

Data Driven Learning Based Robust Control for Nonlinear Electromechanical Systems: A Case Study on SRV02

Chandar Kumar^{a,*}, Dur Muhammad Soomro^a, Najeeb Ur Rehman Malik^b

^a Department of Electrical Engineering, Faculty of Engineering and Applied Science, DHA Suffa University, Karachi – Pakistan

^b Faculty of Computing Information and Technology, DHA Suffa University, Karachi – Pakistan

* Corresponding author: Chandar Kumar, Email: chander.malhi@yahoo.com

Received: 07 August 2025, Revised: 16 September 2025, Accepted: 29 September 2025, Published: 01 October 2025

KEY WORDS

Neural network-based ILSMC
SRV02 motor
Iterative learning control
Sliding mode control
Robust tracking
Electromechanical systems

ABSTRACT

In this paper, a data-driven neural network-based iterative learning sliding mode control (NN-ILSMC) scheme is proposed for precise position and velocity control of the SRV02 motor-driven system. Controlling electromechanical systems like the SRV02, which exhibit second-order nonlinear behavior under significant parametric uncertainties, remains challenging in industrial applications; particularly when model inaccuracies degrade control performance. Traditional model-based ILSMC requires accurate system identification, which limits its effectiveness under high levels of uncertainty. This paper develops and validates a novel NN-ILSMC strategy that leverages neural network approximation to enhance learning from tracking errors, eliminating the need for explicit mathematical models. The proposed control approach is tested against model-based ILSMC using a Quanser SRV02 DC motor system subject to 30% parametric uncertainty, Coulomb friction, and time-varying external disturbances. Monte Carlo simulations with 20 trials and 50 learning iterations are conducted to assess performance using position and velocity RMSE metrics. Results demonstrate that the proposed NN-ILSMC achieves an 86.3% reduction in position RMSE and an 87.6% reduction in velocity RMSE compared to the model-based ILSMC. The NN-based data-driven approach provides superior accuracy, robustness, and practical scalability, making it highly suitable for real-time industrial automation systems where dynamic uncertainties and modeling challenges are prevalent.

1. Introduction

High-precision trajectory tracking is a critical requirement in a wide range of nonlinear systems, including robotic manipulators, autonomous vehicles, and industrial automation platforms. Despite significant advances in control theory, achieving consistent tracking performance under uncertain dynamics and external disturbances remains a challenging task, especially in environments where system parameters change over time or modeling accuracy is limited [1–4].

Iterative Learning Control (ILC) has emerged as a powerful framework for systems performing repetitive

tasks, enabling high-precision tracking by refining the control input across iterations [5–7]. Among various ILC approaches, Iterative Learning Sliding Mode Control (ILSMC) has gained attention for its ability to combine the robustness of sliding mode control (SMC) with the adaptation capability of ILC, making it particularly effective in handling disturbances and uncertainties [8–10]. These methods have been successfully applied in robotic manipulators [11], precision positioning systems [12], and motor-driven platforms [13], where high accuracy and robustness are essential.

However, most existing ILSMC strategies are model-based, requiring accurate knowledge of system dynamics to construct the control law [14–16]. These approaches often depend on precise modeling of inertia, damping, and input gain matrices to compute the control signal [17, 18]. While such methods yield strong performance in simulation, they face practical limitations in real-world implementation due to modeling errors, parameter drift, and unmodeled dynamics [19–21].

To address these limitations, recent studies have explored data-driven ILC approaches that learn the control input directly from data, without requiring system identification [22–26]. Additionally, the application of a neural network-based control strategy has been proposed to realize real-time approximation of the system uncertainties, where robustness is developed based on the adaptive approximating properties of a neural network (NNs) [27, 28, 29, 30]. These methods are particularly useful in environments where system dynamics are unknown or subject to change [31, 32]. In addition to conventional control, intelligent metaheuristic optimization methods like hybrid grey wolf-particle swarm optimization [33], moth-flame using PSO to transmit power [34], and gorilla troop optimization to solve engineering design problems [35] also reveal the efficiency of the learning-based and bio-inspired methods in addressing nonlinear and uncertain engineering issues.

Notable examples include iterative input shaping [36], neural network-based ILC [27, 28], and model-free PID tuning [37]. However, many of these approaches either lack strong theoretical guarantees or are not robust to high-frequency disturbances and measurement noise [38–39].

SMC is known for its strong robustness to matched disturbances and uncertainties [40–42]. Higher-order SMC techniques, such as the super-twisting algorithm, further reduce chattering while maintaining robustness [43–45]. However, most of these strategies are non-learning-based, and their performance remains static across iterations unless manually retuned [46].

Simultaneously, nonlinear control with reinforcement learning (RL)-based controllers has become more widespread due to the ability of policies to be learned by interacting with the system. Recent studies have integrated RL with sliding mode or Lyapunov methods to increase adaptability [47–49]. However, such approaches typically require large data sets and extensive training and are not well-suited to situations where noise or safety is an issue. Likewise, adaptive SMC schemes like fixed-time and super-twisting have been proposed to counter uncertainties

and enhance convergence [50]. Although such techniques are effective, they may still suffer from chattering concerns or structural dependence. In contrast, neural learning of data-driven ILC offers a more stable trade-off among adaptability, robustness, and theoretical guarantees.

Recent efforts have attempted to combine ILC with SMC to achieve both learning capability and robustness. For instance, [3] introduced a composite energy function-based ILC for uncertain systems, while [24] proposed a second-order ILSMC for robotic manipulators. Although these methods are effective, they are based on system models or cannot accommodate in any adaptive manner complex uncertainties in real time [24, 27, 29].

In this paper, a recent Neural Network-based Iterative Learning Sliding Mode Control (NN-ILSMC) framework is proposed. This architecture combines the robustness of SMC, the convergence properties of ILC, and the adaptive approximation capability of neural networks. The proposed NN-ILSMC employs a two-layer feed forward neural network configured to estimate system uncertainties during online execution, where the control law consists of a model-free SMC component and an iterative learning term. This approach eliminates the need for explicit system modeling while providing reliable tracking performance in the presence of parametric variations, unmodeled dynamics, and external disturbances [15, 27, 29, 30]. A comparative analysis between the model-based ILSMC and the proposed NN-ILSMC is presented. The NN-ILSMC leverages online learning via mini-batch gradient descent and incorporates robustness-enhancing mechanisms such as projection-based updates and dead-zone modifications to ensure stability and mitigate chattering [42, 43].

The integration of model-free learning, neural network-based adaptation, and robust control seems to be an effective way of precision tracking in uncertain conditions, whereas the possible extension to industrial and robotic systems are anticipated upon this study.

2. System Description and Control Objective

Consider a class of nonlinear dynamical systems characterized by a single-input single-output structure, governed by the following set of differential equations:

$$\begin{aligned} \dot{x}_i(t) &= x_{i+1}(t), i = 1, \dots, m-1 \\ \dot{x}_m(t) &= \theta^T(x, t)\xi(x, t) + b(x, t)u(t) \\ &\quad + d(t) \end{aligned}$$

$$y(t) = x_1(t), t \in [0, T] \quad (1)$$

In this formulation, the system state vector is defined as $x(t) = [x_1, x_2, \dots, x_m]^T$, while $u(t)$ and $y(t)$ denote the control input and measured output respectively. The function $b(x, t)$ represents a known, non-zero scalar function that captures the input gain dynamics. The term $\theta(x, t) \in \mathbb{R}^{p \times 1}$ is an unknown, time-varying parameter vector that must be learned during the control process, and $\xi(x, t) \in \mathbb{R}^{p \times 1}$ is a known regressor matrix function that encodes the system's structural information. The disturbance $d(t) \in \mathbb{R}$ represents an unspecified external perturbation acting on the system dynamics.

The control objective is to ensure that the system output $y(t)$ asymptotically follows a desired reference trajectory $y_d(t)$, which is assumed to be sufficiently smooth and linearly realizable up to the m^{th} derivative over the finite time interval $[0, T]$. To achieve this, we define the tracking error as $e(t) = y_d(t) - y(t)$, and we impose the following assumptions to ensure the feasibility and convergence of the control design.

First, it is assumed that all higher-order derivatives of the desired trajectory $y_{d(i)}(t)$ for $i = 1, \dots, m$ are well-defined and bounded over the interval $[0, T]$. Second, the unknown perturbation $d(t)$ is bounded in magnitude, satisfying the inequality:

$$|d(t)| \leq b_d, \forall t \in [0, T], \quad (2)$$

where $b_d > 0$ is a known bound. Third, at the beginning of each iteration, the system is initialized such that the tracking error and all its derivatives up to the m^{th} order vanish, i.e.,

$$e(0) = \dot{e}(0) = \ddot{e}(0) = \dots = e^{(m)}(0) = 0, \quad (3)$$

and the sliding surface $\sigma(0)$ is initialized to zero. This ensures consistent starting conditions across iterations and facilitates the learning process.

The ultimate goal is to synthesize a robust iterative learning control law $u(t)$, referred to as the second-order sliding mode iterative learning control (SSMILC), that guarantees the closed-loop system's output $y(t)$ converges to the desired trajectory $y_d(t)$ with arbitrary precision ϵ , such that:

$$\forall t \in [0, T], |y_d(t) - y(t)| \leq \epsilon, \quad (4)$$

To achieve this, a switching surface $\sigma(t)$ is introduced, defined as a linear combination of the tracking error and its derivatives:

$$\sigma(t) = c_1 e(t) + c_2 e'(t) + \dots + c_m e^{(m-1)}(t) = i = \sum_{i=1}^m c_i e^{(i)}, \quad (5)$$

Where $c_m = 1$ and the coefficients c_i for $i = 1, \dots, m-1$ are chosen such that the polynomial $s^m - 1 + c_m - 1s^{m-1} + \dots + c_1$ is Hurwitz, ensuring stability of the sliding dynamics. The time derivative of the switching surface is then derived accordingly, and the system dynamics are expressed in terms of the sliding variable $\sigma(t)$, enabling the design of a robust control law that drives the system states to the sliding surface in finite time and maintains them there despite uncertainties and disturbances.

This theoretical framework serves as the foundation for the design of the ILSMC control strategy, which integrates both sliding mode control and iterative learning mechanisms to achieve high-precision tracking in the presence of model uncertainties and external disturbances.

3. Model-Based ILSMC Design

The foundation of the proposed control strategy is rooted in second-order SMC concepts, originally developed in [15]. Based on this framework, the k^{th} iteration of the continuous SSMILC law is formulated as:

$$u_k(t) = b^{-1}(x_k, t) \left(\sum_{i=1}^m c_i y_d^{(i)}(t) - \sum_{i=1}^m c_i x_{i+1,k} - \dot{\theta}_k^T(t) \xi(x_k, t) - v_k(t) + \alpha_1 |\sigma_k|^{2/3} \text{sgn}(\sigma_k) + \alpha_3 \sigma_k(t) \right) \quad (6)$$

In this expression, k denotes the iteration index, $x_k(t) = [x_{1,k}, x_{2,k}, \dots, x_{m,k}]^T$ is the system state vector at the k^{th} iteration, and α_1, α_3 are positive control coefficients. The function $b(x_k, t)$ represents the input gain, assumed to be known and non-singular. The switching surface $\sigma_k(t)$ is defined as a linear combination of the tracking error and its derivatives, and serves as the core element of the sliding mode dynamics.

The parameter estimation term $\hat{\theta}_k(t)$ is introduced to account for the unknown system dynamics (x, t) . This estimate is updated iteratively using the following learning law:

$$\begin{aligned}\tilde{\theta}_k(t) = & \hat{\theta}_k(t) - \theta_k(t) \\ & - q\xi(x_k, t) \left(\frac{4\eta}{3} |\sigma_k|^{1/3} \text{sgn}(\sigma_k) \right. \\ & \left. + \gamma \sigma_k(t) \right) \quad (7)\end{aligned}$$

Here, η , and γ are positive constants that govern the learning rate and adaptation behavior. The regression matrix $\xi(x_k, t)$ encodes the system's structural information and facilitates the parameter estimation process.

To further enhance robustness against disturbances and unmodeled dynamics, an integral term $v_k(t)$ is incorporated into the control law. This term evolves according to the differential equation:

$$\dot{v}_k(t) = -\beta_1 \sigma_k(t) - \beta_2 |\sigma_k|^{1/3} \text{sgn}(\sigma_k) \quad (8)$$

where β_1 and β_2 are positive design parameters. The integral action helps to compensate for the effects of bounded disturbances $d(t)$, thereby improving the overall tracking performance.

Substituting the control law (6) into the system dynamics and sliding surface equations leads to the following expression for the time derivative of the sliding variable:

$$\begin{aligned}\dot{\sigma}_k(t) = & -\alpha_3 \sigma_k(t) + \dot{\theta}_k^T(t) \xi(x_k, t) + \\ & v_k(t) - d_k(t) - \alpha_1 |\sigma_k|^{2/3} \text{sgn}(\sigma_k) \quad (9)\end{aligned}$$

Where $\tilde{\theta}_k(t) = \hat{\theta}_k(t) - \theta_k(t)$ denotes the parameter estimation error. The integral term $v_k(t)$ plays a critical role in counteracting the influence of the disturbance $d_k(t)$, thereby improving the overall tracking performance.

It has been shown [15] that the second-order sliding mode is more robust to noise and disturbances than the first-order in view of smoothing of the chattering effect and its reduced control effort necessary for convergence. This property is especially useful in the case of iterative learning frameworks where performance over iterations must be maintained.

The stability and convergence of the designed ILSMC law are also ensured, under the assumptions of the previous section. In particular, the control law (6)-(8) is asymptotically stable in the case of the system (1)-(5) because the output

tracking error of (1) will asymptotically converge to some neighbourhood of zero as $t \rightarrow \infty$. This is precisely stated in the following theorem.

Theorem 1. The second-order sliding mode iterative learning control law (6)-(8) ensures that under the conditions that the model uncertainties of system (1) and the disturbance functions are bounded, the output trajectory of system (1) tracks the desired trajectory $y_d(t)$ asymptotically with an arbitrary accuracy.

4. Neural Network based Iterative Learning Sliding Mode Control (NN-ILSMC)

This section is the design, analysis, and implementation of a new NN-ILSMC framework that can be used to control uncertain nonlinear systems. This proposed architecture combines the stability of the SMC, the convergence behavior of ILC, and adaptive approximation properties of the NNs in order to realize high precision tracking performance, despite uncertainties in the model, and external disturbances.

It should be noted that the model-based ILSMC is formulated in continuous time to facilitate the theoretical analysis under the Lyapunov stability framework for proving convergence, whereas the NN-based ILSMC is implemented in discrete time. This reflects the inherent characteristics of data-driven control, which operates in real-time environments and relies on digital platforms with discrete-time sampling. Although the domain representations differ, the formulations are developed in a coherent manner, and the resulting control performance remains comparable within their respective frameworks.

4.1 Control Architecture

The NN-ILSMC architecture, illustrated in Fig. 1, employs a composite control law that combines a model-free sliding mode term, a neural network-based uncertainty compensator, and an iterative learning component. The equation for the total control input at time instant k is as follows:

$$\begin{aligned}u(k) = & \underbrace{\alpha s(k)}_{\text{Sliding mode term (robustness)}} + \\ & \underbrace{\hat{f}_\theta(\xi(k))}_{\text{NN compensator (uncertainties)}} + \\ & \underbrace{\sum_{i=1}^k \Gamma e(i)}_{\text{Iterative learning (performance)}} \quad (10)\end{aligned}$$

where $s(k)$ is the sliding surface variable, $e(k) = y_d(k) - y(k)$ is the tracking error and $\alpha > 0$ is the gain of SMC. The approximation of the NN is $\hat{f}_\theta(\xi(k))$, and the input vector is given by:

$$\xi(k) = [e(k) \quad e(k-1) \quad y(k) \quad \hat{w}(k) \quad k/N]^T \in \mathbb{R}^6, \quad (11)$$

Where $\hat{w}(k)$ represents an estimation of the disturbances. It is given an iterative rate of improvement of performance that is leveled using a learning gain of $\Gamma \in \mathbb{R}$ which determines the amount of cumulative error feedback.

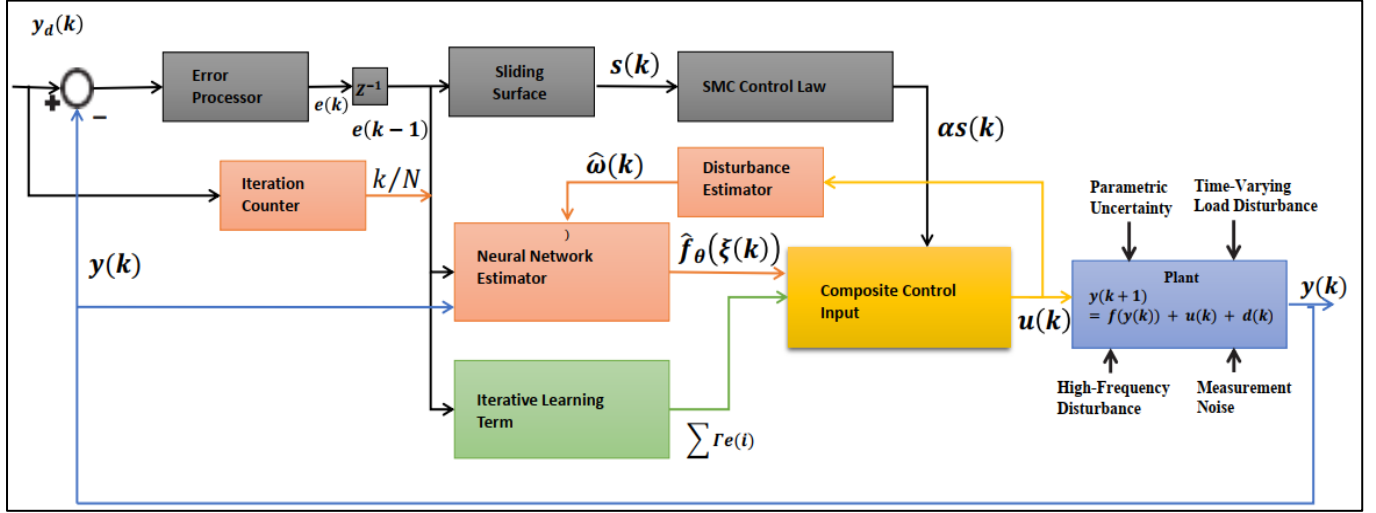


Fig. 1. Block diagram of the proposed NN-ILSMC system.

Theorem 1. In the uncertain nonlinear dynamic system (1), by the use of the above control law, tracking error $e(k)$ will be ensured to converge asymptotically, under the conditions stated as follows

$$\begin{aligned} \alpha &> \frac{J_{max}}{2\eta K_t} (1 + \bar{d}), \\ \|\theta\| &\leq \theta_{max}, \\ \Gamma &< \frac{1-\lambda}{2T_s B_{min}}. \end{aligned} \quad (12)$$

Where α is the learning rate of the neural network, η is sliding mode control gain, θ is the weight vector of the neural network, θ_{max} is the upper bound on the norm of the weight vector, Γ is the adaptation gain or learning gain in the update law, J_{max} is the maximum inertia, K_t is the torque constant, $\bar{d}$ bounds the rate of change of disturbances, $\lambda \in (0,1)$ governs ILC convergence, T_s is the sampling period, and B_{min} is the minimum control effectiveness.

4.2 Neural Network Design

The two-layer fully connected architecture is adopted to perform the NN approximation in the form of $\hat{f}_\theta(\xi)$:

$$\hat{f}_\theta(\xi) = W_2 \sigma(W_1 \xi + b_1) + b_2 \quad (13),$$

where $\sigma(\cdot)$ is activation function. Table 1 shows the detailed specifications of the architecture. The comprehensive uncertainty of the lumped model due to parametric variations, unmodeled dynamics and external disturbances is approximated with a two-layer

fully connected feed forward neural network that is designed as

$$\hat{f}_\theta(\xi) = W_2 \sigma(W_1 \xi + b_1) + b_2, \quad (14)$$

Wherein $W_1 \in \mathbb{R}^{30 \times 6}$ and $W_2 \in \mathbb{R}^{1 \times 30}$ denote the input-to-hidden and hidden-to-output weight matrices respectively, and this is because the input layer (30 nodes), the hidden layer (30 nodes), and the output layer (1 node) are the same as the input, hidden to output and the output weight levels respectively. The hyperbolic tangent inequality, $\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, is used as the input to activation. The whole trainable parameters are concatenated into the vector $\theta = \text{vec}\{W_1, W_2, b_1, b_2\}$ that is online-optimized in terms of the minimization of the estimation error in the control law.

Table 1

Neural Network Architecture Specifications

Parameter	Value
Input dimension	6
Hidden layers	2 layers, each with 30 neurons
Neurons	$\sigma(z) = \tanh(z)$
Activation function	
Learning rate	$\eta(k) = 0.1e^{-0.01k}$
Regularization	$\lambda \ W\ _2^2, \quad \lambda = 10^{-4}$

A two-layered NN containing 30 neurons each in the two layers offers sufficient approximation and is computationally efficient, suitable to numerical simulation case studies.

Remark 1. The input vector $\xi(k)$ provide better numerical consumption and stability of training is:

$$\bar{\xi}_i = \frac{\xi_i}{\sigma_i}, \quad i = 1, \dots, 6, \quad (15)$$

In which $\sigma = [1 \text{ rad}, 2 \text{ rad/s}, 10, 1 \text{ rad}, 2 \text{ rad/s}, 1]^T$ are empirically estimated scaling parameters.

4.3 Online Learning Process

The NN undergoes online adaptation, and mini-batch gradient descent updates are executed periodically during operation. The instantaneous prediction error between the actual control input and the neural network estimate is minimized using an adaptation rule. As depicted in Figure 2, the process starts with the stage of initialization, then periodically computes the desired control signal, updates the mini-batches of the parameters, and resets the buffer.

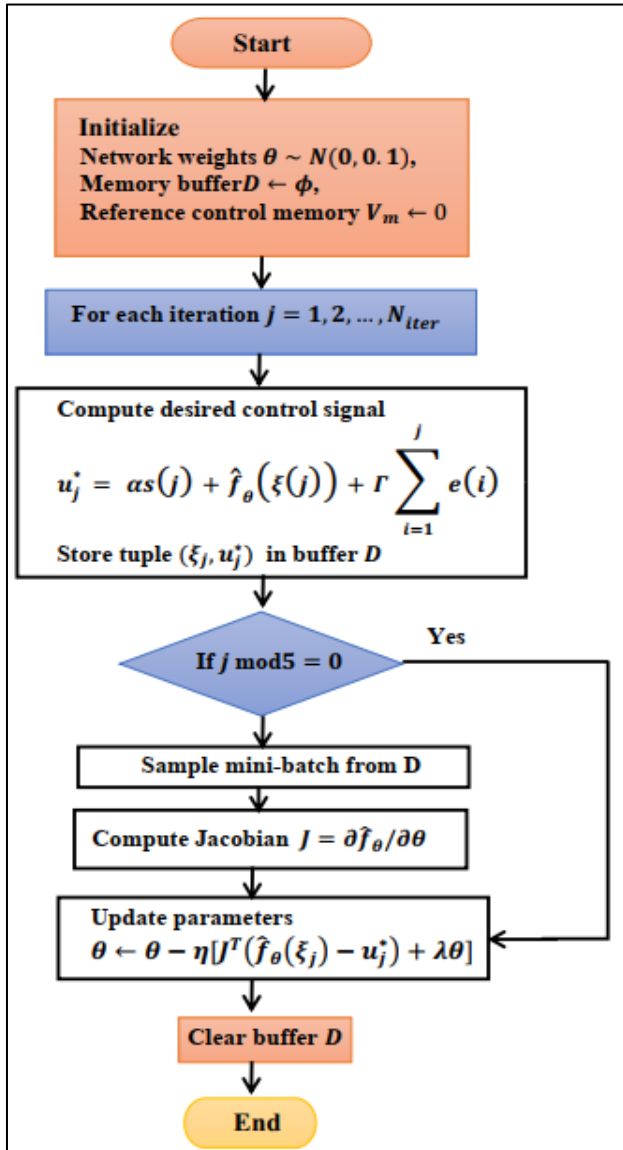


Fig. 2. Flowchart of the Online NN-ILSMC Learning Procedure

This duty-periodic implementation strategy balances real-time execution against computational load and can be efficiently implemented on embedded system.

4.4 Stability Analysis

The discrete-time candidate Lyapunov function to analyze the stability of the closed-loop system takes the form:

$$V(k) = \frac{1}{2} s^2(k) + \frac{1}{2\gamma} \bar{\theta}^T(k) \bar{\theta}(k), \quad (16)$$

The error in the parameter estimation is given by $\bar{\theta}(k) = \theta(k) - \theta^*$ being the optimal weight vector that minimizes the error in approximation, and $\gamma > 0$ is the adaptation gain.

The forward difference, $\Delta V(k) = V(k+1) - V(k)$ has following properties:

$$\Delta V(k) = V(k+1) - V(k) \leq -\alpha s^2(k) - \beta \|\bar{\theta}(k)\| + \frac{1}{\gamma} \bar{\theta}^T(k) (\gamma \phi(k) s(k) + \Delta \bar{\theta}(k)), \quad (17)$$

with $\phi(k) = \partial \hat{f}_\theta / \partial \theta$ the neural net regressor vector. Through the use of the parameter update law:

$$\Delta \bar{\theta}(k) = -\gamma \phi(k) s(k) - \nu \bar{\theta}(k), \quad \nu > 0 \quad (18),$$

and Substitution into (17) yields the following:

$$\Delta V(k) \leq -\alpha s^2(k) - \nu \|\bar{\theta}(k)\|^2 + \beta \|\bar{\theta}(k)\| \quad (19)$$

This inequality can be expressed as follows using standard bounding techniques and completing the square:

$$\Delta V(k) \leq -\eta V(k) + \epsilon, \quad (20)$$

There exist some positive constant $\eta > 0$ and some small bound ϵ . This means that $V(k)$ must be ultimately uniformly bounded, and thus the sliding variable $s(k)$ and variations $\bar{\theta}(k)$ to the parameters are bound. Therefore, the tracking error $e(k)$ will asymptotically approach a vicinity of the zero point which makes the system robust to uncertainties.

4.5 Robustness Enhancements

To enhance robustness in practical implementations and prevent parameter drift during online learning, two key modifications are introduced in the adaptation mechanism. First, a projection-based parameter update law is employed to constrain the parameters within a predefined compact set, thereby mitigating the risk of unbounded growth due to measurement noise, unmodeled dynamics, or insufficient excitation. Second, a dead-zone modification is incorporated into the adaptation law to halt parameter updates when the

system state is sufficiently close to the sliding manifold specifically, when the sliding variable s satisfies $|s| < \delta$, with $\delta = 0.01$. The resulting adaptation law is given as:

$$\Delta\theta = \begin{cases} -\gamma\phi_s, & \text{if } |s| > 0.01, \\ 0, & \text{otherwise.} \end{cases} \quad (21)$$

This formulation effectively attenuates chattering and suppresses unnecessary parameter fluctuations near steady-state conditions, thereby enhancing the smoothness and robustness of the control law without compromising asymptotic convergence.

5. Uncertainty and Disturbance Modeling

To rigorously evaluate controller robustness, we consider four classes of perturbations: parametric uncertainties, load disturbances, high-frequency sinusoidal disturbance, and measurement noise. The system inertia J and damping coefficient B are subject to bounded parametric variations expressed as:

$$J_{\text{real}} = J_{\text{nom}}(1 + \Delta_J), B_{\text{real}} = B_{\text{nom}}(1 + \Delta_B) \quad (22)$$

Where $\Delta_J, \Delta_B \in [-0.3, 0.3]$ represent $\pm 30\%$ deviations from nominal values. A time-varying load disturbance $\tau_{\text{load}}(t) = -0.005 \sin(2\pi t)$ is introduced at $t = 5$ second to simulate sudden external loading. High-frequency actuator disturbance is modeled as $\Delta_\omega = 0.01 \sin(100t)$, characteristic of real-world electromechanical imperfections. Sensor measurements are corrupted by additive Gaussian white noise $\mathcal{N}(0, 0.01^2)$, yielding observed states $x_{\text{meas}} = x + \mathcal{N}(0, 0.01)$.

The overarching uncertainty framework guarantees that all three controllers model-based ILSMC and neural network based ILSMC are assessed under the same rigorous uncertainty framework, which allows for a significant evaluation across different robustness strategies.

6. Numerical Simulation

To demonstrate the effectiveness of the proposed data-driven ILSMC and to compare it with the model-based ILSMC, numerical simulations are performed using the SRV02 motor-driven system. The simulation environment is designed to correspond to SRV02 physical specifications Table 2, and includes parametric uncertainty, external disturbances, and measurement noise.

Table 2

Nominal Parameters of the Srv02 System

Symbol	Description	Value	Unit
J_{cq}	Equivalent moment of inertia	2.08×10^{-32}	$\text{kg} \cdot \text{m}^2$
B_{cq}	Equivalent viscous damping coefficient	0.015	$\text{N} \cdot \text{m} \cdot \text{s} / \text{rad}$
R_m	Armature resistance	2.6	Ω
K_t	Torque constant	7.68×10^{-3}	$\text{N} \cdot \text{m} / \text{A}$
K_m	Back EMF constant	7.68×10^{-3}	$\text{V} \cdot \text{s} / \text{rad}$
η_g	Gearbox efficiency	0.9	—
K_g	Gear ratio	70	—

6.1 Simulation Framework

The system is simulated over some finite time horizon $t \in [0, 10]$ seconds with a constant sampling time $dt = 0.01$ s. All simulations are of 50 learning iterations long, which is of sufficient duration for the control law to converge. Statistical performance is evaluated with regards to different initial conditions and parameter uncertainties using a Monte Carlo configuration of 20 independent simulation runs.

The system dynamics are governed by the state-space model:

$$\dot{x}(t) = Ax(t) + Bu(t) + D\tau_{ml}(t), y(t) = Cx(t), \quad (23)$$

with matrices defined as:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -16.606 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 576.92 \end{bmatrix}, C = [1 \quad 0], \\ D = \begin{bmatrix} 0 \\ -480.77 \end{bmatrix}.$$

6.2 Simulation Results

We express the model-based ILSMC and the NN-ILSMC results in terms of its position and velocity tracking error, over the time horizon of the simulation, to quantitatively evaluate both control strategies. These performance metrics make it possible to objectively compare the convergence, tracking precision, and robustness.

The root mean square error (RMSE) for position and velocity are given as:

$$RMSE_{\theta} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\theta(i\Delta t) - \theta_d(i\Delta t))^2}, \quad (24)$$

$$RMSE_{\omega} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\omega(i\Delta t) - \omega_d(i\Delta t))^2}, \quad (25)$$

where $\theta(i\Delta t)$ and $\omega(i\Delta t)$ represent the actual position and angular velocity at time step i , $\theta_d(i\Delta t)$ and $\omega_d(i\Delta t)$ denote the desired reference trajectories, and N corresponds to the total number of time steps in the simulation. All of these measures form a global measure of tracking precision, which takes into account the entire period of the simulation.

To evaluate statistical consistency and robustness against parametric uncertainty, we conducted $N_{MC} = 20$ Monte Carlo simulations, in which both the system parameters and initial conditions were randomly varied. The mean and standard deviation of the RMSE values across all trials are reported to illustrate the variability in performance.

7. Result and Discussion

In Table 3, the results of all three controllers are summarized. It can be observed that the NN approach has far much less tracking errors compared to the model-based. The improvements in particular are determined as a percentage of the reduction in RMSE of the proposed NN-ILSMC compared to the model-based ILSMC with the formula:

$$\text{Improvement (\%)} = \frac{RMSE_{\text{baseline}} - RMSE_{\text{proposed}}}{RMSE_{\text{baseline}}} \times 100 \quad (24)$$

In accordance with this., the position RMSE error is reduced by 86.3 percent and the velocity RMSE is reduced by 87.6 percent. These advances point out the increased completeness and effectiveness of the proposed data-driven approach relative to other methods.

Table 3

Average Tracking RMSE ($\pm$ Standard Deviation) over 20 Monte Carlo trials.

Controller	Pos. RMSE (rad)	Vel. RMSE (rad/s)	Pos. Improv. (%)	Vel. Improv. (%)
Model-Based ILSMC	0.2047 $\pm$ 0.3951	0.2541 $\pm$ 0.4195	-	-
Neural Network based ILSMC	0.0281 $\pm$ 0.0483	0.0314 $\pm$ 0.0569	86.3%	87.6%

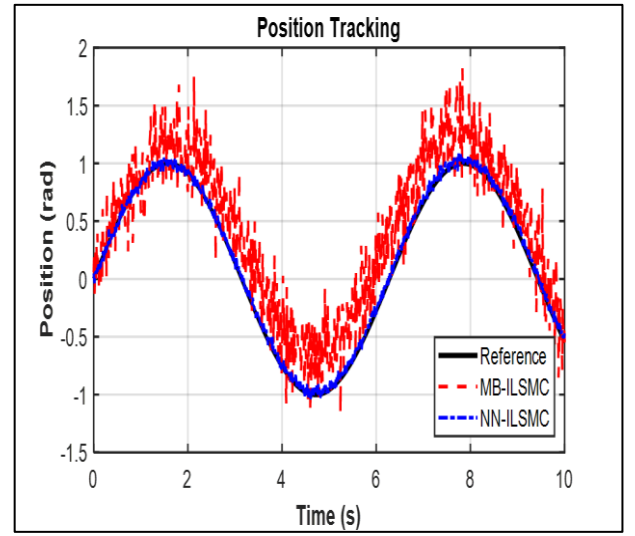


Fig. 3. Position Tracking Comparison of Model-Based and NN-ILSMC Based Controllers

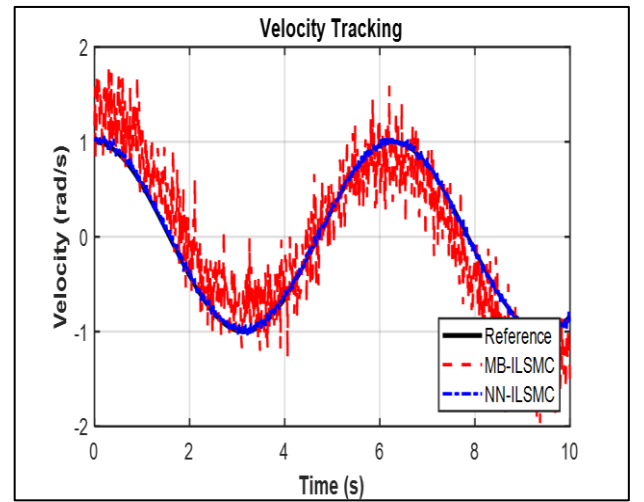


Fig. 4. Velocity Tracking Comparison of Model-Based and NN-ILSMC Based Controllers

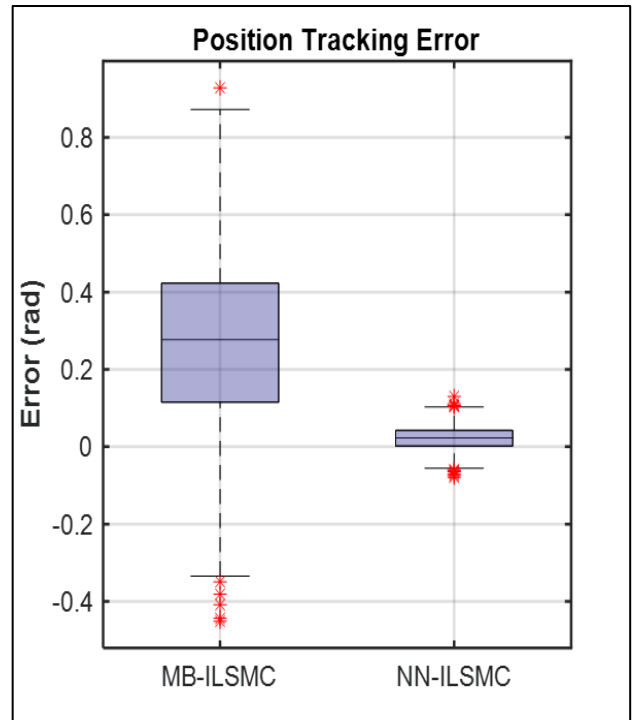


Fig. 5. Position RMSE Distribution across Model-Based and NN-ILSMC Based Controllers

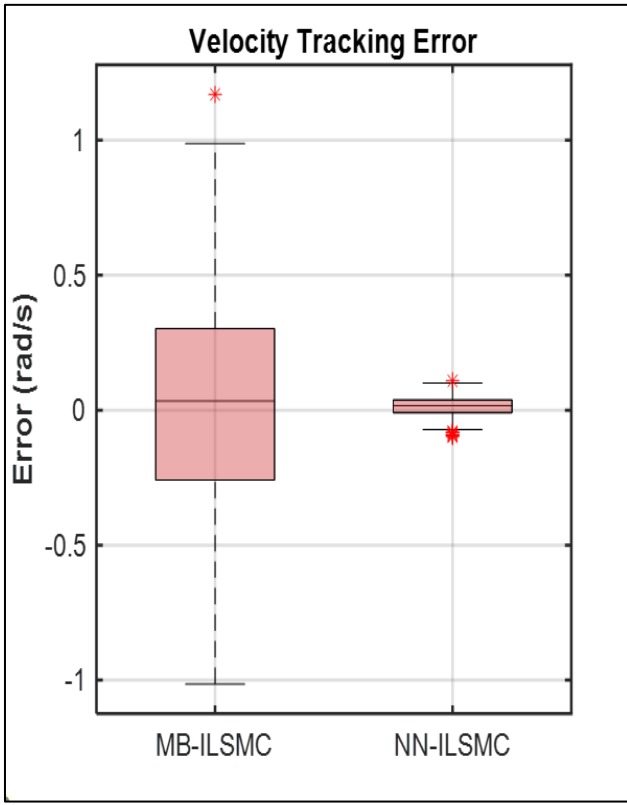


Fig. 6. Velocity RMSE Distribution across Model-Based and NN-ILSMC Based Controllers

The Figures 3, 4, 5 and 6 give a complete performance analysis of both controllers MB-ILSMC and NN-ILSMC.

In Figure 3, NN-ILSMC maintains reference trajectory alignment for the entire period, and their marking is the closest to it with the smallest deviation. MB-ILSMC is in stark contrast, doing very poorly with large tracking errors and not closely following the reference trajectory.

In Figure 4, NN-ILSMC demonstrates excellent velocity tracking, smoothly following the reference with minimal error. MB-ILSMC again performs poorly, with significant velocity errors and an inability to track rapid changes, further highlighting its instability.

Table 4:
Comparative Analysis with Data-Driven Learning Techniques

Feature	Data-Driven Framework (Sontag's Formula / GMM) [42]	Proposed NN-ILSMC
Learning Signal	Batch-processed demonstration trajectories	Iterative tracking-error minimization
Update Law	Offline constrained optimization	Online error-driven neural weight adaptation. $\theta \leftarrow \theta - \eta [J^T(\hat{f}_\theta(\xi_j) - u_j^*) + \lambda \theta]$.
Temporal Basis	Predefined dataset time steps	Real-time control cycles
Adaptation Scope	Fixed after training	Continually updated during operation
Robustness Source	Implicit in offline model fit	Sliding mode robustness combined with adaptive NN learning

As observed from Figures 5 and 6 illustrating the statistical distribution of tracking errors, NN-ILSMC achieves the lowest Position RMSE (0.0281 ± 0.0483 rad) and Velocity RMSE (0.0314 ± 0.0569 rad/s), confirming its superior precision and consistency. On the other hand, MB-ILSMC exhibits much higher errors (Position RMSE: 0.2047 ± 0.3951 rad, Velocity RMSE: 0.2541 ± 0.4195 rad/s) with greater variability, indicating unreliable performance.

This is due to the capability of the proposed NN-ILSMC integrated with adaptive learning from data, exhibiting enhanced adaptability to system uncertainties and disturbances compared to conventional model-based controllers that rely on accurate plant models. [42] Presents a framework to address nonlinear dynamics where explicit models do not exist; the contribution of our method builds on this by embedding theoretical foundations derived from iterative learning and sliding mode control architectures into neural network adaptation. The inherent robustness from principle-driven sliding mode control combined with the adaptability of neural network models allows this hybrid design to surpass existing state-of-the-art alternatives under parametric uncertainties and noise in terms of tracking accuracy and resilience. Extensive simulation results demonstrate improvements over [42], with potential for extension to MIMO systems and time-varying disturbances.

7.1 Fundamental Learning Processes

In order to further clarify the methodological differences, Table4 compares the main learning principles in the data-driven framework in [42] and the proposed NN-ILSMC. Such a contrast shows the transition between offline, batch-oriented optimization, and online, adaptive error-driven parameter estimation within iterative learning schemes, to enable faster convergence and enhanced error robustness in real-time control task.

Conclusion

The paper developed a NN-ILSMC strategy for input-output based model-free systems, enabling model-free control for precise trajectory tracking. The proposed control approach utilizes feedback-based control and error-driven feedback iterative learning for control refinement. The controller learns control adjustments for model-free control while accommodating parametric uncertainties, external and unmodeled dynamics, and other factors. Convergence was shown theoretically with discrete-time Lyapunov methods. Simulations conducted on the Quanser SRV02 motor-driven system yielded a position tracking accuracy improvement of 86.3% and a velocity tracking robustness improvement of 87.6% in comparison to the model-based ILSMC. The limitations of MB-ILSMC are largely due to its dependence on the correct modeling of plants; unmodeled dynamics and externality activate its vulnerability, which could be important when dealing with uncertain environments. However, the proposed NN-ILSMC can adapt online by tracking the tracking errors, which ensures the reliable operation of the model during uncertainties. These results confirm the robustness of the proposed method and its effectiveness for systems with uncertainties or time-varying dynamics. Future work includes exploration of adaptive learning rate approaches, integration of the proposed control method with time-varying learning techniques and further adjusting parameters for optimal performance, and testing on the actual SRV02 model.

Conflict of Interest Statement

The authors declare no conflict of interest.

References

- [1] Arimoto, Suguru, Sadao Kawamura, and Fumio Miyazaki. "Bettering operation of robots by learning." *Journal of Robotic Systems* 1.2 (1984): 123-140.
- [2] Moore, Kevin L. *Iterative learning control for deterministic systems*. Springer, 2012.
- [3] Xu, Jian-Xin, and Ying Tan. *Linear and nonlinear iterative learning control*. Springer Berlin Heidelberg, 2003.
- [4] Chen, Yangquan, and Changyun Wen, eds. *Iterative learning control: convergence, robustness and applications*. Springer London, 1999.
- [5] Ardakani, M. Mahdi Ghazaei, Sei Zhen Khong, and Bo Bernhardsson. "On the convergence of iterative learning control." *Automatica* 78 (2017): 266-273.
- [6] Ahn, Hyo-Sung, YangQuan Chen, and Kevin L. Moore. "Iterative learning control: Brief survey and categorization." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 37.6 (2007): 1099-1121.
- [7] Xu, Jian-Xin. "A survey on iterative learning control for nonlinear systems." *International Journal of Control* 84.7 (2011): 1275-1294.
- [8] Owens, D. H., S. P. Freeman, and E. Rogers. "A common framework for discrete-Lyapunov design of model reference adaptive iterative learning control algorithms." *IEEE Transactions on Control Systems Technology* 16.5 (2008): 996-1008.
- [9] Deutschmann-Olek, Andreas, Georg Stadler, and Andreas Kugi. "Stochastic iterative learning control for lumped- and distributed-parameter systems: A Wiener-filtering approach." *IEEE Transactions on Automatic Control* 66.8 (2020): 3856-3862.
- [10] Wang, Yan, Rajesh Rajamani, and Ali Zemouche. "Sequential LMI approach for the design of a BMI-based robust observer state feedback controller with nonlinear uncertainties." *International Journal of Robust and Nonlinear Control* 28.4 (2018): 1246-1260.
- [11] Hameed, Watheq Najah, and Jasim Oudah Khawwaf. "Robust sliding mode control for 2-DoF robot manipulator position control system." *IET Conference Proceedings CP906* 2024.34 (2024).
- [12] Saab, Samer Said, et al. "Iterative learning control: Practical implementation and automation." *IEEE Transactions on Industrial Electronics* 69.2 (2021): 1858-1866.
- [13] Tayebi, Abdelhamid. "Analysis of two particular iterative learning control schemes in frequency and time domains." *Automatica* 43.9 (2007): 1565-1572.
- [14] Utkin, Vadim, Jürgen Guldner, and Jingxin Shi. *Sliding mode control in electro-mechanical systems*. CRC Press, 2017.
- [15] Kumar, Chandar, Dur Muhammad Soomro, and Najeeb Ur Rehman Malik. "Computationally efficient iterative learning control design for uncertain nonlinear systems applied on SRVO2 motor-driven machine." *Journal of Independent Studies and Research Computing* 23.1 (2025): 23-31.
- [16] Levant, Arie. "Quasi-continuous high-order sliding-mode controllers." *42nd IEEE*

- [17] Davila, Jorge, Leonid Fridman, and Arie Levant. "Second-order sliding-mode observer for mechanical systems." *IEEE Transactions on Automatic Control* 50.11 (2005): 1785-1789.
- [18] Zuo, Ying, Chunyan Lai, and K. Lakshmi Varaha Iyer. "A review of sliding mode observer-based sensorless control methods for PMSM drive." *IEEE Transactions on Power Electronics* 38.9 (2023): 11352-11367.
- [19] Anelone, Anet J. N., Peter Kim, and Sarah K. Spurgeon. "Sliding mode control theory interprets elite control of HIV." *Feedback Control for Personalized Medicine*. Academic Press, 2022. 151-171.
- [20] Baek, Seungmin, et al. "An adaptive model uncertainty estimator using delayed state-based model-free control and its application to robot manipulators." *IEEE/ASME Transactions on Mechatronics* 27.6 (2022): 4573-4584.
- [21] Wang, Yulin, and Tesheng Hsiao. "Fast-update iterative learning control for performance enhancement with application to motion systems." *IEEE Access* 10 (2022): 79458-79468.
- [22] Memon, Furqan, and Cheng Shao. "Data-driven optimal PID type ILC for a class of nonlinear batch process." *International Journal of Systems Science* 52.2 (2021): 263-276.
- [23] Nikooienejad, Nastaran, Mohammad Maroufi, and S. O. Reza Moheimani. "Iterative learning control for high-speed rosette trajectory tracking." *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019.
- [24] Ba, Dang Xuan, Nguyen Trung Thien, and Joonbum Bae. "A novel iterative second-order neural-network learning control approach for robotic manipulators." *IEEE Access* 11 (2023): 58318-58332.
- [25] Kumar, Chandar, et al. "Data-Driven Control Design for Complex Systems: Theory, Challenges, and Novel Applications." *Sukkur IBA Journal of Computing and Mathematical Sciences* 8.1 (2024): 56-73.
- [26] Kumar, C., Soomro, D. M., & Malik, N. U. R. (2025). Data-Driven Stabilization: A Robust Control Approach with Linear Matrix Inequalities. *Sir Syed University Research Journal of Engineering & Technology*, 15(1), 64–70.
- [27] Meng, Deyuan, and Jingyao Zhang. "Design and analysis of data-driven learning control: An optimization-based approach." *IEEE Transactions on Neural Networks and Learning Systems* 33.10 (2021): 5527-5541.
- [28] Li, Yue, Xin Xin, and Yuhang Yan. "A signal compensation-based balance control for the rotary inverted pendulum system." *Journal of Vibration and Control* 30.15-16 (2024): 3538-3558.
- [29] Truong, Hoai-Vu-Anh, Duc-Thien Tran, and Kyoung Kwan Ahn. "A neural network based sliding mode control for tracking performance with parameters variation of a 3-DOF manipulator." *Applied Sciences* 9.10 (2019): 2023.
- [30] Sun, Ting, Cheng Liu, and Xuegang Wang. "Distributed anti-windup NN-sliding mode formation control of multi-ships with minimum cost." *ISA transactions* 138 (2023): 49-62.
- [31] Radac, Mircea-Bogdan, Radu-Emil Precup, and Emil M. Petriu. "Constrained data-driven model-free ILC-based reference input tuning algorithm." *Acta Polytechnica Hungarica* 12.1 (2015): 137-160.
- [32] Molzahn, Daniel K., et al. "A survey of distributed optimization and control algorithms for electric power systems." *IEEE Transactions on Smart Grid* 8.6 (2017): 2941-2962.
- [33] Shaikh, M. S., Lin, H., Xie, S., et al. (2025). An intelligent hybrid grey wolf–particle swarm optimizer for optimization in complex engineering design problem. *Scientific Reports*, 15, 18313.
- [34] Shaikh, M. S., Zhang, Y., Khan, M. U., et al. (2023). A hybrid moth–flame algorithm with particle swarm optimization with application in power transmission and distribution. *Decision Analytics Journal*, 6, 100182.
- [35] Shaikh, Muhammad Suhail, et al. "Optimizing Welded Beam Design Using Gorilla Troop Optimization: An Innovative Engineering Approach." 2025 12th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE, 2025.
- [36] Hou, Zhong-Sheng, and Zhuo Wang. "From model-based control to data-driven control: Survey, classification and perspective." *Information Sciences* 235 (2013): 3-35.

- [37] Levant, Arie. "Principles of 2-sliding mode design." *Automatica* 43.4 (2007): 576-586.
- [38] Lee, Hoon, and Vadim I. Utkin. "Chattering suppression methods in sliding mode control systems." *Annual Reviews in Control* 31.2 (2007): 179-188.
- [39] Norrlof, Mikael. "An adaptive iterative learning control algorithm with experiments on an industrial robot." *IEEE Transactions on Robotics and Automation* 18.2 (2002): 245-251.
- [40] Sun, Shu-Ting, Xiao-Dong Li, and Ren-Xin Zhong. "An open-closed-loop iterative learning control approach for nonlinear switched systems with application to freeway traffic control." *International Journal of Systems Science* 48.13 (2017): 2752-2763.
- [41] Moreno, Jaime A., and Marisol Osorio. "Strict Lyapunov functions for the super-twisting algorithm." *IEEE Transactions on Automatic Control* 57.4 (2012): 1035-1040.
- [42] Trung, Cao Thanh, Nguyen Hoai Nam, and Nguyen Doan Phuoc. "A model-free controller for uncertain robot manipulators with matched disturbances." *Vietnam Journal of Science and Technology* 61.1 (2023): 166-176.
- [43] Edwards, Christopher, and Yuri B. Shtessel. "Adaptive continuous higher order sliding mode control." *Automatica* 65 (2016): 183-190.
- [44] Aarnoudse, Leontine, and T. A. E. Oomen. "Random learning leads to faster convergence in 'Model-Free' ILC." (2024).
- [45] Yu, Xinghuo, and Man Zhihong. "Fast terminal sliding-mode control design for nonlinear dynamical systems." *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications* 49.2 (2002): 261-264.
- [46] Son, Tong Duy, Goele Pipeleers, and Jan Swevers. "Robust monotonic convergent iterative learning control." *IEEE Transactions on Automatic Control* 61.4 (2015): 1063-1068.
- [47] Becerra-Mora, Y.A., Acosta, J.Á. Data-driven learning and control of nonlinear system dynamics. *Nonlinear Dyn* 113, 6605–6625 (2025). <https://doi.org/10.1007/s11071-024-10149-1>
- [48] Mosharafian, Sahand, et al. "A deep reinforcement learning-based sliding mode control design for partially-known nonlinear systems." 2022 European Control Conference (ECC). IEEE, 2022.
- [49] Youssef, Omnia, Mohamed Wafa, and Raafat Shalaby. "Reinforcement learning-enhanced adaptive sliding mode control for nonlinear systems." *Complex & Intelligent Systems* 11.8 (2025): 351.
- [50] Ahmed, Saim, Ahmad Taher Azar, and Ibraheem Kasim Ibraheem. "Nonlinear system controlled using novel adaptive fixed-time SMC." *AIMS Mathematics* 9.4 (2024): 7895-7916.