Predictive Reference Design for Robust and Optimal Motion Control of Differential-Drive Robots Under Perturbations

Sabeur BEN HADJ MOHAMED [1]*, Adel MAHJOUB[2], Bouthaina DAMMAK[3], Anouar BENAMOR[4]

[1] Faculty of Sciences of Monastir, University of Monastir, Avenue of the environment, 5019, Monastir, Tunisia
belhadj.med.saber@gmail.com (**Corresponding author*)

[2] Higher Institute of Applied Sciences and Technology of Kairouan, University of Kairouan,
Route périphérique Dar El Amen, 3100, Kairouan, Tunisia
adel.mahjoub@isigk.rnu.tn

[3] Department of Computer Science, Applied College, Princess Nourah bint Abdulrahman University,
P.O. Box 84428, Riyadh, 11671, Saudi Arabia
Badammak@pnu.edu.sa

[4] Department of Electrical Engineering, National Engineering School of Monastir, University of Monastir,
Avenue Ibn El Jazzar, 5019, Monastir, Tunisia
Benamor_anouar@yahoo.com

Abstract: In natural environments, interception represents a predatory behavior as the hunter anticipates the prey's path and moves toward a future position of the target rather than following it directly. This approach generally reduces both the time required to capture the prey and the associated energy consumption. Based on this concept, this paper introduces an interception-based controller-independent approach for tracking the trajectory of a differential-drive mobile robot. The proposed method is implemented and a fair comparison is carried out with two classical approaches and a recent hybrid prescribed-time controller (HPTC). Furthermore, the gains of the controllers are tuned using a Particle Swarm Optimization algorithm. The simulation results demonstrate that the proposed interception-based strategy provides an improved tracking accuracy and a faster convergence in comparison with the classical trajectory tracking methods.

Keywords: Interception, Trajectory Tracking, Mobile Robots, Sliding Mode Control, Prescribed-time Control, PSO Optimization, Optimal reference trajectory tracking.

1. Introduction

Over the past several decades, mobile robot control has emerged as an active and significant research field, particularly focusing on systems capable of autonomous navigation and task execution. Among the various control objectives, trajectory tracking and target interception remain critical and challenging problems due to the nonlinear and dynamic nature of mobile robots. As a result, extensive research efforts have led to the development of diverse control strategies aimed at achieving a high performance, including fast convergence and accurate tracking with minimal errors. Several studies have explored different control approaches for addressing mobile robot positioning and motion objectives. For instance, Quiroga et al. (2022) investigated the use of reinforcement learning techniques for the position control of a simulated mobile robot, demonstrating their effectiveness in both obstacle-free environments and scenarios involving obstacles. More recently, Park (2025) focused on the problem of driving a mobile robot to a desired posture within a minimum-time framework, proposing a dynamically adaptive proportional–integral–derivative controller optimized by using a genetic algorithm to enhance the control performance.

In order to enhance the trajectory tracking capabilities of a mobile robot, several research works have adopted intelligent control approaches. Trujillo et al. (2023) introduced a controller based on neural networks to improve the tracking behavior. Similarly, Park (2025) and Quiroga et al. (2022) applied artificial intelligence techniques with the objective of reducing the time required for a robot to reach its desired motion or posture, with the genetic algorithm–based tuning of control parameters reported in the former study. And although these researches achieved a satisfactory tracking performance and time reduction, they have not focused on minimizing the convergence error.

In addition to intelligent control techniques, model predictive control (MPC) has been widely adopted for mobile robot control. Bouzoualegh, Guechi, & Kelaiaia (2018), for example, addressed the control of a differential-drive mobile robot by designing a model predictive controller based on the minimization of a quadratic criterion function. Their approach relied on linearizing the mobile robot's nonlinear dynamic model through an input-output linearization framework, enabling the predictive controller to generate suitable control actions.

https://doi.org/10.24846/v35i1y202609

The problems of tracking and intercepting moving targets have also attracted considerable attention in the robotics literature. (Belkhouche and Belkhouche, 2004) introduced an intercept-based control strategy, called guidance approach, that enables a wheeled mobile robot to successfully intercept a moving object when the mobile robot is faster than the moving target. In parallel, sliding mode control has been widely employed to enhance robustness against model uncertainties and external disturbances, as demonstrated in (Bkekri et al., 2019). More recently, Flores-Campos et al. (2024) presented a new interception algorithm for a manipulator robot for intercepting a moving object. To reach the interception objective in a prescribed time, the authors used the combination of a robust second-order sliding mode control with a terminal attractor based on a designed time base generator, while Hameed et al. (2023) developed a nonlinear sliding mode control approach for achieving trajectory tracking for a two-wheeled mobile robot. Although the aforementioned studies emphasized robustness aspects, they did not explicitly address the reduction of convergence or residual tracking errors, which remains an important performance criterion in mobile robot interception tasks. In contrast, Nguyen and Kim (2025) addressed the problem of controlling an uncertain four-wheeled mobile robot affected by wheel-surface interaction phenomena, including WSS (Wheel Slipping and Skidding). Despite the robustness achieved and the convergence error that the authors have mentioned, it remains to be verified whether this strategy works with any controller that can be used.

Song et al. (2023) presents an accessible review of prescribed-time control highlighting the relationships among finite-time, fixed-time predefined-time and prescribed-time control.

This paper addresses the trajectory tracking problem for a differentially driven mobile robot. A novel interception-based trajectory tracking strategy is proposed, in which the desired orientation of the robot is defined toward a predicted virtual reference point. The proposed approach operates at the error-transformation level and is independent of the specific control law. By anticipating the future motion of the reference robot, the strategy aims to improve the convergence speed and tracking accuracy while avoiding online optimization.

The effectiveness of the proposed strategy is evaluated through a simulation using a first-order sliding mode controller and a prescribed-time control scheme. For a fair comparison, controller gains are tuned using a particle swarm optimization (PSO) algorithm. The simulation results under circular reference trajectories demonstrate the advantages of the proposed interception-based strategy in terms of tracking performance and convergence behavior.

The remainder of this paper is organized as follows. Section 2 presents the kinematic and dynamic models of the mobile robot. Section 3 introduces the proposed interception-based strategy and its integration with the considered control schemes. Further on, Section 4 reports and discusses the simulation results. Finally, Section 5 concludes this paper and outlines possible future research directions.

2. Problem Formulation

The following scenario is considered: a mobile robot is positioned within a 2D reference frame, where it is oriented counterclockwise at a certain angle relative to the X-axis. As shown in Figure 1, the robot's posture errors are represented by its linear position errors (x, y) and its orientation error denoted by the angle θ, all illustrated in the global coordinate frame OXY.

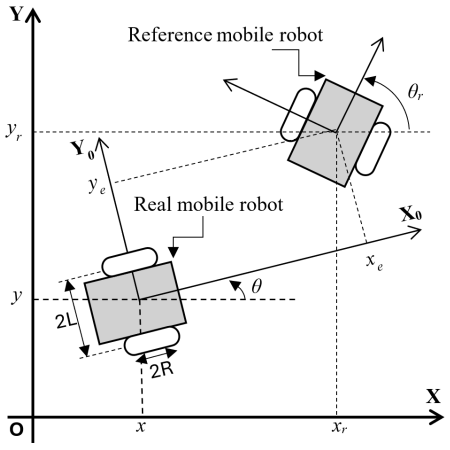

Figure 1. Mobile robot posture errors

The kinematic model of the mobile robot is defined as:

$$\dot{q} = S(q)V \tag{1}$$

where $q = (x, y, \theta)^T$ is the real robot posture, $S(q) = \begin{bmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{bmatrix}$ is a Jacobian matrix, and $V = [v \quad \omega]^T$ is the robot velocity control input

vector with v being the linear velocity and ω the angular velocity.

The vectors of the desired posture and the desired velocity are expressed as:

$$q_d = (x_d, y_d, \theta_d)^T,\ V_d = (v_d, \omega_d)^T \tag{2}$$

The posture error in the robot's local frame is defined as follows:

$$q_e = (x_e, y_e, \theta_e)^T \tag{3}$$

where:

$$\begin{cases} x_e = cos\theta(x_d - x) + sin\theta(y_d - y) \\ y_e = -sin\theta(x_d - x) + cos\theta(y_d - y) \\ \theta_e = \theta_d - \theta \end{cases} \tag{4}$$

$$q_e = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix} \tag{5}$$

$$q_e = \Re(\theta) \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix}.$$

with $\Re(\theta)$ being the orthogonal rotation matrix.

The nonholonomic constraint governing the slip-free rolling of the wheel on the ground can be expressed as follows:

$$\dot{x}sin\theta + \dot{y}cos\theta = 0 \tag{6}$$

Using equations (1) and (6), the derivative of equation (5) is given as follows:

$$\dot{q}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} y_e\omega - v + v_d cos\theta_e \\ -x_e\omega + v_d sin\theta_e \\ \omega_d - \omega \end{bmatrix} \tag{7}$$

To achieve a realistic simulation of the mobile robot, both the robot's dynamic model and a corresponding dynamic controller are incorporated into the block diagram of the control strategy as shown in Figure 2:

$$\begin{cases} \left(m + \frac{2}{R^2} I_w\right)\dot{v} - m_c d\omega^2 = \frac{1}{R}(\tau_R + \tau_L) \\ \left(I + \frac{2L^2}{R^2} I_w\right)\dot{\omega} + m_c d\omega v = \frac{L}{R}(\tau_R - \tau_L) \end{cases} \tag{8}$$

where

$[\tau_R, \tau_L]^T$ is the torque input to the mobile robot, L represents half of the track width, R is the wheel radius, d is the distance from the center of mass to the middle point between the right and left wheel, m and m_c denote the total mass of the robot and the robot mass without its wheels and actuators, respectively, I_w denotes the moment of inertia of each wheel and I represents the total moment of inertia.

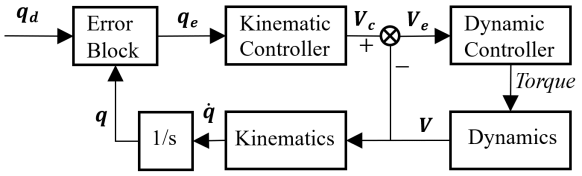

Figure 2. Closed loop of the controlled system

In practical operating conditions, a perfect velocity tracking cannot be achieved. Consequently, the dynamic controller introduces auxiliary velocity tracking errors, denoted by $V_e = [v_e \ \omega_e]^T$, which are assumed to be bounded by known constants. These residual errors can be interpreted as lumped uncertainties and external disturbances acting on the kinematic model.

As a result, the closed-loop kinematic model can be expressed as follows:

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} (v_c + v_e)cos\theta \\ (v_c + v_e)sin\theta \\ (\omega_c + \omega_e) \end{bmatrix} \tag{9}$$

where $V_c = [v_c \ \omega_c]^T$ denotes the velocity control input. After straightforward algebraic manipulations, the effect of the auxiliary tracking error appears as an additional term on the right-hand side of the error dynamics expression:

$$\dot{q}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} y_e(\omega_c + \omega_e) - (v_c + v_e) + v_d cos\theta_e \\ -x_e(\omega_c + \omega_e) + v_d sin\theta_e \\ \omega_d - (\omega_c + \omega_e) \end{bmatrix} \tag{10}$$

3. The Proposed Approach and Main Results

3.1 Optimal Reference Trajectory Tracking

This subsection introduces the proposed desired orientation angle, which is derived based on the principles of interception and trajectory tracking.

Numerous studies have contributed to advancements in mobile robot control, particularly those focusing on trajectory tracking performance (e.g. Goto & Martins, 2024; Lin et al., 2021). These works vary significantly in their approaches

to calculating posture errors, with a particular distinction regarding the methods used for defining the desired orientation angle.

Some authors consider the desired angle as the orientation of the reference mobile robot (e.g. Lin et al., 2021):

$$\theta_{d1} = \theta_r \tag{11}$$

$$\theta_{e1} = \theta_r - \theta \tag{12}$$

Others consider the desired angle as the direction pointing towards the reference mobile robot (e.g. Cao et al., 2024; Hassan, Abed & Al-Hussaibi, 2023):

$$\theta_{d2} = \theta_{TR} = tan^{-1}\left(\frac{y_d - y}{x_d - x}\right) \tag{13}$$

"*atan2*" can be used to return the four-quadrant inverse tangent, as follows:

$$\theta_{d2} = \theta_{TR} = atan2((y_d - y),(x_d - x)) \tag{14}$$

$$\theta_{e2} = \theta_{TR} - \theta \tag{15}$$

The proposed approach defines the desired orientation angle as the direction pointing toward a virtual mobile robot, $\theta_{d3} = \theta_{TVR}$.

Figure 3 illustrates the proposed approach, where the virtual mobile robot represents the predicted position of the reference robot after a time period Δt; more specifically, Δt denotes the minimum-time interception of the moving reference mobile robot such that $\lim_{t \to \Delta t} q_e = 0$.

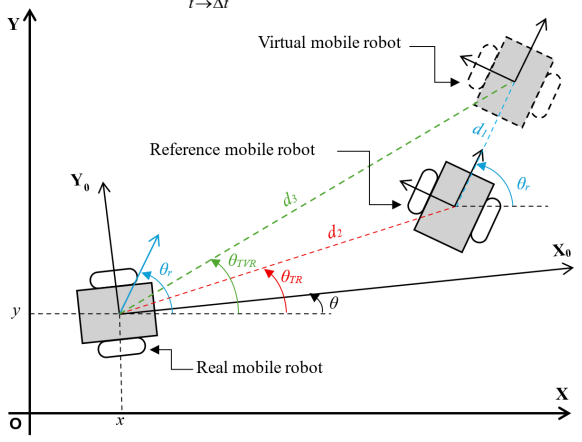

Figure 3. The proposed strategy: $\theta_{d3} = \theta_{TVR}$

Assumption: the linear velocities of the actual mobile robot and of the reference mobile robot are positive.

Further on, the following terms are defined:

$d_1 = v_d \Delta t$ - the distance between the reference mobile robot and its predicted position;

$d_2 = \sqrt{(y_d - y)^2 + (x_d - x)^2}$ - the distance between the reference mobile robot and the actual mobile robot;

$d_3 = v\Delta t$ - the distance between the virtual mobile robot and the actual mobile robot.

$$\theta_{d3} = \theta_{TVR} = \theta_{TR} + \theta_p \tag{16}$$

where θ_p is the angle resulting from the prediction.

So, it is readily seen that by applying a rotation of angle $-\theta_{TR}$ in Figure 3, the configuration shown in Figure 4 is obtained.

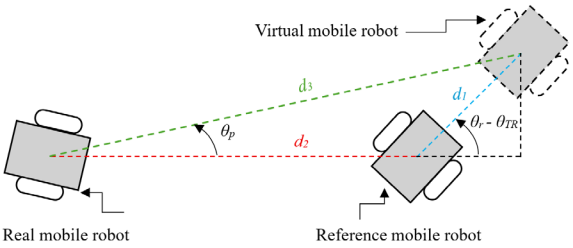

Figure 4. Rotation of $-\theta_{TR}$ in Figure 3

Then, the following is obtained:

$$\theta_p = \sin^{-1}\left(\frac{v_d}{v} sin(\theta_r - \theta_{TR})\right) \tag{17}$$

where $v \neq 0$ and $-1 \leq sin(\theta_r - \theta_{TR}) \leq 1$.

The Inverse Sine "$\sin^{-1}$" returns one value of θ_p in the interval $[-\pi/2, \pi/2]$, however there are two solutions: θ_p or $\pi - \theta_p$.

$$|\theta_p| \in [0, |\theta_r - \theta_{TR}|]$$

If $\theta_p \geq 0$ then $\theta_{TR} \leq \theta_d \leq \theta_r$

If $\theta_p \leq 0$ then $\theta_r \leq \theta_d \leq \theta_{TR}$

If $\theta_p = 0$ then $\theta_d = \theta_r = \theta_{TR}$

According to Figure 4, the following is obtained:

$$d_3 \cos(\theta_p) = d_1 \cos(\theta_r - \theta_{TR}) + d_2$$

$$\cos(\theta_p) V \Delta t = \cos(\theta_r - \theta_{TR}) v_d \Delta t + d_2$$

$$\Delta t = \frac{d_2}{cos(\theta_p) v - cos(\theta_r - \theta_{TR}) v_d}$$

$\Delta t \geq 0$ leads to:

$$\cos(\theta_p) v - \cos(\theta_r - \theta_{TR}) v_d > 0$$

$$\cos(\theta_p) v > \cos(\theta_r - \theta_{TR}) v_d$$

The angle error $\theta_{e3} = \theta_{TVR} - \theta$ can also be expressed as:

$$\theta_{e3} = atan2((y_d - y),(x_d - x)) + sin^{-1}\left(\frac{v_d}{v} sin(\theta_r - \theta_{TR})\right) - \theta \tag{18}$$

The angle error is minimized when $|\theta_{e3}| > \pi$ to avoid the long path when the mobile robot rotates.

If $\theta_{e3} > \pi$ then $\theta_{e3min} = \theta_{e3} - 2\pi$ (19)

If $\theta_{e3} < -\pi$ then $\theta_{e3min} = \theta_{e3} + 2\pi$ (20)

So, $\theta_{e3min} \in [-\pi, \pi]$.

In the proposed framework, the error transformation block is explicitly decoupled from the control law design. This modular structure enables the proposed approach to be combined with a wide range of control laws and mobile robot models.

3.2 Kinematic Controller

Comparative simulations are conducted using different posture error-based strategies in combination with a First-Order Sliding Mode Controller (FOSMC) and a prescribed-time controller.

Figure 5 depicts the overall simulation flowchart, illustrating the complete control loop and highlighting how the posture error is computed at each time step using the proposed desired orientation angle.

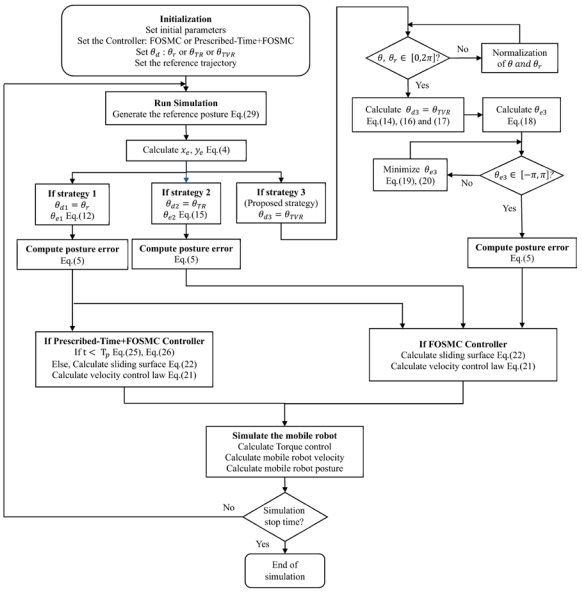

Figure 5. The overall simulation flowchart

3.3 First-Order Sliding Mode Controller (FOSMC)

The trajectory tracking control law proposed by Martins & Bertol (2022) utilizes a first-order sliding mode control-based approach, incorporating a non-singular coordinate transformation. This method enhances stability and control in proximity to the target by preventing singularities typically encountered with regard to Cartesian coordinates.

Thus, the control law is expressed as:

$$V_c = \begin{bmatrix} v_c \\ w_c \end{bmatrix} = -B_{0_\sigma}^{-1} A_{0_\sigma} - \mathrm{G}\, sat(\sigma^*) \tag{21}$$

where:

$$\begin{aligned} A_{0_\sigma} &= \frac{\partial \sigma}{\partial q_e} A_0 = \begin{bmatrix} \lambda_1 v_d cos(\theta_e) \\ \lambda_2 v_d sin(\theta_e) + \lambda_3 \omega_d \end{bmatrix} \\ B_{0_\sigma} &= \frac{\partial \sigma}{\partial q_e} B_0 = \begin{bmatrix} -\lambda_1 & \lambda_1 y_e \\ 0 & -\lambda_2 x_e - \lambda_3 \end{bmatrix} \\ \sigma^*(q_e, t) &= \begin{bmatrix} \sigma_1^* \\ \sigma_2^* \end{bmatrix} \\ \sigma^*(q_e, t) &= \begin{bmatrix} -\lambda_1^2 x_e \\ \lambda_1^2 x_e y_e - (\lambda_2 x_e + \lambda_3)(\lambda_2 y_e + \lambda_3 \theta_e) \end{bmatrix} \end{aligned} \tag{22}$$

The 'sign' function was substituted, in (Mohamed, Mahjoub & Benamor, 2025), with a saturation function in order to mitigate the chattering effect, where "$sat(\sigma^*)$" is component-wise discontinuous and defined as:

$$sat(\sigma^*) = \begin{bmatrix} \mathrm{sat}(\sigma_1^*) & \mathrm{sat}(\sigma_2^*) \end{bmatrix}^{\mathrm{T}}$$

$$sat(\sigma_i^*, \phi) = \begin{cases} \dfrac{\sigma_i^*}{\phi}, & \text{if } |\sigma_i^*| \le \phi \\ sign(\sigma_i^*), & \text{if } |\sigma_i^*| > \phi \end{cases}$$

where ϕ is the boundary layer thickness and $i = 1,2$.

$$G = diag([g_1 \;\; g_2]) \tag{23}$$

The condition below must be satisfied to guarantee that B_{0_σ} is always nonsingular:

$$\lambda_2 = \lambda_3 \kappa \text{, with } 0 \le \kappa \le \frac{1}{|x_e| + 1} \tag{24}$$

3.4 Hybrid Prescribed-Time Controller (HPTC)

In (Nasab, Badiei & Asemani, 2025), the authors proposed a prescribed-time controller to achieve a desired trajectory in a predefined time to ensure error dynamics convergence to an equilibrium point for wheeled mobile robots. Afterward, they used a proportional controller enforcing the system errors to remain at the equilibrium point after the prescribed time t_p.

In this work, this proportional action is replaced by a first-order sliding mode controller (FOSMC) (equation 21), the gains of which are optimized using PSO. The resulting hybrid scheme preserves the prescribed-time controller behavior while benefiting from the robustness properties of sliding mode control.

For $t \in [0, t_p]$ the control law is expressed as:

$$v_c = v_r \cos(\phi_e) + \frac{y_e\left(v_r \sin(\phi_e) - \frac{\gamma_1 y_e}{t - t_p}\right)}{x_e} - \frac{\gamma_2 x_e}{t - t_p} \quad (25)$$

$$\omega_c = \frac{v_r \sin(\phi_e) - \frac{\gamma_1 y_e}{t - t_p}}{x_e} \quad (26)$$

where γ_1, γ_2 are given positive constants and t_p is the user-defined prescribed time, so that:

$\lim_{t \to t_p} x_e = 0, \lim_{t \to t_p} y_e = 0$ and $\lim_{t \to t_p} \phi_e = 0$.

3.5 Optimal Controllers

In this subsection, a MATLAB script is implemented using the built-in "particleswarm" Matlab function to determine the optimal controller gain values within the Simulink framework.

Simulink served as the platform for designing the mobile robot control architecture, as its block-oriented structure enables model-driven simulations and enhanced capabilities for system analysis and debugging.

The tuning parameters of the sliding mode and prescribed-time controller have a direct impact on the effectiveness of the control strategy in achieving a rapid convergence and an accurate trajectory tracking. To optimize the controller parameters, the Particle Swarm Optimization (PSO) algorithm is adopted in this study, where it systematically adjusts the control gains by minimizing a predefined objective function. An Integral Time Squared Error (ITSE) criterion, formulated using the posture error vector, is chosen as the objective function and it can be expressed as follows:

$$J_{ITSE} = \int t q_e^2 dt \quad (27)$$

Each particle in the swarm repesents a potential solution, whose position encodes a particular combination of controller gain parameters. Throughout the optimization procedure, the particles traverse the search space with assigned velocities and they iteratively update their positions. As a result of this collective behavior, the swarm progressively converges toward the global optimum.

The optimization procedure consists of several main steps. First, a population of particles is randomly initialized within predefined parameter bounds, and both the personal best position of each particle and the global best position of the swarm are recorded. At each iteration, particle velocities are updated according to the following equation:

$$v_i = W \cdot v_i + y_1 \cdot u_1 \cdot (p_i - x_i) + y_2 \cdot u_2 \cdot (g - x_i) \quad (28)$$

where W denotes the inertia weight, y_1 and y_2 are the cognitive and social acceleration coefficients, and $u_1, u_2 \in (0,1)$ are uniformly distributed random variables. The particle positions are then updated by using $x_i = x_i + v_i$ and boundary constraints are enforced for keeping all the parameters within admissible ranges. The updated positions are evaluated by using the cost function, and the personal and global best solutions are updated accordingly. To improve the convergence performance, the inertia weight is adaptively adjusted based on the evolution of the global best solution. The algorithm iterates until a predefined stopping criterion, such as a maximum number of iterations, is satisfied.

By performing the offline optimization of the controller gains using PSO, the proposed control system demonstrates an enhanced tracking accuracy and an accelerated convergence, while preserving a low computational complexity during robot simulation.

4. Validation and Comparative Study

To demonstrate the effectiveness and advantages of the proposed control strategy, a simulation study was conducted on circular trajectory tracking, thereby validating the performance of the adopted approach.

The values for the initial robot states x, y and θ are 6m, 2m, and π, respectively. The initial states

of the reference robot are $x_d = x_r = 4m$, $y_d = y_r = 0$ and $\theta_r = \pi/2$.

The expected circular trajectory can be expressed as:

$$\begin{cases} x_r = v_d cos(\omega_d t) \\ y_r = v_d sin(\omega_d t) \\ \theta_r = \omega_d t \end{cases} \quad (29)$$

where v_d=4m/s and ω_d=1rad/s

It is assumed that the center of mass of the mobile robot is located at the midpoint between the left and right wheel (d=0), and that the moment of inertia of each wheel is neglected (I_w=0). The robot parameters listed in Table 1 are employed in the simulation study.

Table 1. Robot parameters

L (m)	**R** (m)	**m** (Kg)	**I** (Kg·m^2)
0.15	0.03	4.5	3

The parameters of the kinematic controllers included in Table 2 remain constant throughout the simulation experiments and their values were chosen as follows:

Table 2. Constant parameters of the FOSMC and prescribed-time controller

FOSMC controller			
$\phi = 0.5$	$\lambda_1 = 1$	$\lambda_2 = 6$	$\lambda_3 = 1$
Prescribed-time controller			
$\phi_e = 0.2$			

Gain tuning was performed in two stages. First, the sliding mode controller gains were optimized for each posture error-based strategy. Then, for the hybrid prescribed-time scheme, one posture error-based strategy, namely $\theta_{d1} = \theta_r$ was used, then both the sliding mode gains and the prescribed-time parameters were optimized.

Tables 3 and 4 report the optimized gain values obtained, along with the corresponding cost function values. The proposed method exhibits a superior efficiency with respect to the objective function, achieving a minimum value of 0.0588. Based on this result, the proposed method significantly outperforms both the first and second classical strategies, and also the Hybrid Prescribed-Time Controller, for which the corresponding cost values were 0.1152, 0.4175 and 0.079, respectively.

Table 3. Optimal gains for the FOSMC controller

Desired orientation angle	g_1	g_2	J_{ITSE}
$\theta_{d1} = \theta_r$	32.5166	44.9584	0.1152
$\theta_{d2} = \theta_{TR}$	0.9444	82.9444	0.4175
Proposed $\theta_{d3} = \theta_{TVR}$	13.1741	32.7398	0.0588

Table 4. Optimal gains for the HPTC with $\theta_{d1} = \theta_r$

Υ_1	Υ_2	t_p	J_{ITSE}
0.01	0.01	0.001	0.0790
g_1	g_2		
43.514	92.6234		

To evaluate the robustness of the proposed approach, bounded perturbations were injected into the linear and angular velocity signals of the actual mobile robot during the time interval from 3 s to 4 s, and they are expressed as follows:

$$\begin{aligned} \Delta v(t) &= 4\sin(2t - \pi) \\ \Delta\omega(t) &= 3\sin(1t - \pi) \end{aligned} \quad (30)$$

Figure 6 illustrates the X–Y trajectory tracking responses of the considered controllers under velocity perturbations, showing that stable tracking is maintained despite the injected disturbances by using the proposed strategy.

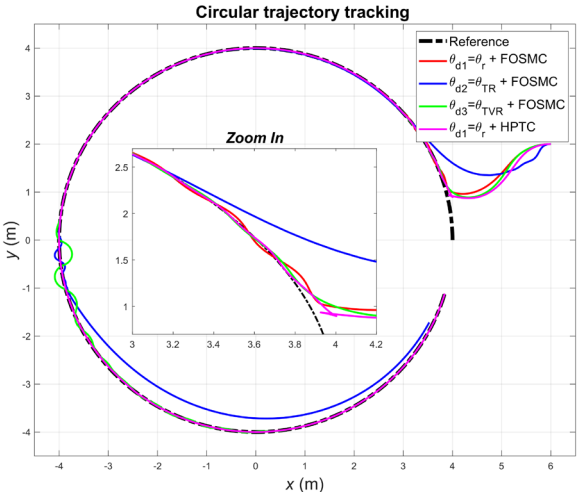

Figure 6. Circular trajectory tracking

The convergence times of the posture error $q_e = (x_e, y_e, \theta_e)^T$ to its steady-state value are extracted from Figures 7 and 8 and included in Table 5.

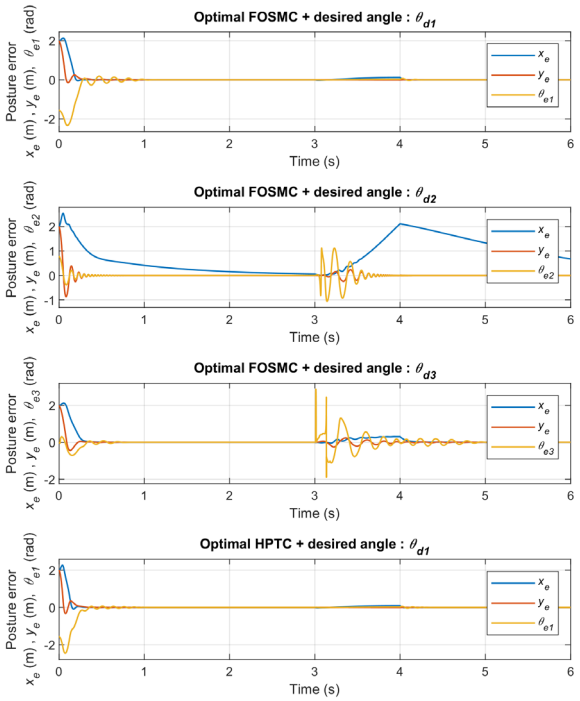

Figure 7. Posture error

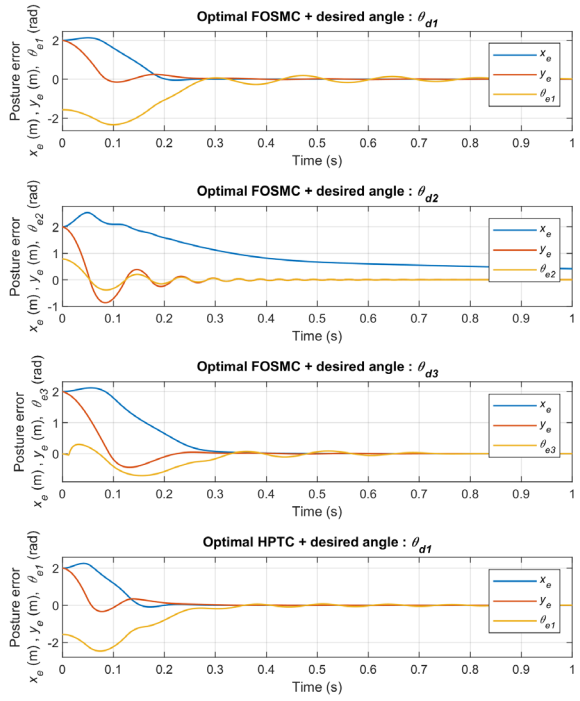

Figure 8. Posture error - Zoom

During the initial phase of the simulation, as illustrated in Figure 8, the residual errors reached the steady-state neighborhood significantly faster for the proposed method than for the other three competing methods.

As shown in Table 5, the proposed method converged in 0.7s, while the second strategy achieved the lowest convergence time.

Table 5. Convergence time (T_c) in seconds

Desired orientation angle + controller	T_c (s)
$\theta_{d1} = \theta_r$ + FOSMC	1.2
$\theta_{d2} = \theta_{TR}$ + FOSMC	3
Proposed $\theta_{d3} = \theta_{TVR}$ + FOSMC	0.7
$\theta_{d1} = \theta_r$ + HPTC	0.8

Also, as shown in Figure 7, the injection of a velocity perturbation revealed a distinct trade-off in the proposed strategy architecture.

While the proposed strategy maintained the reference trajectory, the posture error figures show a more pronounced deviation in comparison with the first strategy and the prescribed-time control strategy.

During the transient phase, as it is clear in Figure 9, the optimal FOSMC-based strategy ($\theta_{d3} = \theta_{TVR}$) exhibits fewer oscillation around the reference trajectory.

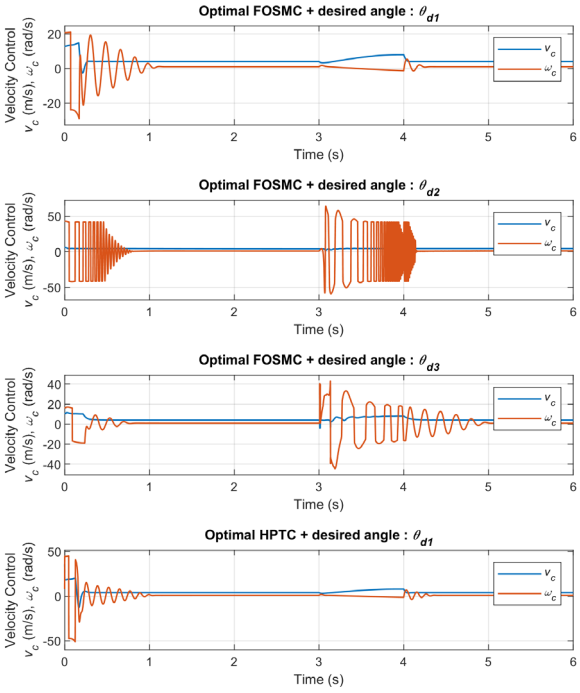

Figure 9. Velocity control signals

In summary, the proposed strategy is the most effective for applications requiring a rapid, smooth, and energy-efficient initial tracking. While the prescribed-time controller provides a superior "hardness" against perturbations, the proposed method provides a superior balance between fast convergence and reduced oscillations in the presence of disturbances. This makes

it particularly suitable for applications where steady-state precision and a smooth start-up are prioritized over high-impact disturbance rejection.

5. Conclusion

This study presented a novel interception-based trajectory tracking control scheme for differential-drive mobile robots, along with an innovative gain optimization approach that effectively integrates the global optimization capabilities of the particle swarm optimization (PSO) algorithm implemented in MATLAB with the block-based modeling framework provided by Simulink. The obtained results demonstrated that the proposed method achieved a substantial reduction in the convergence time, while markedly attenuating the control signal oscillations during the transient phase in comparison with three other conventional approaches from the literature. These improvements were primarily attributed to the proposed robot trajectory tracking scheme and the structured gain optimization strategy, which constitute the main contributions of this work. Future research will focus on conducting a real-time experimental validation of this approach on an embedded control platform in order to assess its robustness against physical disturbances and unmodeled dynamics. Moreover, extending the proposed framework by incorporating adaptive or learning-based schemes is anticipated to accommodate a wider range of operating conditions while preserving a high tracking performance.

Acknowledgement

This work was supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R847), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

REFERENCES

Belkhouche, F. & Belkhouche, B. (2004) On the tracking and interception of a moving object by a wheeled mobile robot. In: *IEEE Conference on Robotics, Automation and Mechatronics*, 2004, 1-3 December 2004, *Singapore*. New York, USA, IEEE. pp. 130–135.

Bkekri, R., Benamor, A., Alouane, M.A. et al. (2019) Robust adaptive super twisting controller: methodology and application of a human-driven knee joint orthosis. *Industrial Robot: the international journal of robotics research and application*. 46 (4), 481–489. https://doi.org/10.1108/IR-09-2018-0198.

Bouzoualegh, S., Guechi, E.-H. & Kelaiaia, R. (2018) Model Predictive Control of a Differential-Drive Mobile Robot. *Acta Universitatis Sapientiae Electrical and Mechanical Engineering*. 10(1), 20–41. https://doi.org/10.2478/auseme-2018-0002.

Cao, T.N.T., Pham, B.T., Tran, H.D et al. (2024) Non-singular terminal sliding mode control for trajectory-tracking of a differential drive robot. In: Truong, N.V. & Ha, Q. (eds.). *International Conference on Energy, Infrastructure and Environmental Research EIER 2024*, Les Ulis, France. E3S Web of Conferences. Vol. 496. EDP Sciences. https://doi.org/10.1051/e3sconf/202449602005.

Flores-Campos, J.A., Torres-San-Miguel, C.R., Paredes-Rojas, J.C. et al. (2024) Prescribed Time Interception of Moving Objects' Trajectories Using Robot Manipulators. *Robotics*. 13(10), Art. ID 145. https://doi.org/10.3390/robotics13100145.

Goto, W.J.N. & Martins, N.A. (2024) Leader-follower formation tracking for differential-drive wheeled mobile robots with uncertainties and disturbances based on immune fuzzy quasi-sliding mode control. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*. 46(2), Art. ID 91. https://doi.org/10.1007/s40430-023-04650-8.

Hameed, I.A., Abbud, L.H., Abdulsaheb, J.A. et al. (2023) A New Nonlinear Dynamic Speed Controller for a Differential Drive Mobile Robot. *Entropy*. 25(3), 514. https://doi.org/10.3390/e25030514.

Hassan, I.A., Abed, I.A. & Al-Hussaibi, W.A. (2023) Path Planning and Trajectory Tracking Control for Two-Wheel Mobile Robot. *Journal of Robotics and Control* (*JRC*). 5(1), 1–15. https://doi.org/10.18196/jrc.v5i1.20489.

Lin, L., Wu, P., He, B. et al. (2021) The sliding mode control approach design for nonholonomic mobile robots based on non-negative piecewise predefined-time control law. *IET Control Theory & Applications*. 15(9), 1286–1296. https://doi.org/10.1049/cth2.12122.

Martins, N.A. & Bertol, D.W. (eds.) (2022) *Wheeled mobile robot control: Theory, Simulation, and Experimentation* (*Studies in Systems*, *Decision and Control*, vol. 380). Cham, Switzerland, Springer.

Mohamed, S. B. H., Mahjoub, A. & Benamor, A. (2025) Optimal Sliding Mode Control for Differential Drive Mobile Robots. In: 2025 *International Conference on Control, Automation and Diagnosis* (*ICCAD*), *1-3 July 2025, Barcelona, Spain*. New York, USA, IEEE. 10.1109/ICCAD64771.2025.11182005.

Nasab, A.A., Badiei, R. & Asemani, M.H. (2025) Safe prescribed time controller for wheeled mobile robots by using control barrier functions as a safety filter. *ISA Transactions*. 162, 119–133. https://doi.org/10.1016/j.isatra.2025.04.024.

Nguyen, V.-C. & Kim, S.H. (2025) A novel fixed-time prescribed performance sliding mode control for uncertain wheeled mobile robots. *Scientific Reports*. 15 (1), Art. ID 5340. https://doi.org/10.1038/s41598-025-89126-6.

Park, J.-H. (2025) Optimal Controller Design for a Mobile Robot Using Genetic Algorithm and Adaptive PID Controller. *IEEE Access*. 13, 86167–86184. https://doi.org/10.1109/ACCESS.2025.3570472.

Quiroga, F., Hermosilla, G., Farias, G. et al. (2022) Position Control of a Mobile Robot through Deep Reinforcement Learning. *Applied Sciences*. 12(14), Art. ID 7194. https://doi.org/10.3390/app12147194.

Song, Y., Ye, H. & Lewis, F. L. (2023) Prescribed-Time Control and Its Latest Developments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. 53(7), 4102–4116. https://doi.org/10.1109/TSMC.2023.3240751.

Trujillo, D., Morales, L.A., Chávez, D. et al. (2023) Trajectory Tracking Control of a Mobile Robot using Neural Networks. *Emerging Science Journal*. 7(6), 1843–1862. https://doi.org/10.28991/ESJ-2023-07-06-01.