International Journal of Mechatronics and Applied Mechanics
2025, Issue 19, Vol. 1

PERFORMANCE COMPARISON OF FAST TERMINAL AND PRESCRIBED PERFORMANCE SLIDING MODE CONTROL IN ROBOTIC SYSTEMS

Nhu Thanh Vo[1] [0000-0001-5383-3119], Hoai Nam Le[1] [0000-0001-5215-8289]*
[1]Faculty of Mechanical Engineering, The University of Danang – University of Science and Technology, 54 Nguyen Luong Bang, Lien Chieu, Danang, Vietnam
Email: lehoainam@dut.udn.vn

Abstract - This paper presents two advanced Sliding Mode Control methods for trajectory tracking of a two-degree-of-freedom (2-DoF) robot: Sliding Mode Control combined with Fast Terminal (FTSMC) and SMC integrated with Prescribed Performance Control (PPSMC). The first method, FTSMC, aims to improve the transient response and convergence speed while addressing the chattering issue and maintaining robustness. The second method, PPSMC, ensures precise trajectory tracking, guarantees convergence time, and manages tracking errors within predefined performance boundaries. The stability of both control methods is verified using Lyapunov stability theory. Extensive simulation results for a 2-DoF robot tracking a circular trajectory validate the effectiveness of the proposed methods. The results demonstrate that FTSMC enhances tracking performance and reduces overshoot, while PPSMC ensures smoother control signals and stable performance within the desired bounds.

Keywords: Sliding Mode Control; Fast Terminal; Prescribed Performance; 2-DoF robot; Trajectory tracking control.

1. Introduction

A robotic system is a programmable automated device designed to perform industrial tasks under a control system based on predetermined instructions. These systems complement or replace human workers in industrial production by performing complex and repetitive tasks with high precision, thereby improving efficiency and reducing human exposure to hazardous working environments. The applications of robotic systems have expanded significantly to encompass increasingly sophisticated tasks, including precision welding, complex assembly operations, and medical procedures [1-2].

The two-degree-of-freedom (2-DoF) robot is a fundamental nonlinear system widely utilised in research and education for control theory studies. Developing controllers for 2-DoF robots has attracted significant attention from researchers, with a primary focus on achieving stable control [1, 3] and precise trajectory tracking [4]. Various control methods have been explored to enable 2-DoF robots to follow desired trajectories. The proportional-integral-derivative (PID) controller, renowned for its simplicity and widespread industrial application [5-8], exhibits limited effectiveness in nonlinear systems.

Sliding Mode Control (SMC) has emerged as a prominent approach in robot trajectory tracking control [9-14]. This strategy offers robustness against system uncertainties and disturbances, making it particularly effective for nonlinear systems [15]. SMC directs system states onto a predefined sliding surface and maintains them there, ensuring desired performance despite external perturbations. Building upon the fundamental principles of SMC, researchers have developed several advanced control methodologies, including: adaptive SMC, which dynamically adjusts control parameters to accommodate system variations [16]; proportional-derivative SMC, which combines SMC's robustness with PD control's simplicity to enhance transient response [17]; and fuzzy SMC, which incorporates fuzzy logic for improved handling of uncertainties and nonlinearities [18]. These developments have significantly expanded SMC's applicability in modern robotic control systems.

Fast Terminal Sliding Mode Control (FTSMC) is an advanced control strategy that enhances transient

https://doi.org/10.17683/ijomam/issue19

response and convergence speed through finite-time convergence to equilibrium points [19-21]. It employs a terminal sliding surface that introduces a nonlinear convergence rate, enabling rapid and precise state convergence even under disturbances and uncertainties. Whilst traditional SMC is robust, it often suffers from chattering—a phenomenon characterised by rapid control mode switching, which can lead to mechanical wear and reduce precision. FTSMC mitigates this issue by incorporating a terminal attractor that smooths the convergence process whilst preserving robustness. This feature makes FTSMC particularly valuable in high-precision applications, such as advanced robotics and aerospace systems, where consistent performance under dynamic conditions is essential.

Prescribed Performance Control (PPC) is an advanced control strategy designed to ensure that system performance adheres to predefined specifications throughout its operational cycle [22]. Unlike traditional control methods that primarily focus on stability and general performance acceptability, PPC explicitly delineates desired performance characteristics, including transient behaviour, convergence rate, and steady-state error. By establishing these parameters, PPC transforms the control challenge into a constrained optimisation problem, requiring the controller to operate within defined bounds to achieve the desired outcomes.

PPC is particularly effective in applications demanding stringent performance standards, such as aerospace systems, automotive controls and high-precision manufacturing. Its ability to manage dynamic changes and external disturbances whilst maintaining adherence to specified performance criteria enhances its applicability in real-world scenarios with frequently varying operating conditions. By integrating PPC, systems achieve enhanced reliability and efficiency, ensuring optimal performance whilst minimising deviations from desired outcomes [23].

Previous research [24] proposed a Nonsingular Fast Terminal Sliding Mode Control with Prescribed Performance (PPNFTSMC) for robotic manipulators, integrating PPC and FTSMC into a unified framework to enhance transient and steady-state performance. Their methodology introduces a novel error performance index and utilises hyperbolic tangent transformations to achieve robust control and rapid convergence. In contrast, this paper examines FTSMC and PPSMC as distinct control strategies, providing a focused analysis of their individual strengths and specific applications. By analysing these methods separately, we provide deeper insights into their performance, particularly under external disturbances, contributing a complementary perspective to the combined approaches in previous studies.

This paper presents a modern PPSMC [25-27] strategy for a 2-DoF robot, extending the traditional SMC framework by integrating prescribed performance guarantees. Unlike conventional SMC methods that primarily minimise tracking error and ensure stability, PPSMC ensures that the tracking error remains within predefined performance bounds throughout operation, even in the presence of uncertainties and external disturbances [28].

Recent studies have explored related robotic control approaches, including approximate continuous fixed-time terminal SMC with prescribed performance [29], adaptive sliding mode disturbance rejection control with prescribed performance [30], and fast convergent nonsingular terminal sliding mode with adaptive prescribed performance control [31]. This work addresses existing research gaps by introducing a more generalised and robust PPSMC strategy that explicitly accounts for external disturbances, ensuring consistent performance in dynamic and uncertain environments.

The proposed approach is specifically designed for robotic manipulators, offering several advantages over existing methods, including enhanced operational flexibility, reduced computational complexity, and improved system robustness. Through comprehensive analytical examination and detailed simulations, this research demonstrates the effectiveness of the PPSMC strategy, particularly its capability to maintain precise trajectory tracking under diverse operating conditions.

Contributions of this paper:

- Development of a more robust and generalised PPSMC approach that ensures precise tracking performance even under external disturbances.
- Comprehensive simulation results, evaluating system performance for circular trajectories with disturbances, analysed using numerical metrics such as Root Mean Square Error (RMSE) and maximum error.
- Demonstration of significant performance improvements over traditional SMC and FTSMC, offering superior results in RMSE and maximum error.
- Enhanced ability of PPSMC to handle external disturbances, making substantial advancements in precision and robustness for complex robotic systems.

This paper is structured as follows:

- Section 2 details the dynamic modelling of the 2-DoF robot, providing a foundational framework for understanding system behaviour.
- Section 3 explores the development of advanced control strategies, specifically the integration of SMC with PPC, forming the basis of our proposed PPSMC method.
- Section 4 presents comprehensive simulation results obtained through MATLAB and Simulink, demonstrating the effectiveness and robustness of the proposed control strategies under various operating conditions.

- Section 5 concludes with a summary of the key findings, emphasising the significant improvements in performance and stability achieved through these advanced control techniques.

2. Dynamic Modelling of the 2-DoF Robot

The robotic manipulator consists of two rotational joints positioned within a defined coordinate system, as illustrated in Figure 1. A 2-DoF robot example [32-33], depicted in Figure 1, is described in the Oxy coordinate system. The primary link has a length of r_1 and a mass of m_1, whilst the secondary link has a length of r_2 and a mass of m_2. Both links rotate about the z-axis, with their respective rotation angles denoted as q_1 for the primary link and q_2 for the secondary link. These links are connected by a joint located at the end of the first link. The torques applied at the primary and secondary joints are represented by τ_1 and τ_2, respectively, with g denoting the gravitational acceleration constant.

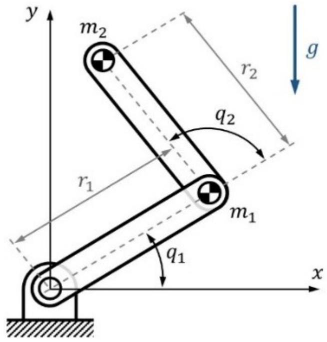


Figure 1: 2-DoF robot

The 2-DoF robot operates within the Oxy coordinate system, with the centres of gravity for both links positioned at their respective endpoints. The links are characterised by their respective masses, m_1 and m_2, and the moment of inertia for each link is considered negligible. This simplification facilitates the development of the dynamic model, allowing for a more straightforward derivation of the control laws whilst maintaining model reliability. By disregarding the moment of inertia, the analysis focuses on the dominant dynamics governed by the mass and positional parameters, which are crucial for trajectory tracking control applications. This approximation is commonly used in control system design when inertial effects are minimal, ensuring the model remains mathematically tractable whilst preserving the essential behaviour of the system.

Assumption 1 *In this study, we assume ideal conditions, where actuators operate within their limits and measurement noise is negligible. However, in practical scenarios, actuators may experience saturation, and measurement noise could degrade control performance.*

Let (x_{m1}, y_{m1}) and (x_{m2}, y_{m2}) represent the coordinates of m_1 and m_2, respectively. The position and velocity of m_1 are given by:

$$\begin{bmatrix} x_{m1} \\ y_{m1} \end{bmatrix} = \begin{bmatrix} r_1 \cos q_1 \\ r_1 \sin q_1 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} \dot{x}_{m1} \\ \dot{y}_{m1} \end{bmatrix} = \begin{bmatrix} -r_1 \sin q_1 \\ r_1 \cos q_1 \end{bmatrix} \cdot \dot{q}_1 \tag{2}$$

The position and velocity of m_2 are given by:

$$\begin{bmatrix} x_{m2} \\ y_{m2} \end{bmatrix} = \begin{bmatrix} r_1 \cos q_1 + r_2 \cos(q_1 + q_2) \\ r_1 \sin q_1 + r_2 \sin(q_1 + q_2) \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} \dot{x}_{m2} \\ \dot{y}_{m2} \end{bmatrix} = \begin{bmatrix} -r_1 \sin q_1 - r_2 \sin(q_1 + q_2) & -r_2 \sin(q_1 + q_2) \\ r_1 \cos q_1 + r_2 \cos(q_1 + q_2) & r_2 \cos(q_1 + q_2) \end{bmatrix} \cdot \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \tag{4}$$

The kinetic energy of the first and second links, corresponding to the kinetic energy at m_1 and m_2, are given by:

$$KE_1 = \frac{1}{2} m_1 (\dot{x}_{m1}^2 + \dot{y}_{m1}^2) \tag{5}$$

$$KE_2 = \frac{1}{2} m_2 (\dot{x}_{m2}^2 + \dot{y}_{m2}^2) \tag{6}$$

Combining equations (5) and (6), the total kinetic energy of the system is given by:

$$\begin{aligned} KE &= KE_1 + KE_2 \\ &= \frac{1}{2} [m_1 (\dot{x}_{m1}^2 + \dot{y}_{m1}^2) + m_2 (\dot{x}_{m2}^2 + \dot{y}_{m2}^2)] \end{aligned} \tag{7}$$

Equations (2) and (4) can be substituted into the kinetic energy equation (7):

$$KE = \frac{1}{2}(m_1 + m_2) r_1^2 \dot{q}_1^2 + \frac{1}{2} m_2 r_2^2 (\dot{q}_1^2 + \dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2) + m_2 r_1 r_2 \cos q_2 \, (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) \tag{8}$$

The total potential energy of the system is determined by the potential energies at the centres of mass m_1 and m_2 of the respective links.

$$PE_1 = m_1 g y_{m1} = m_1 g r_1 \sin q_1 \tag{9}$$

$$\begin{aligned} PE_2 = m_2 g y_{m2} = m_2 g [&r_1 \sin q_1 \\ &+ r_2 \sin(q_1 + q_2)] \end{aligned} \tag{10}$$

From equations (9) and (10), the total potential energy of the system is given by:

$$PE = PE_1 + PE_2 = m_1 g r_1 \sin q_1 + m_2 g[r_1 \sin q_1 + r_2 \sin(q_1 + q_2)] \quad (11)$$

The dynamic equation of the robot is established using the Euler–Lagrange method as follows:

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q}_i}\right) - \frac{\delta L}{\delta q_i} = Q_i \qquad (i = 1,2) \quad (12)$$

where $L = KE - PE$, with KE representing the total kinetic energy, PE representing the system's total potential energy, Q_i being the generalised force, and the variables q_i $(i = 1,2)$ referred to as generalised coordinates.

From equations (7) and (11), the Lagrangian of the system can be obtained as follows:

$$L = \frac{1}{2}(m_1 + m_2) r_1^2 \dot{q}_1^2 + \frac{1}{2} m_2 r_2^2 (\dot{q}_1^2 + \dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2) + m_2 r_1 r_2 \cos q_2 (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) - m_1 g r_1 \sin q_1 - m_2 g[r_1 \sin q_1 - r_2 \sin(q_1 + q_2)] \quad (13)$$

Based on the Euler-Lagrange equation (12) for the variables q_1 and q_2, we obtain:

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q}_1}\right) - \frac{\delta L}{\delta q_1} = \tau_1 \quad (14)$$

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q}_2}\right) - \frac{\delta L}{\delta q_2} = \tau_2 \quad (15)$$

After simplification, the force applied at joint 1 is given by:

$$\tau_1 = [(m_1 + m_2) r_1^2 + m_2 r_2^2 + 2 m_2 r_1 r_2 \cos q_2]\ddot{q}_1 + (m_2 r_2^2 + m_2 r_1 r_2 \cos q_2)\ddot{q}_2 - 2 m_2 r_1 r_2 \dot{q}_1 \dot{q}_2 \sin q_2 - m_2 r_1 r_2 \dot{q}_1^2 \sin q_2 + (m_1 + m_2) g r_1 \cos q_1 + m_2 g r_2 \cos(q_1 + q_2) \quad (16)$$

The force applied at joint 2 is given by:

$$\tau_2 = (m_2 r_2^2 + m_2 r_1 r_2 \cos q_2)\ddot{q}_1 + m_2 r_2^2 \ddot{q}_2 - m_2 r_1 r_2 \dot{q}_1 \dot{q}_2 \sin q_2 + m_2 g r_2 \cos(q_1 + q_2) \quad (17)$$

Equations (16) and (17) yield the general equation of the 2-DoF robot:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \quad (18)$$

where:

- $q = [q_1, q_2]^T$ is the vector representing the angular position;
- $\dot{q} = [\dot{q_1}, \dot{q_2}]^T$ is the vector representing the angular velocity;
- $M(q) \in R^{2\times 2}$ is the generalised inertia matrix:

$$M(q) = \begin{bmatrix} a & c \\ b & d \end{bmatrix} \quad (19)$$

with: $a = (m_1 + m_2) r_1^2 + m_2 r_2^2 + 2 m_2 r_1 r_2 \cos q_2$; $b = c = m_2 r_2^2 + m_2 r_1 r_2 \cos q_2$; $d = 0$;

- $C(q,\dot{q}) \in R^{2\times 2}$ is the Coriolis and centrifugal force matrix:

$$C(q,\dot{q}) = \begin{bmatrix} -2 m_2 r_1 r_2 \dot{q}_2 \sin q_2 & -m_2 r_1 r_2 \dot{q}_2 \sin q_2 \\ -m_2 r_1 r_2 \dot{q}_2 \sin q_2 & 0 \end{bmatrix} \quad (20)$$

- $G(q) \in R^{2\times 1}$ is the gravitational vector:

$$G(q) = \begin{bmatrix} (m_1 + m_2) g r_1 \cos q_1 + m_2 g r_2 \cos(q_1 + q_2) \\ m_2 g r_2 \cos(q_1 + q_2) \end{bmatrix} \quad (21)$$

where $\tau = [\tau_1, \tau_2]^T \in R^{2\times 1}$ are the joint torques.

For the calculation and control design, equation (18) is rewritten as follows:

$$\ddot{q} = M^{-1}(q)[-C(q,\dot{q})\dot{q} - G(q) + \tau] \quad (22)$$

and,

$$\ddot{q} = f(q,\dot{q}) + g(x)\tau \quad (23)$$

where $f(q,\dot{q}) = M^{-1}(q)[-C(q,\dot{q})\dot{q} - G(g)]$, and $g(q) = M^{-1}(q)$.

Assuming that $u = \tau$, $x = [x_1, x_2]^T$ where $x_1 = q = [q_1, q_2]^T$, $x_2 = \dot{q} = [\dot{q_1}, \dot{q_2}]^T$, we obtain equation (24), which is described in the state space:

$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = f(x) + g(x)u \end{cases} \quad (24)$$

in which $f(x) \in R^2$ and $g(x) \in R^2$ are smooth nonlinear vector fields. This assumption allows the system dynamics to be expressed in a state-space form, which is essential for the application of state-space control techniques such as SMC. The direct relationship between the control input and the generalised torque simplifies the design and analysis of the controller, ensuring that the control law can effectively regulate the system's state variables, such as positions and velocities.

The system dynamics $f(x)$ can be represented as:

$$f(x) = f_n(x) + \Delta f(x) \quad (25)$$

where:

- $f_n(x)$ represents the nominal part of the system dynamics, which is the known or estimated dynamics based on the system's model.
- $\Delta f(x)$ represents the uncertain part of the system dynamics, accounting for discrepancies, unmodelled dynamics, and external disturbances.

After developing the mathematical model of the 2-DoF robot, the following section focuses on designing control strategies. This section delves into various methodologies aimed at enhancing stability and performance, including the synthesis of controllers and the evaluation of their effectiveness through simulation results.

3. Controller Design for the 2-DoF Robot

Equation (24) is nonlinear, where the input u represents the torque applied to the robot but is unknown. Therefore, controlling the force applied to the joints is essential to achieve the desired final position. In this section, we discuss three controllers for the 2-DoF robot: Sliding Mode Control (SMC), Fast Terminal SMC (FTSMC), and Prescribed Performance SMC (PPSMC).

Figure 2 illustrates the control architecture for a 2-DoF robot, where the goal is to track a reference trajectory in Cartesian space. The reference position, denoted by $[X_d, Y_d]$, is first input into an inverse kinematics block, which calculates the desired joint angles (q_d), velocities ($\dot{q_d}$), and accelerations ($\ddot{q_d}$) required to achieve the desired end-effector position. These values are then fed into the proposed controller, which generates the appropriate control signal u to command the robot joints. The actual joint angles q and velocities $\dot{q}$ are measured and fed back into the controller for continuous adjustment. Simultaneously, the forward kinematics block converts the actual joint angles into Cartesian coordinates $[X, Y]$, representing the end-effector's actual trajectory. This actual trajectory is then compared to the reference trajectory to evaluate the controller's tracking performance, ensuring precise and accurate trajectory tracking. The proposed controller in this architecture can be implemented as either SMC, FTSMC, or PPSMC, depending on the control strategy chosen to achieve precise trajectory tracking.

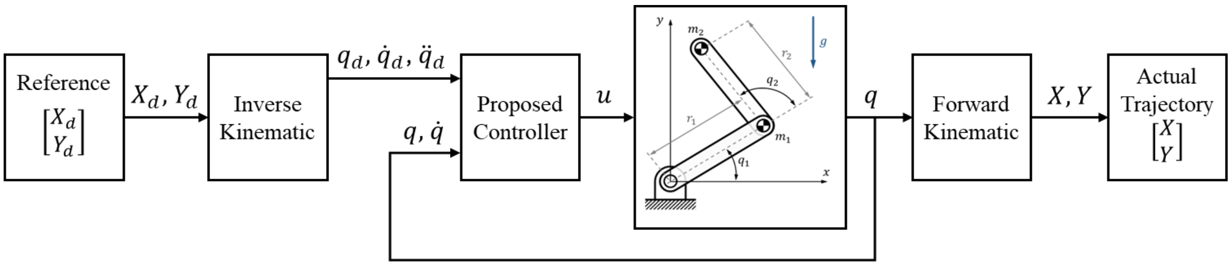


Figure 2: Control architecture

3.1. Sliding Mode Control

The objective of the SMC method is to drive the tracking error towards the sliding surface and subsequently ensure that it moves along this surface to reach the origin.

Let $e = x_1 - x_d$ and $\dot{e} = x_2 - \dot{x_d}$ be the trajectory tracking error vector and the corresponding velocity error vector, with $x_d = q_d = [q_{1d}, q_{2d}]^T \in R^2$ being the desired trajectory vector of the link angles. The sliding mode surface is designed as follows:

$$s = \dot{e} + \lambda e \quad (26)$$

where $s \in R^{2\times 2}$ represents the sliding variables and where $\lambda = diag(\lambda_1, \lambda_2)$ represents a positive definite diagonal matrix. This assumption is foundational to the SMC method, which relies on driving the system's state to a sliding surface and ensuring that it remains there. The choice of λ as a positive definite matrix ensures the convergence of the error dynamics, providing robustness against disturbances.

This assumption is standard in SMC design, where the sliding surface plays a critical role in achieving the desired tracking performance.

The first time derivative of s is as follows:

$$\dot{s} = \ddot{e} + \lambda\dot{e} = (\dot{x}_2 - \ddot{x}_d) + \lambda\dot{e} \quad (27)$$

From equations (24) and (27):

$$\dot{s} = \ddot{e} + \lambda\dot{e} = f(x) + g(x)u - \ddot{x}_d + \lambda\dot{e} \quad (28)$$

Based on a sliding mode function (26), the Lyapunov function is chosen as:

$$V = \frac{1}{2}s^T s \quad (29)$$

Derivative of both sides of (29), we have:

$$\dot{V} = s^T\dot{s} \quad (30)$$

In accordance with the Lyapunov stability criterion, for the system to be stable, a Lyapunov function must always be greater than 0, and its derivative must be less than or equal to 0.

Based on the reaching control law, we choose the following:

$$\dot{s} = -k_1 s - k_2 sign(s) \quad (31)$$

where $k_1, k_2 \in R^{2\times 2}$ are the positive definite diagonal matrices.

By substituting equation (31) into (30), the following equation can be obtained:

$$\dot{V} = s^T\left(-k_1 s - k_2 sign(s)\right)\dot{s} \\ = -k_1 s - k_2 sign(s) \quad (32)$$

which leads to:

$$\dot{V} = -k_1|s| - s^T k_2 s \leq 0, \forall s \quad (33)$$

By substituting equation (27) into (31), we obtain:

$$-k_1 s - k_2 sign(s) = f(x) + g(x)u - \ddot{x_d} + \lambda\dot{e} \quad (34)$$

From equation (34), the control law u for SMC is designed as follows:

$$u = -g(x)^{-1}[f(x) - \ddot{x_d} + \lambda\dot{e} + k_1 s \\ + k_2 sign(s)] \quad (35)$$

where $k_1 s$ provides the necessary feedback to drive the system towards the sliding surface and $k_2 sign(s)$ is introduced to counteract the uncertain dynamics $\Delta f(x)$ and maintain system robustness.

3.2. Fast Terminal Sliding Mode Control

FTSMC extends the conventional SMC method by incorporating a terminal attractor. This additional component generates a control signal that accelerates the system's convergence to the desired state, achieving faster response times than standard SMC.

The sliding surface of the FTSMC, which is based on the tracking error, is designed as follows:

$$s = \dot{e} + \alpha e + \beta e^{\frac{a}{b}} \quad (36)$$

where $\alpha, \beta \in R^{2\times 2}$ are positive definite diagonal matrices and where $a, b (a < b)$ are positive odd integers.

Remark 1 *The FTSMC method enhances the convergence speed and transient performance, but the terminal sliding surfaces used in FTSMC may encounter singularities when the sliding variable reaches critical values, potentially leading to undefined or excessively large control inputs. Addressing this challenge requires careful design of the terminal sliding surfaces to ensure that the control input remains bounded and that the system remains stable across all operating conditions.*

The trajectory tracking error $e(0) \neq 0$ reaching $e = 0$ within the time interval is given by:

$$t_s = \frac{b}{\alpha(b-a)} \ln \frac{\alpha e(0)^{\frac{a-b}{b}} + \beta}{\beta} \quad (37)$$

By appropriately designing α, β, a, and b, the system state can achieve the equilibrium point in a finite time t_s.

The first-time derivative of equation (36) is as follows:

$$\dot{s} = \ddot{e} + \alpha\dot{e} + \beta\frac{a}{b} e^{\frac{a}{b}-1}\dot{e} \\ = (\dot{x_2} - \ddot{x_d}) + \alpha\dot{e} + \beta\frac{a}{b} e^{\frac{a}{b}-1}\dot{e} \quad (38)$$

Substitute equation (24) into (38):

$$\dot{s} = f(x) + g(x)u - \ddot{x_d} + \alpha\dot{e} + \beta\frac{a}{b} e^{\frac{a}{b}-1}\dot{e} \quad (39)$$

The Lyapunov function can be defined as:

$$V = \frac{1}{2} s^T s \quad (40)$$

Take the derivative of equation (40):

$$\dot{V} = s^T \dot{s} \quad (41)$$

For the reaching control law, we choose:

$$\dot{s} = -k_3 s - k_4 sign(s) \quad (42)$$

where $k_3, k_4 \in R^{2\times 2}$ are positive definite diagonal matrices.

Substitute equation (39) into (42):

$$-k_3 s - k_4 sign(s) = f(x) + g(x)u - \ddot{x_d} + \alpha\dot{e} \\ + \beta\frac{a}{b} e^{\frac{a}{b}-1}\dot{e} \quad (43)$$

From equation (43), the control law u of the FTSMC is designed as follows:

$$u = -g(x)^{-1}\left[f(x) - \ddot{x_d} + \alpha\dot{e} + \beta\frac{a}{b} e^{\frac{a}{b}-1}\dot{e} \\ + k_3 s + k_4 sign(s)\right] \quad (44)$$

By substituting equation (42) into (41) yields:

$$\dot{V} = s^T\left(-k_3 s - k_4 sign(s)\right) \quad (45)$$

which leads to:

$$\dot{V} = -k_3|s| - s^T k_4 s \leq 0, \forall s \quad (46)$$

This implies the stability of the system. Note that $k_3 s$ is used to ensure that the system states converge rapidly to the sliding surface, and $k_4 sign(s)$ is introduced to counteract the effects of $\Delta f(x)$, thereby ensuring that the control system remains robust even in the presence of uncertainties and external disturbances.

3.3. Prescribed Performance Sliding Mode Control

3.3.1. Prescribed Performance function and error transformation

First, a prescribed performance function is chosen to ensure the desired performance and to stabilize the tracking error within a predefined boundary:

$$\psi(t) = (\psi_0 - \psi_\infty)\varepsilon^{-\mu t} + \psi_\infty \quad (47)$$

where ε is Euler's number, ψ_0 is the initial value, ψ_∞ is the final value, and $\psi_0 > \psi_\infty > 0$ and $\mu > 0$ are exponential constants.

The following condition is used to maintain the trajectory tracking error within a specified range:

$$-\delta\psi(t) < e(t) < \delta\psi(t), \forall t \geq 0 \quad (48)$$

where $0 < \delta < 1$ is a constant.

As shown in Figure 3, this condition holds provided that $e(t)$ remains within the boundary defined by equation (48) for $t \geq 0$.

In Figure 3, $-\delta\psi(t)$ represents the lower bound of the undershoot, whereas $\delta\psi(t)$ represents the upper bound of the overshoot. The parameter μ indicates the convergence speed. Therefore, designing parameters such as μ, δ, ψ_0 and ψ_∞ to optimize the steady state and performance is possible.

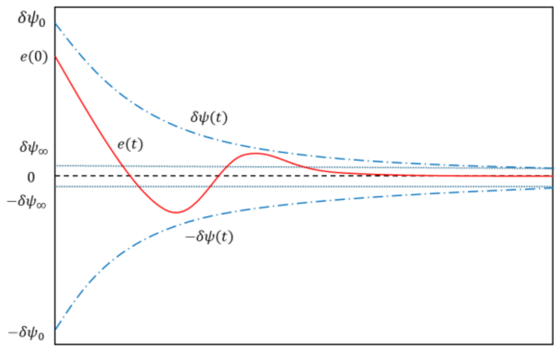


Figure 3: Prescribed performance function

Although equation (48) provides a useful framework for ensuring that $e(t)$ meets the prescribed performance criteria, directly developing controllers for this constrained system is challenging. A commonly used approach is error transformation, which converts equation (48) into the following "unconstrained" formulation:

$$e(t) = \sigma(z)\psi(t) \tag{49}$$

where z is a transformed error; $\sigma(z)$ is a transformed function; and $\sigma(z)$ is defined as follows:

$$\sigma(z) = \frac{\delta\varepsilon^{z} - \delta\varepsilon^{-z}}{\varepsilon^{z} + \varepsilon^{-z}} \tag{50}$$

Due to the transformation error z, being blocked, it is calculated as follows:

$$z = \sigma^{-1}\left[\frac{e(t)}{\psi(t)}\right] = \frac{1}{2}\ln\frac{v(t)+\delta}{\delta - v(t)} \tag{51}$$

where σ^{-1} is the inverse function and where $v(t) = \frac{e(t)}{\psi(t)}$ is the normalized tracking error. With $v(t) = \frac{e(t)}{\psi(t)}$, the first and second derivatives of z yield:

$$\begin{cases} \dot{z} = r\left(\dot{e} - \frac{e\dot{\psi}}{\psi}\right) \\ \ddot{z} = \dot{r}\vartheta_1(e,t) + r[\ddot{e} + \vartheta_2(e,t)] \end{cases} \tag{52}$$

where:

- $r = \frac{1}{2\psi}\left(\frac{1}{v+\delta} - \frac{1}{v-\delta}\right)$;
- $\vartheta_1(e,t) = \dot{e} - \frac{e\dot{\psi}}{\psi}$;
- $\vartheta_2(e,t) = -\frac{\dot{e}\dot{\psi}}{\psi} - \frac{e\ddot{\psi}}{\psi} + \frac{e\dot{\psi}^2}{\psi^2}$;
- $\dot{r} = -\frac{\dot{\psi}}{2\psi^2}\left(\frac{1}{v+\delta} - \frac{1}{v-\delta}\right) - \frac{\dot{e}\psi - e\dot{\psi}}{2\psi^3}\left[\frac{1}{(v+\delta)^2} - \frac{1}{(v-\delta)^2}\right]$.

3.3.2. Prescribed Performance Sliding Mode Controller Design

We can rewrite $\ddot{z}$ of equation (52) as:

$$\ddot{z} = \dot{r}\vartheta_1(e,t) + r[(\ddot{x}_2 - \ddot{x}_d) + \vartheta_2(e,t)] \tag{53}$$

By substituting equation (24) into (53) yields:

$$\ddot{z} = \dot{r}\vartheta_1(e,t) + r[f(x) + g(x)u - \ddot{x}_d + \vartheta_2(e,t)] \tag{54}$$

The sliding surface of PPSMC based on the tracking error can be designed as follows:

$$s = \dot{z} + cz \tag{55}$$

where $c \in R^{2\times 2}$ is a positive definite diagonal matrix.

Remark 2 *PPSMC involves a nonlinear error transformation based on the prescribed performance function, which can lead to singularities when the tracking error approaches the limits of the prescribed performance bounds. These singularities may cause the control input to become unbounded, affecting the stability of the control system. It is crucial to design the error transformation function and the associated control laws in a manner that prevents the occurrence of such singularities, ensuring robust and stable control performance.*

The time derivative of equation (55) is as follows:

$$\dot{s} = \ddot{z} + c\dot{z} \tag{56}$$

Substitute equation (54) into (56):

$$\dot{s} = \dot{r}\vartheta_1(e,t) + r[f(x) + g(x)u - \ddot{x}_d + \vartheta_2(e,t)] + c\dot{z} \tag{57}$$

The Lyapunov function can be chosen:

$$V = \frac{1}{2}s^T s \tag{58}$$

The time derivative of the Lyapunov function (58) is as follows:

$$\dot{V} = s^T\dot{s} \tag{59}$$

From equation (57) and equation (59), we have:

$$\dot{V} = s^T\left[\dot{r}\vartheta_1(e,t) + r\big(f(x) + g(x)u - \ddot{x}_d + \vartheta_2(e,t)\big) + c\dot{z}\right] \tag{60}$$

The control law is denoted as:

$$u = \hat{u} + g(x)^{-1}r^{-1}[-k_5 s - k_6 sign(s)] \tag{61}$$

where $k_5, k_6 \in R^{2\times 2}$ are positive definite diagonal matrices.

And,

$$\hat{u} = -g(x)^{-1}r^{-1}\left[\dot{r}\vartheta_1(e,t) + r\big(f(x) - \ddot{x}_d + \vartheta_2(e,t)\big) + c\dot{z}\right] \tag{62}$$

Substituting equations (61) and (62) into (60) yields:

$$\dot{V} = s^T[-k_5 s - k_6 sign(s)] \tag{63}$$

which leads to:

$$\dot{V} = -k_5|s| - s^T k_6 s \leq 0, \forall s \tag{64}$$

This implies the stability of the system. The control law u of PPSMC is defined as:

$$u = -g(x)^{-1} r^{-1} \left[\dot{r}\vartheta_1(e,t) + r\left(f(x) - \ddot{x}_d + \vartheta_2(e,t)\right) + c\dot{z} - k_5 s - k_6 sign(s) \right] \tag{65}$$

in which $k_5 s$ is utilized to guide the system state along the prescribed performance surface, ensuring that it meets the defined performance criteria, and $k_6 sign(s)$ is incorporated to handle the uncertain dynamics $\Delta f(x)$, providing the necessary robustness to ensure that the control system remains stable and effective despite uncertainties.

Remark 3 *The control gains k_1, k_2, k_3, k_4 (for SMC and FTSMC) and k_5, k_6 (for PPSMC) are chosen as positive definite diagonal matrices. Selecting the control gains as positive definite diagonal matrices is crucial for ensuring the stability and performance of the control system. These gains are designed to balance the trade-off between response speed and robustness, and their positive definiteness guarantees that the control inputs effectively stabilize the system. This assumption is key to achieving the desired transient and steady-state behaviour in the presence of system uncertainties.*

4. Simulation Results

This section presents the simulation results of the end-effector following a circular trajectory. Three control algorithms—SMC, FTSMC, and

PPSMC—are implemented to track these paths. In addition to trajectory tracking, the performance of the controllers is also evaluated under external disturbances. The evaluation criteria include key metrics such as the settling time, RMSE, and maximum error. The parameters of the 2-DoF robot are listed in Table 1.

Table 2 presents the control parameters used in the implementation of the three control strategies: SMC, FTSMC, and PPSMC.

Table 1. Parameters of the 2-DoF robot

Parameter	Symbol	Unit	Value
Length of link 1	r_1	m	0.3
Length of link 2	r_2	m	0.3
Mass of link 1	m_1	kg	5
Mass of link 2	m_2	kg	3
Gravity acceleration	g	kg/m^2	9.81

Remark 4 *The control parameters for SMC, FTSMC, and PPSMC are critical for ensuring the system's stability and performance. A systematic tuning process is required for each method. The parameters k_1 and k_2 for SMC, k_3 and k_4 for FTSMC, and k_5 and k_6 for PPSMC should be tuned by starting with conservative values and gradually increasing them based on system response. The performance of the system can be enhanced by balancing the trade-off between fast response times and robustness against disturbances.*

Table 2. Selection of control parameters for the SMC, FTSMC, and PPSMC methods

Method	Parameters
SMC	$\lambda = diag(10,10)$, $k_1 = diag(5,5)$, $k_2 = diag(5,5)$
FTSMC	$\alpha = diag(10,10)$, $\beta = diag(3,3)$, $a = 5, b = 7$, $k_3 = diag(5,5)$,$k_4 = diag(5,5)$
PPSMC	$\psi_0 = 0.3$, $\psi_\infty = 0.002$, $\delta = 0.8$, $\mu = 10$, $c = diag(100,100)$, $k_5 = diag(10,10)$, $k_6 = diag(20,20)$

4.1. Tracking Performance for a Circular Trajectory

The desired trajectory for the end-effector of the 2-DoF robot is a circular path with a center at point $(0.3, 0.3)$ and a radius of $0.1\ m$, $\omega = 2\pi f$ with $f = 0.1\ Hz$:

$$\begin{cases} X_d = 0.3 + 0.1\cos(\omega t) \\ Y_d = 0.3 + 0.1\sin(\omega t) \end{cases} \tag{66}$$

The initial values of the system are selected as $q_1(0) = \pi/3\ rad$; $q_2(0) = -\pi/3\ rad$; $\dot{q}_1(0) = 0\ rad/s$; and $\dot{q}_2(0) = 0\ rad/s$. Therefore, the initial values of $X(0)$ and $Y(0)$ are $X(0) = 0.45\ m$ and $Y(0) = 0.26\ m$, respectively.

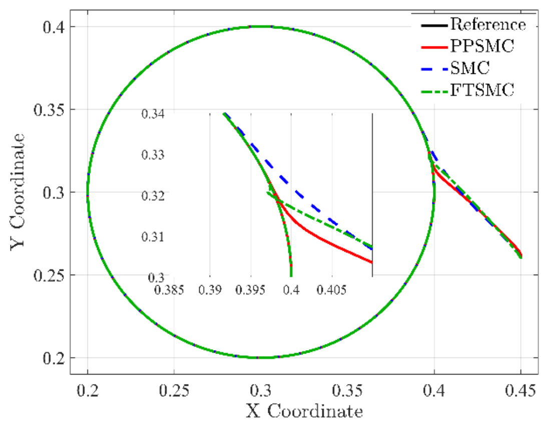
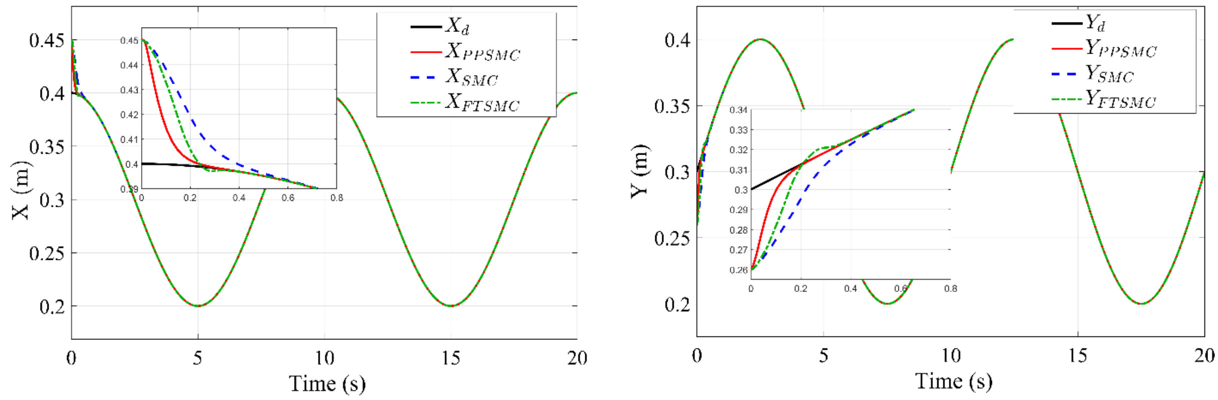
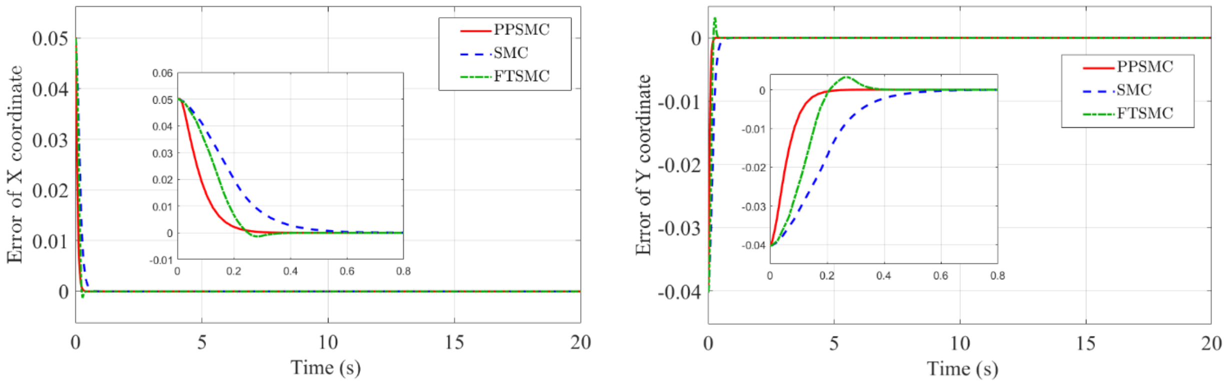


Figure 4: Trajectory of the end-effector



(a) Trajectory along the X-axis *(b) Trajectory along the Y-axis*

Figure 5: Orbit of the end-effector along the axes



(a) Position tracking error along the X-axis *(b) Position tracking error along the Y-axis*

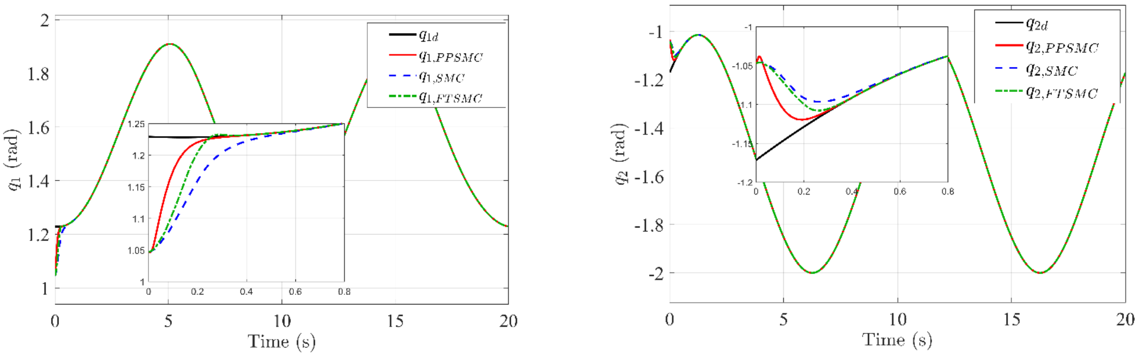
Figure 6: Position tracking error of the end-effector along the axes

Figures 4, 5 and 6 show the trajectory of the end effector of the 2-DoF robot along the X- and Y-axes and the tracking error for each axis as it executes a circular path under the influence of three control algorithms, SMC, FTSMC and PPSMC. The end-effector follows the desired circular trajectory when starting at the initial point $(X_0, Y_0) = (0.45, 0.26)$.

The trajectory responses of the controllers closely follow the desired path. Upon closer examination, we observe that the FTSMC controller has the fastest response, reaching the desired trajectory more quickly than the others. The PPSMC follows it and then the SMC. However, the FTSMC response is overshooting, whereas the reactions of the PPSMC and SMC are more stable.

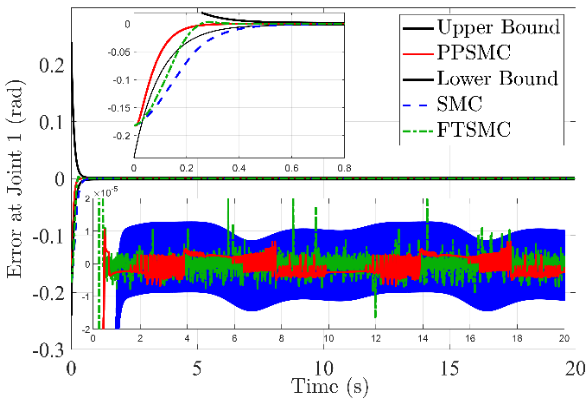
Figures 7 and 8 show the angular trajectory's response and the angular trajectory's tracking error for the different controllers.

The angular trajectories of all the controllers closely follow the desired trajectory. The angular response of the FTSMC controller is faster than that of the PPSMC and SMC controllers but is overshot. Upon closer inspection, it is evident that PPSMC achieves better convergence performance, staying within the predefined performance bounds.

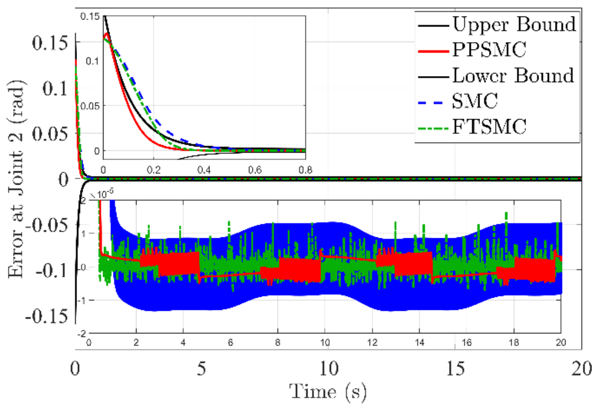


(a) The trajectory of q_1 *(b) The trajectory of q_2*

Figure 7: Trajectory of q

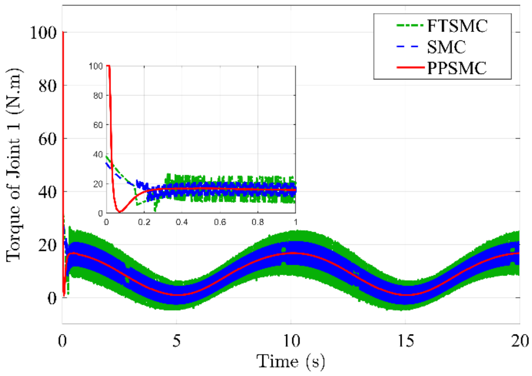


(a) Position tracking error of q_1

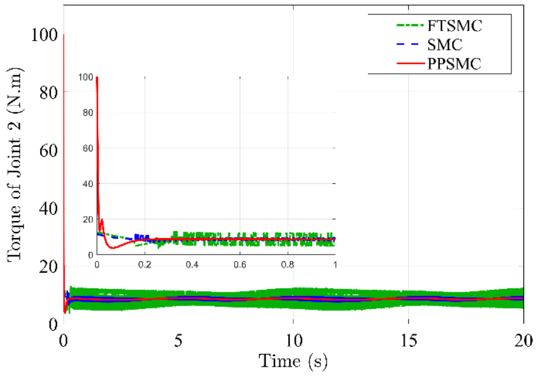


(b) Position tracking error of q_2

Figure 8: Position tracking error of the joint angles



(a) Control signal u_1



(b) Control signal u_2

Figure 9: Control signal

Figure 9 shows the control torque from three controllers for each joint of the 2-DoF robot. Even though the control torque applied to the joints is relatively high, the PPSMC provides smoother control torque without chattering in the control signals of the FTSMC and SMC controllers.

Table 3 presents the root mean square error (RMSE) and maximum error for the trajectory tracking of the end-effector in the circular path. Among the three control methods, PPSMC has the best performance, with the lowest RMSE and maximum error in both Cartesian coordinates and joint angles. Specifically, PPSMC achieves an RMSE of $7.8 \times 10^{-4}\ m$ in the X-axis and $6.2 \times 10^{-4}\ m$ in the Y-axis, significantly outperforming FTSMC and SMC.

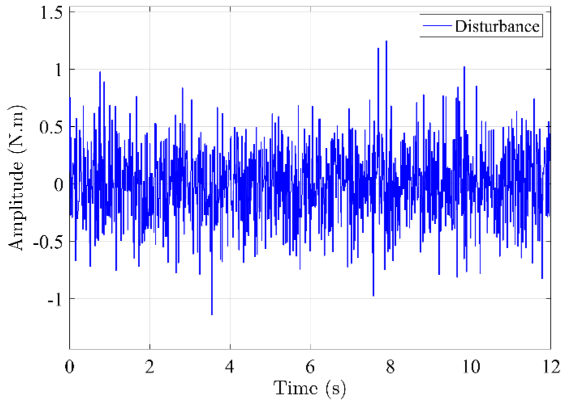
The errors in the joint angles (q_1 and q_2) also follow a similar trend, with PPSMC maintaining lower values than those of FTSMC and SMC. Overall, PPSMC demonstrates superior accuracy and robustness in tracking the circular trajectory, whilst FTSMC performs moderately well, and SMC shows the highest tracking errors among the three methods.

Table 3. RMSE and maximum error

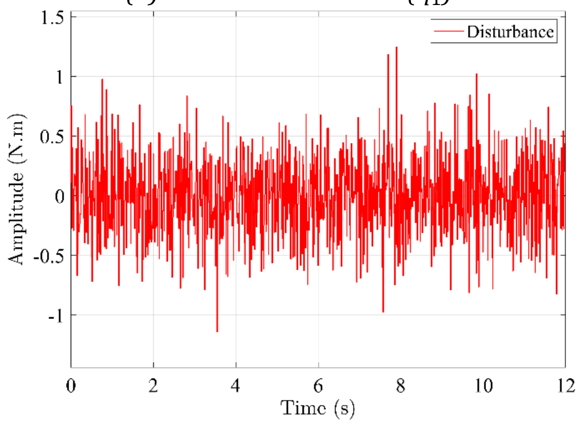
	RMSE			**Max Error**		
	PPSMC	**FTSMC**	**SMC**	**PPSMC**	**FTSMC**	**SMC**
$X(m)$	7.8×10^{-4}	2.8×10^{-3}	6.2×10^{-3}	5.0×10^{-2}	5.0×10^{-2}	5.0×10^{-2}
$Y(m)$	6.2×10^{-4}	2.3×10^{-3}	4.8×10^{-3}	4.0×10^{-2}	4.0×10^{-2}	4.0×10^{-2}
$q_1(rad)$	2.8×10^{-3}	9.9×10^{-3}	2.3×10^{-2}	1.8×10^{-1}	1.8×10^{-1}	5.0×10^{-1}
$q_2(rad)$	2.0×10^{-3}	6.1×10^{-3}	1.6×10^{-2}	1.8×10^{-1}	1.8×10^{-1}	5.0×10^{-1}

4.2 Control Performance under External Disturbances

In this subsection, the control performance of the system under external disturbances is evaluated. Two disturbances are applied to the system: one acting on link 1 (q_1) and the other on link 2 (q_2). The amplitude and frequency of the disturbances are shown in Figures 10a and 10b. These disturbances simulate external forces that may impact the tracking accuracy of the robotic in real-world scenarios.



(a) Disturbance on link 1 (q_1)



(b) Disturbance on link 2 (q_2)

Figure 10: Disturbance

Figure 11 presents the trajectory of the end-effector whilst tracking a circular path under these disturbances. As shown in the figure, the proposed PPSMC maintains the closest trajectory to the reference, with minimal deviation, outperforming both FTSMC and SMC. Despite the disturbances, PPSMC demonstrates superior robustness, keeping the tracking error within the prescribed performance bounds.

The RMSE values provide further evidence of the controller's effectiveness. For the X coordinate, PPSMC achieves an RMSE of $4.097 \times 10^{-3}\ m$, which is lower than that of SMC ($2.224 \times 10^{-3}\ m$) but higher than that of FTSMC ($1.296 \times 10^{-3}\ m$). In the Y coordinate, PPSMC records an RMSE of $2.986 \times 10^{-3}\ m$, outperforming both SMC ($1.808 \times 10^{-3}\ m$) and FTSMC ($1.107 \times 10^{-3}\ m$). These results highlight the strong tracking performance of PPSMC, especially in the presence of disturbances.

The RMSE, which quantifies the average magnitude of tracking errors, is a crucial metric for evaluating the precision of trajectory tracking. In industrial applications, where high accuracy and consistency are essential, achieving lower RMSE values directly translates to improved system reliability and performance. Our results demonstrate that the proposed PPSMC approach consistently achieves a lower RMSE than existing methods do, highlighting its effectiveness in ensuring precise trajectory tracking.

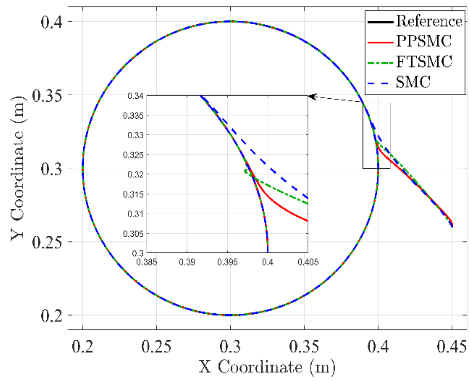
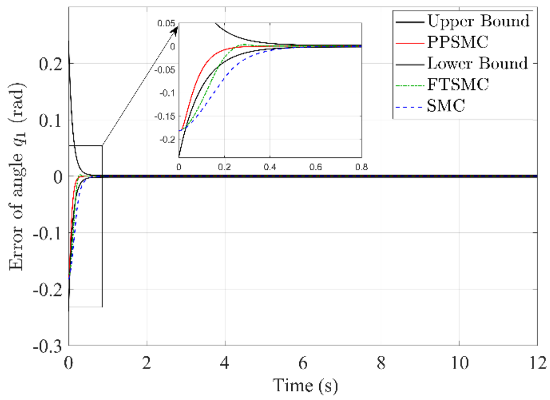
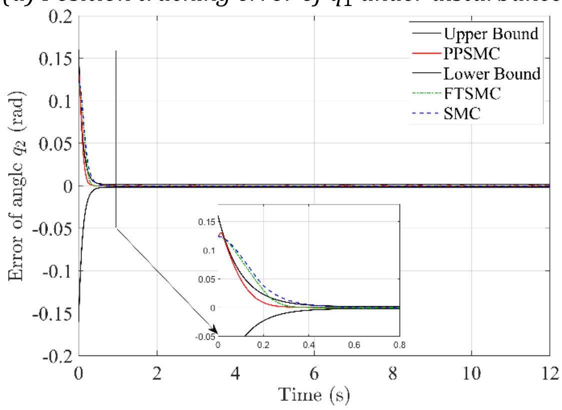


Figure 11: Trajectory of the end-effector under disturbance



(a) Position tracking error of q_1 under disturbance



(b) Position tracking error of q_2 under disturbance

Figure 12: Position tracking error of joint angles under disturbance

Figure 12 shows the tracking errors of the angular positions q_1 and q_2. The results show that PPSMC consistently achieves the lowest position tracking errors, staying within the upper and lower bounds. Compared with PPSMC, FTSMC achieves moderate performance but with greater errors, especially in the early stages of the trajectory.
The SMC shows the largest tracking errors, indicating reduced performance under external disturbances.

These results demonstrate that the proposed PPSMC method is highly effective in maintaining precise control and trajectory tracking even in the presence of external disturbances. By keeping the tracking error within predefined bounds, PPSMC ensures both robustness and accuracy, making it a suitable control strategy for robotic systems operating in uncertain environments.

5. Conclusions

This study has demonstrated the effectiveness of an advanced Sliding Mode Control method that integrates SMC with PPC for a 2-DOF robot. The proposed PPSMC method, developed via error transformation and a prescribed performance function, achieves precise tracking, ensures timely convergence, and maintains tracking errors within predefined performance boundaries. The analysis and simulation results indicate that the integration of PPC with SMC significantly outperforms both the standalone SMC and the FTSMC methods. By incorporating PPC, the limitations of SMC in guaranteeing consistent performance and convergence times have been effectively addressed, leading to enhanced overall system performance.

Future work will focus on integrating various techniques with SMC to increase its robustness in the presence of actuator saturation, measurement noise, and potential singularity issues. Anti-windup techniques will be explored to mitigate the effects of actuator saturation, whereas noise filtering and estimation methods, such as Kalman filters and extended state observers, will be employed to improve measurement accuracy. Additionally, the design of nonsingular sliding surfaces, particularly in FTSMC and PPSMC, will be a key focus to prevent control input from becoming unbounded or undefined. Adaptive control strategies and robust control techniques will be further investigated to ensure that the control system maintains optimal performance under varying operating conditions and uncertainties without encountering singularities.

References

[1] A.-D. Pham, B.H. Thai, P.V. Dang, N.T. Vo. “Analysis of the parametric configuration impact on BallBot control performance”. Int. J. Mech. Syst. Dyn., vol. 4, pp. 446-460, 2024. doi:10.1002/msd2.12133

[2] M.M. Gichane, J.B. Byiringiro, J.B., M. Benoussaad, *et al.* “Human-Centric Framework for Robotic Assembly for Industry of the Future”. *Int. J. Precis. Eng. Manuf.,* 2024. https://doi.org/10.1007/s12541-024-01176-4.

[3] A.-D. Pham and H.-J. Ahn. "Evaluation of input shaping methods for the nonlinear vibration system using a Furuta pendulum,” Journal of the Korean Society for Precision Engineering, vol. 38(11), pp. 827-833, 2020. doi: 10.7736/JKSPE.020.056.

[4] A.-T. Nguyen, A. Dequidt, V.-A. Nguyen, L. Vermeiren, and M. Dambrine, "Fuzzy descriptor tracking control with guaranteed L∞ error-bound for robot manipulators," Trans. Inst. Meas. Control, vol. 43, no. 6, pp. 1404–1415, 2021.

[5] N. M. Ghaleb and A. A. Aly, "Modelling and control of 2-DOF robot arm," Int. J. Emerg. Eng. Res. Technol., vol. 6, no. 11, pp. 24–31, 2018.

[6] M. A. A. Talib, A. S. Khamarazaman, M. Y. Salimun, J. Jaafar, and R. L. A. Shauri, "PID position control for 2-DOF robotic finger," in 2013 IEEE 4th Control and System Graduate Research Colloquium, 2013, pp. 152–157.

[7] N. A. Elkhateeb and R. I. Badr, "Novel PID tracking controller for 2DOF robotic manipulator system based on artificial bee colony algorithm," Electr. Control Commun. Eng., vol. 13(1, pp. 55–62, 2017.

[8] V.-A. Nguyen, L. Vermeiren, A. Dequidt, A.-T. Nguyen, M. Dambrine, and L. Cung, "Takagi-Sugeno fuzzy descriptor approach for trajectory control of a 2-DOF serial manipulator," in 2018 13th IEEE Conf. Industrial Electronics and Applications (ICIEA), 2018, pp. 1284–1289.

[9] H. N. Le, M.-K. Pham, D.-H. Pham, and T.-V.-A. Nguyen, "Trajectory tracking and stabilization of two-wheeled balancing mobile robot with hierarchical and sliding mode control," Int. J. Dyn. Control, vol. 13, no. 1, pp. 1–14, 2025.

[10] S. Ilgen, A. Durdu, E. Gülbahçe, and A. Cakan, "Optimal tuning of the SMC parameters for a two-link manipulator cosimulation control," Elektronika ir Elektrotechnika, 2021.

[11] D.-B. Pham, Q.-T. Dao, and T.-V.-A. Nguyen, "Optimized Hierarchical Sliding Mode Control for the Swing-Up and Stabilization of a Rotary Inverted Pendulum," Automation, vol. 5, no. 3, 2024.

[12] U. Zakia, M. Moallem, and C. Menon, "PID-SMC controller for a 2-DOF planar robot," in 2019 Int. Conf. Electrical, Computer and Communication Engineering (ECCE), 2019, pp. 1–5.

[13] M. Huseyinoglu and T. Abut, "Dynamic model and control of 2-DOF robotic arm," Eur. J. Technol. (EJT), vol. 8, no. 2, pp. 141–150, 2018.

[14] T.-V.-A. Nguyen, Q.-T. Dao, and N.-T. Bui, "Optimized fuzzy logic and sliding mode control for stability and disturbance rejection in rotary inverted pendulum," Sci. Rep., vol. 14(1), pp. 31116, 2024.

[15] N.T. Vo, A.-D. Pham, D.M.T. Nguyen, "Hierarchical Sliding Mode Control with Oscillation Compensation for Low-Cost Inverted Pendulum Systems Using Hardware-In-Loop," Arch. of Mech. Eng., vol. 72(1), pp. 31-55, 2025.

[16] A. Eltayeb, M. F. Rahmat, M. A. Eltoum, S. I. M. H. Sanhoury, and M. A. M. Basri, "Adaptive sliding mode control design for the 2-DOF robot arm manipulators," in 2019 Int. Conf. Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), 2019, pp. 1–5.

[17] M. Rahmani, H. Komijani, and M. H. Rahman, "New sliding mode control of 2-DOF robot manipulator based on extended grey wolf optimizer," Int. J. Control, Autom. Syst., vol. 18, pp. 1572–1580, 2020.

[18] K. Lochan and B. K. Roy, "Control of two-link 2-DOF robot manipulator using fuzzy logic techniques: a review," in Proc. 4th Int. Conf. Soft Comput. Problem Solving: SocProS 2014, vol. 1, Springer, 2014, pp. 499–511.

[19] L. Hu, P. Zhao, and Z. Lu, "Trajectory tracking control based on global fast terminal sliding mode for 2-DOF manipulator," in 2011 2nd Int. Conf. Intelligent Control and Information Processing, vol. 1, 2011, pp. 65–69.

[20] T. N. Truong, A. T. Vo, and H. J. Kang, "A backstepping global fast terminal sliding mode control for trajectory tracking control of industrial robotic manipulators," IEEE Access, vol. 9, pp. 31921–31931, 2021.

[21] A. T. Vo and H. J. Kang, "An adaptive neural nonsingular fast-terminal sliding-mode control for industrial robotic manipulators," Appl. Sci., vol. 8, no. 12, p. 2562, 2018.

[22] V.-V. Dinh, D.-H. Mai, M.-D. Duong, and Q.-T. Dao, "Prescribed Performance Function Based Sliding Mode Control of Opposing Pneumatic Artificial Muscles to Enhance Safety," J. Appl. Sci. Eng., vol. 27, no. 2, pp. 2117–2126, 2024.

[23] X. Bu, "Prescribed performance control approaches, applications and challenges: A comprehensive survey," Asian J. Control, vol. 25, no. 1, pp. 241–261, 2023.

[24] X. Zhang and R. Wang, "Nonsingular fast terminal sliding mode control of robotic manipulator with prescribed performance," J. Eng. Appl. Sci., vol. 71, no. 1, pp. 1–17, 2024.

[25] A. T. Vo, T. N. Truong, and H. J. Kang, "A Modified Adaptive Prescribed Performance Control Method for Motion Control Problems of Robotic Manipulators," Int. J. Mech. Eng. Robotics Res., vol. 11, no. 12, 2022.

[26] M. Wang and A. Yang, "Dynamic learning from adaptive neural control of robot manipulators with prescribed performance," IEEE Trans. Syst. Man Cybern. Syst., vol. 47, no. 8, pp. 2244–2255, 2017.

[27] S. Zheng, X. Niu, and C. Peng, "Adaptive supertwisting-like sliding mode control with prescribed performance for robot manipulators," J. Mech. Med. Biol., vol. 19, no. 08, p. 1940053, 2019.

[28] H. N. Le, P. V. Dang, A.-D. Pham, and N. T. Vo. "System identifications of a 2DOF pendulum controlled by QUBE-servo and its unwanted oscillation factors," Archive of Mechanical Engineering, 67(4):435–450, 2020.

[29] H. Sai, Z. Xu, C. Xia, and X. Sun, "Approximate continuous fixed-time terminal sliding mode control with prescribed performance for uncertain robotic manipulators," Nonlinear Dyn., vol. 110, no. 1, pp. 431–448, 2022.

[30] C. Jing, H. Xu, and X. Niu, "Adaptive sliding mode disturbance rejection control with prescribed performance for robotic manipulators," ISA Trans., vol. 91, pp. 41–51, 2019.

[31] Y. Zhang, L. Fang, T. Song, and M. Zhang, "Adaptive nonsingular fast terminal sliding mode based on prescribed performance for robot manipulators," Asian J. Control, vol. 25, no. 4, pp. 3253–3268, 2023.

[32] S. M. Mahil and A. Al-Durra, "Modelling analysis and simulation of 2-DOF robotic manipulator," in 2016 IEEE 59th Int. Midwest Symp. Circuits Syst. (MWSCAS), 2016, pp. 1–4.

[33] A. A. Okubanjo, O. K. Oyetola, M. O. Osifeko, O. O. Olaluwoye, and P. O. Alao, "Modelling of 2-DOF robot arm and control," Futo J. Ser. (FUTOJNLS), vol. 3, no. 2, pp. 80–92, 2017.