IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Journal Européen des Systèmes Automatisés
Vol. 59, No. 4, April, 2026, pp. 925-941
Journal homepage: http://iieta.org/journals/jesa

Traction-Adaptive Super-Twisting Sliding Mode Control for Mecanum Robotaxi Tracking under Wheel-Speed Constraints: Modeling, Estimation, and Benchmarking

Mohammed Ali Alfayizi

Department of Electrical Engineering, College of Engineering, Qassim University, Buraydah 52571, Saudi Arabia

Corresponding Author Email: m.alfayizi@qu.edu.sa

https://doi.org/10.18280/jesa.590406

Received: 25 January 2026
Revised: 20 April 2026
Accepted: 28 April 2026
Available online: 30 April 2026

Keywords:
Mecanum-wheeled mobile robot, omnidirectional kinematics, sliding mode control, super-twisting algorithm, adaptive control systems, traction estimation, actuator saturation, trajectory tracking

ABSTRACT

This work investigates constraint-aware trajectory tracking for a Mecanum-wheeled mobile robot motivated by robotaxi-style operation in smart-city environments. Owing to its omnidirectional mobility, the platform is well-suited to low-speed maneuvering and curbside docking in constrained service areas, but it is also sensitive to traction variation and slip. To evaluate these effects, a benchmark-oriented comparative study is conducted over five scenarios: CurbDocking, RoundedSquareLoop, FrictionMap, PayloadEvent, and DockingSlipShock. A traction-adaptive super-twisting sliding mode controll (ASTSMC) with online traction estimation is compared against a baseline Proportional-Integral-Derivative (PID) inner-loop velocity controller under realistic wheel-speed limits. MATLAB simulation results show a clear robustness-precision tradeoff: ASTSMC improves robustness to traction variations and provides more reliable online traction estimation during service motion, whereas the tuned PID baseline achieves tighter terminal docking precision under strict wheel-speed constraints. These results support traction-adaptive robust control as a practical candidate for omnidirectional mobile robots operating on variable-surface conditions.

1. INTRODUCTION

The rapid development of autonomous transportation, on-demand mobility services, and smart-city infrastructure has heightened interest in mobile robotic platforms capable of operating safely and precisely in constrained urban environments. While most public attention has focused on car-like autonomous vehicles, low-speed service platforms with omnidirectional mobility are also highly relevant for last-mile transportation, curbside pickup assistance, indoor-outdoor logistics, and precision docking tasks in dense service areas. In these applications, the ability to translate laterally without reconfiguring the steering is a major advantage. For this reason, Mecanum-wheeled mobile robots have attracted growing interest as candidate platforms for compact robotaxi-style or service-mobility systems operating in narrow, cluttered spaces [1-7].

Despite this maneuverability advantage, Mecanum platforms remain challenging to control accurately in practice. Their omnidirectional motion relies on roller-ground interaction, and the resulting body motion is sensitive to load transfer, changes in floor or road conditions, micro-slip, actuator nonlinearities, and mismatches between nominal wheel kinematics and effective platform motion [8-14]. These effects are especially important in realistic service situations, such as curb approach, docking near painted or polished surfaces, payload changes, and localized low-friction regions. Consequently, a controller designed only from nominal kinematics may provide acceptable tracking in mild conditions but may lose repeatability, robustness, or terminal precision when traction conditions vary during motion.

The literature on wheeled and omnidirectional mobile robots offers several useful directions, but each addressed only part of this practical problem. Classical works established the structural and kinematic foundations of wheeled mobile robots and provided the modeling tools needed to describe holonomic and omnidirectional motion [8-12]. More recent comparative studies have emphasized that Mecanum and omniwheel systems exhibit performance limitations strongly influenced by wheel geometry, contact quality, and implementation details [13, 14]. These studies underscore the need for control laws that remain effective when actual motion deviates from the nominal wheel-speed-to-body-velocity mapping.

A large class of practical mobile-robot controllers still relies on Proportional-Integral-Derivative (PID)-family designs for their simplicity, low implementation burden, and direct tunability [15]. For Mecanum systems, PID and fuzzy-PID variants remain attractive in embedded applications and can achieve good performance when operating conditions remain close to nominal. However, their robustness may degrade when traction changes quickly or when actuator feasibility becomes the dominant factor near aggressive maneuvers or terminal docking. In parallel, more advanced methods such as neural tracking control, sensor-fusion-aided path following, and predictive control have been developed to improve trajectory tracking under uncertainty or sensing imperfections. For example, neural control has been used to improve Mecanum tracking performance [16], and sensor-fusion-based path

following has been shown to improve state estimation under nonuniform measurements and practical nonlinearities such as saturation and wheel sliding [17]. Observer-based predictive control has also recently been used to combine disturbance estimation with explicit handling of physical constraints in Mecanum robots, while sliding-mode-observer-based model predictive tracking has further highlighted the value of estimation-enhanced robust control in this class of systems [18]. These studies are valuable, but they generally emphasize estimation, optimization, or tracking accuracy rather than a benchmark-oriented study of traction adaptation under strict wheel-speed feasibility constraints. In contrast, the present work combines online traction estimation, adaptive super-twisting control, and wheel-speed-feasible command projection within a common benchmark framework against a practically relevant PID baseline. Robust nonlinear control is another important line of work. Sliding mode control (SMC) is well known for its invariance to matched uncertainties and disturbances [19-21]. Higher-order variants, especially the super-twisting algorithm, are attractive because they preserve strong robustness while reducing the chattering associated with first-order switching laws [22-24]. Adaptive-gain super-twisting methods further extend this idea by allowing the control gains to change online when the uncertainty level is unknown or time-varying [25, 26]. For mobile robots, observer-based and sliding-mode-based methods have shown strong disturbance rejection and robustness to modeling errors [27-28]. Recent studies also continue to explore sliding-mode-based control design for Mecanum omnidirectional robots, including optimal-control-oriented SMC formulations that evaluate trajectory error, execution time, and energy-related performance [29]. Nevertheless, for Mecanum-wheeled systems, relatively few studies examine the combined problem of traction variability represented directly in the body-velocity channels, online traction estimation for slip-aware control adaptation, actuator-feasible implementation under wheel-speed bounds, and standardized comparison against a practically meaningful baseline across multiple service scenarios.

This gap is important because omnidirectional service robots operate in a regime where robustness and precision must be evaluated together rather than separately. A controller may show strong disturbance rejection in unconstrained simulations yet still lose docking accuracy when wheel-speed limits become active. Conversely, a simple baseline controller may achieve good terminal accuracy under mild conditions but offer weaker resilience to traction variations during route-following motion. Therefore, from an application perspective, the relevant question is not whether one controller is universally superior, but rather how different controller structures trade off robustness, traction awareness, control aggressiveness, and terminal accuracy under the same physical constraints.

Motivated by this issue, this paper investigates a two-loop control architecture for a four-Mecanum-wheeled mobile robot designed for low-to-moderate-speed robotaxi-style operation in smart-city service environments. The outer loop generates pose-to-velocity references, while the inner loop is implemented in two alternative forms for fair comparison: a baseline PID body-velocity controller and a traction-adaptive super-twisting sliding mode controll (ASTSMC) with online traction estimation. In the proposed formulation, traction variation is modeled as channel-wise multiplicative uncertainty in the feasible body-velocity input, and the estimated traction level is used to adapt the super-twisting gains. To preserve implement ability, all controller-generated commands are mapped through wheel-speed constraints and projected back to feasible body motion.

The main contributions of this work are as follows. First, a slip-aware ASTSMC framework is formulated for a Mecanum platform, with explicit enforcement of wheel-speed feasibility and online traction estimation. Second, a benchmark-oriented evaluation protocol is constructed so that both the proposed ASTSMC and the PID baseline operate under the same outer-loop reference generator, actuator limits, and scenario definitions. Third, five representative service scenarios are considered, including curb docking, repeated curvature tracking, spatially varying friction, payload-related traction change, and terminal slip shock. This enables quantification of not only tracking error but also terminal docking performance, traction-estimation quality, saturation statistics, and control activity. The resulting comparison reveals a practically important robustness-precision tradeoff: ASTSMC improves robustness and traction awareness during traction-varying service motion, whereas the tuned PID baseline can remain preferable for the tightest terminal docking tasks when the motion becomes strongly constraint-dominated.

The remainder of the paper is organized as follows. Section 2 presents the mathematical model, the wheel-speed feasibility mapping, the slip-affected body-velocity representation, and the traction estimator. Section 3 details the outer-loop reference generation and the proposed ASTSMC inner-loop design. Section 4 discusses boundedness, feasibility, and a Lyapunov-based interpretation of the proposed controller. Section 5 introduces the benchmark suite and evaluation protocol. Section 6 presents the simulation results. Section 7 discusses their practical implications. Section 8 concludes the paper and outlines future work.

2. MATHEMATICAL MODEL

This section presents the mathematical model of the four-Mecanum-wheeled robot used for control design and benchmarking. It introduces planar rigid-body kinematics, wheel-speed mapping, actuator-feasibility projection, slip-affected body-velocity model, and the online traction estimator. For clarity, the main symbols are defined at first use, and a concise nomenclature is provided at the end of the paper.

2.1 Frames, pose, and rigid-body kinematics

Let $\{I\}$ denote the inertial (world) frame and $\{B\}$ the robot body frame attached to the platform center. The planar pose is

$$q = [x \quad y \quad \theta]^\top \tag{1}$$

where, q $\in \mathbb{R}^3$, x and y are the inertial-frame coordinates in meters, and θ is the yaw angle in radians. The body-frame velocity is

$$v = [v_x \quad v_y \quad \omega]^\top \tag{2}$$

where, $v \in \mathbb{R}^3$, v_x and v_y are translational velocities expressed in $\{B\}$ in m/s, and ω is the yaw rate in rad/s. The rigid-body planar kinematics are given by [9]

$$\dot{q} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} v \tag{3}$$

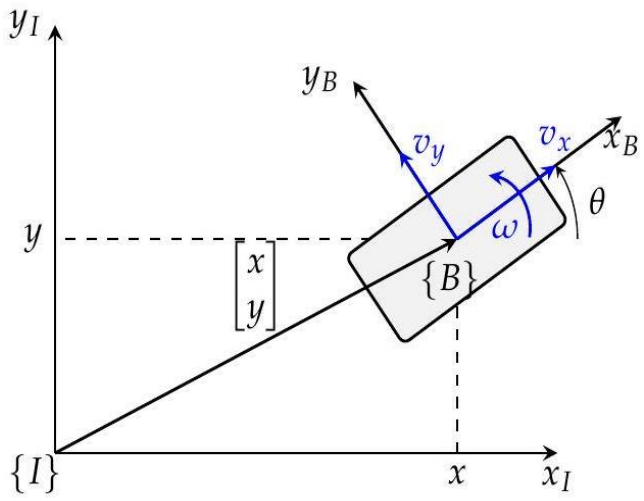

Figure 1. Planar pose and reference frames

Figure 1 illustrates the adopted coordinate convention. The inertial frame $\{I\}$ defines the global coordinates (x, y), while the body frame $\{B\}$ is attached to the robot at yaw angle θ. The body-frame velocity v therefore represents the platform motion in the robot frame rather than in the inertial frame.

2.2 Four-Mecanum wheel kinematic mapping

A Mecanum wheel enables omnidirectional motion through angled rollers, originally introduced in the seminal patent by Ilon [11]. Consider a four-wheel Mecanum platform with wheel radius r and geometric parameters L_x (half-length) and L_y (half-width), as described in Figure 2, and define $L = L_x + L_y$. Let the wheel-speed vector be

$$\Omega = [\omega_1 \quad \omega_2 \quad \omega_3 \quad \omega_4]^\top \tag{4}$$

For the common X-roller configuration, Eq. (5) gives a widely used inverse-kinematics mapping from the body velocity v to the wheel-speed vector Ω [7].

$$\Omega = \frac{1}{r}\begin{bmatrix} 1 & -1 & -L \\ 1 & 1 & L \\ 1 & 1 & -L \\ 1 & -1 & L \end{bmatrix}\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \tag{5}$$

Eq. (6) gives the corresponding forward mapping from the wheel-speed vector back to the body velocity [7].

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \frac{r}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{L} & \frac{1}{L} & -\frac{1}{L} & \frac{1}{L} \end{bmatrix}\Omega \tag{6}$$

In other words, Eq. (5) computes the wheel speeds required for a desired body motion, whereas Eq. (6) reconstructs the effective body motion associated with a given wheel-speed vector. Any equivalent sign convention may be used, provided that the wheel numbering and the pair Eqs. (5) and (6) remain mutually consistent.

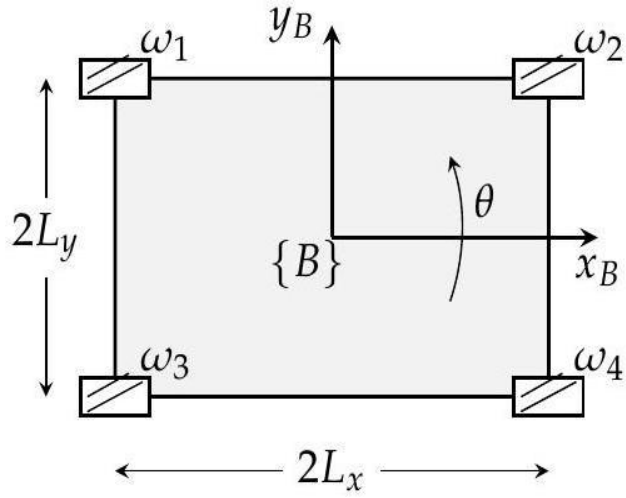

Figure 2. Four-Mecanum-wheel platform geometry and wheel numbering used for the kinematic mappings in Eqs. (5) and (6)

2.3 Actuator constraints and feasible command projection

Wheel angular velocities are bounded as in Eq. (7):

$$|\omega_i| \le \Omega_{\max}, i = 1, \ldots, 4 \tag{7}$$

Optionally, wheel slew (acceleration) may be limited in discrete time as in Eq. (8):

$$|\omega_i[k] - \omega_i[k-1]| \le \dot{\Omega}_{\max}\Delta t \tag{8}$$

Let $u \in \mathbb{R}^3$ denote the controller-generated body command before actuator constraints are applied. The implementation maps $u$the wheel speeds via Eq. (5) and applies the wheel-speed limitations in Eqs. (7) and (8), and then maps the limited wheel speeds back to a feasible body velocity using Eq. (6). This yields

$$\Omega^\star = \mathcal{S}\big(\mathrm{IK}(u)\big), u^\star = \mathrm{FK}(\Omega^\star) \tag{9}$$

where, IK (·) is the inverse-kinematic mapping in Eq. (5), FK (·) denotes the forward kinematic mapping in Eq. (6), and $\mathcal{S}(\cdot)$ is the saturation/slew-limiting operator. Thus, u is the ideal controller request, whereas $u^\star$ is the actuator-feasible body command actually delivered to the robot model. This projection ensures compatibility with physical wheel-speed limits.

2.4 Slip-affected body-velocity model

In urban mobility operations, Mecanum platforms may experience variations in traction due to micro-slip, painted lines, wet patches, and localized surface changes. These effects are modeled as multiplicative traction uncertainties acting on the body-velocity channels according to

$$v = \Lambda u^\star + d, \Lambda \triangleq \mathrm{diag}\big(\lambda_x, \lambda_y, \lambda_\omega\big) \tag{10}$$

where, $u^\star$ is the feasible input from Eq. (9), $d \in \mathbb{R}^3$ is a bounded disturbance term, and λ_x, λ_y, and λ_ω are unknown traction coefficients satisfying λ_x , λ_y , λ_ω $\in$ [λ_{min} ,1], associated with the longitudinal, lateral, and yaw channels,

respectively. Eq. (10) should therefore be interpreted channel-wise, with each traction coefficient scaling the effective control authority of its corresponding body-motion channel.

Remark on the traction model: The diagonal traction model in Eq. (10) is an engineering approximation that captures the dominant reduction in effective longitudinal, lateral, and yaw control authority under slip and surface variation. Although Mecanum wheel-roller contact can introduce additional coupling effects, the diagonal form is adopted here to provide a simple, interpretable, and reproducible benchmark-oriented model.

2.5 Online traction estimation

To enable slip-aware adaptation, the traction matrix Λ is estimated online using measured body velocity v and the feasible command $u^\star$. Motivated by normalized-gradient adaptive identification laws with projection onto admissible bounds [30], the following discrete-time update is adopted:

$$\hat{\lambda}_i[k+1] = \Pi_{[\lambda_{\min},1]} * \left(\hat{\lambda}_i[k] + \gamma_i \frac{u_i^\star[k]\big(v_i[k] - \hat{\lambda}_i[k]u_i^\star[k]\big)}{\varepsilon + (u_i^\star[k])^2}\right), i \in \{x, y, \omega\} \tag{11}$$

where, γ_i are adaptation gains, ε prevents division by zero, and Π denotes projection onto the admissible interval for each coefficient. The gains γ_i determine the estimator's responsiveness, with larger values yielding faster adaptation but greater sensitivity to noise.

The estimator is applied channel-wise to remain consistent with the diagonal traction model in Eq. (10) and to preserve a simple, interpretable adaptation structure for benchmarking purposes. The projection bounds also prevent nonphysical estimates and improve numerical robustness.

The robot geometry, actuator limits, and simulation settings used in the benchmark study are summarized in Table 1.

Table 1. Robot geometry, actuator limits, and simulation settings

Item	Value
Wheel radius	$r = 0.05$ m
Half-length / half-width	$L_x = 0.20$ m, $L_y = 0.15$ m
Derived parameter	$L = L_x + L_y = 0.35$ m
Wheel-speed limit	$\lvert\omega_i\rvert \le \Omega_{\max} = 40$ rad/s, $i = 1, \dots, 4$
Sampling time / horizon	$\Delta t = 2 \times 10^{-4}$ s, $T = 40$ s
Initial pose	$q(0) = [0 \quad 0 \quad 0]^\top$
Velocity noise (nominal)	$\sigma_v = [0 \quad 0 \quad 0]^\top$

3. CONTROLLER DESIGN

3.1 Outer loop: Pose-to-velocity reference generation

The outer loop converts pose errors into a body-frame velocity reference. Let $q_d = [x_d, y_d, \theta_d]^\top$ be the desired pose and $\dot{\theta}_d$ the desired yaw rate. Define the inertial-frame position error

$$e_p \triangleq \begin{bmatrix} x_d - x \\ y_d - y \end{bmatrix} \tag{12}$$

and transform it to the body frame using the rotation matrix $R(\theta)$ [9]:

$$e_b \triangleq R^\top(\theta) e_p, R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{13}$$

Let $e_\theta \triangleq \mathrm{wrap}(\theta_d - \theta)$ denote the wrapped heading error. The velocity reference is generated by feedforward plus proportional feedback:

$$\begin{aligned} v_{x,\mathrm{ref}} &= v_{d,x}^b + k_x e_{b,x}, \\ v_{y,\mathrm{ref}} &= v_{d,y}^b + k_y e_{b,y}, \\ \omega_{\mathrm{ref}} &= \dot{\theta}_d + k_\theta e_\theta \end{aligned} \tag{14}$$

where, $\left[v_{d,x}^b, v_{d,y}^b\right]^\top = R^\top(\theta)\dot{p}_d$ is the feedforward velocity expressed in $\{B\}$ and $\dot{p}_d = [\dot{x}_d \; \dot{y}_d]^\top$.

Docking speed shaping and comfort limiting.

For curbside docking, distance-based shaping is applied to reduce speed near the goal, improving passenger comfort and safety:

$$\sigma_d \triangleq \min\left(1, \frac{\|e_p\|}{r_d}\right), \begin{bmatrix} v_{x,\mathrm{ref}} \\ v_{y,\mathrm{ref}} \end{bmatrix} \leftarrow \sigma_d \begin{bmatrix} v_{x,\mathrm{ref}} \\ v_{y,\mathrm{ref}} \end{bmatrix} \tag{15}$$

A similar shaping can be applied to ω_{ref} using a yaw-radius $r_\theta > 0$. Additionally, to avoid abrupt changes in commanded velocity, a discrete-time slew limiter can be applied:

$$v_{\mathrm{ref}}[k] = v_{\mathrm{ref}}[k-1] + \mathrm{sat}_{\Delta v_{\max}}\left(v_{\mathrm{ref}}^{\mathrm{raw}}[k] - v_{\mathrm{ref}}[k-1]\right) \tag{16}$$

where, $v_{\mathrm{ref}}^{\mathrm{raw}}$ is given by Eqs. (14) and (15), and $\mathrm{sat}_{\Delta v_{\max}}(\cdot)$ acts elementwise with componentwise rate limits.

3.2 Inner loop: Slip-aware adaptive super-twisting sliding mode control

The inner loop regulates body velocity v to track v_{ref} under the slip-affected model Eq. (10). SMC is well known for robustness to matched uncertainties and disturbances [19-21]. Define the velocity tracking error (sliding variable) as

$$s \triangleq v - v_{\mathrm{ref}} \tag{17}$$

To reduce chattering in practice, a boundary-layer saturation is employed:

$$\sigma \triangleq \mathrm{sat}\left(\frac{s}{\phi}\right), \phi = [\phi_x \quad \phi_y \quad \phi_\omega]^\top, \phi_x, \phi_y, \phi_\omega > 0 \tag{18}$$

where the division is elementwise. The saturation operator is also applied elementwise, with

$$\mathrm{sat}(\xi) = \begin{cases} \xi, & |\xi| \le 1 \\ \mathrm{sgn}(\xi), & |\xi| > 1 \end{cases}$$

Super-twisting structure.

Second-order sliding mode methods, including super-twisting, achieve finite-time convergence of s and $\dot{s}$ without requiring $\dot{s}$ measurement, under standard boundedness assumptions [23, 24]. Let

$$z = \begin{bmatrix} z_x \\ z_y \\ z_\omega \end{bmatrix} \in \mathbb{R}^3$$

denote the super-twisting internal state, updated by

$$\dot{z} = -K_2\sigma, K_2 = \mathrm{diag}\left(k_{2,x}, k_{2,y}, k_{2,\omega}\right) \tag{19}$$

Using the estimated traction matrix

$$\hat{\Lambda} = \mathrm{diag}\left(\hat{\lambda}_x, \hat{\lambda}_y, \hat{\lambda}_\omega\right)$$

The commanded body input is formed as a traction-compensated reference term plus a super-twisting correction:

$$\begin{gathered} u = \hat{\Lambda}^{-1} v_{\mathrm{ref}} - K_1(|s|^{1/2} \odot \sigma) + z, \\ K_1 \triangleq \mathrm{diag}\left(k_{1,x}, k_{1,y}, k_{1,\omega}\right) \end{gathered} \tag{20}$$

where, $\odot$ denotes elementwise multiplication, and

$$|s|^{1/2} = \begin{bmatrix} |s_x|^{1/2} \\ |s_y|^{1/2} \\ |s_\omega|^{1/2} \end{bmatrix}$$

is applied elementwise.

In this implementation, traction compensation is applied only to the reference (equivalent) term through $\hat{\Lambda}^{-1} v_{\mathrm{ref}}$, while the super-twisting correction remains unscaled. This choice avoids excessive amplification of the discontinuous correction when the estimated traction becomes small.

Traction-adaptive gains. Because traction variations in Eq. (10) effectively modify the available control authority in each channel, adaptive-gain super-twisting is used to strengthen the corrective action when the estimated traction decreases [25, 26]. For each channel $i \in \{x, y, \omega\}$, define the scheduling variable

$$\rho_i \triangleq \frac{1}{\hat{\lambda}_i + \delta_i}, \delta_i = \delta > 0 \tag{21}$$

The gains are then scheduled as

$$\begin{gathered} k_{1,i}[k] = \mathrm{clip}\left(k_{10,i}\rho_i[k]^{\alpha_i}, k_{1,i}^{\min}, k_{1,i}^{\max}\right), \\ k_{2,i}[k] = \mathrm{clip}\left(k_{20,i}\rho_i[k]^{\beta_i}, k_{2,i}^{\min}, k_{2,i}^{\max}\right), i \\ \in \{x, y, \omega\} \end{gathered} \tag{22}$$

where, $k_{10,i} > 0$ and $k_{20,i} > 0$ are nominal base gains, $\alpha_i > 0$ and $\beta_i > 0$ are channel-wise scheduling exponents, and

$$\mathrm{clip}(x, x_{\min}, x_{\max}) = \min(\max(x, x_{\min}), x_{\max}).$$

The role of these parameters is as follows. The gain $k_{1,i}$ primarily scales the instantaneous nonlinear correction in channel i, whereas $k_{2,i}$ governs the buildup rate of the internal state z_i, which supports disturbance rejection and sustained corrective action. When $\hat{\lambda}_i$ decreases, the scheduling variable ρ_i increases; therefore, both $k_{1,i}$ and $k_{2,i}$ increase, which compensates for the loss of effective control authority in that channel. The lower bounds prevent the gains from becoming too small under nominal conditions, while the upper bounds prevent excessive aggressiveness, actuator stress, and high-frequency control activity in low-traction conditions.

Table 2. Outer-loop reference generation and inner-loop controller parameters

Component	Parameters
Outer loop (pose → v_{ref})	$k_x = 0.7, k_y = 0.7, k_\theta = 1.2.$
Docking comfort limits	$v_{\mathrm{Dock}} = 0.05$ m/s, $\omega_{\mathrm{Dock}} = 0.15$ rad/s. Enabled; $\dot{v}_{\mathrm{ref,\,max}} = [0.6 \quad 0.6 \quad 1.2]^{\mathsf{T}}$, with translational entries in m/s^2 and yaw entry in rad/s^2.
Reference slew limiting	
Proportional-Integral-Derivative (PID) inner loop (body velocity)	$e_p = [0.40 \quad 0.40 \quad 0.60]^{\mathsf{T}}, K_i = [0.05 \quad 0.05 \quad 0.02]^{\mathsf{T}}, K_d = [0 \quad 0 \quad 0]^{\mathsf{T}};$ $I_{\max} = [0.30 \quad 0.30 \quad 0.20]^{\mathsf{T}}; \tau_d = 0.02$ s.
ASTSMC inner loop (adaptive super twisting)	$k_{10} = [0.6 \quad 0.6 \quad 0.5]^{\mathsf{T}}, k_{20} = [3.0 \quad 3.0 \quad 2.5]^{\mathsf{T}}; \alpha = [0.9 \quad 0.9 \quad 0.8]^{\mathsf{T}},$ $\beta = [0.9 \quad 0.9 \quad 0.8]^{\mathsf{T}}; \phi = [0.04 \quad 0.04 \quad 0.03]^{\mathsf{T}}; \delta = 0.05; \rho_{\max} = 2.5;$ $\phi_{\mathrm{DockMult}} = 2.0; gain_{\mathrm{DockMult}} = 0.65.$ $k_{1,\min} = [0.3 \quad 0.3 \quad 0.2]^{\mathsf{T}}, k_{1,\max} = [3.0 \quad 3.0 \quad 2.8]^{\mathsf{T}};$ $k_{2,\min} = [1.0 \quad 1.0 \quad 0.8]^{\mathsf{T}}, k_{2,\max} = [16 \quad 16 \quad 12]^{\mathsf{T}}.$

The nominal gains $k_{10,i}$ and $k_{20,i}$ are selected from stable near-nominal tracking trials so that the inner loop remains faster than the outer loop and the wheel-speed projection is mostly inactive away from severe traction loss. The clipping bounds are then chosen to preserve robustness while limiting excessive wheel speed demand. Channel-wise exponents α_i and β_i allow different adaptation sensitivities in the translational and yaw channels, consistent with the parameter choices reported in Table 2. Discrete-time implementation and wheel feasibility.

With sampling time Δt, Eq. (19) is implemented as

$$z[k+1] = z[k] - \Delta t K_2[k]\sigma[k] \tag{23}$$

and $u[k]$ is computed from Eq. (20). The commanded body input is then converted to wheel speeds using Eq. (5), constrained by Eqs. (7) and (8), and projected back to a feasible body command using Eqs. (6) and (9) before being applied to the slip model Eq. (10). This ensures that the implemented closed loop respects actuator limitations while preserving the practical robustness advantages of super-twisting sliding-mode control, provided sufficient control authority remains available under the wheel-speed constraint [24].

4. STABILITY AND BOUNDEDNESS DISCUSSION

The present study is benchmark- and application-oriented, focusing on actuator-feasible implementation under traction variability. A complete nonlinear proof for the Mecanum contact with wheel-roller coupling is outside the scope of this paper. Nevertheless, the proposed closed-loop architecture admits a practical boundedness argument under explicit assumptions, together with a Lyapunov-based interpretation of ASTSMC behavior in the unconstrained case. This analysis

clarifies what can and cannot be guaranteed once wheel-speed projection is applied.

4.1 Assumptions

Assume: (i) the effective traction coefficients are bounded, $\lambda_{\min} \le \lambda_i(t) \le 1$ for $i \in \{x, y, \omega\}$; (ii) the disturbance $d(t)$ in (10) is bounded; (iii) the traction estimator uses projection so that $\hat{\lambda}_i(t) \in [\lambda_{\min}, 1]$; and (iv) the wheel-speed limiter enforces $|\omega_j(t)| \le \Omega_{\max}$ for all wheels $j = 1, \ldots, 4$.

4.2 Bounded feasible commands under wheel-speed projection

Let $\Omega_{\mathrm{cmd}} = \mathrm{IK}(u(t))$ be the wheel-speed command generated by the controller through Eq. (5). The implementation applies the saturation/slew operator $\mathcal{S}(\cdot)$:

$$\Omega^\star(t) = \mathcal{S}(\Omega_{\mathrm{cmd}}(t)), \|\Omega^\star(t)\|_\infty \le \Omega_{\max} \tag{24}$$

and then computes the feasible body command

$$u^\star(t) = FK(\Omega^\star(t))$$

through Eq. (6). Since FK(·) is linear, bounded wheel speeds imply bounded feasible commands:

$$\|u^\star(t)\| \le c_u \Omega_{\max} \tag{25}$$

for some constant $c_u > 0$ determined by r and $L = L_x + L_y$ in (6).

4.3 Bounded adaptive gains and super-twisting dynamics

The adaptive severity index in Eq. (21) satisfies

$$\rho_i(t) = \frac{1}{\hat{\lambda}_i(t) + \delta_i} \le \frac{1}{\lambda_{\min} + \delta_i} \tag{26}$$

because $\hat{\lambda}_i(t)$ is projected and $\delta_i > 0$. The gain laws Eq. (22) explicitly clip $k_{1,i}$ and $k_{2,i}$ within finite bounds, hence $K_1(t)$ and $K_2(t)$ are bounded for all t. Moreover, $\sigma = \mathrm{sat}(s/\phi)$ satisfies $\|\sigma\|_\infty \le 1$. Therefore, the discrete-time internal-state update Eq. (23) is bounded-increment and produces bounded $z[k]$ for bounded sampling time and bounded $K_2[k]$.

4.4 What can and cannot be guaranteed

Under the above assumptions, the implemented closed loop guarantees (i) actuator feasibility ($|\omega_j| \le \Omega_{\max}$), (ii) bounded feasible commands $u^\star$, (iii) bounded adaptive gains, and (iv) bounded traction estimates. Classical super-twisting analysis provides finite-time convergence of the sliding variable under standard matched-uncertainty assumptions when the unconstrained control is applied. However, once wheel-speed projection is active, the applied input is a saturated/projection-modified version of the ideal control. Therefore, while the architecture preserves feasibility and boundedness, global finite-time convergence cannot be claimed without additional assumptions on how often the projection is active and on the available control authority under $\Omega_{\max}$. This distinction is consistent with the application-oriented goal of this paper: to quantify the robustness-precision tradeoff under realistic wheel-speed limits.

4.5 Lyapunov-based stability interpretation of a traction-adaptive super-twisting sliding-mode controller

A Lyapunov-style interpretation can be given for the proposed ASTSMC in the ideal, unconstrained case, which helps explain the practical behavior observed in the constrained implementation.

Consider each body-velocity channel separately and define the sliding variable

$$s_i = v_i - v_{i,\,\mathrm{ref}}, i \in \{x, y, \omega\} \tag{27}$$

Under the slip-affected model Eq. (10), the traction coefficient $\lambda_i \in [\lambda_{\min}, 1]$ acts as an unknown but strictly positive input gain, while the disturbance term is matched in the same channel. This structure is compatible with the standard super-twisting setting, in which bounded matched uncertainties are rejected by second-order sliding-mode action.

For the unconstrained continuous-time ASTSMC dynamics, a standard super-twisting Lyapunov candidate can be interpreted channel-wise as

$$V_i = \frac{1}{2} z_i^2 + c_{1,i}|s_i| + c_{2,i}|s_i|^{3/2}, c_{1,i}, c_{2,i} > 0 \tag{28}$$

and the composite function

$$V = \sum_{i \in \{x,y,\omega\}} V_i \tag{29}$$

Under the usual super-twisting assumptions of bounded matched disturbance and strictly positive bounded input gain, one obtains

$$\dot{V} \le -\eta_1 \sum_i |s_i|^{1/2} - \eta_2 \sum_i |z_i|, \eta_1, \eta_2 > 0 \tag{30}$$

which implies finite-time convergence of the sliding variable s and the internal state z in the ideal unconstrained case. In the present controller, the adaptive scheduling laws Eq. (22) increase the effective gains when $\hat{\lambda}_i$ decreases, thereby preserving robustness margins under low-traction conditions, while gain clipping ensures implementability.

However, the actual robot implementation is not unconstrained: the commanded body input is mapped to wheel speeds, limited by (7) and (8), and projected back to a feasible body command through (9). Therefore, once wheel-speed projection is active, the applied input differs from the ideal super-twisting command, and the classical finite-time convergence result cannot be claimed globally for the full constrained discrete-time closed loop. In this case, the appropriate interpretation is one of practical stability: feasibility is guaranteed by construction, all closed-loop signals remain bounded, and the tracking error is practically confined to a neighborhood of the origin whose size depends on the available control authority under $\Omega_{\max}$ and the current traction level. This viewpoint is consistent with benchmark results, in which ASTSMC improves robustness to traction variability during service motion, while terminal docking precision becomes constraint-dominated under strict wheel-

speed limits.

5. BENCHMARK SUITE AND EVALUATION PROTOCOL

This section defines the evaluation methodology used to validate and compare the proposed ASTSMC against a PID baseline for a Mecanum mobile robot platform motivated by robotaxi-style operation. To ensure a fair comparison, both controllers use the same robot parameters, outer-loop pose-to-velocity reference generator, sampling time, initial conditions, actuator limits, wheel-speed constraints, and benchmark scenarios.

5.1 Controllers under comparison

Two inner-loop body-velocity controllers are evaluated while keeping the same outer-loop reference generator and the same wheel-speed feasibility layer for both cases:

(i) a baseline PID inner-loop body-velocity controller, widely used in mobile robotics because of its simplicity and tunability; (ii) the proposed ASTSMC, which combines super-twisting action with online traction estimation to improve robustness under slip and traction variation.

Thus, the comparison is designed so that any observed difference arises primarily from the inner-loop control structure rather than from unequal reference generation, scenario definition, or actuator handling.

5.2 Benchmark scenarios

Five scenarios are considered:

- CurbDocking: precise pose docking to a target curb location (low-speed accuracy emphasis).
- RoundedSquareLoop: continuous tracking with repeated curvature changes (transient response emphasis).
- FrictionMap: spatially varying traction/slip parameters to emulate heterogeneous road conditions.
- PayloadEvent: sudden payload changes leading to inertial/traction mismatch.

  The push disturbance, traction-estimator parameters, and scenario-to-traction-mode mapping are summarized in Table 3.
- DockingSlipShock: short-duration traction loss during the final docking phase.

Table 3. Disturbance settings, traction estimator parameters, and scenario-to-traction-mode mapping

Item	Value
Push disturbance	Enabled; applied at $t = 18$ s for 0.2 s with $\Delta v = [0.15 \quad -0.10 \quad 0.05]^{\top}$.
Traction estimator (projection)	$\lambda_{\min} = 0.25; \gamma = [10 \quad 10 \quad 8]^{\top}; u_{\min} = 0.03; \varepsilon > 0$ (as in (11)).
Scenario → traction mode	CurbDocking → mapA; RoundedSquareLoop → mapA; FrictionMap → mapB; PayloadEvent → event; DockingSlipShock → dockShock.

5.3 Wheel-speed constraints

Wheel angular velocities are constrained by the limit

$$|\omega_i| \le \Omega_{\max}, i = 1, \dots, 4.$$

In all reported benchmarks, this wheel-speed magnitude constraint is enforced at every sampling instant. The optional discrete-time wheel slew limit introduced in Eq. (8) is not activated in the benchmark results reported here unless explicitly stated otherwise. Accordingly, controller-generated body commands are first mapped to wheel speeds, then passed through the wheel-speed feasibility layer, and finally projected back to a feasible body command before being applied to the slip-affected model. This ensures that both controllers are evaluated under the same actuator constraints and that only realizable commands are delivered to the robot model.

5.4 Performance metrics

For each run, the following metrics are reported:

- RMSE (x, y, θ) over the full trajectory,
- planar tracking error $\mathrm{RMSE}_{xy} = \sqrt{\mathrm{RMSE}_x^2 + \mathrm{RMSE}_y^2}$,
- final docking position and heading error for docking scenarios,
- peak wheel speed and saturation statistics to verify constraint compliance,
- traction-estimation RMSE, and
- control activity (TV) as a relative indicator of command aggressiveness.

For the traction estimator, the channel-wise estimation error is evaluated by comparing the true traction coefficients $\lambda_x, \lambda_y, \lambda_\omega$ with their corresponding estimates $\hat{\lambda}_x, \hat{\lambda}_y, \hat{\lambda}_\omega$ over the simulation horizon. When a combined translational traction-estimation metric is reported, it is computed from the longitudinal and lateral channels using a root-mean-square aggregation consistent with the notation used later in the robustness summary presented later in this section.

5.5 Implementation and reproducibility

All simulations are executed in MATLAB. Each benchmark returns time histories of pose, reference pose, body-frame tracking errors, wheel speeds, and traction estimates. A single script executes all scenarios for both controllers and generates the time histories and quantitative metrics used in the comparative analysis. The benchmark design follows the general principle of providing standardized, repeatable tracking tests [9]. Additional robustness runs are performed under measurement noise and discrete-time wheel-speed command delay to assess sensitivity to sensing and implementation non-idealities.

5.6 Benchmark parameters and traction map definition

The benchmark traction profiles are defined as follows (all values are channel-wise in the body frame as $\lambda = [\lambda_x, \lambda_y, \lambda_\omega]^{\top}$):

- **mapA** (mild spatial variation): nominal

$$\lambda = [0.95, 0.80, 0.95]^{\top};$$

tile patch if $3 < x < 5$ and $1 < y < 3$, then

$$\lambda = [0.85, 0.65, 0.90]^{\top}.$$

- **mapB** (harsher spatial variation): nominal

$$\lambda = [0.92, 0.75, 0.95]^{\mathsf{T}};$$

wet region if $2 < x < 4$ and $0 < y < 2$, then

$$\lambda = [0.70, 0.55, 0.85]^{\mathsf{T}};$$

tile region if $4 < x < 6$ and $2 < y < 4$, then

$$\lambda = [0.80, 0.60, 0.88]^{\mathsf{T}}.$$

- **event** (payload/traction drop):

$$\lambda = [0.95, 0.80, 0.95]^{\mathsf{T}} \text{ for } t \leq 18 \text{ s}$$

and

$$\lambda = [0.75, 0.55, 0.85]^{\mathsf{T}} \text{ for } t > 18 \text{ s}.$$

6. RESULTS

6.1 Benchmark protocol and metrics

The proposed ASTSMC is evaluated against a baseline PID body-velocity controller using the same outer-loop pose-to-velocity reference generator and the same wheel-speed constraint. All simulations enforce the actuator bound $|\omega_i| \leq 40$ rad/s for each wheel. Performance is quantified using: (i) planar tracking RMSE, $\mathrm{RMSE}_{xy} = \sqrt{\mathrm{RMSE}_x^2 + \mathrm{RMSE}_y^2}$; (ii) final docking position error and final heading error; (iii) traction-estimation RMSE; and (iv) wheel-speed constraint and control activity statistics, including peak $|\omega|$, saturation fraction, total saturation time, and total variation (TV).

6.2 Scenario results

For each scenario, the summary figures report: (i) $x - y$ path tracking; (ii) body-frame position errors (e_x^b, e_y^b); (iii) true vs. estimated traction components; and (iv) wheel angular speeds ω_i.

6.2.1 CurbDocking

Figures 3 and 4 show the curbside docking behavior under PID and ASTSMC, respectively. The corresponding subplots illustrate docking-path convergence, body-frame error evolution, traction estimation, and wheel-speed activity during the final approach. Consistent with Table 4, the PID baseline achieves tighter terminal docking accuracy in this scenario, whereas ASTSMC provides better traction-estimation accuracy according to Table 5.

6.2.2 RoundedSquareLoop

Figures 5 and 6 present sustained route tracking on a rounded-square loop, emphasizing transient behavior at corners and steady tracking along straight segments. In this sustained service-motion benchmark, ASTSMC achieves lower tracking RMSE than PID and also yields better traction-estimation accuracy.

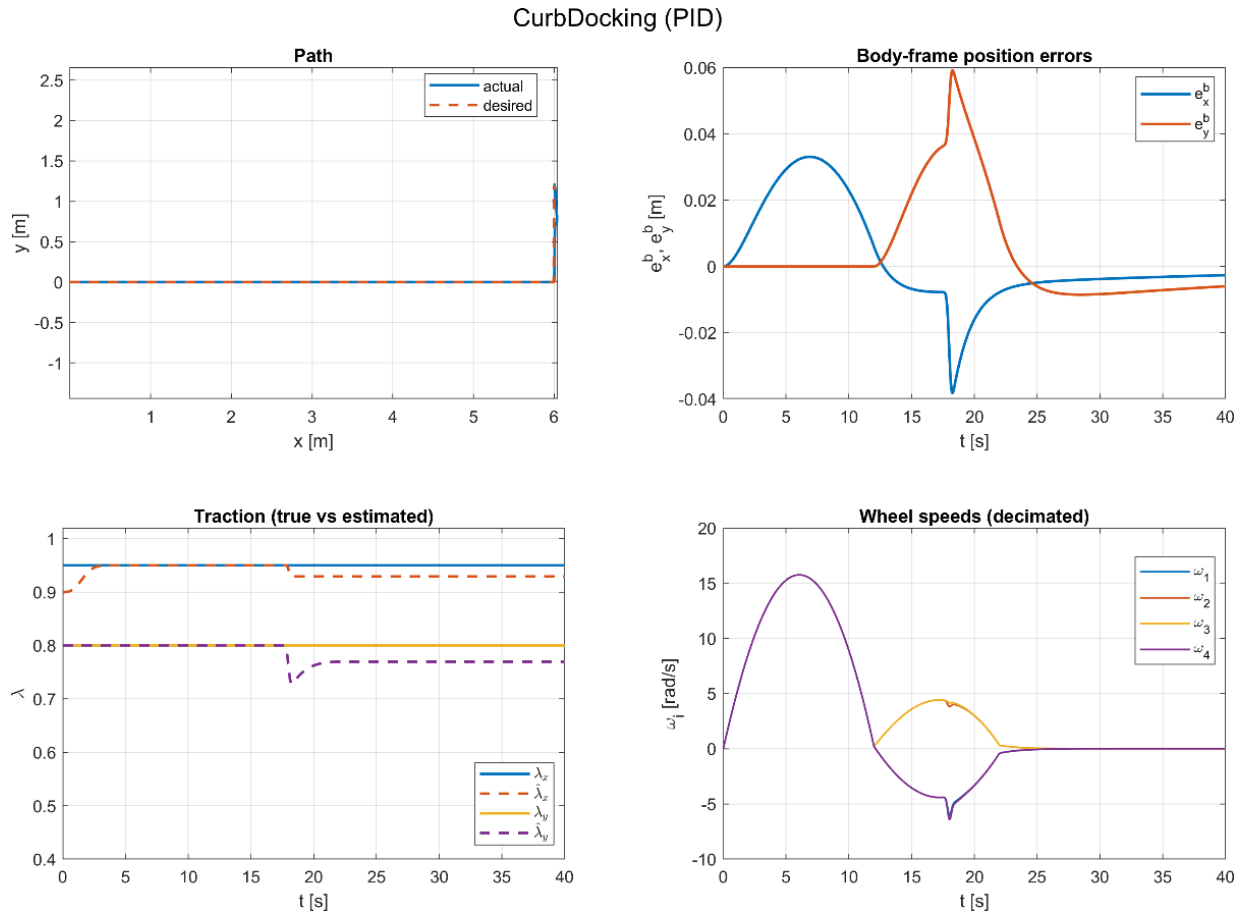

Figure 3. CurbDocking benchmark (Proportional-Integral-Derivative (PID) baseline)

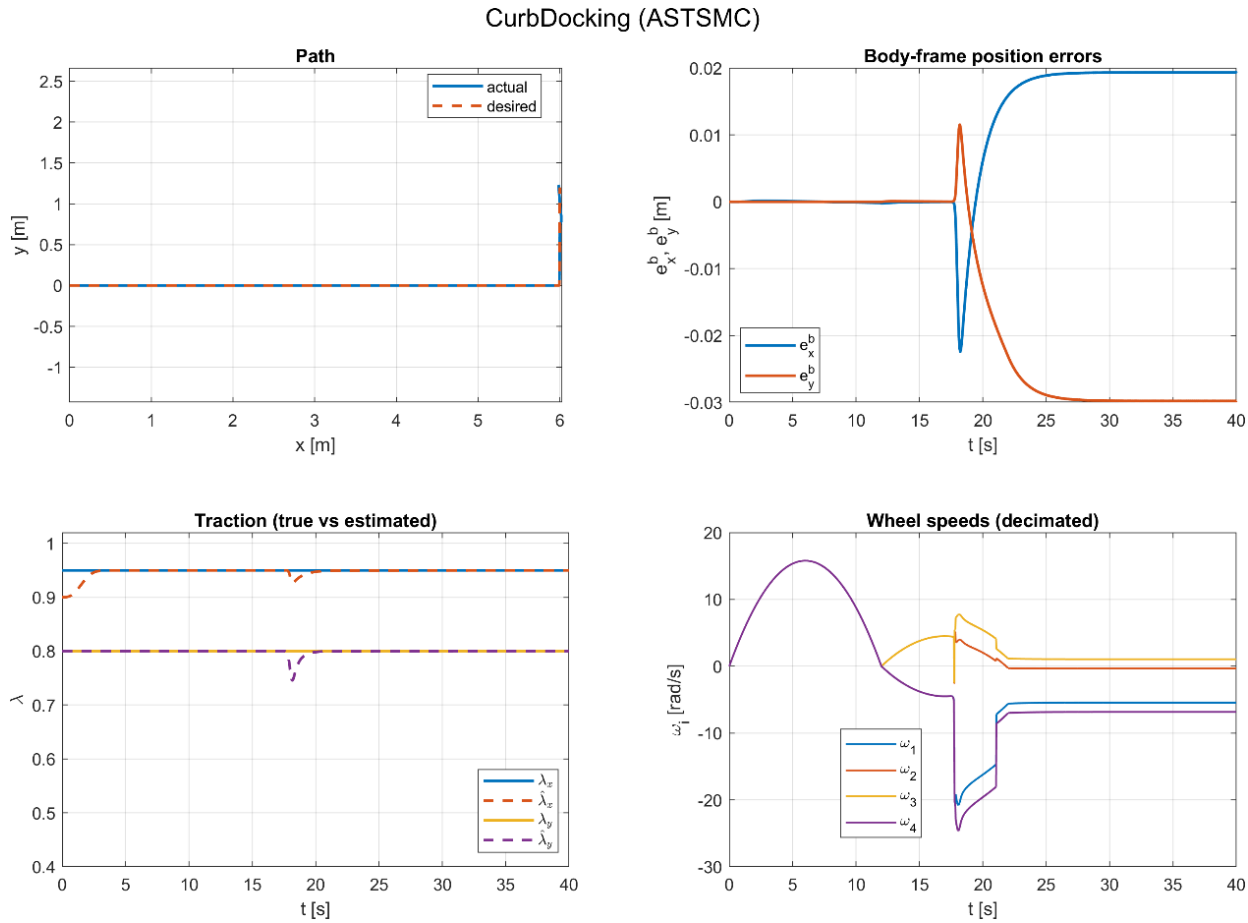

Figure 4. CurbDocking benchmark (adaptive super-twisting sliding mode controller (ASTSMC) proposed)

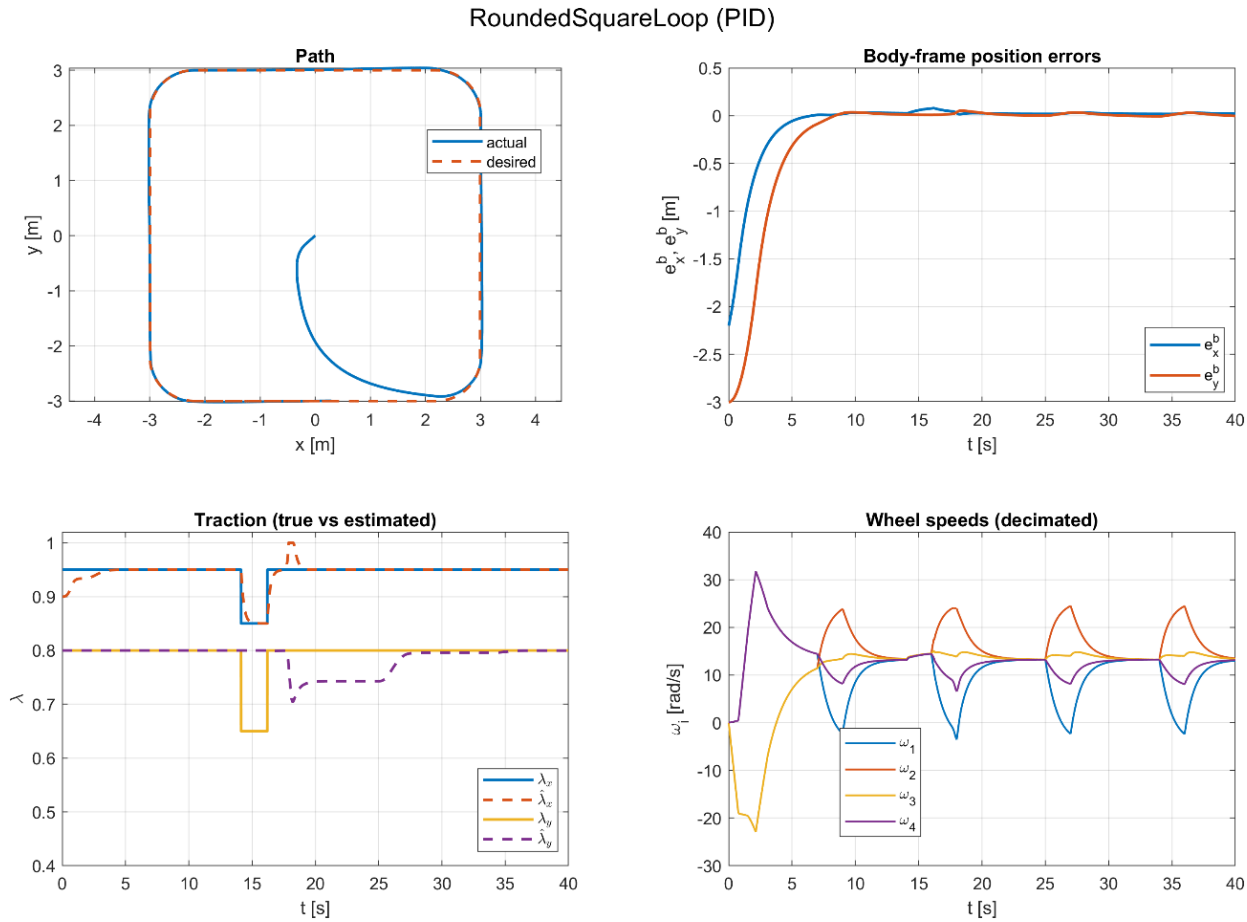

Figure 5. RoundedSquareLoop benchmark (Proportional-Integral-Derivative (PID) baseline)

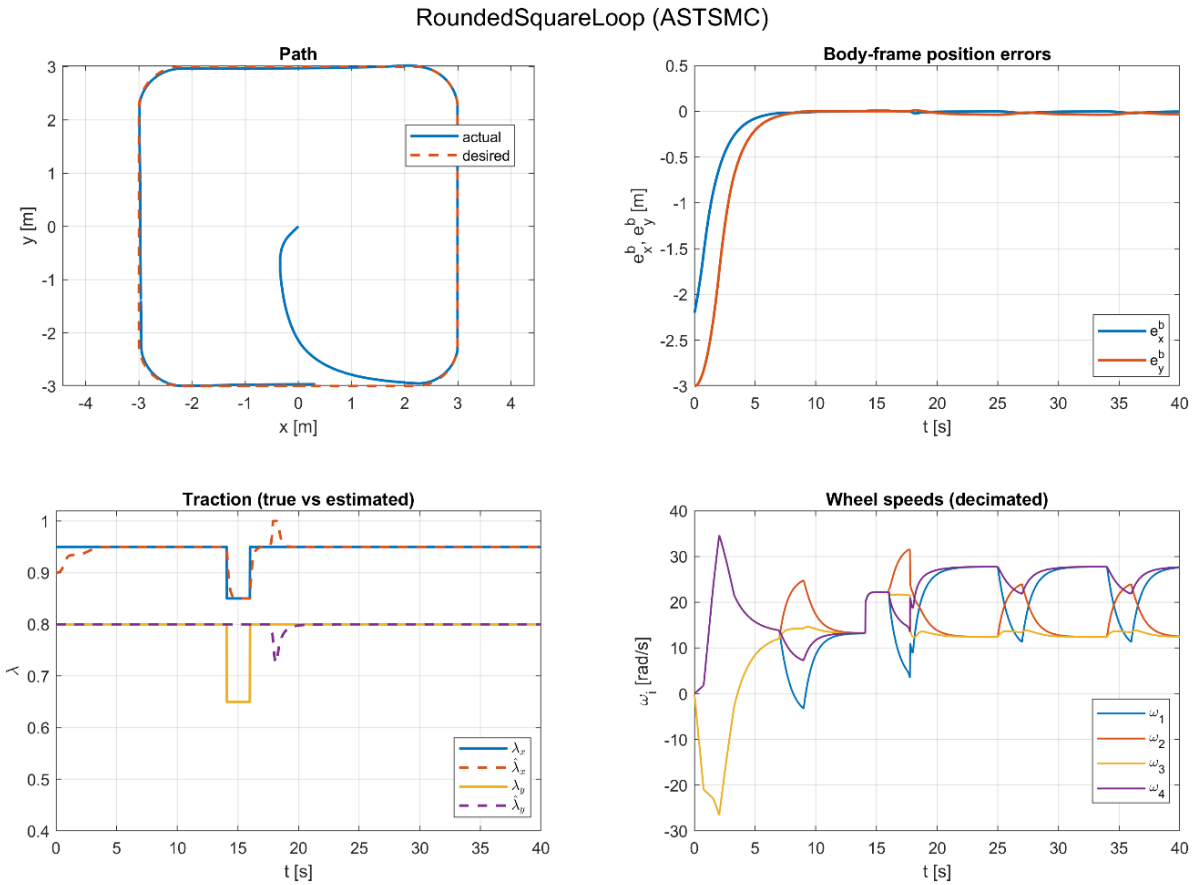

Figure 6. RoundedSquareLoop benchmark (adaptive super-twisting sliding mode controller (ASTSMC) proposed)

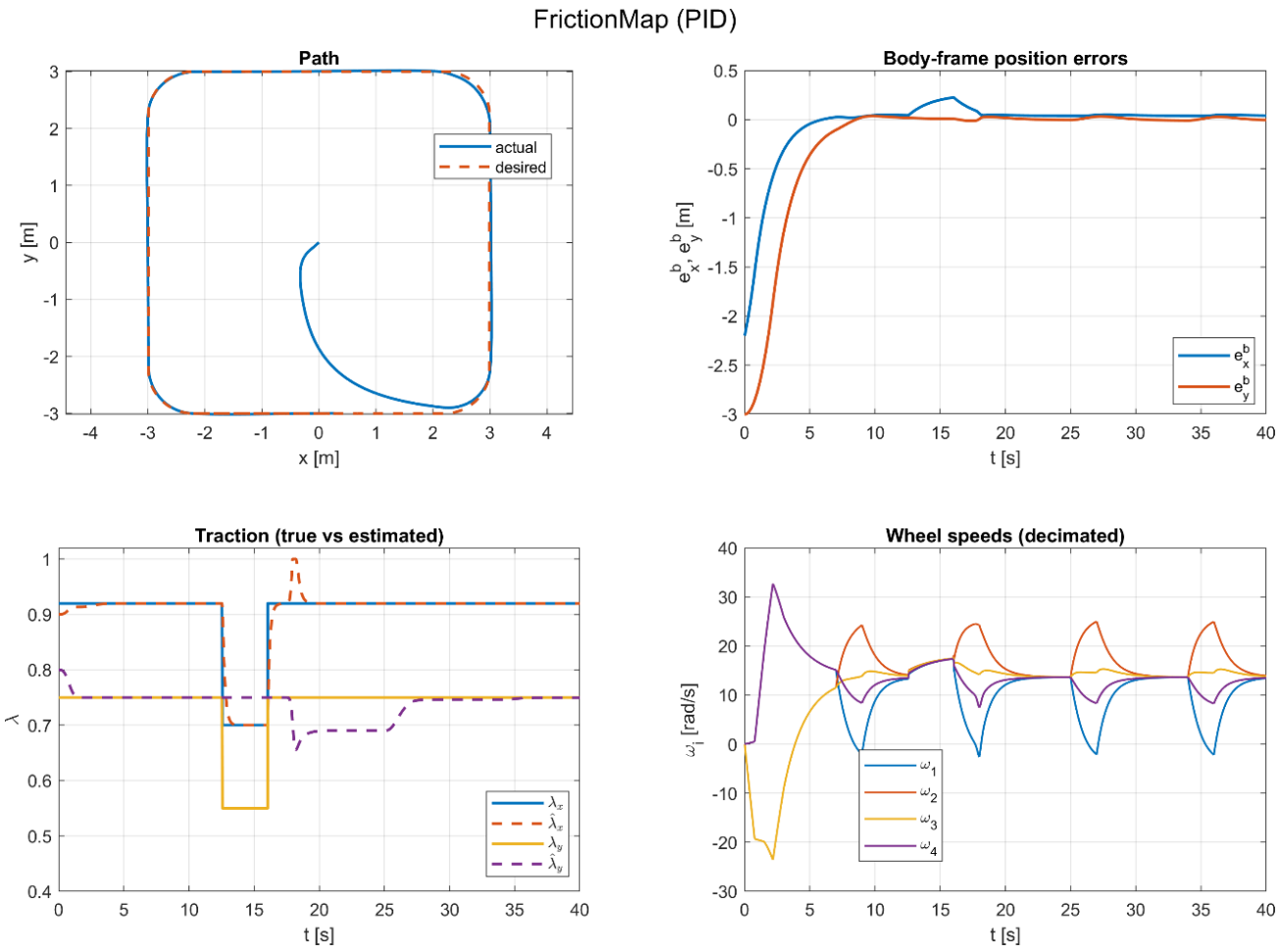

Figure 7. FrictionMap benchmark (Proportional-Integral-Derivative (PID) baseline)

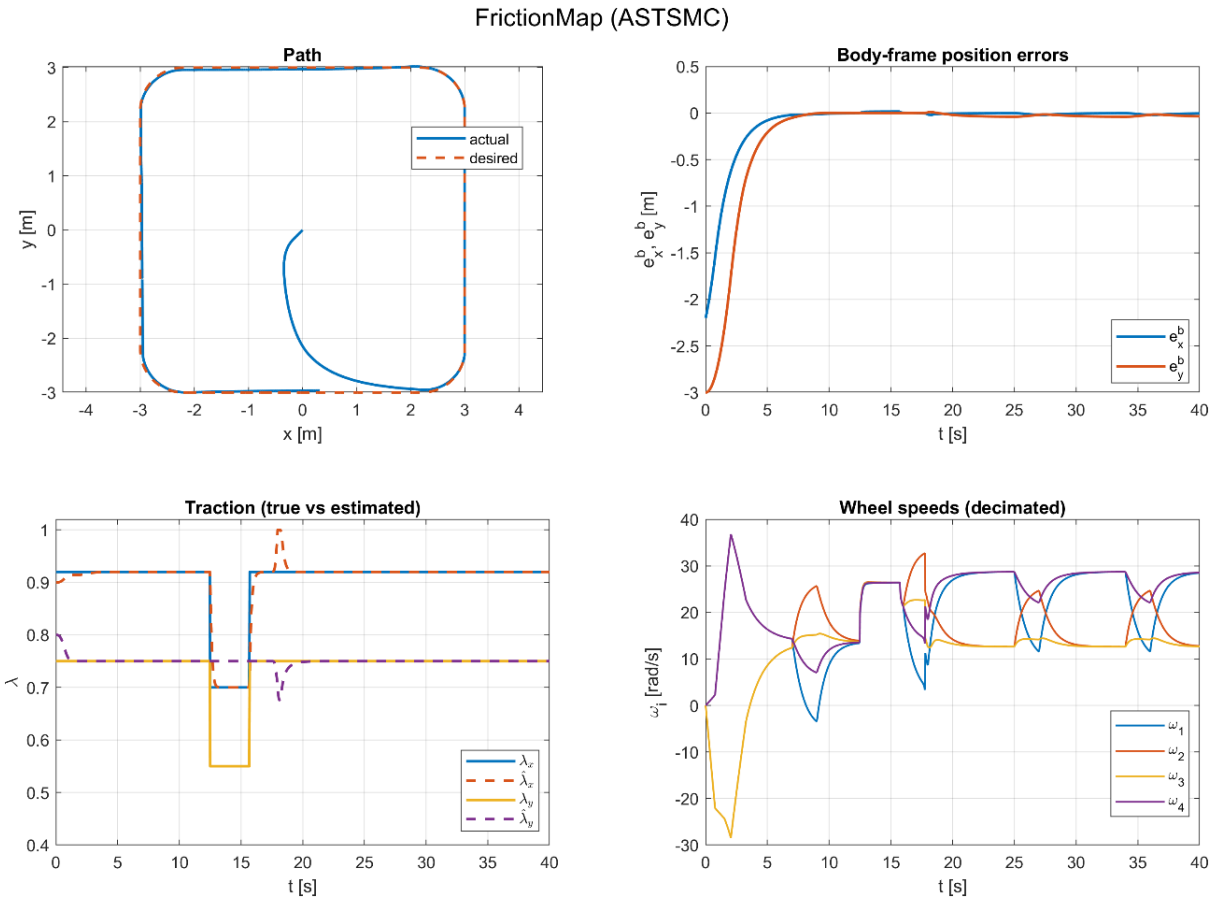

Figure 8. FrictionMap benchmark (adaptive super-twisting sliding mode controll (ASTSMC) proposed)

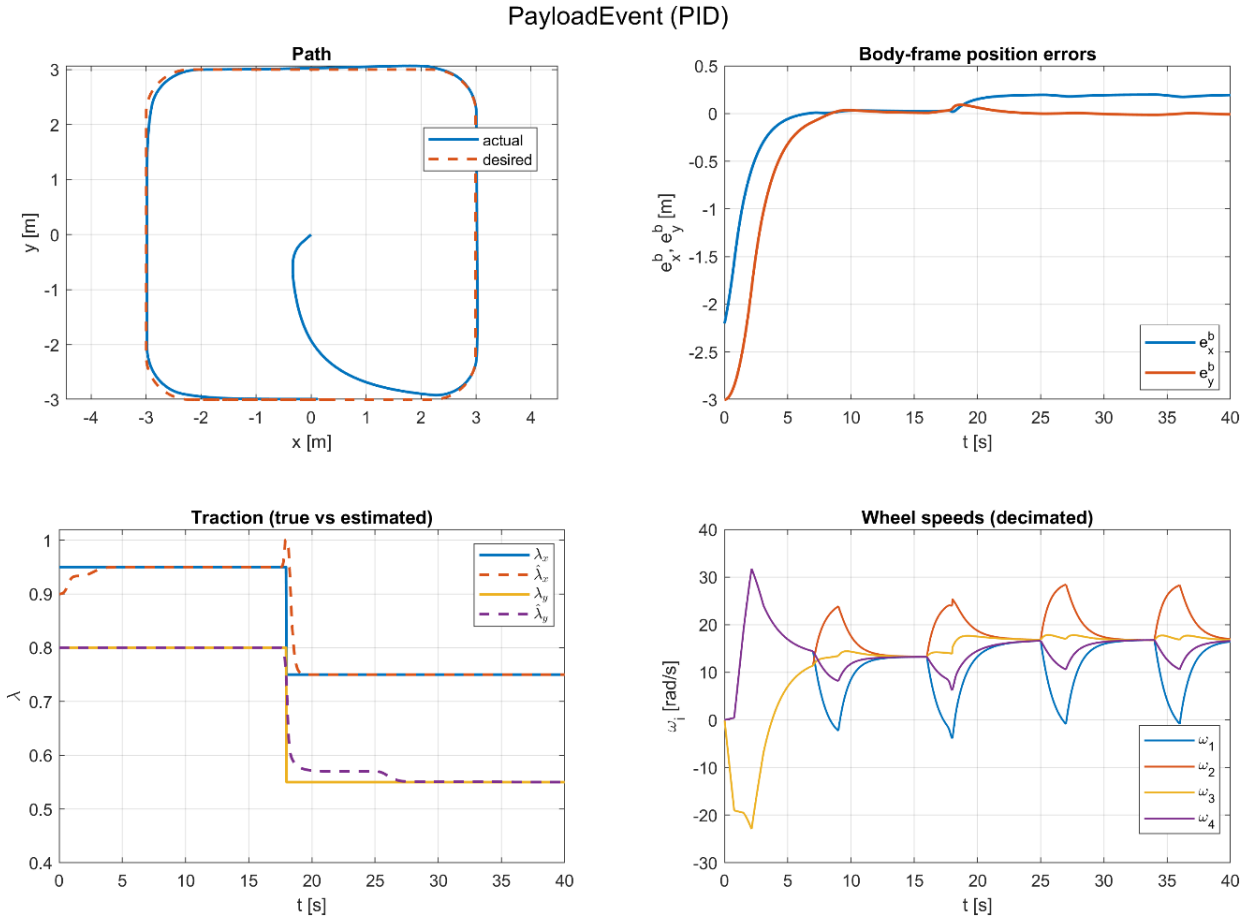

Figure 9. PayloadEvent benchmark (Proportional-Integral-Derivative (PID) baseline)

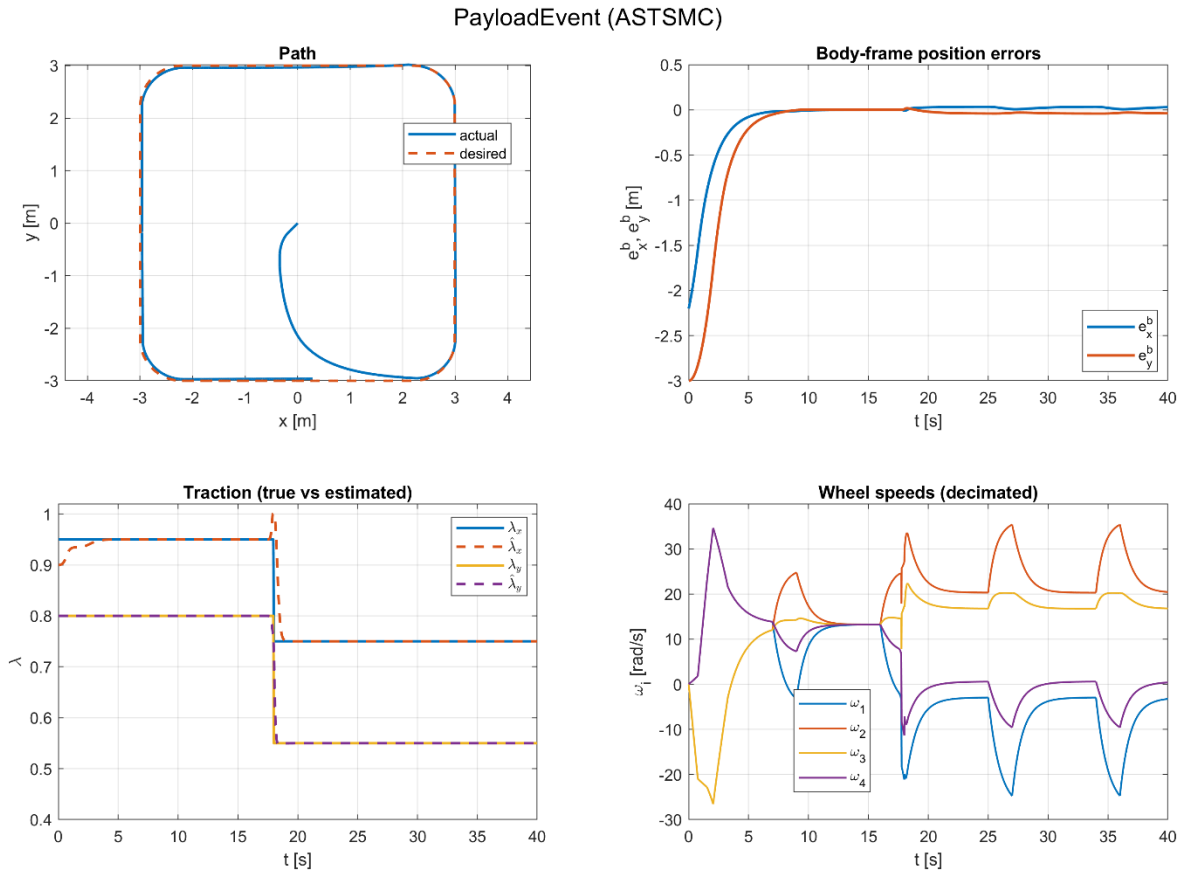

Figure 10. PayloadEvent benchmark (adaptive super-twisting sliding mode controll (ASTSMC) proposed)

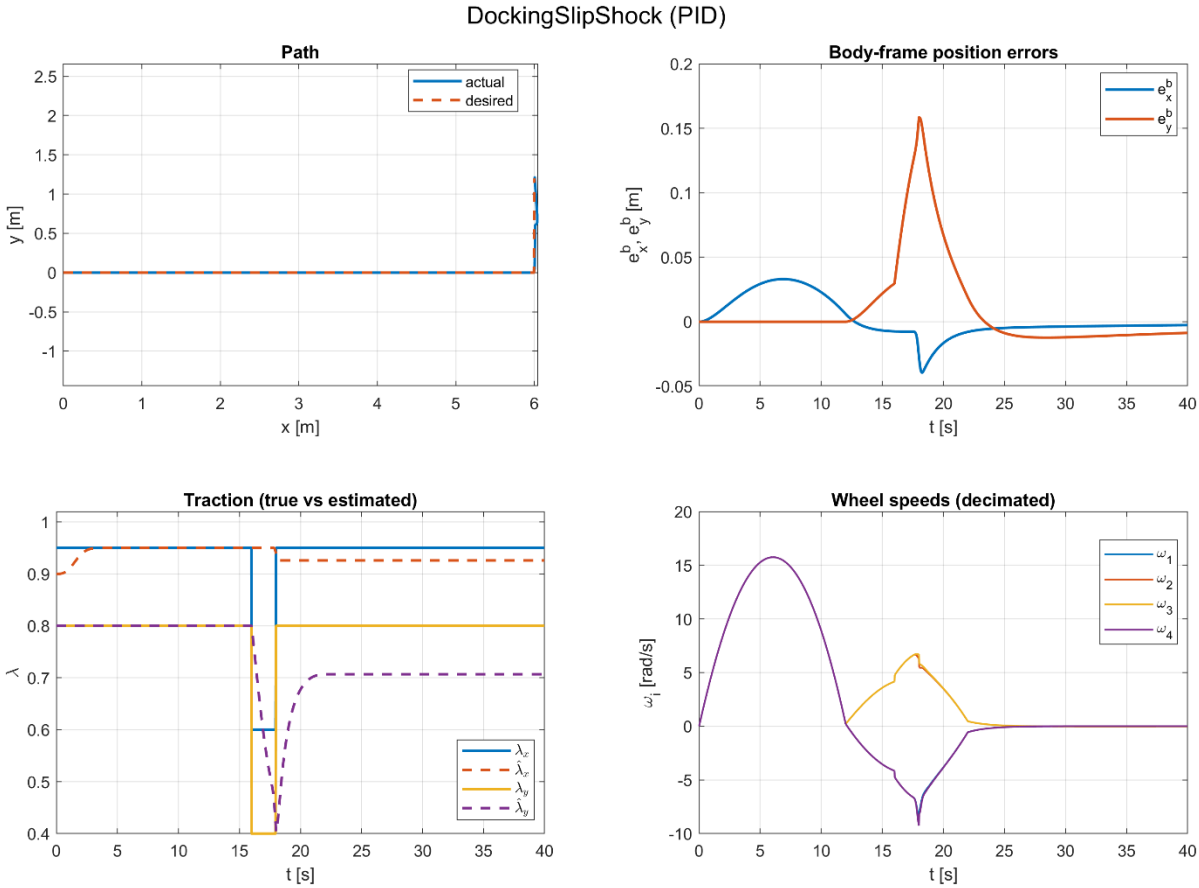

Figure 11. DockingSlipShock benchmark (Proportional-Integral-Derivative (PID) baseline)

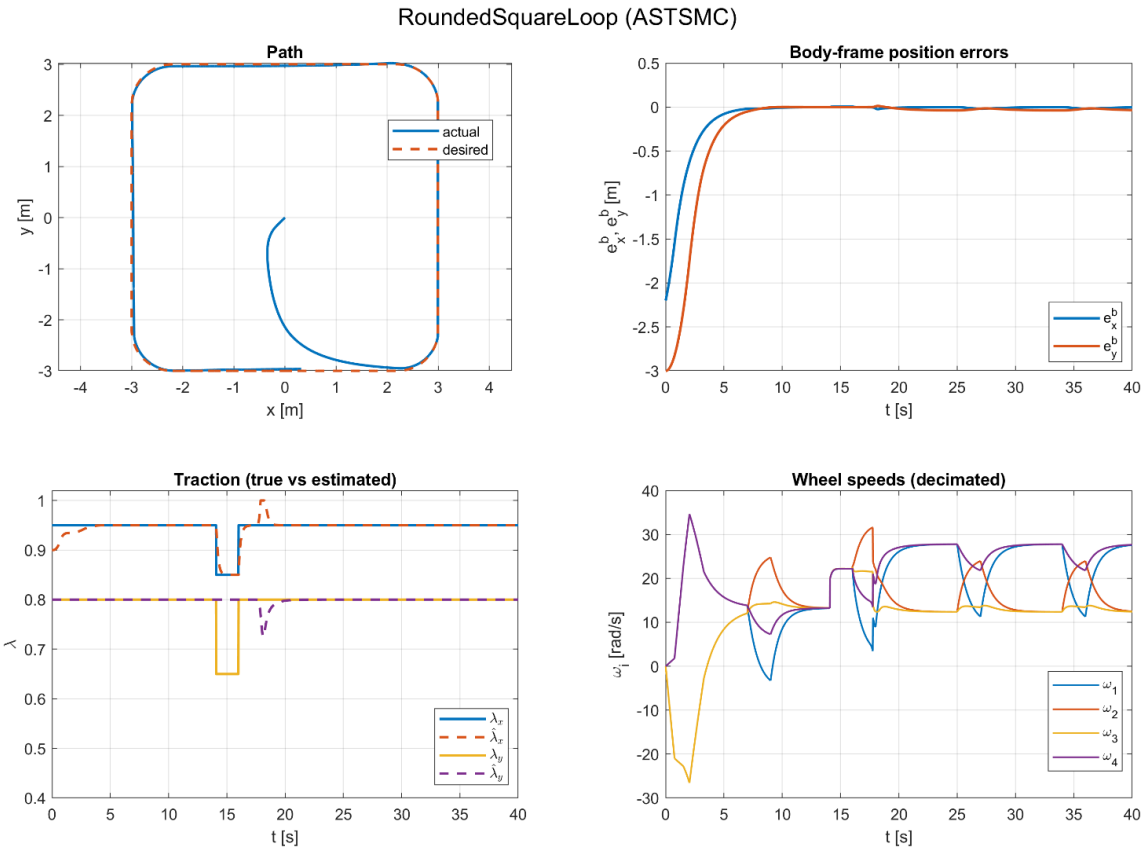

Figure 12. DockingSlipShock benchmark (adaptive super-twisting sliding mode controll (ASTSMC) proposed)

6.2.3 FrictionMap

Figures 7 and 8 evaluate robustness under spatially varying traction. The traction subplot is particularly important here because it validates the estimator's response to traction-induced changes. The quantitative results confirm that ASTSMC outperforms PID in both tracking RMSE and traction-estimation RMSE in this spatially varying traction scenario.

6.2.4 PayloadEvent

Figures 9 and 10 apply a payload/traction-change event during service motion. This benchmark stresses disturbance rejection and estimator adaptation during a sudden change in operating conditions. Table 4 shows that ASTSMC provides better tracking performance and markedly smaller final position error after the payload-related traction change, while Table 5 also indicates improved traction-estimation accuracy.

6.2.5 DockingSlipShock

Figures 11 and 12 test docking under an impulsive slip-like traction disturbance. This benchmark highlights both constraint handling and the estimator/controller's capability to respond rapidly during the terminal docking phase. In this terminal docking disturbance case, PID again retains tighter terminal accuracy, whereas ASTSMC preserves better traction-estimation accuracy and exhibits higher peak wheel-speed demand.

6.3 Quantitative summary

Tables 4 and 5 summarize tracking/docking performance and robustness/constraint statistics across all scenarios. Overall, Table 4 confirms a robustness–precision tradeoff: the PID baseline achieves tighter terminal docking accuracy in CurbDocking and DockingSlipShock, whereas ASTSMC achieves lower tracking RMSE in RoundedSquareLoop, FrictionMap, and PayloadEvent.

In addition to nominal benchmarks, supplementary robustness tests were performed to assess sensitivity to sensing and implementation non-idealities. Specifically, additional tests were conducted under (i) additive body-velocity measurement noise and (ii) a discrete-time wheel-speed command delay representing embedded computation and communication latency. Table 6 shows that the qualitative tradeoff identified in the nominal benchmarks remains broadly consistent: ASTSMC retains a robustness advantage in traction-varying service-motion scenarios, where PID remains preferable for the highest terminal docking precision under strict wheel-speed limits.

7. DISCUSSION

The benchmark results under the realistic wheel-speed bound $|\omega_i| \leq 40$ rad/s highlight a clear tradeoff between (i) terminal docking accuracy and (ii) robustness to traction variability with reliable online traction estimation. Table 4 summarizes tracking and docking performance, while Table 5 reports traction-estimation RMSE, wheel-speed saturation statistics, and control activity.

The Lyapunov-based interpretation in Section 4.5 clarifies these observations: when wheel-speed projection is mostly inactive, the super-twisting mechanism effectively rejects traction-induced uncertainty and matched disturbances; when projection becomes active near terminal docking, the closed loop becomes input-limited, and precision is dominated by feasible control authority under $\Omega_{\max}$. Therefore, the observed docking tradeoff is consistent with constrained robust control

behavior rather than a contradiction of super-twisting robustness.

Table 4. Tracking and docking summary under $|\omega_i| \leq 40$ rad/s

Scenario	RMSE $_{xy}$ (PID)	RMSE $_{xy}$ (ASTSMC)	Final pos. (m) (PID)	Final pos. (m) (ASTSMC)	Final yaw (deg) (PID)	Final yaw (deg) (ASTSMC)	Peak $\|\omega\|$ (PID)	Peak $\|\omega\|$ (ASTSMC)
CurbDocking	0.022	0.024	0.007	0.035	0.01	0.75	15.75	24.65
RoundedSquareLoop	0.743	0.716	0.021	0.034	0.24	0.28	31.78	34.66
FrictionMap	0.755	0.716	0.042	0.036	0.24	0.28	32.79	36.83
PayloadEvent	0.755	0.716	0.195	0.049	0.37	0.14	31.78	36.58
DockingSlipShock	0.039	0.066	0.009	0.101	0.01	0.18	15.75	40.00

Note: RMSE = Root Mean Square Error; PID = Proportional-Integral-Derivative; ASTSMC = adaptive super-twisting sliding mode control

Table 5. Robustness and constraint summary: Traction estimation RMSE and wheel speed statistics

Scenario	RMSE$_{\lambda,xy}$ (PID)	RMSE$_{\lambda,xy}$ (ASTSMC)	Sat. % (PID)	Sat. % (ASTSMC)	Sat. Time (s) (PID)	Sat. Time (s) (ASTSMC)	TV (PID)	TV (ASTSMC)
CurbDocking	0.031	0.011	0.0000	0.0000	0.0000	0.0000	1.68e + 02	3.69e + 06
RoundedSquareLoop	0.045	0.036	0.0000	0.0000	0.0000	0.0000	5.01e + 02	7.95e + 06
FrictionMap	0.068	0.059	0.0000	0.0000	0.0000	0.0000	5.28e + 02	8.88e + 06
PayloadEvent	0.027	0.023	0.0000	0.0000	0.0000	0.0000	5.17e + 02	8.81e + 06
DockingSlipShock	0.128	0.087	0.0000	0.0026	0.0000	0.0010	1.93e + 02	7.56e + 06

Note: RMSE = Root Mean Square Error; PID = Proportional-Integral-Derivative; ASTSMC = adaptive super-twisting sliding mode control; TV = Total variation

Table 6. Robustness to measurement noise and actuation delay (Nominal vs Noise vs Delay)

Scenario	Ctrl	RMSE $_{xy}$ (Nom)	RMSE $_{xy}$ (Noise)	RMSE $_{xy}$ (Delay)	FinalPos (Nom)	FinalPos (Noise)	FinalPos (Delay)
CurbDocking	PID	0.022	0.022	0.022	0.007	0.007	0.007
RoundedSquareLoop	PID	0.743	0.743	0.743	0.021	0.021	0.021
FrictionMap	PID	0.755	0.755	0.755	0.042	0.042	0.042
PayloadEvent	PID	0.755	0.755	0.755	0.195	0.195	0.195
DockingSlipShock	PID	0.039	0.039	0.039	0.009	0.009	0.009
CurbDocking	ASTSMC	0.024	0.016	0.035	0.035	0.000	0.051
RoundedSquareLoop	ASTSMC	0.716	0.733	0.716	0.034	0.170	0.038
FrictionMap	ASTSMC	0.716	0.735	0.716	0.036	0.019	0.037
PayloadEvent	ASTSMC	0.716	0.757	0.739	0.049	0.157	0.287
DockingSlipShock	ASTSMC	0.066	0.024	0.063	0.101	0.001	0.091

Note: RMSE = Root Mean Square Error; PID = Proportional-Integral-Derivative; ASTSMC = adaptive super-twisting sliding mode control

7.1 Docking scenarios: Accuracy vs. constraint-dominated behavior

In curbside docking maneuvers (CurbDocking and DockingSlipShock), the PID baseline achieves tighter terminal position error compared with ASTSMC (Table 4). This is expected because docking is a low-speed, precision-oriented task in which command feasibility and transient shaping dominate performance. The wheel-speed constraint clips aggressive wheel commands, and any controller that relies on strong corrective action near the end of the maneuver can experience reduced effectiveness when the constraint becomes active. This behavior is evident in the wheel-speed subplots of Figures 3-12, where the commanded motion approaches the actuator limits more frequently during the final approach for the robust controller.

Importantly, even when docking accuracy is less favorable, ASTSMC maintains higher traction-estimation quality (Table 5), supporting the paper's main objective: slip/traction awareness for robotaxi-style operation, where road conditions can change rapidly near curbside regions (painted markings, polished asphalt, and localized wet patches).

7.2 Service-motion scenarios: Robustness under traction variability

In service-motion scenarios that emulate continuous operation (RoundedSquareLoop, FrictionMap, and PayloadEvent), the traction-adaptive design goal of ASTSMC becomes more pronounced. The body-frame error subplots and path-tracking plots (Figures 5-10) show that ASTSMC sustains stable tracking despite traction variations induced by spatial friction changes and event-type disturbances. This is consistent with the purpose of using a sliding-mode-based inner loop: to reduce sensitivity to uncertain traction scaling and disturbance-like effects in the velocity dynamics.

A key contribution is that ASTSMC consistently produces lower RMSE for traction estimation than the PID baseline (Table 5). This is relevant for future autonomy functions because traction estimates can be used for supervisory

decisions such as speed reduction near low-friction zones, safer docking, or constraint-aware planning. In addition, the supplementary noise and delay tests in Table 6 preserve the same qualitative trend, indicating that the proposed ASTSMC remains more robust in traction-varying service-motion scenarios, whereas PID remains preferable for the tightest terminal docking tasks under the tested non-idealities.

7.3 Control activity, saturation, and practical implications

Both controllers explicitly enforce the wheel-speed constraint. However, Table 5 indicates that ASTSMC exhibits higher saturation-related statistics and substantially larger control activity (TV) than PID. Large TV values are characteristic of sliding-mode-type laws in discrete-time implementations, even with boundary layers, because switching-like corrective actions increase high-frequency content. While this behavior can improve robustness to matched uncertainties, it may also raise practical concerns for deployment, including increased actuator wear and heating, amplified sensitivity to measurement noise, excitation of unmodeled dynamics (e.g., roller compliance and drivetrain backlash), and reduced passenger comfort through increased jerk.

Several mitigation mechanisms are already incorporated in the proposed implementation: (i) a boundary-layer saturation (18) to soften discontinuities; (ii) adaptive-gain clipping 22(a) and (b) to prevent excessive gains under low traction; (iii) explicit wheel-speed feasibility projection (9), together with outer-loop feasibility scaling, to ensure actuator-realizable commands; and (iv) saturation-aware update of the super-twisting internal state to reduce windup when wheel limits are active. In addition, the two-loop architecture requires the outer-loop reference generator to remain constraint-feasible and the inner loop to be sufficiently faster than the outer loop, especially in omnidirectional maneuvers that combine lateral motion and yaw correction.

TV should therefore be interpreted as a relative indicator of command aggressiveness under fixed sampling, sensing, and actuator conditions. In particular, measurement noise and implementation non-idealities can increase wheel-speed variation even when the qualitative tracking behavior remains similar; hence, TV comparisons are most informative when made under the same experimental conditions. Overall, the results suggest that ASTSMC is well-suited to service-motion robustness under traction variability, while docking maneuvers benefit from additional softening or scheduling. A practical implementation may therefore employ docking-mode gain scheduling (larger boundary layer and reduced gains near the goal) or a hybrid terminal strategy to retain traction awareness without unnecessary high-frequency control action.

7.4 Summary of implications for Mecanum robotaxi control

Overall, the results support the following conclusions. First, the proposed traction-adaptive ASTSMC improves traction estimation and maintains robust tracking under traction variability, matching the intended smart-city operating conditions. Second, the PID baseline can be preferable for final docking precision under strict wheel-speed limits, suggesting that a practical deployment may use either (i) a dedicated docking-mode gain schedule for ASTSMC, or (ii) a hybrid strategy where ASTSMC is used for route/service motion and PID (or softened SMC) is used for the terminal docking phase. These findings strengthen the case for traction-aware robust control as a candidate inner-loop strategy for omnidirectional robot platforms operating on real roads.

8. CONCLUSIONS

This work investigated constraint-aware trajectory tracking for a four-Mecanum-wheeled mobile robot, motivated by future robotaxi-style operation under traction variability. A two-loop control structure was considered, where an outer-loop pose-to-velocity reference generator provides feasible body-velocity commands and an inner loop executes either a baseline PID velocity controller or a traction-adaptive super-twisting sliding mode.

ASTSMC augmented with online traction estimation. All benchmarks explicitly enforced realistic wheel-speed limits, ensuring that the evaluation reflects actuator-feasible motion.

Across the service-motion scenarios with spatially varying traction and event-type disturbances, the traction-adaptive ASTSMC demonstrated greater robustness to traction changes and consistently improved traction estimation quality compared with the PID baseline. In docking-oriented scenarios, the PID baseline achieved smaller terminal docking errors, reflecting the constraint-dominated and precision-focused nature of curbside docking. These results suggest that a practical omnidirectional robotaxi controller may benefit from using traction-adaptive ASTSMC during service motion, while employing docking-mode scheduling or a hybrid terminal strategy when the task becomes strongly precision- and constraint-dominated.

8.1 Limitations and future work

This study has several limitations that motivate future work. First, controller tuning in a cascaded architecture should explicitly enforce time-scale separation, with an inner-loop bandwidth significantly higher than that of the outer loop; systematic gain-selection methods (e.g., frequency-domain or optimization-based tuning under actuator constraints) will be explored. Second, docking is a terminal, low-speed maneuver where actuator feasibility and transient shaping dominate; future work will investigate docking-mode scheduling for ASTSMC (e.g., softened gains, terminal sliding surfaces, or hybrid switching between robust and precision modes) to close the docking-accuracy gap while retaining traction awareness. Third, the evaluation is currently simulation-based; future work will add sensor noise, delays, and unmodeled effects (e.g., wheel compliance and slip coupling) and extend the analysis to Monte Carlo, which runs over traction maps and disturbance realizations. Finally, the released benchmark suite will be expanded to include additional robotaxi-relevant scenarios (e.g., curb approach with lane markings, stop-and-go passenger loading, and mixed-traffic speed profiles) and will be validated on experimental hardware once a Mecanum platform testbed becomes available.

ACKNOWLEDGMENT

During the preparation of this manuscript, ChatGPT (OpenAI) was used solely for language editing and writing

support. The author reviewed, revised, and approved the final manuscript and accepts full responsibility for its content.

REFERENCES

[1] Maldonado Silveira Alonso Munhoz, P.A., da Costa Dias, F., Kowal Chinelli, C., Azevedo Guedes, A.L., et al. (2020). Smart mobility: The main drivers for increasing the intelligence of urban mobility. Sustainability, 12(24): 10675. https://doi.org/10.3390/su122410675

[2] Brzeziński, Ł. (2024). Social aspects of smart urban mobility. Encyclopedia, 4(2): 864-873. https://doi.org/10.3390/encyclopedia4020055

[3] Mutavdžija, M., Kovačić, M., Buntak, K. (2024). Moving towards sustainable mobility: A comparative analysis of smart urban mobility in Croatian cities. Sustainability, 16(5): 2004. https://doi.org/10.3390/su16052004

[4] Eruyar, M., Özen, H. (2026). Developing a sustainable urban mobility maturity model. Sustainability, 18(2): 689. https://doi.org/10.3390/su18020689

[5] Allam, Z., Sharifi, A. (2022). Research structure and trends of smart urban mobility. Smart Cities, 5(2): 539-561. https://doi.org/10.3390/smartcities5020029

[6] Mavlutova, I., Atstaja, D., Grasis, J., Kuzmina, J., Uvarova, I., Roga, D. (2023). Urban transportation concept and sustainable urban mobility in smart cities: A review. Energies, 16(8): 3585. https://doi.org/10.3390/en16083585

[7] Doroftei, I., Grosu, V., Spinu, V. (2007). Omnidirectional mobile robot—design and implementation. In: Habib, M.K. (ed) Bioinspiration and Robotics: Walking and Climbing Robots. InTech (I-Tech Education and Publishing), Vienna, Austria, pp. 512-531.

[8] Campion, G., Bastin, G., D'Andrea-Novel, B. (1996). Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. IEEE Transactions on Robotics and Automation, 12(1): 47-62. https://doi.org/10.1109/70.481750

[9] Siegwart, R., Nourbakhsh, I.R., Scaramuzza, D. (2011). Introduction to Autonomous Mobile Robots. MIT Press.

[10] Muir, P.F., Neuman, C.P. (1987). Kinematic modeling of wheeled mobile robots. Journal of Robotic Systems, 4(2): 281-340. https://doi.org/10.1002/rob.4620040209

[11] Ilon, B.E. (1975). U.S. Patent No. 3,876,255. Washington, DC: U.S. Patent and Trademark Office.

[12] Gfrerrer, A. (2008). Geometry and kinematics of the mecanum wheel. Computer-Aided Geometric Design, 25: 784-791. https://doi.org/10.1016/j.cagd.2008.07.008

[13] Tagliavini, L., Colucci, G., Botta, A., Cavallone, P., Baglieri, L., Quaglia, G. (2022). Wheeled mobile robots: State of the art overview and kinematic comparison among three omnidirectional locomotion strategies. Journal of Intelligent & Robotic Systems, 106(3): 57. https://doi.org/10.1007/s10846-022-01745-7

[14] Zeidis, I., Zimmermann, K. (2019). Dynamics of a four-wheeled mobile robot with Mecanum wheels. ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik, 99(12): e201900173. https://doi.org/10.1002/zamm.201900173

[15] Åström, K.J., Hägglund, T. (2006). Advanced PID Control. ISA, Research Triangle Park, NC, USA.

[16] Szeremeta, M., Szuster, M. (2022). Neural tracking control of a four-wheeled mobile robot with mecanum wheels. Applied Sciences, 12(11): 5322. https://doi.org/10.3390/app12115322

[17] Pizá, R., Carbonell, R., Casanova, V., Cuenca, Á., Salt Llobregat, J.J. (2022). Nonuniform dual-rate extended Kalman-filter-based sensor fusion for path-following control of a holonomic mobile robot with four mecanum wheels. Applied Sciences, 12(7): 3560. https://doi.org/10.3390/app12073560

[18] Wang, D., Gao, Y., Wei, W., Yu, Q., Wei, Y., Li, W., Fan, Z. (2024). Sliding mode observer-based model predictive tracking control for Mecanum-wheeled mobile robot. ISA Transactions, 151: 51-61. https://doi.org/10.1016/j.isatra.2024.05.050

[19] Utkin, V.I. (1977). Variable structure systems with sliding modes. IEEE Transactions on Automatic Control, 22(2): 212-222. https://doi.org/10.1109/TAC.1977.1101446

[20] Utkin, V.I. (1992). Sliding Modes in Control and Optimization. Springer, Berlin/Heidelberg, Germany.

[21] Edwards, C., Spurgeon, S.K. (1998). Sliding Mode Control: Theory and Applications. CRC Press. , Boca Raton, FL, USA.

[22] Levant, A. (2003). Higher-order sliding modes, differentiation and output-feedback control. International Journal of Control, 76(9-10): 924-941. https://doi.org/10.1080/0020717031000099029

[23] Levant, A. (2007). Principles of 2-sliding mode design. Automatica, 43(4): 576-586. https://doi.org/10.1016/j.automatica.2006.10.008

[24] Shtessel, Y., Edwards, C., Fridman, L., Levant, A. (2014). Sliding Mode Control and Observation. Birkhäuser, Boston, MA, USA. https://doi.org/10.1007/978-0-8176-4893-0

[25] Shtessel, Y., Taleb, M., Plestan, F. (2012). A novel adaptive-gain supertwisting sliding mode controller: Methodology and application. Automatica, 48(5): 759-769. https://doi.org/10.1016/j.automatica.2012.02.024

[26] Utkin, V.I., Poznyak, A.S. (2013). Adaptive sliding mode control with application to super-twist algorithm: Equivalent control method. Automatica, 49: 39-47. https://doi.org/10.1016/j.automatica.2012.09.008

[27] Ren, C., Li, X., Yang, X., Ma, S. (2019). Extended state observer-based sliding mode control of an omnidirectional mobile robot with friction compensation. IEEE Transactions on Industrial Electronics, 66: 9480-9489. https://doi.org/10.1109/TIE.2019.2892678

[28] Zhang, M., Huang, J., Cao, Y. (2019). Adaptive super-twisting control for mobile wheeled inverted pendulum systems. Applied Sciences, 9(12): 2508. https://doi.org/10.3390/app9122508

[29] Nghiem, G.K., Le, T.L., Phung, T.C., Dinh, Q.H. (2025). A study on the optimal control strategy using sliding mode controller for Mecanum-wheeled omnidirectional mobile robot. Measurement, 255: 118113. https://doi.org/10.1016/j.measurement.2025.118113

[30] Ioannou, P.A., Sun, J. (1996). Robust Adaptive Control. PTR Prentice-Hall, Upper Saddle River, NJ, USA.

NOMENCLATURE

Symbol	Description
ASTSMC	Adaptive super-twisting sliding mode control
PID	Proportional-integral-derivative
IK	Inverse kinematic mapping
FK	Forward kinematic mapping
RMSE	Root mean square error
TV	Total variation
q	Robot pose vector
x, y	Inertial-frame coordinates (m)
θ	Yaw angle (rad)
s	
v	Body-frame velocity vector
ω_i	Angular speed of wheel i (rad/s)
r	Wheel radius (m)
L_x, L_y	Platform half-length and half-width (m)
L	Geometric parameter, $L = L_x + L_y$ (m)
Ω_{max}	Maximum wheel angular speed (rad/s)
Λ	Traction matrix
λ_x, λ_y, λ_ω	Traction coefficients
$\hat{\lambda_i}$	Estimated traction coefficient
v_{ref}	Reference body velocity