Memorias del XX Congreso Latinoamericano de Control Automático (CLCA 2025)
13-17 de Octubre, 2025. Cancún, Quintana Roo, México

Peak Response Mitigation in a Mecanum-Wheeled Mobile Robot Using Active Disturbance Rejection Control with S-Curve Motion Profiles

José C. Ortiz Hernández *, **David I. Rosas Almeida** *,

* *Facultad de Ingeniería Mexicali, Universidad Autónoma de Baja California, Blvd. Benito Juárez S/N, Parcela 44, 21280 Mexicali, B.C., México (e-mail: jose.carlos.ortiz.hernandez@uabc.edu.mx, drosas@uabc.edu.mx).*

Abstract: Mecanum-wheeled mobile robots offer notable flexibility for navigating in conglomerated environments, as they can move without reorientation, in contrast to car-like robots constrained by kinematics. However, their control remains challenging due to the wheels' strong susceptibility to slippage and high-speed vibrations, leading to positioning and orientation errors. This manuscript presents an improved motion control using Active Disturbance Rejection Control and S-Curve motion profiles to mitigate peak responses. The structure is validated through simulations and experiments, and its performance is compared against step inputs and trapezoidal profiles.

Keywords: Mecanum mobile robot, Active Disturbance Rejection Control, Trapezoidal, S-Curve motion profile, Smooth reference

1. INTRODUCTION

1.1 Motivation

The growing interest in improving maneuverability across various applications has driven the design and development of mobile robots with omnidirectional motion, Shao et al. (2016). Several types of wheels have been designed to achieve omnidirectional motion, Akkad and Almusawi (2023). Among these, the Mecanum wheel patented by Ilon (1973), has been widely adopted due to its high payload capacity and easy installation, since it can be fixed to the chassis, which simplifies the robot design substantially and allows for forward, backward, lateral, diagonal, and rotational displacements, all while ideally maintaining the robot's original orientation. Such maneuverability makes them attractive for navigation in conglomerated environments, Giurgiu et al. (2022). Notwithstanding these core strengths, the Mecanum wheel is highly susceptible to slippage and vibration, leading to significant positioning errors, and the demand for advanced motion control techniques, Taheri and Zhao (2020). A wide variety of motion control techniques have been explored to overcome these issues. For example, the Proportional-Integral-Derivative (PID) (Hernández and Almeida (2024b)), and novel variations, such as Self-Tuning Fuzzy PID (Malayjerdi et al. (2018)), modified PID with Kalman Filter (Le et al. (2025)), and Neural-PID tracking controllers (Szeremeta and Szuster (2022)). In addition to PID-based methods, more sophisticated approaches have also been proposed, including Sliding Mode Control Sun et al. (2024), Backstepping Control Jiang et al. (2022), and Model Predictive Control Moreno-Caireta et al. (2021). However, despite the several control techniques developed in this robotics field, a crucial aspect of control performance remains underexplored: the generation of reference trajectories. From an academic perspective, many studies rely on step inputs, which can induce abrupt jerks, Han (2009). A common alternative is using trapezoidal motion profiles, also known as ramp-like references, which mitigate these issues. Nevertheless, even these profiles introduce discontinuities in acceleration, resulting in residual jerks, Hernández and Almeida (2024a).

1.2 Research Question

Based on the background, the rest of this manuscript is tailored to the following research question (RQ)

- **RQ**: Can the motion performance of a Mecanum-wheeled mobile robot be enhanced by providing S-Curve reference motion profiles in combination with the Active Disturbance Rejection Control (ADRC)?

1.3 Contribution

This manuscript provides an enhanced motion control that combines S-Curve reference motion profiles with the ADRC to achieve a smooth response and more reliable performance in a Mecanum mobile robot, which will aid in mitigating jerks, vibrations, and slippage. The summarized contributions of this manuscript are as follows:

⋆ The first author acknowledges the Secretariat of Science, Humanities, Technology and Innovation of México for the grant number 6511be9fef569262a03fd45d.

https://doi.org/10.58571/CNCA.AMCA.2025.005

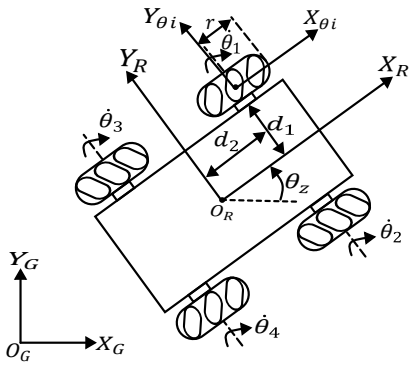

Fig. 1. Main parameters of the mobile robot for modeling.

- We provide a point of view to mitigate position errors and orientation angle deviations in Mecanum-wheeled mobile robots based on smooth reference motion profiles.
- We implement S-Curve motion profiles as reference inputs to provide a smoother response in the vehicle.
- We integrate the proposed S-Curve profiles with an Active Disturbance Rejection Control (ADRC) scheme to improve tracking performance under dynamic uncertainties and external disturbances.

1.4 Paper distribution

The remainder of this manuscript is organized as follows. The mathematical modeling is provided in Section 2. The complete control structure is presented in Section 3. In Section 4, simulation results are illustrated. The real-world experiments are in Section 5. Finally, conclusions and future research directions are in Section 6.

2. MATHEMATICAL MODELING

Consider a top view of a square-shaped mobile robot with four motorized Mecanum wheels, as shown in Figure 1, where X_G, O_G, Y_G denotes the fixed coordinate frame and X_R, O_R, Y_R represents the robot's body coordinate frame and is centered at the robot's geometric center, describing its real-time pose. The wheels are labeled as $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4$, corresponding to their rotational velocities in [rad/s] units. The wheel radius is r in [mm], and the distances from the robot center to the wheel axes along the X_R and Y_R directions are d_1 and d_2, respectively, in [mm]. The robot's configuration space is then denoted by its rectangular components x, y, and the robot's orientation θ_z, or just θ, with respect to the global frame. Local coordinate frames $O_{\theta_i}, X_{\theta_i}, Y_{\theta_i}$ are also assigned to each wheel to facilitate force and velocity decomposition. The following sections present the kinematic and dynamic models.

2.1 Kinematic model

Assuming pure rolling without slippage, the velocity relationship between the robot platform and its four Mecanum wheels is derived through force decomposition on the wheels and rollers. According to Taheri et al. (2015) we have

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\\ \dot{\theta}_4\end{bmatrix} = \underbrace{\frac{1}{r}\begin{bmatrix}1 & -1 & -(d_1+d_2)\\ 1 & 1 & (d_1+d_2)\\ 1 & 1 & -(d_1+d_2)\\ 1 & -1 & (d_1+d_2)\end{bmatrix}}_{\mathbf{J}}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}_z\end{bmatrix}, \tag{1}$$

which is called forward kinematics. The right-hand side terms $\dot{x}$, $\dot{y}$, and $\dot{z}$ are the global velocities associated with the body of the robot. Here, $\dot{x}$ and $\dot{y}$ represent the instantaneous longitudinal and lateral linear velocities of the mobile robot platform, respectively, given in [m/s] units. Conversely, the term $\dot{z}$ represents the instantaneous rotational velocity of the mobile robot platform, given in [rad/s]. The term $\mathbf{J}$ is a non-square matrix, and by obtaining its pseudo-inverse, denoted as $\mathbf{J}^*$, and ensuring the property that $\mathbf{J}^*\mathbf{J} = I \in \mathbb{R}^{3x3}$, we transform (1) into the inverse kinematics and have

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}_z\end{bmatrix} = \underbrace{\frac{r}{4}\begin{bmatrix}1 & 1 & 1 & 1\\ -1 & 1 & 1 & -1\\ \frac{-1}{d} & \frac{1}{d} & \frac{-1}{d} & \frac{1}{d}\end{bmatrix}}_{\mathbf{J}^*}\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\\ \dot{\theta}_4\end{bmatrix}, \tag{2}$$

with $d = (d_1 + d_2)$. Table 1 summarizes the main kinematic parameters for the mobile robot. The kinematic model establishes the relationship between the platform's linear and angular velocities and the rotational speeds of the individual wheels. In the following section, a dynamic motion model is derived.

Table 1. Main parameters

Parameter	Value	Parameter	Value
Weight [kg]	4.7	Distance d_1 [mm]	255
Height [mm]	90	Distance d_2 [mm]	85
Length [mm]	578	Number of wheels	4
Width [mm]	260	Rollers per wheel	8
Wheel radius r [mm]	30	Wheel thickness [mm]	32

2.2 Dynamic model

The equations of motion for the four-wheeled Mecanum robot on a flat surface are derived via the Lagrangian method, accounting for viscous friction as energy dissipation. The formulation is given by

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}(q,\dot{q})}{\partial \dot{q}}\right) - \frac{\partial \mathcal{L}(q,\dot{q})}{\partial q} = \tau - \frac{\partial D}{\partial \dot{q}}, \tag{3}$$

where the term $\mathcal{L}(q,\dot{q})$ refers to the Lagrangian function, which is defined as

$$\mathcal{L}(q,\dot{q}) = \mathcal{K}(q,\dot{q}) - \mathcal{P}(q). \tag{4}$$

The potential energy term is neglected if the robot operates on a planar surface, then, the kinetic energy is given by

$$\mathcal{K}(q,\dot{q}) = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}I_p\dot{\theta}_z^2 + \frac{1}{2}I_\omega\sum_{i=1}^{4}\dot{\theta}_i^2, \tag{5}$$

where m, I_p, and I_ω denote the robot's mass [kg], platform inertia [kg·m^2], and wheel inertia [kg·m^2], respectively. In addition, identical wheels, motors, and drivers are assumed. From (3), the energy loss due to the viscous friction term is expressed by

$$D = \frac{1}{2}\mu\left[\dot{x}^2 \;\; \dot{y}^2 \;\; \dot{\theta}_z^2\right]^T,$$

 https://doi.org/10.58571/CNCA.AMCA.2025.005

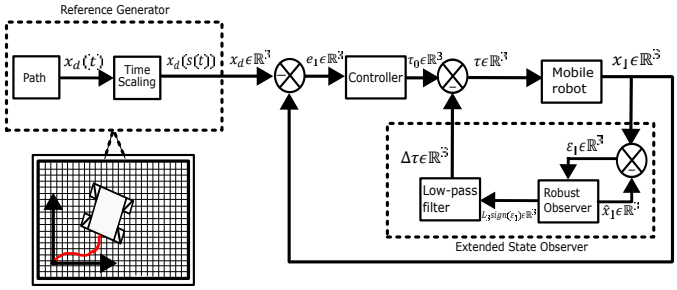

Fig. 2. The complete control scheme using Active Disturbance Rejection Control with time scaling and S-Curve motion profile.

where μ is

$$\mu = \begin{bmatrix} \mu_x & 0 & 0 \\ 0 & \mu_y & 0 \\ 0 & 0 & \mu_\theta \end{bmatrix},$$

where μ_x, μ_y, and μ_θ are the viscous friction coefficients along each coordinate direction, given in [kg·s]. By defining the generalized coordinates as $q = [x \ y \ \theta_z]^T$, subsequently, $\dot{q} = [\dot{x} \ \dot{y} \ \dot{\theta}_z]^T$, and combining (1), (3), (4), and (5), we yield the following compact model

$$\mathbf{M}\ddot{q} + \mu\dot{q} = \tau, \tag{6}$$

where

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{12} & 0 \\ 0 & 0 & m_{13} \end{bmatrix}, \quad \ddot{q} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta}_z \end{bmatrix}, \quad \tau = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_{\theta_z} \end{bmatrix}, \tag{7}$$

with $m_{11} = [m + \frac{4}{r^2} I_w]$, $m_{12} = [m + \frac{4}{r^2} I_w]$, $m_{13} = [I_p + \frac{4(d_1+d_2)^2}{r^2} I_w]$.

2.3 Modeling uncertainties

Issues such as model parameter uncertainty and external disturbances often arise in real-world applications. Thus, from (6) consider

$$(\mathbf{M} + \Delta\mathbf{M})\ddot{q} + (\mu + \Delta\mu)\dot{q} = \tau + \delta(t), \tag{8}$$

here, M and μ represent the nominal, known plant parameters, while $\Delta\mathbf{M}$ and $\Delta\mu$ denote the parametric uncertainties capturing unknown dynamics. The term $\delta(t)$ models time-varying external disturbances. Due to the linearity property of the Lagrangian systems, we yield to

$$\mathbf{M}\ddot{q} + \mu\dot{q} = \tau + \Gamma(t.\ddot{q}, \dot{q}), \tag{9}$$

with $\Gamma(t, \ddot{q}, \dot{q}) = \delta(t) - \Delta\mathbf{M}\ddot{q} - \Delta\mu\dot{q}$. For simplicity, consider $\Gamma(t, \ddot{q}, \dot{q}) = \Gamma(\cdot)$

3. ACTIVE DISTURBANCE REJECTION CONTROL

The ADRC structure, shown in Figure 2, consists of three main components: (i) a reference generator using S-Curve motion profiles to reduce abrupt responses and minimize slippage; (ii) an Extended State Observer that estimates and filters disturbances for real-time compensation; and (iii) a controller with disturbance rejection.

3.1 Reference generator

Definition 3.1. *A **path** is a time-independent function that defines the desired configuration of the robot, denoted as $x_d(t)$, where $t \in [0, T]$. A **time-scaling** function $s : [0, T] \to [0, 1]$ maps time to a normalized progression along the path. By combining both, the resulting reference $x_d(s(t))$ describes the desired robot's motion over time Lynch and Park (2017).*

A convenient form for the time scaling $s(t)$ mentioned above is a polynomial time scaling. In this regard, we have

Third-Order Polynomial (Cubic Time Scaling) A cubic polynomial satisfies the position and velocity constraints. Its expression is

$$s(t) = 3\left(\frac{t}{T}\right)^2 - 2\left(\frac{t}{T}\right)^3, \tag{10}$$

where $T > 0$ denotes the total duration of the motion along the trajectory, see Figure 3. It represents the time required to traverse the entire path from the initial to the final configuration, with $t \in [0, T]$ being the current time during motion execution. Thus, taking account (10), the complete reference generator with the path and S-Curve time scaling is given by

$$x_d(t) = \begin{cases} x_d(s(t)), & 0 \le t \le T \\ x_d(s(T) = 1), & t \ge T \end{cases} \tag{11}$$

3.2 Extended State Observer

Consider the plant (9). By defining $x_1 = q$ and $x_2 = \dot{q}$ as state variables, we yield to the following state-space representation

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= \mathbf{M}^{-1}[-\mu x_2 + \tau + \Gamma(\cdot)], \\ y &= x_1, \end{aligned} \tag{12}$$

where $y \in \mathbb{R}^3$ is the measurable output. To estimate the disturbance term $\Gamma(\cdot)$ and compensate for it, this work adopts the Extended State Observer philosophy, a core item in the ADRC framework, by implementing a robust observer, followed by a low-pass filter. The observer design is based on the structure proposed in Rosas Almeida et al. (2007) and is given by

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 + C_1(y - \hat{y}), \\ \dot{\hat{x}}_2 &= \mathbf{M}^{-1}[-\mu\hat{x}_2 + \tau] + C_2(y - \hat{y}) + C_3\,\mathrm{sign}(y - \hat{y}), \\ \hat{y} &= \hat{x}_1, \end{aligned} \tag{13}$$

where $\hat{x}_1 \in \mathbb{R}^3$ and $\hat{x}_2 \in \mathbb{R}^3$ represent the estimated states of the observer. The matrices C_1, C_2, and C_3 are diagonal and defined as $C_1 = \mathrm{diag}\{c_{11}, c_{12}, c_{13}\}$, $C_2 = \mathrm{diag}\{c_{21}, c_{22}, c_{23}\}$, and $C_3 = \mathrm{diag}\{c_{31}, c_{32}, c_{33}\}$. The discontinuous term is $\mathrm{sign}(y - \hat{y}) = [\mathrm{sign}(y_x - \hat{y}_x), \mathrm{sign}(y_y - \hat{y}_y), \mathrm{sign}(y_{\theta_z} - \hat{y}_{\theta_z})]^T$. To analyze the convergence behavior of the observer, we define the estimation errors as $\epsilon_1 = x_1 -$

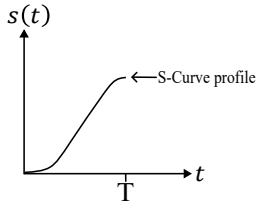

Fig. 3. Behavior of the third-order polynomial time scaling which is used for S-Curve reference motion profile.

 https://doi.org/10.58571/CNCA.AMCA.2025.005

$\hat{x}_1$ and $\epsilon_2 = x_2 - \hat{x}_2$. The observer error dynamics are then described by

$$\dot{\epsilon}_1 = \epsilon_2 - C_1\epsilon_1,$$
$$\dot{\epsilon}_2 = -\mathbf{M}^{-1}\mu\epsilon_2 - C_2\epsilon_1 - C_3\,\mathrm{sign}(\epsilon_1) + \mathbf{M}^{-1}\Gamma(\cdot).$$

By introducing the coordinate transformation $z_1 = \epsilon_1$ and $z_2 = \epsilon_2 - C_1\epsilon_1$, the error dynamics can be rewritten as

$$\begin{aligned}\dot{z}_1 &= z_2,\\ \dot{z}_2 &= -(\mathbf{M}^{-1}\mu + C_1)z_2 - (\mathbf{M}^{-1}\mu + C_2)z_1\\ &\quad - C_3\,\mathrm{sign}(z_1) + \mathbf{M}^{-1}\Gamma(\cdot).\end{aligned} \tag{14}$$

Following the procedure in Rosas Almeida et al. (2007), suitable gain values C_1, C_2, and C_3 are selected to ensure that the origin $z_1 = z_2 = 0$ is asymptotically stable, leading to the convergence of $\hat{x}_1 \to x_1$ and $\hat{x}_2 \to x_2$. From (14), it is observed that in the sliding regime $z_1 = z_2 = 0$, a second-order sliding mode arises, with the equivalent control given by $u_{eq} = \Gamma(\cdot)$. As established in Utkin (2013), this equivalent control corresponds to the slow-varying component of the discontinuous term in (14). Hence, the disturbance can be extracted using a low-pass filter, provided its cutoff frequency exceeds the bandwidth of the slow dynamics. Based on this, we propose a second-order Butterworth low-pass filter.

$$\frac{Y_i(s)}{U_i(s)} = \frac{\omega_{c_i}^2}{s^2 + \sqrt{2}\omega_{c_i} + \omega_{c_i}^2}, \tag{15}$$

with $i = x, y, \theta$. The term ω_{ci} is the cutoff frequency of each filter in [rad/s] units. The challenge in selecting ω_{ci} lies in setting a cutoff frequency that balances response speed and noise attenuation, ensuring minimal delay while maintaining signal integrity. The filter input is given by $\mathrm{sign}(y - \hat{y})$, and denoting the output of the filter as $y_f \in \mathbf{R}^3$ we have

$$\lim_{t\to\infty} y_f = \hat{\Gamma}(\cdot) \approx \Gamma(\cdot). \tag{16}$$

3.3 Controller

Lemma 1. *Consider a Mecanum-wheeled mobile robot model described by (12), with the ESO items (13) and (15). The control objective is given by*

$$\lim_{t\to\infty} |x_1 - x_d(t)| = 0.$$

The control input to achieve an exponentially stable equilibrium point is

$$\tau = \underbrace{\mathbf{M}[-K_1e_1 - K_2e_2 + \mu\mathbf{M}^{-1} + \ddot{x}_d]}_{\tau_0} + \underbrace{\mathbf{M}\hat{\Gamma}(\cdot)}_{\Delta\tau}, \tag{17}$$

where τ_0 is the nominal controller term, and $\Delta\tau$ is the real-time disturbance rejection term. The gain matrices $K_1, K_2 \in \mathbb{R}^{3x3}$ are positive definite with $K_1 = \mathrm{diag}\{k_{11}, k_{12}, k_{13}\}$ and $K_2 = \mathrm{diag}\{k_{21}, k_{22}, k_{23}\}$.

Proof. Let us define the error as $e_1 = x_1 - x_d(t)$ and $e_2 = x_2 - \dot{x}_d(t)$, where $x_d(t) \in \mathbb{R}^3$ contains the reference trajectory generated by both the path and time scaling. The error dynamics are

$$\dot{e}_1 = e_2,$$
$$\dot{e}_2 = -\mathbf{M}^{-1}[-\mu(e_2 + \dot{x}_d) + \tau + \Gamma(\cdot)]. \tag{18}$$

Substituting (17) into (18) yields the closed-loop error dynamics

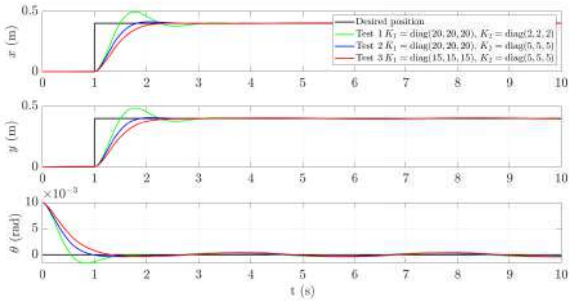

Fig. 4. Robot model response using step-input reference (simulation).

$$\dot{e}_1 = e_2,$$
$$\dot{e}_2 = -K_1e_1 - K_2e_2 + \Gamma(\cdot) - \hat{\Gamma}(\cdot). \tag{19}$$

Ideally, if the disturbance estimation were perfect, we would have

$$\lim_{t\to\infty} |\Gamma(\cdot) - \hat{\Gamma}(\cdot)| = 0.$$

Thus, this term is treated as a vanishing disturbance. Under these conditions, with an appropriate selection of matrices K_1 and K_2, the system's origin in (19) can be exponentially stable. Nevertheless, with a real-world approach, due mainly to the lag introduced by the filtering in the disturbance recovery stage, it stands out that

$$\lim_{t\to\infty} |\Gamma(\cdot) - \hat{\Gamma}(\cdot)| \leq \rho_0,$$

where $\rho_0 > 0$ is a small constant. According to the stability results for perturbed systems presented in Khalil (2002), the system trajectories converge to a small neighborhood of the origin of (19). □

4. SIMULATION RESULTS

Simulation results are presented to validate the closed-loop stability of the proposed control scheme. Using the reference (11), the ESO elements (13)–(15), and the controller (17), the following gains were applied: $C_1 = \mathrm{diag}\,250$, $C_2 = \mathrm{diag}\,15$, $C_3 = \mathrm{diag}\,2.5$, and $\omega_{ci} = \mathrm{diag}\,17.5, 25, 20$. Performance under the ADRC framework with the S-Curve reference was compared against step and trapezoidal inputs, with motion along the upper-left diagonal. Simulations were programmed in Matlab and Simulink. Figure 4 illustrates the behavior of the robot model when a step input is used as a reference. As shown, the response exhibits noticeable overshoot. Although this can be mitigated through trade-off-based tuning, it results in a significantly longer convergence time. The trapezoidal profile in Figure 5 improves settling time but still produces slight overshoot due to acceleration discontinuities. The S-curve profile in Figure 6 achieves the smoothest response, eliminating jerks and improving system reliability, and the error plot signal is shown in Figure 7.

5. REAL-WORLD EXPERIMENTS

This section presents real-world experiments evidence for the complete control structure depicted in previous sections. The experiments were conducted with a data acquisition dSPACE DS1103, and the control algorithms were programmed using MATLAB and Simulink. Figure 8 illustrates the prototype developed for this implementation. It

 https://doi.org/10.58571/CNCA.AMCA.2025.005

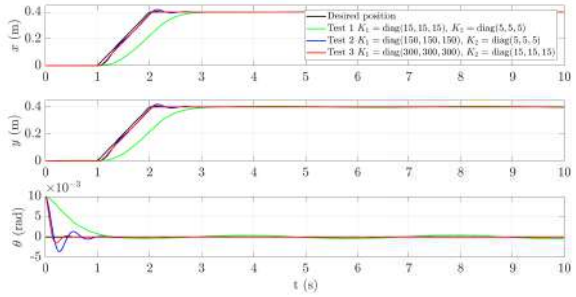

Fig. 5. Robot model response using trapezoidal motion profile (simulation).

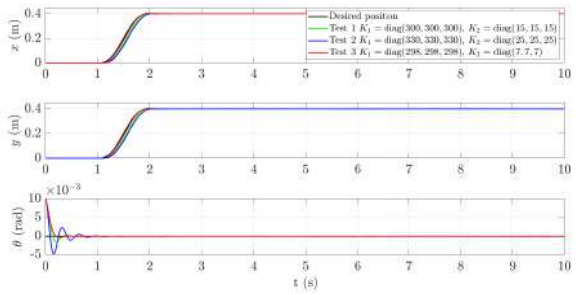

Fig. 6. Robot model response using S-Curve motion profile (simulation).

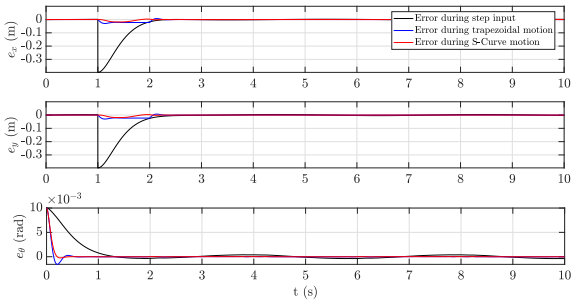

Fig. 7. Error history signals for Test 3 (simulation).

includes a rectangular platform mounted on four Mecanum wheels, directly coupled with direct current (DC) motors, enabling mobility. The DC motors include encoders for velocity and position estimation. In addition, the robot contains an Inertial Measurement Unit (IMU) for angle orientation measurement. Three reference signals were tested to evaluate the system performance: a step input, a trapezoidal velocity profile, and an S-Curve profile. The desired position computed was the upper-left displacement for x and y coordinates, while θ was intended to remain the same from start to end. The corresponding responses of the robot in the x, y, and θ coordinates are shown in Figures 9, 10, and 11. These plots compare the desired position with the actual measured position. The results show that the robot can track all references with good accuracy. However, differences in tracking behavior can be observed depending on the reference type employed. For instance, as summarized in Table 2, the S-curve reference yielded the smallest overshoot $M_p(i)\%$ across all coordinates $i = x, y, \theta$, with values of only 4.87%, 5.02%, and

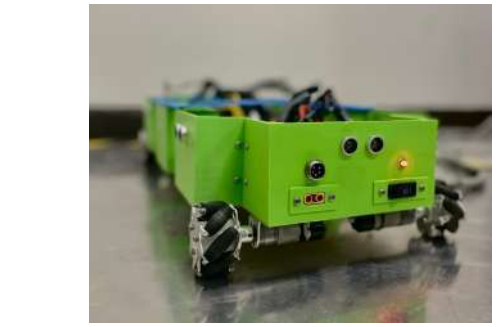

Fig. 8. Appreciative view of the prototype.

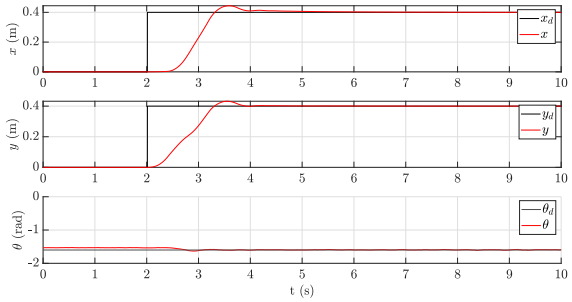

Fig. 9. Real robot response using step-input reference (real-world experiments).

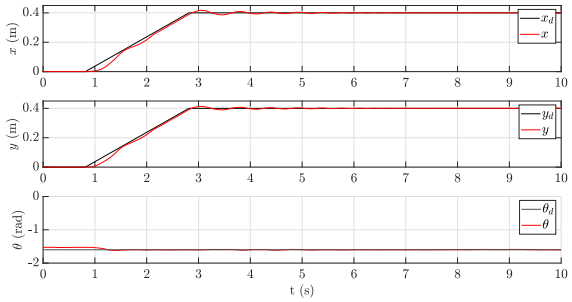

Fig. 10. Real robot response using trapezoidal profile reference (real-world experiments).

Table 2. Overshoot response comparison

Coordinate	Step-input	Trapezoid	S-Curve
$M_p(x)\%$	11.07%	8.66%	**4.87%**
$M_p(y)\%$	7.99%	7.50%	**5.02%**
$M_p(\theta)\%$	4.58%	4.74%	**2.60%**

2.60% for x, y, and θ, respectively. In contrast, the step-input reference produced higher overshoot, especially in the x-axis (11.07%). Table 3 presents each coordinate and reference type's Root Mean Square Error (RMSE). Again, the S-curve profile showed favorable results in the x-axis, with the lowest RMSE of 3.14×10^{-5} m. It is important to report that the trapezoidal profile performed slightly better for the y-axis case. Nevertheless, trapezoidal and S-Curve methods remained within relatively acceptable bounds for real-world implementations.

 https://doi.org/10.58571/CNCA.AMCA.2025.005

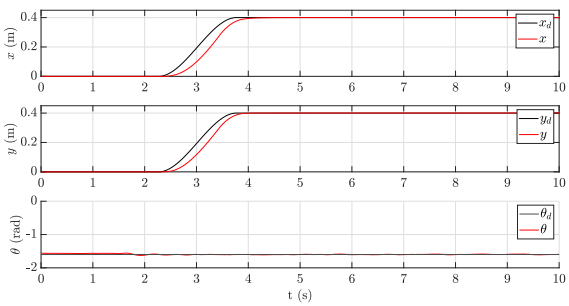

Fig. 11. Real robot response using S-Curve profile reference (real-world experiments).

6. CONCLUSION AND FUTURE RESEARCH DIRECTION

This manuscript addressed the challenges of enhancing motion control in Mecanum mobile robots with an approach based on S-Curve reference profile generation and Active Disturbance Rejection Control. The S-curve profile outperformed both step and trapezoidal references. For the x-axis, overshoot was reduced by 56.0% (from 11.07% to 4.87%) compared to the step input, and by 43.4% (from 8.6% to 4.87%) relative to the trapezoidal input. On the y-axis, improvements were 37.2% and 33.1%, respectively, for the orientation θ, overshoot decreased by 43.3% and 45.7%. These findings indicate that the S-Curve profile not only smoothens the reference trajectory but contributes significantly to minimizing transient deviations, thus enhancing implementation reliability for more scalable applications and overall control accuracy.

Future research directions include

- Extending the displacement range to evaluate the controller's performance under broader operational conditions.
- Performing multiple experimental trials and computing statistical metrics to evaluate repeatability and robustness provides stronger evidence of the advantages of the S-curve motion profile.
- Applying the S-curve profile directly to actuator signals within the control law to mitigate energy peaks in the control plant.

REFERENCES

Akkad, H. and Almusawi, H. (2023). A comparative review of omnidirectional wheel types for mobile robotics. *Rochester, NY*.

Giurgiu, T., Bârsan, G., Virca, I., and Pupăză, C. (2022). Mecanum wheeled platforms for special applications. In *International conference KNOWLEDGE-BASED ORGANIZATION*, volume 28, 44–51.

Han, J. (2009). From pid to active disturbance rejection control. *IEEE transactions on Industrial Electronics*, 56(3), 900–906.

Table 3. Root Mean Square Error comparison

Coordinate	Step-input	Trapezoid	S-Curve
$RMSE_{(x)}$	0.0011 (m)	$2.7131e^{-4}$ (m)	$3.1448e^{-5}$ (m)
$RMSE_{(y)}$	$3.7086e^{-4}$ (m)	$1.8441e^{-4}$ (m)	$6.3554e^{-4}$ (m)
$RMSE_{(\theta)}$	0.0071 (rad)	0.0044 (rad)	0.0630 (rad)

Hernández, J.C.O. and Almeida, D.I.R. (2024a). Experimental control approach of a mecanum-wheeled mobile robot for slippage error and energy consumption reduction on different surfaces. *Journal of Mechanical Science and Technology*, 1–10.

Hernández, J.C.O. and Almeida, D.I.R. (2024b). Kinematic control in a four-wheeled mecanum mobile robot for trajectory tracking. *The Journal of Engineering*, 2024(9), e70006.

Ilon, B. (1973). Directionally stable self propelled vehicle. US Patent 3,746,112.

Jiang, M., Chen, L., Wang, Y., and Wu, H. (2022). Adaptive backstepping control for mecanum-wheeled omnidirectional vehicle using neural networks. *IEEJ Transactions on Electrical and Electronic Engineering*, 17(3), 378–386.

Khalil, H. (2002). *Nonlinear Systems*. Pearson Education. Prentice Hall, New Jersey.

Le, T.L., Khang, N.G., and Thien, V.D. (2025). A study on the kalman filter based pid controller for mecanum-wheeled mobile robot. In *Journal of Physics: Conference Series*, volume 2949, 012029. IOP Publishing.

Lynch, K.M. and Park, F.C. (2017). *Modern robotics*. Cambridge University Press.

Malayjerdi, E., Kalani, H., and Malayjerdi, M. (2018). Self-tuning fuzzy pid control of a four-mecanum wheel omni-directional mobile platform. In *Electrical Engineering (ICEE), Iranian Conference on*, 816–820. IEEE.

Moreno-Caireta, I., Celaya, E., and Ros, L. (2021). Model predictive control for a mecanum-wheeled robot navigating among obstacles. *IFAC-PapersOnLine*, 54(6), 119–125.

Rosas Almeida, D.I., Alvarez, J., and Fridman, L. (2007). Robust observation and identification of ndof lagrangian systems. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 17(9), 842–861.

Shao, J., He, T., Jiang, J., and Zhang, Y. (2016). Recent patents on omni-directional wheel applied on wheeled mobile robot. *Recent Patents on Mechanical Engineering*, 9(3), 215–221.

Sun, Z., Li, Z., Xie, H., Zheng, Y., Zheng, J., and Chen, B. (2024). Precise trajectory tracking of mecanum-wheeled omnidirectional mobile robots via a novel fixed-time sliding mode control approach. *Control Theory and Technology*, 22(4), 596–611.

Szeremeta, M. and Szuster, M. (2022). Neural tracking control of a four-wheeled mobile robot with mecanum wheels. *Applied Sciences*, 12(11), 5322.

Taheri, H., Qiao, B., and Ghaeminezhad, N. (2015). Kinematic model of a four mecanum wheeled mobile robot. *International journal of computer applications*, 113(3).

Taheri, H. and Zhao, C.X. (2020). Omnidirectional mobile robots, mechanisms and navigation approaches. *Mechanism and Machine Theory*, 153, 103958.

Utkin, V.I. (2013). *Sliding modes in control and optimization*. Springer Science & Business Media.

 https://doi.org/10.58571/CNCA.AMCA.2025.005