time stability proof. As a result, trajectories started from any initial conditions will reach the origin within a finite time bounded by $t_r \le 2V_0^{1/2} / \ell$.

Remark 1. If the control parameters $\gamma_1, \gamma_2, c, \lambda_1, \lambda_2$ are appropriately selected, the path-following error trajectories will converge to zero in a finite time.

3.3. Particle Swarm Optimization

PSO is a swarm-based stochastic optimization approach inspired by the collective behavior of birds flocking or fish schooling. PSO has gained widespread use in engineering problems because of its simplicity, fast convergence, and capability to handle nonlinear, non-convex search spaces. Unlike gradient-based methods, PSO does not require differentiability of the objective function which greatly reduces the computational costs [57].

In PSO, a set of particles (swarm) which are candidate solutions explores the search space. Each particle adjusts its trajectory in terms of its personal best experience and the global best experience of the swarm.

Let the position and the velocity of particle *i* at iteration *k* be:

$$X_i^k = (x_{i1}^k, x_{i2}^k, \dots, x_{in}^k) \tag{57}$$

$$V_i^k = (v_{i1}^k, v_{i2}^k, \dots, v_{in}^k) \tag{58}$$

where n is the dimension of the search space. Also, each particle remembers its personal best position:

$$P_i^k = (p_{i1}^k, p_{i2}^k, \dots, p_{in}^k) \tag{59}$$

and the swarm discerns the global best position:

$$G^k = (g_1^k, g_2^k, \dots, g_n^k) \tag{60}$$

Then the new updates for velocity and position will be as follows:

$$v_{in}^{k+1} = w \cdot v_{in}^{k} + c_1 \cdot r_1 \cdot (p_{in}^{k} - x_{in}^{k}) + c_2 \cdot r_2 \cdot (g_n^{k} - x_{in}^{k}) \tag{61}$$

$$x_{in}^{k+1} = x_{in}^{k} + v_{in}^{k+1} \tag{62}$$

where w is the inertia weight that balances between exportation and exploration, c_1 and c_2 are cognitive and social acceleration coefficients, and r_1, r_2 are zero-mean stochastic process, respectively.

In this paper, the core of the optimization problem is to obtain the SMC parameters defined within a bounded range with an appropriate performance index. A swarm set serving as the SMC gains are initialized randomly and then evaluated based on the following cost function:

$$J = \int_0^t \left(q_1 e_x^2 + q_2 e_y^2 + q_3 e_\theta^2 + \beta_1 u_1^2 + \beta_2 u_2^2\right) d\tau \tag{63}$$

where q_1, q_2, q_3, β_1, β_2 are positive constants representing the weight of each term. The velocities and positions for each particle are evaluated with Eqs. (61) and (62). The cost function (63) will be again calculated and the personal best position for each particle along with the global best position of the swarm will be calculated with Eqs. (57) to (60). This process will be repeated until the convergence criteria are reached. The set of optimal gains obtained offline will then be utilized in the sliding mode controllers for velocity and the angular velocity of the robot.

4. Control Barrier Function

Consider the following nonlinear system:

$$\dot{x} = f(x) + g(x)u, \quad x(t_0) = x_0 \tag{64}$$

where $x \in \mathbb{R}^n$ is the state vector representing the time varying variables of the system, $f(x) \in \mathbb{R}^n$ is the vector of nonlinear system dynamics, $g(x) \in \mathbb{R}^{n \times m}$ is the coupling matrix between states and control signals and $u \in \mathbb{R}^m$ is the control input vector, respectively. A forward invariant set $\mathbb{C}$ can be defined for such a system so that it spans the space into three regions [58]:

$$\mathbb{C} = \{x \in \mathbb{R}^n \mid h(x) > 0\} \tag{65}$$

$$\partial(\mathbb{C}) = \{x \in \mathbb{R}^n \mid h(x) = 0\} \tag{66}$$

$$\mathrm{In}(\mathbb{C}) = \{x \in \mathbb{R}^n \mid h(x) < 0\} \tag{67}$$

where $h:[n,1] \times 1$ is a continuously differentiable function forming the obstacle geometry, $\partial(\mathbb{C})$ and $\mathrm{In}(\mathbb{C})$, respectively, represent the perimeter and the interior of h.

Definition 1. [59] Any continuously differentiable function $h(x)$ is called a CBF if there exists an extended class K function $\alpha(\alpha(0) = 0)$ such that the following condition is satisfied for all $x \in \mathbb{R}^n$:

$$\sup_{u \in \mathbb{R}^m} \left[\dot{h}(x) + \alpha h(x)\right] \geq 0 \tag{68}$$

This condition guarantees that there always exists a controller u keeping trajectories in the safe set $\mathbb{C}$. Consequently, the safe controller u_{CBF} is defined as:

$$u_{CBF} = \left\{u : \dot{h}(x) + \alpha h(x) \geq 0\right\} \tag{69}$$

The existence of the CBF permits one to design a safe control u_{CBF} for the nonlinear dynamics (64).

Theorem 2. If any continuous function $h(x)$ is defined satisfying Eq. (68), then any designed controller u_{CBF} guarantees that set $\mathbb{C}$ is forward invariant.

Proof. Since Eq. (68) is satisfied, then any u_{CBF} will ensure that:

$$\dot{h}(x) + \alpha h(x) \geq 0 \tag{70}$$

Then by integration, one can write:

$$\int_{h_0}^{h} \frac{dh}{h} \geq -\alpha \int_0^t dt \tag{71}$$

Hence,

$$h(x) \geq h_0 e^{-\alpha t} \tag{72}$$

As a result, $\mathbb{C}$ is forward invariant.

Lemma 3. [60] Assume a predefined nominal controller $u_{nom}(x)$ for the system described in Eq. (64) is designed such that the trajectory tracking is achieved. The following optimization problem can be solved to obtain the CBF control signals such that the closed-loop states remain within the safe set $\mathbb{C}$:

$$\begin{gathered} u_{CBF}^{*} = \arg\min_u \| u - u_{nom} \|^2 \\ \text{Subj. to: } \dot{h}(x) + \alpha h(x) \geq 0 \end{gathered} \tag{73}$$

In cases where the CBF has relative degree 2 with respect to the control signals, the following conditions must be satisfied to ensure safety [22]:

$$1.\ h(0) \geq 0 \tag{74}$$

$$2.\ \dot{h}(0) + \alpha h(0) \geq 0 \tag{75}$$

$$3.\ \ddot{h}(x) + 2\alpha\dot{h}(x) + \alpha^2 h(x) > 0 \tag{76}$$

Then, the safe controller can be obtained by solving the following optimal control problem:

$$\begin{gathered} u^*_{\text{CBF}} = \arg\min_u \| u - u_{\text{nom}} \|^2 \\ \text{Subj to: } \ddot{h}(x) + 2\alpha\dot{h}(x) + \alpha^2 h(x) > 0 \end{gathered} \tag{77}$$

which guarantees the obstacle-free path following. The primary objective of this paper is to design a safe control law that enables the robot to move along an obstacle-free trajectory, minimizing tracking errors in each time step. The barrier function, in this paper, is defined as follows:

$$h = (X - X_c)^2 + (Y - Y_c)^2 - R^2 \tag{78}$$

where (X_c, Y_c) is the center of the cylinder with its radius equals R. Then, its first-time derivative will be:

$$\dot{h} = 2(X - X_c)\upsilon\cos(\theta) + 2(Y - Y_c)\upsilon\sin(\theta) \tag{79}$$

It can be seen that with a single control signal υ, the robot cannot change its path with constant steering. Therefore, to change the vehicle orientation, a second control signal ω is required. As a result, the relative degree will be 2 with respect to the angular velocity of the robot. Then, the second time derivative of h will be calculated as:

$$\ddot{h} = 2\upsilon^2 + 2\upsilon\omega\big((X - X_c)\cos(\theta) + 2\upsilon\omega(Y - Y_c)\sin(\theta)\big) = 2\upsilon^2 + \omega\dot{h} \tag{80}$$

The safe control law will be then obtained by minimizing the cost function (77) with obstacle constraints (74) to (76) and (78) to (80). Since we proved in **Theorem 1** that the nominal adaptive SMC ensures the trajectory tracking, then the optimization problem (73) guarantees the existence of a safe control u_{CBF} such that the conditions (68) will always be satisfied. Then our designed barrier function $h(x)$ satisfies conditions (74) to (76). As a result, $h(x)$ is a forward invariant set and the designed optimal safe controller will ensure obstacle avoidance.

The closed-loop framework to which the proposed control is applied is shown in Fig. 2. The combination of error signals calculated from the deviation between the reference trajectories and the output signals obtained from GPS and IMU sensors forms the sliding surfaces. The SMC parameters will be tuned by the PSO and the adaptive laws (40) and (41). The optimal adaptive SMC will then be applied to the mobile robot subjected to unknown external disturbances.

5. Simulation Study

The simulation outcomes of the proposed controller are expressed in this section. The optimization problem for PSO and CBF have been solved with *particleswarm* and *YALMIP* built-in toolboxes in MATLAB R2022b. The convergence of PSO has been checked by 50 iterations with the weighting parameters considered in performance index J to be: $q_1 = q_2 = q_3 = 5, \beta_1 = 2, \beta_2 = 10$, respectively. The *YALMIP* solver for calculating safe control inputs is considered *moment* in order to prevent getting stuck in local minima. The α function, on the other hand, for performing the optimization task equals 2.25. The sampling frequency for all simulations is considered $f_s = 50Hz$ and the simulation time is 15s, respectively. The saturation limit for linear velocity is $\upsilon_{\max} = 3.5$ and for angular velocity is 2, preventing high control values and instability.

5.1. Control scheme

The desired reference position the mobile robot must follow is an elliptical trajectory expressed as in Eqs. (81) and (82):

$$X_d = 6\cos(0.5t) \tag{81}$$

$$Y_d = 3\sin(0.5t) \tag{82}$$

And the reference heading angle can be calculated as $\theta_d = \text{atan}(\dot{Y}_d / \dot{X}_d)$. Simulations have been performed by a comparison perspective with a PSO-based PD control integrated with CBF and a Back-stepping based CBF to verify the superiority of our control framework.

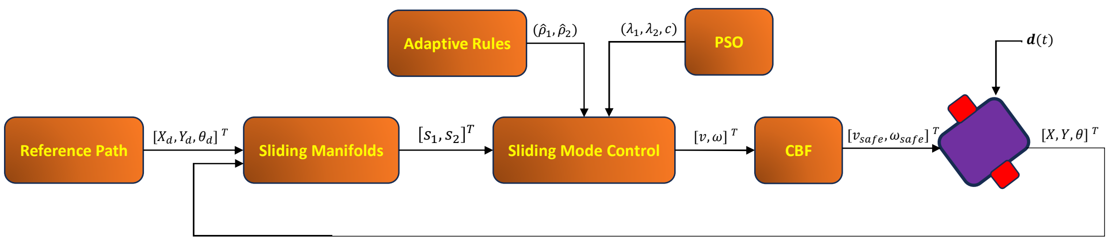

Fig. 2. System block diagram.

Table 1. Control parameters required for simulations.

Parameter	Value	Parameter	Value
γ_1	0.5	k_{px}	3.7233
γ_2	1	k_{dx}	0.8001
λ_1	0.2862	k_{py}	1.6151
λ_2	1	k_{dy}	0.0773
c	1	$k_{p\theta}$	4
k_{xBSC}	2	k_{yBSC}	4
$k_{\theta BSC}$	1		

The proposed control gains used for adaptive laws, PD control, and the PSO parameters obtained via optimization are listed in Table 1. Additionally, the obstacle geometry is considered a static cylinder with the following properties: $(X_c, Y_c, r) = (-4, 2, 0.4)m$. Furthermore, the external disturbances which must be compensated for by the proposed optimal adaptive SMC are mathematically expressed as a combination of constant and time-varying disturbances in Eqs. (83):

$$d_X = \begin{cases} 0.1\cos(0.1t) & t < 3 \\ 0 & t \geq 3 \end{cases}$$

$$d_Y = \begin{cases} 0 & t < 6 \\ 0.05 & 6 \leq t \leq 9 \\ 0.05\sin(0.1t) & t > 9 \end{cases} \tag{83}$$

$$d_\theta = \begin{cases} 0 & t < 10 \\ 0.07 & t \geq 10 \end{cases}$$

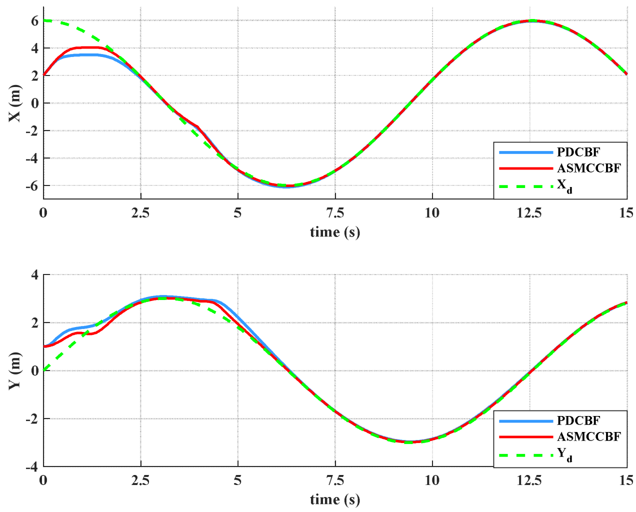

Fig. 3. Robot position X and Y.

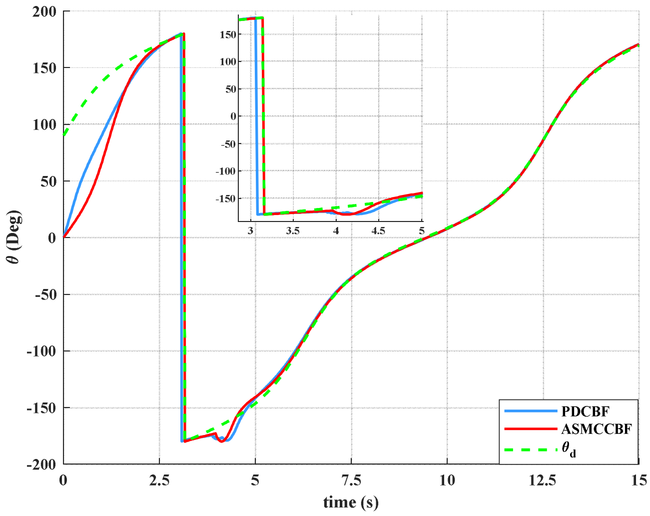

Fig. 4. Mobile robot orientation.

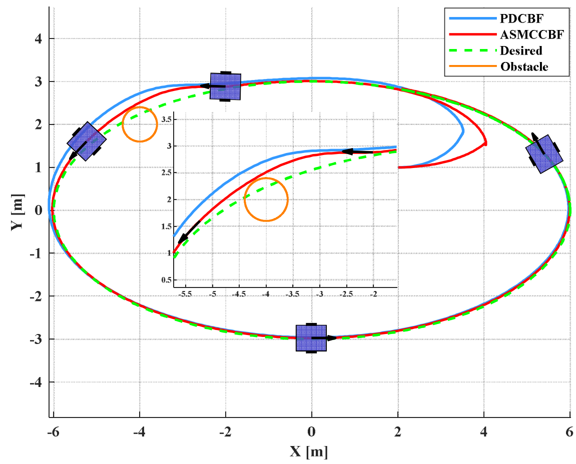

Fig. 5. Robot phase trajectory in XY plane.

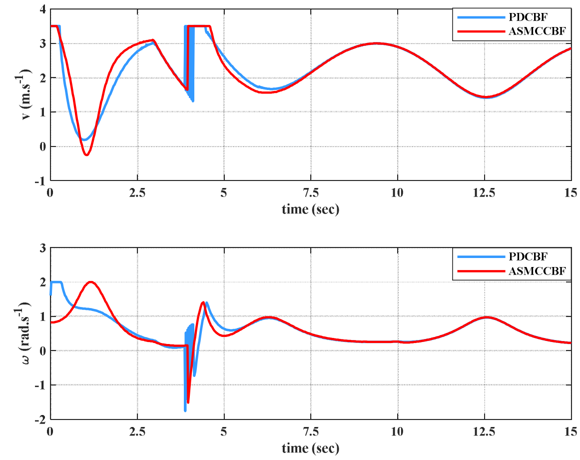

Fig. 6. Control signals.

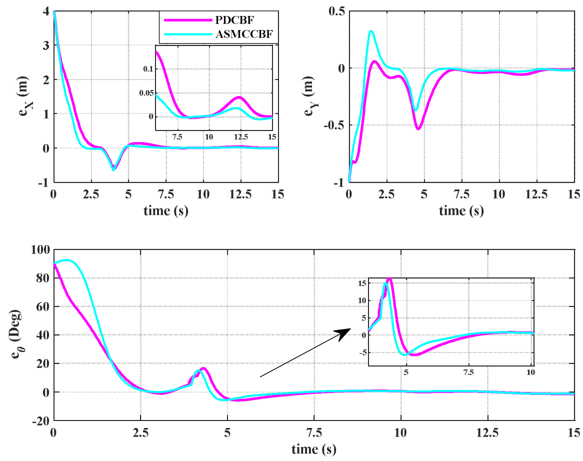

Fig. 7. Error signals.

The simulation results are shown in Figs. 3 to 7. The proposed control is marked with ASMCCBF, while the PD controller is marked with PDCBF in all figures, respectively. According to Fig. 3, it is shown that the global position of the robot, X, Y, exactly matches the desired trajectory with the proposed control scheme, and the settling time is shorter than that of the PDCBF. Moreover, whereas both control methods are able to navigate the autonomous robot safely without any collision with the cylinder, the proposed ASMCCBF results in reduced angular deviation relative to the PDCBF. Figure 4 displays the orientation of the robot obtained via the suggested control approach. The heading angle is mapped in the range $[-\pi, \pi]$ so that it remains bounded. It can be observed from Fig. 4 that both control schemes show appropriate performance in the tracking of the desired heading angle. However, the output behavior in our proposed control technique has better performance in attenuation of overshoots, compensating for disturbances, and the correct following of the desired angle, before and after confronting the obstacle. Figure 5 illustrates the phase plane trajectory. It is obvious that the proposed control unit is significantly capable of navigating the vehicle through the prescribed path and changing its orientation and position when facing the obstacle with almost zero deviation from the reference lateral and longitudinal displacement. Compared with PDCBF, our proposed control algorithm has way better performance in capturing the sudden behavior change of the desired path, and in compensating for the effect of external unknown time-varying disturbances.

Table 2. RMSE results.

RMSE	PDCBF	ASMCCBF
X	0.12749	0.11772
Y	0.13997	0.078428
θ	0.060488	0.048034

Figure 6 demonstrates the values of input signals obtained via the proposed control unit. In relation to Fig. 6, linear and angular velocities converge to their steady-state values in a finite time. Also, it can be seen that the oscillations in the proposed controller are much smaller than those of the PDCBF, especially when the robot needs to change its orientation. Furthermore, both controllers have acceptable performance in which no saturation has occurred. Figure 7 indicates the tracking errors for the state variables of the closed-loop system. It can be seen that the settling time is faster with the proposed control scheme, and smaller overshoots were produced when the orientation of the robot was changed to avoid an obstacle, in comparison with PDCBF. This highlights the potential of the SMC framework in disturbance rejection and obstacle avoidance.

A numerical comparison study with the RMSE criteria which is defined as $\mathrm{RMSE} = \sqrt{e^2 / N}$ where e is the tracking error, is carried out to evaluate the tracking accuracy and the results are summarized in Table 2 for all degrees of freedom of the mobile robot.

The numerical results indicate that both controllers achieved satisfactory tracking performance for the desired trajectory. For the longitudinal direction, the PDCBF controller yielded an RMSE of 0.12749 , while the ASMCCBF achieved a slightly smaller value of 0.11772 . This explains that the ASMCCBF improves tracking precision in the forward motion, though the difference is relatively moderate. In the lateral direction, the performance difference becomes more pronounced. The RMSE of the PDCBF controller was 0.13977 while the ASMCCBF reduced this error remarkably to 0.078428 . This highlights the superior robustness of the ASMCCBF approach in handling lateral dynamics, which are typically more sensitive to disturbances and modeling uncertainties. With regard to the heading angle, the RMSE values are 0.060488 for the PDCBF and 0.048034 for the ASMCCBF which means that the ASMCCBF outperforms the PDCBF, providing smoother and more accurate tracking.

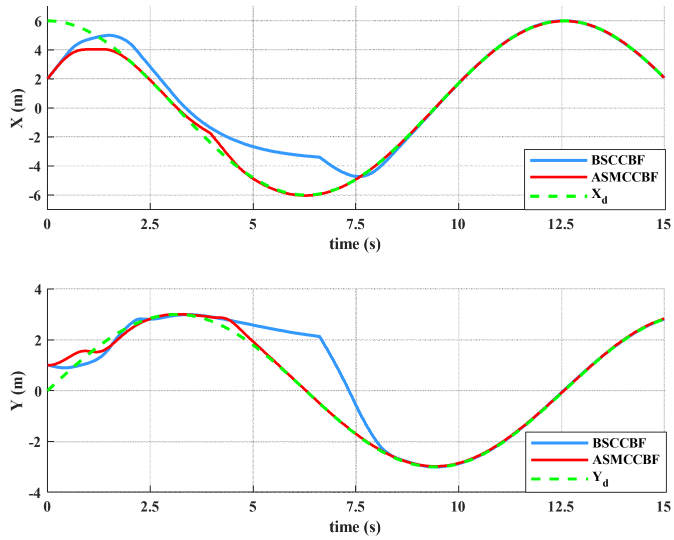

Fig. 8. Robot position in ASMCCBF and BSCCBF.

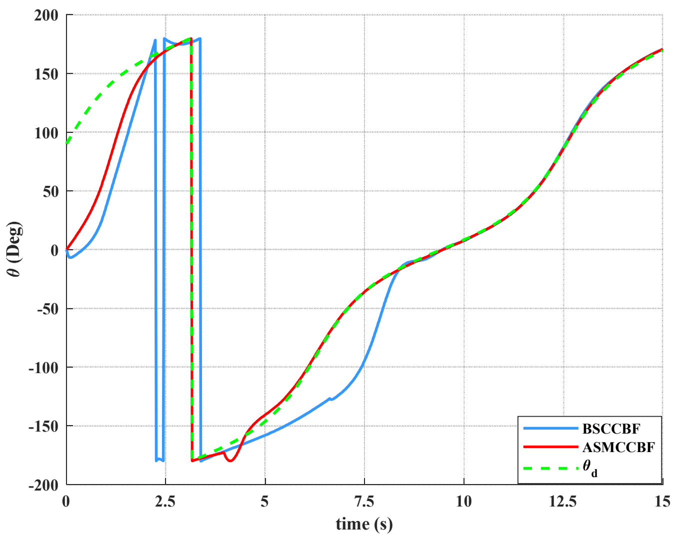

Fig. 9. robot orientation illustration.

To further demonstrate the efficacy of the proposed ASMCCBF, the closed-loop performance has been also compared via a Backstepping controller to which CBF is applied (BSCCBF). The results are shown in Figs. 8 and 9. The comparative analysis between the recommended ASMCCBF and BSCCBF reveals a distinct performance hierarchy, particularly in the context of nonholonomic two-wheeled mobile robots. In the translational dynamics governing the X and Y positions, the ASMCCBF framework demonstrates a fundamental advantage rooted in its inherent invariance to matched uncertainties and disturbances. While both methods incorporate CBF to guarantee forward invariance of the safe set (obstacle avoidance), the Backstepping approach relies on a recursive Lyapunov design that is inherently sensitive to the accuracy of the virtual control derivatives. Conversely, the SMC+CBF approach decouples the robustness handling from the safety filter. The result is a trajectory that adheres to the safe set boundaries with the rigidity of a mechanical constraint while preserving the crisp, overdamped transient response characteristic of sliding mode dominance, thereby eliminating the oscillatory energy waste evident in the BSC position plots. Hence, the positions for the proposed control method have better behavior in response time, smaller settling time, and way much less overshoot.

The disparity in performance becomes even more pronounced when examining the orientation dynamics, a critical degree of freedom for differential-drive robots where heading error directly couples into lateral position drift. The high overshoot observed in the BSCCBF orientation tracking, according to Fig. 9, is symptomatic of a phase lag introduced by the virtual control law's attempt to simultaneously stabilize the kinematic chain while respecting the CBF safety constraints. In BSC, the control torque is a function of both the orientation error and the derivative of the stabilizing function for the position; when the CBF intervenes to prevent a collision, it introduces a discontinuity in the desired yaw rate profile, which the Backstepping integrator chain struggles to follow without transient ringing. ASMCCBF sidesteps this issue through the sliding mode's reaching phase and subsequent high-frequency switching. The SMC law enforcers a first-order decay of the heading error that is agnostic to the smoothness of the reference signal imposed by the CBF filter. Because the sliding surface dynamics are of reduced order, the output response of the orientation angle is effectively decoupled from the transient perturbations of the obstacle avoidance maneuver.

5.2. Combined linear and constant trajectory

The reference signal in this case is given as:

$$X_d(t) = t \tag{84}$$

$$Y_d(t) = \begin{cases} \dfrac{t}{2}, & t \le 6 \\ 3, & t > 6 \end{cases} \tag{85}$$

The disturbance signals and the obstacle geometry are the same as the previous case of simulations. For the sake of obstacle avoidance, we have compared the results of this case with a Model Predictive Control (MPC) in which the CBF condition $\ddot{h} + \alpha^2 h + 2\alpha\dot{h} \ge 0$ is deemed to be the nonlinear constraint. The optimization problem of the MPC to obtain the safe controller is as follows:

$$J = \int_0^t \left(e^T Q e + u^T R u\right) d\tau \tag{86}$$

$$\text{Subj. to}: \dot{x} = Ax + Bu \tag{87}$$

$$\ddot{h} + \alpha^2 h + 2\alpha\dot{h} \ge 0 \tag{88}$$

The outcomes are exhibited in Figs. 10 and 11. According to Fig. 10, the positions X and Y in the proposed controller has smoother behavior along with faster settling time. Despite the fact that the output orientation in MPC scheme has smaller overshoots when the robot encounters the obstacle, the tracking performance in SMC after passing the obstacle demonstrates lower steady state error.

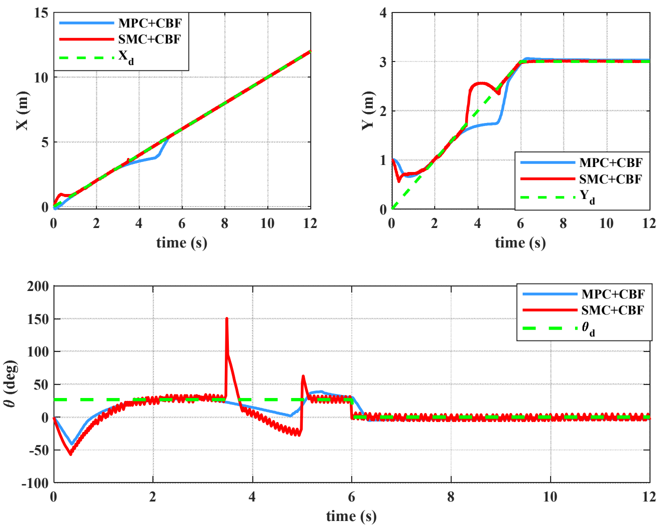

Fig. 10. The output states of the mobile robot in the proposed method compared with MPC.

Table 3. ISE results for SMC and MPC.

ISE	MPCCBF	ASMCCBF
X	0.7982	0.1379
Y	0.7994	0.5442
θ	0.7482	0.9688

In relation to Table 3, the Integral Squared Error (ISE) results are compared for the implemented control methods, ASMCCBF and MPCCBF. It is clear that in the case of position trajectory tracking, our proposed methodology has superior performance over the MPCCBF, but when it comes to the orientation following, MPC has converged faster and the produced control signals are smoother.

The phase plane trajectory is shown in Fig. 11. As illustrated, the proposed method exhibits zero steady state error after the vehicle passes the obstacle. Both controllers, however, shows acceptable tracking performance before the robot faces the obstacle. Overall, our sliding mode control approach combined with CBF has smoother transient and steady state error with respect to the MPC method.

6. Computational Complexity Analysis

In this section, we have added the analysis of the computational complexity which addresses the concern about the real-time evaluation and implementation. According to Fig. 12, we have measured and evaluated the total step time, controller-only computation time, and CBF solver time for all three controllers (SMC + CBF, Backstepping + CBF, and PD + CBF) at the 50 Hz sampling rate. All controllers are shown to have achieved 100% real-time feasibility with average CPU utilization below 2.5%. Additionally, we counted the floating-point operations (FLOPs) required by each controller per step, indicated in the Fig. 12. The SMC requires 33 FLOPs, Backstepping requires 12 FLOPs, and PD requires 19 FLOPs. All controllers have ORDER (1) time and space complexity per step.

According to Fig. 13, the worst-case step time for all controllers is well below 20 *ms* sampling period, providing a computational margin of over 75%. This confirms that the proposed approach is real-time implementable on standard embedded hardware. The computational complexity analysis presented in this article provides strong evidence, as shown in the figures, that the proposed methodology is implementable for real-time deployment.

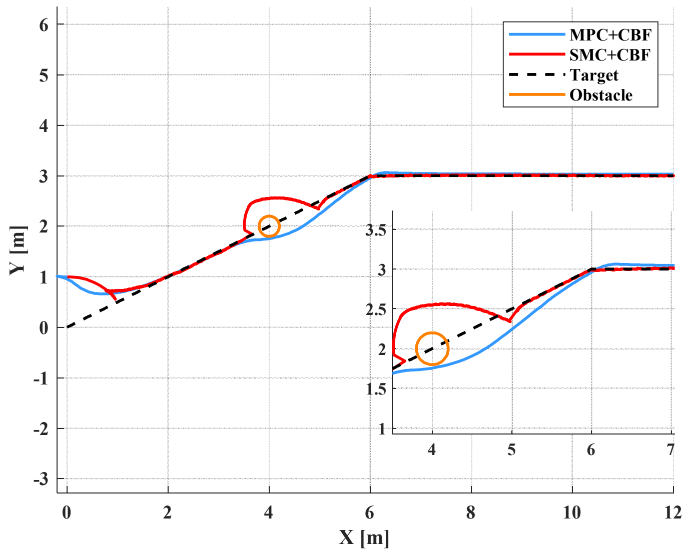

Fig. 11. Phase plane of wheeled robot.

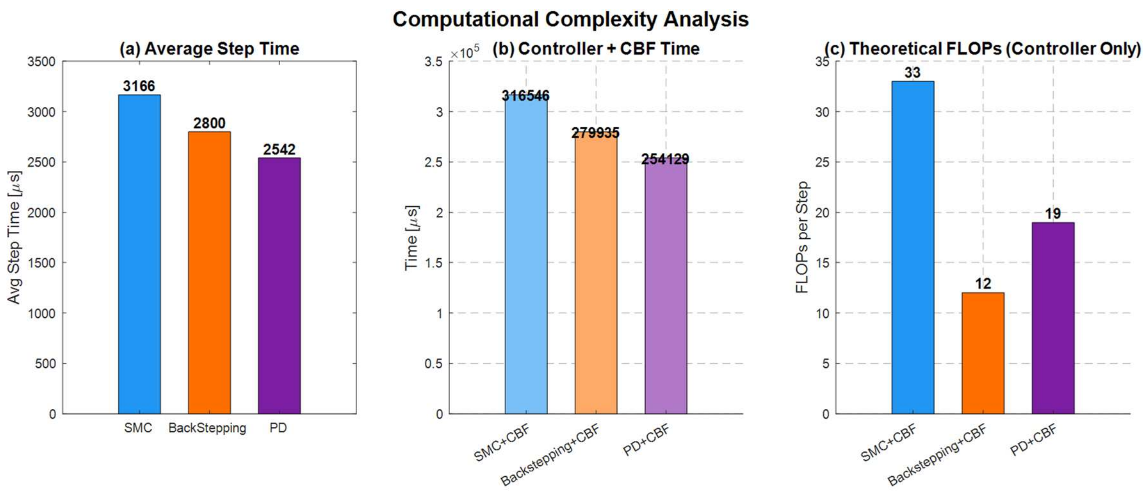

Fig. 12. Complexity analysis bar chart.

Table 4. System and controller specifications regarding HIL test.

Specification	Laptop (Dynamics)	Arduino Module (Controller)
Processor	Intel® Cor™ i5 CPU @ 4.20 GHz	8-bit MPU dual processor
Installed Memory (RAM)	16 GB	4 MB

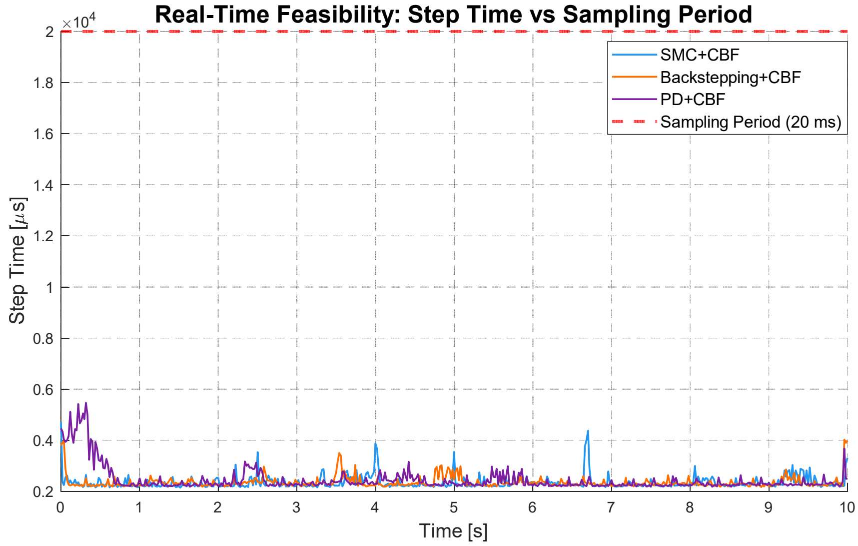

Fig. 13. The real-time feasibility analysis of all controllers.

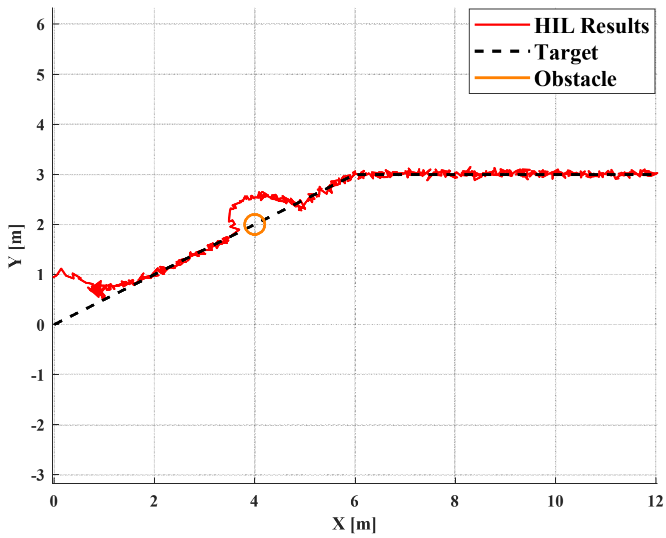

Fig. 14. Phase plane results for the path of the robot with HIL.

7. Hardware in the Loop Simulations

To bridge the gap between pure simulation and real-world deployment, we have implemented Hardware-in-the-Loop (HIL) simulations to evaluate our closed-loop control scheme in a practical implementation. In our HIL setup, the control algorithm ran in an Arduino UNO microcontroller, communicating with a high-fidelity dynamic model of the mobile robot and virtual sensors. This has allowed us to verify the controller's real-time computability, robustness to timing constraints, and correct handling of sensor-like noise and discretization effects. While HIL does not capture all physical uncertainties, it provides a substantially more realistic validation than offline numerical simulations alone. The requirements and specifications are listed in the following table.

In this Hardware-in-the-Loop setup, the microcontroller acts as the main controller: it receives sensor measurements and calculates the control commands for trajectory tracking. The mobile robot dynamics are simulated on a laptop, which outputs the vehicle's position and orientation, all corrupted by additive sensor noise. The measurement noise for each sensor is modeled as zero-mean white noise with the following covariance matrices: $R_{GPS} = 0.002$, $R_{Encoder} = 0.0025$, respectively. A sampling delay of 1 *ms* is assumed for both measurement sensors. The proposed SMC-CBF controller algorithm has been compiled on the microcontroller module. The data in real-time are transferred between the laptop (Dynamics simulator) and the Arduino module over a serial USB link at a baud rate of 9600 *bps*.

Figure 14 presents the HIL results obtained from the implemented HIL SMC-CBF controller. The phase plane plot shows the evolution of the robot's trajectory in the XY plane during an obstacle-avoidance maneuver. The state trajectory converges smoothly to the desired equilibrium point while clearly circumventing the unsafe region defined by the CBF constraint. No violation of the safety boundary occurs, the robot could manage to avoid collision, and the sliding surface is reached rapidly, confirming that the real-time implementation preserves the finite-time convergence property of the proposed SMC.

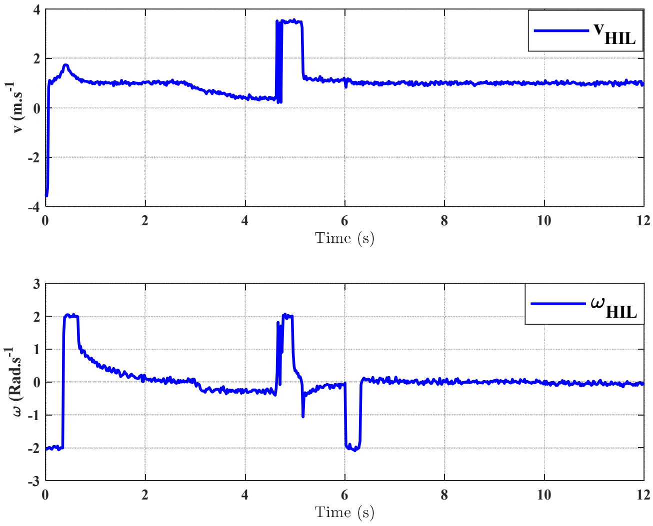

Fig. 15. Control signals obtained by HIL simulations.

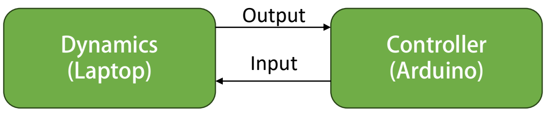

Fig. 16. HIL Setup for mobile robot.

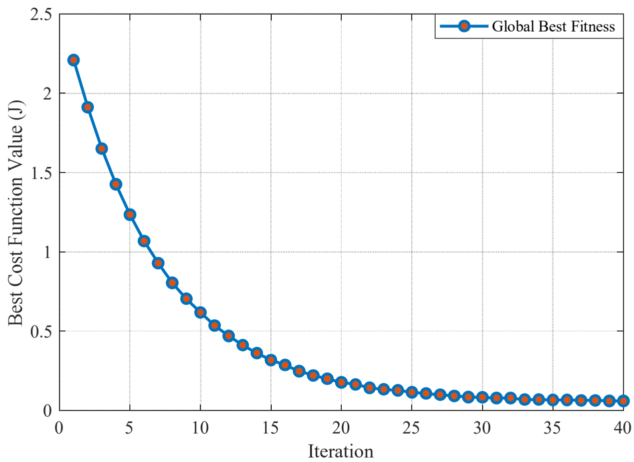

Fig. 17. Analysis of PSO optimization process.

The corresponding control inputs, linear velocity and angular velocity, are shown in Fig. 15. Both signals remain within the actuator limits throughout the maneuver. The SMC component introduces a high-frequency switching action, visible as a slight chattering in ω when the system is near the sliding surface, while the CBF-mediated modification smoothly adjusts the velocity command v only when the robot approaches an obstacle. Importantly, the HIL results closely match those from the pure numerical simulations, with negligible timing jitter and no data loss over the serial port, demonstrating that the controller is computationally feasible for real-time embedded deployment. The schematic of the HIL setup along with the communication between the Laptop and Arduino is shown in Fig. 16.

8. Sensitivity Analysis

To address the concern regarding parameter sensitivity, we have conducted additional simulations to evaluate the robustness of the proposed SMC-CBF. Regarding the PSO optimization process, we have expanded the manuscript to include a detailed analysis. Figure 17 demonstrates the convergence curve of the PSO algorithm, which illustrates the best fitness value over iterations. This confirms that the optimization process effectively minimizes the cost function and converges smoothly to an optimal set of control parameters without oscillation. These additions have been incorporated into the revised manuscript. We perturbed the fast control parameters λ_1, λ_2 and the sliding manifold parameters c by $\pm 20\%$ to represent modeling or tuning uncertainties. The results shown in Fig. 18 demonstrate that the adaptive switching gains effectively compensate for these parameter variations, maintaining stable and accurate trajectory tracking without requiring gain retuning, and the Euclidean norm of tracking error will converge to zero exponentially. Finally, we have tested the controller robustness under varying initial conditions by using different starting positions and orientations. The outcomes are displayed in Fig. 19. The system successfully converged to the desired trajectory in all scenarios, proving the controller's low sensitivity to initialization errors, and the adaptive control unit helps the robot move on the predefined path without collision or impact to the obstacle.

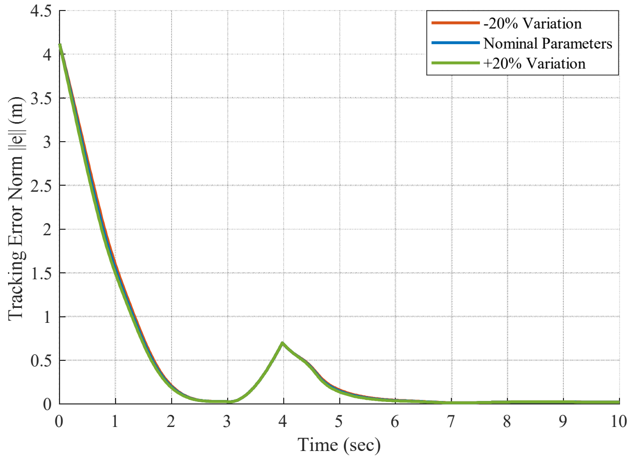

Fig. 18. Sensitivity to SMC Parameter Variations.

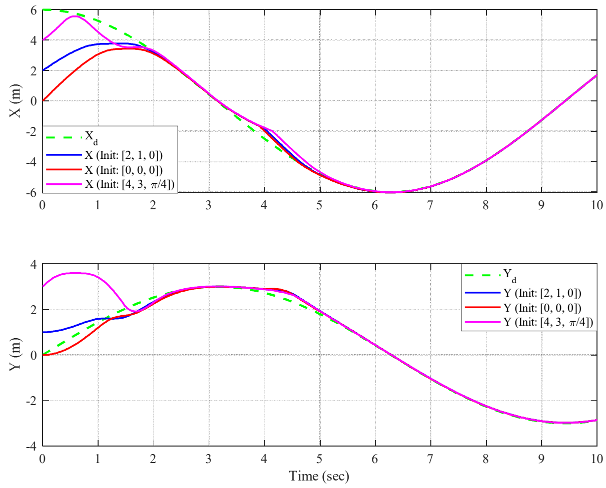

Fig. 19. Sensitivity to Initial Condition Variations.

9. Conclusion

This study introduced an adaptive SMC framework augmented with a CBF for the reliable navigation of a two-wheeled mobile robot operating in an environment with unknown external disturbances. The proposed closed-loop scheme successfully combined the robustness of the SMC with an adaptive mechanism that dynamically tuned the switching control gains, eliminating the necessity of predefining the disturbance bounds and enhancing disturbance rejection capabilities. Additionally, the use PSO provided a wise procedure for fine-tuning the control parameters that noticeably reduced steady-state values of tracking errors and diminished the occurrence of large overshoots, improving the closed-loop framework performance. The integration of the CBF ensured safety by enforcing obstacle avoidance as it minimized the deviation between the generated safe control inputs and the nominal commands. Extensive simulation studies validated the effectiveness of the ASMCCBF which resulted in precise trajectory tracking, collision-free navigation, and consistent control efficiency even under uncertain operating conditions. The developed controller offers a robust, adaptive, and optimization-enhanced solution that can be expanded to a wide range of robotic platforms and practical navigation tasks in uncertain and dynamic environments for real-time implementation which will be considered by the authors as future works. Experimental validation is also planned as future work once the necessary hardware infrastructure becomes available.

Author Contributions

N. Firouzi performed the Conceptualization, Formal Analysis, Software, Validation, and Writing-Original draft; H. Khazal performed the Investigation, Resources, Data Curation, and Writing-Review and editing; Y.K. Khdir performed the Conceptualization, Investigation, Data Analysis, and Writing-Original draft; P. Podulka performed the Conceptualization, Software, Investigation, Supervision, and Writing-Review and Editing; A.A. Faqihi and H.M. Hadidi performed the Investigation, HIL simulations, Data curation, Resources, and Writing-Review and editing. The manuscript was written through the contribution of all authors. All authors discussed the results, reviewed, and approved the final version of the manuscript.

Acknowledgments

Not applicable.

Conflict of Interest

The authors declared no conflict of interest to disclose.

Funding

No funding is received for this article.

Data Availability Statements

No data is used for this article.

References

[1] Raj, R., Kos, A., A Comprehensive Study of Mobile Robot: History, Developments, Applications, and Future Research Perspectives, *Applied Sciences*, 12(14), 2022, 6951.
[2] Tang, Y., Zakaria, M.A., Younas, M., Path Planning Trends for Autonomous Mobile Robot Navigation: A Review, *Sensors*, 25(4), 2025, 1206.
[3] Ben Jabeur, C., Seddik, H., Design of a PID optimized neural networks and PD fuzzy logic controllers for a two-wheeled mobile robot, *Asian Journal of Control*, 23(1), 2021, 23–41.
[4] Ni, J., Wang, Y., Li, H., Du, H., Path Tracking Motion Control Method of Tracked Robot Based on Improved LQR Control, *2022 41st Chinese Control Conference (CCC)*, 2022, 2888–2893.
[5] Vu, T.M., Moezzi, R., Cyrus, J., Hlava, J., Model Predictive Control for Autonomous Driving Vehicles, *Electronics (Basel)*, 10(21), 2021, 2593.
[6] Hassani, I., Rekik, C., Backstepping Controller for Mobile Robot in Presence of Disturbances and Uncertainties, *International Journal of Robotics and Control Systems*, 3(4), 2023, 934–954.
[7] Moudoud, B., Aissaoui, H., Diany, M., Fuzzy adaptive sliding mode controller for electrically driven wheeled mobile robot for trajectory tracking task, *Journal of Control and Decision*, 9(1), 2022, 71–79.
[8] Liang, Z., Wang, X., Guo, X., Adaptive Backstepping Sliding Mode Control of Nonholonomic Mobile Robot with Input Delays Based on RBF Neural Network, *2025 37th Chinese Control and Decision Conference (CCDC)*, 2025, 6431–6436.
[9] Huang, H., Gao, J., Backstepping and Novel Sliding Mode Trajectory Tracking Controller for Wheeled Mobile Robots, *Mathematics*, 12(10), 2024, 1458.
[10] Amarasiri, N., Barhorst, A.A., Gottumukkala, R., Sharp Curve Trajectory Tracking of a Universal Omni-Wheeled Mobile Robot Using a Sliding Mode Controller, *ASME Letters in Dynamic Systems and Control*, 5(2), 2025, 024501.
[11] Li, Z., Zhai, J., Super-twisting sliding mode trajectory tracking adaptive control of wheeled mobile robots with disturbance observer, *International Journal of Robust and Nonlinear Control*, 32(18), 2022, 9869–9881.
[12] Phuc, P.T., Tho, T.P., Hai, N.D.X., Thinh, N.T., Design of Adaptive Fuzzy Sliding Mode Controller for Mobile Robot, *International Journal of Mechanical Engineering and Robotics Research*, 10, 2021, 54–59.
[13] Ha, V.T., Thi Thuong, T., Ngoc Truc, L., Trajectory tracking control based on genetic algorithm and proportional integral derivative controller for two-wheel mobile robot, *Bulletin of Electrical Engineering and Informatics*, 13(4), 2024, 2348–2357.
[14] Pérez-Juárez, J.G. et al., Kinematic Fuzzy Logic-Based Controller for Trajectory Tracking of Wheeled Mobile Robots in Virtual Environments, *Symmetry (Basel)*, 17(2), 2025, 301.
[15] Nasir, M., Fareh, R., Ghommam, J., Gad, O., Khadraoui, S., Bettayeb, M., Adaptive Fractional-Order Sliding Mode Control for Mobile Robots: A PSO-Based Optimization Framework, *2025 IEEE 22nd International Multi-Conference on Systems, Signals & Devices (SSD)*, 2025, 1111–1116.
[16] Beharu Arega, T., Mersha Tesfa, Y., Abdissa, C.M., Three-Wheeled Mobile Robot Trajectory Tracking Control Using Nonlinear PID Controller Based Neural Network Combined with Backstepping Controller, *IEEE Access*, 13, 2025, 100167–100182.
[17] Raj, S., Trajectory Tracking and Obstacle Avoidance for Mobile Robot using Laguerre based Model Predictive Control, *2025 International Conference on Computer, Electrical & Communication Engineering (ICCECE)*, 2025, 1–4.
[18] Xiao, H., Chen, C., Zhang, G., Chen, C.L.P., Reinforcement learning-driven dynamic obstacle avoidance for mobile robot trajectory tracking, *Knowledge-based Systems*, 297, 2024, 111974.
[19] Katona, K., Neamah, H.A., Korondi, P., Obstacle Avoidance and Path Planning Methods for Autonomous Navigation of Mobile Robot, *Sensors*, 24(11), 2024, 3573.
[20] Zheng, K., Autonomous Obstacle Avoidance and Trajectory Planning for Mobile Robot Based on Dual-Loop Trajectory Tracking Control and Improved Artificial Potential Field Method, *Actuators*, 13(1), 2024, 37.
[21] Cao, Y., Mohamad Nor, N., An improved dynamic window approach algorithm for dynamic obstacle avoidance in mobile robot formation, *Decision Analytics Journal*, 11, 2024, 100471.
[22] Nasab, A.A., Badiei, R., Asemani, M.H., Safe prescribed time controller for wheeled mobile robots by using control barrier functions as a safety filter, *ISA Transactions*, 162, 2025, 119–133.
[23] Xu, L., Chen, W., Li, H., Sliding Mode Control with Control Barrier Functions for Uncertain Nonlinear Systems, *IEEE Transactions on Automatic Control*, 67(8), 2022, 4235–4242.
[24] Li, Z., Wang, Y., CBF-Based Sliding Mode Control for Mobile Robot Obstacle Avoidance, *Robotics and Autonomous Systems*, 161, 2023, 104345.
[25] Chen, H., Liu, J., Zhang, P., Adaptive Sliding Mode Control Barrier Functions for Safety-Critical Systems, *Automatica*, 148, 2023, 110772.
[26] Chinelato, D.S., Angélico, C.A., Sliding Mode Control Barrier Function, *Asian Journal of Control*, 24(6), 2022, 3177–3190.
[27] Nguyen, H., Vu, M., Unifying Hierarchical Sliding Mode Control and Control Barrier Functions for Tilt Angle Constraint of a Ball-Balancing Robot, *IET Cyber-Systems and Robotics*, 7(1), 2025, 42–54.
[28] Syntakas, G., Vlachos, K., Safe Sliding Mode Control for Marine Vessels Using Higher-Order Control Barrier Functions and Fast Projection, *arXiv:2512.24281*, 2025.
[29] Sawarkar, S., Sangeerth, Saharsh, Jagtap, S., Sliding Mode Control for Safe Trajectory Tracking with Moving Obstacles Avoidance, *arXiv:2604.24518*, 2026.
[30] Obeid, H., Fridman, L.M., Laghrouche, S., Harmouche, M., Barrier Function-Based Adaptive Sliding Mode Control, *Automatica*, 93, 2018, 286–293.
[31] Obeid, H., Laghrouche, S., Fridman, L.M., Harmouche, M., Barrier Function-Based Adaptive Higher Order Sliding Mode Control, *Automatica*, 119, 2020, 109065.
[32] Obeid, H., Laghrouche, S., Fridman, L.M., BFASM: Barrier Function Based Adaptive Sliding Mode Control for Uncertain Systems with Input Saturation, *IEEE Access*, 10, 2022, 10514–10523.
[33] Wang, Y., Liu, Z., Chen, X., Barrier Function-Based Fuzzy Adaptive Super-Twisting Integral Terminal Sliding Mode Control for Robotic Manipulators, *Symmetry*, 17(3), 2025, 412.
[34] Zhang, J., Li, H., Zhou, Q., Saturated Adaptive Barrier Sliding Mode Control with State Constraints for Nonlinear Systems, *IET Control Theory & Applications*, 18(5), 2024, 621–634.
[35] Liu, R., Zhao, X., Wang, F., Barrier Function-Based Fixed-Time Fractional Order Sliding Mode Control for Nonlinear Systems, *Scientific Reports*, 15, 2025, 8421.
[36] Taylor, A.J., Ames, A.D., Adaptive Safety with Control Barrier Functions, *Proceedings of the American Control Conference (ACC)*, 2020, 1393–1399.
[37] Duwaish, H., Al-Dhaifallah, M., Particle Swarm-Based Sliding Mode Control with Chattering Reduction, *Actuators*, 13(4), 2024, 128.
[38] Kim, S., Park, J., Lee, B., Online Sliding Mode Gain Tuning Using Particle Swarm Optimization for Mobile Robot Control, *Electronics*, 13(7), 2024, 1312.
[39] Kim, D., Park, J., Kim, H.J., Online Adaptive Inter-Control-Constraint CBF for Safety-Critical Control, *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2025, 3121–3127.
[40] Wu, Y., Li, Z., Zhu, J., How to Adapt Control Barrier Functions? A Learning-Based Approach, *arXiv:2504.03038*, 2025.
[41] Teji, M.D., Zou, T., Zeleke, D.S., Backstepping-Based Nonsingular Terminal Sliding Mode Control for Finite-Time Trajectory Tracking of a Skid-Steer Mobile Robot, *Robotics*, 13(12), 2024, 180.
[42] Nguyen, V., Vu, M.T., Kim, S.H., A Finite-Time Non-Singular Fast Terminal Sliding Mode Control of Wheeled Mobile Robots with Prescribed Performance, *IET Control Theory & Applications*, 19(1), 2025, 70013.

[43] Hachani, S., Nechadi, E., Type-2 Fuzzy Logic-Based Robot Navigation in Uncertain Environments: Simulation and Real-World Implementation, *Journal of Robotics and Control (JRC)*, 6(1), 2025, 437–445.
[44] Ahmadlou, N., Mirzaei, M., Rafatnia, S., Information fusion-based estimation of model uncertainties and disturbances for constrained predictive control of autonomous wheeled mobile robot, *International Journal of Dynamics and Control*, 13(5), 2025, 192.
[45] Huang, X., You, Z., Zhang, W., Fang, Z., Yi, D., Chen, R., Stabilization control of wheeled mobile robot based on neural sliding mode under wheel slip conditions, *International Journal of Dynamics and Control*, 13(2), 2025, 56.
[46] Tran, D.L., Cao, N.T.D., Phan, V.D., Duong, D.T., Ho, S.P., Advanced Trajectory Tracking Control for Wheeled Mobile Robots Under Actuator Faults and Slippage, *Bulletin of Electrical Engineering & Informatics*, 14(3), 2025, 1746–1757.
[47] Zheng, Y., Zheng, J., Shao, K., Zhao, H., Xie, H., Wang, H., Adaptive Trajectory Tracking Control for Nonholonomic Wheeled Mobile Robots: A Barrier Function Sliding Mode Approach, *IEEE/CAA Journal of Automatica Sinica*, 11(4), 2024, 1007–1021.
[48] Yesmin, A., Nath, K., Kumari, K., Sinha, A., Bera, M.K., Adaptive Generalized Super-Twisting Sliding Mode Controller for Wheeled Mobile Robot, *IFAC-PapersOnLine*, 57, 2024, 345–350.
[49] Ayten, K.K., Çiplak, M.H., Dumlu, A., Implementation a fractional-order adaptive model-based PID-type sliding mode speed control for wheeled mobile robot, *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 233(8), 2019, 1067–1084.
[50] Dang, S., Dinh, X., Than Ngoc, T., Nguyen, V., Kim, D., Le, X., Adaptive Sliding Mode Control for Robust Trajectory Tracking of Car-Like Mobile Robots in Autonomous Navigation Systems, *Journal of Applied and Computational Mechanics*, 12(3), 2026, 739–758.
[51] Tzafestas, S.G., Mobile Robot Control and Navigation: A Global Overview, *Journal of Intelligent & Robotic Systems*, 91(1), 2018, 35–58.
[52] Song, Y., Ye, H., Lewis, F.L., Prescribed-time control and its latest developments, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 53(7), 2023, 3677–3690.
[53] Behjameh, M.R., Delavari, H., Vali, A., Global Finite Time Synchronization of Two Nonlinear Chaotic Gyros Using High Order Sliding Mode Control, *Journal of Applied and Computational Mechanics*, 1(1), 2014, 26–34.
[54] Gharagozloo, M.H., Shahmansoorian, A., Chaos Control in Gear Transmission System using GPC and SMC Controllers, *Journal of Applied and Computational Mechanics*, 8(2), 2022, 545–556.
[55] Anjum, Z., Sun, Z., Chen, B., Disturbance-Observer-Based Fault-Tolerant Control of Robotic Manipulator: A Fixed-Time Adaptive Approach, *IET Control Theory & Applications*, 18(11), 2024, 1398–1413.
[56] Hoang, T.H., Nguyen, V., Ho, K., Chu, N., Ngo, Q., Wireless-Controlled Capsule Robots: A Novel Design and Experimental Validation, *Journal of Applied and Computational Mechanics*, 12(1), 2026, 271–288.
[57] Numbi, J., Zioui, N., Tadjine, M., Quantum Particle Swarm Optimization Proportional–Derivative Control for Trajectory Tracking of a Car-like Mobile Robot, *Electronics (Basel)*, 14(5), 2025, 832.
[58] Ames, A.D., Xu, X., Grizzle, J.W., Tabuada, P., Control Barrier Function Based Quadratic Programs for Safety Critical Systems, *IEEE Transactions on Automatic Control*, 62(8), 2017, 3861–3876.
[59] Garg, K. et al., Advances in the Theory of Control Barrier Functions: Addressing practical challenges in safe control synthesis for autonomous and robotic systems, *Annual Review of Control*, 57, 2024, 100945.
[60] Seiler, P., Jankovic, M., Hellstrom, E., Control Barrier Functions with Unmodeled Input Dynamics Using Integral Quadratic Constraints, *IEEE Control Systems Letters*, 6, 2022, 1664–1669.

ORCID iD

Nasser Firouzi https://orcid.org/ 0000-0002-2056-7472
Haider Khazal https://orcid.org/0009-0002-6044-9582
Younis Khalid Khdir https://orcid.org/0000-0001-5861-1283
Przemysław Podulka https://orcid.org/ 0000-0002-1699-3137
Abdullah A. Faqihi http://orcid.org/0009-0006-3512-9850
Haitham M. Hadidi https://orcid.org/0000-0002-4727-1698

How to cite this article: Firouzi N., Khazal H., Khdir Y.K., Podulka P., Faqihi A.A., Hadidi H.M. Barrier Function-Based Adaptive Sliding Mode Control for Disturbed Two-Wheeled Robot Obstacle Avoidance, *J. Appl. Comput. Mech.*, xx(x), 2026, 1–20. https://doi.org/10.22055/jacm.2026.49111.5743