FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

Design of Sliding Mode Controller Integrated with Nonlinear Disturbance Observer and Radial Basis Function Neural Network for Trajectory Tracking of DDMRs Considering Terrain Factor

Nguyen Van Tien[1*], and Do Khac Tiep[2]

[1]*Faculty of Electrical and Electronic Engineering, Vietnam Maritime University, Haiphong, Vietnam; Email: nguyenvantien@vimaru.edu.vn*
[2]*Faculty of Electrical and Electronic Engineering, Vietnam Maritime University, Haiphong, Vietnam; Email: dokhactiep@vimaru.edu.vn*

***Correspondence:** *Nguyen Van Tien; nguyenvantien@vimaru.edu.vn; Tel: +84-976-483-586*

ABSTRACT- This paper presents a robust control design method for Differential Drive Mobile Robots (DDMR) to address the trajectory tracking problem under conditions of disturbances and uncertainties, specifically focusing on terrain-induced variations. The proposed controller utilizes Sliding Mode Control (SMC) to ensure robustness and fast response. To mitigate the inherent chattering phenomenon of SMC and enhance tracking accuracy, two disturbance observers are designed and integrated into the system: A Nonlinear Disturbance Observer (NDO) and a Neural Network Observer (NNO) using Radial Basis Functions (RBF). These observers are tasked with estimating aggregate disturbances, including friction, model uncertainties, and terrain effects, thereby effectively compensating the control signal. This approach allows for a reduction in the switching gain of the SMC, resulting in improved control quality. Simulation results demonstrate that the SMC-NDO-NNO controller significantly improves control performance even in the presence of uncertain disturbances and substantially reduces chattering.

Keywords: Mobile Robot, Nonlinear Disturbance Observer, Sliding Mode Control, Robust Control, Trajectory Tracking.

ARTICLE INFORMATION
Author(s): Nguyen Van Tien, and Do Khac Tiep;
Received: 03/12/2025; **Accepted**: 19/03/2026; **Published**: 30/03/2026;
E- ISSN: 2347-470X;
Paper Id: IJEER0312A01
Citation: 10.37391/ijeer.140125
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-14/ijeer-140125.html

Publisher's Note: FOREX Publication stays neutral with regard to jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

In recent years, mobile robots have become an indispensable tool in numerous applications ranging from industry and services to space exploration and the military. One of the core functional requirements determining robot performance is the capability to track a reference trajectory accurately and stably [1]. However, achieving this in real-world environments presents significant challenges, primarily stemming from uncertain components in the dynamic model (mass errors, moment of inertia) and external disturbances, particularly when the robot operates on uneven terrain. Factors such as friction variations, slope gradients, and minor collisions can cause significant tracking errors, degrading the performance of traditional linear controllers like PID.

Various controllers have been researched for mobile robots, including PID, SMC, and SMC combined with NDO; however, these controllers exhibit certain limitations:

PID Controller: While being the most classic and popular method, PID is inherently a linear controller. It faces significant difficulties when dealing with the inherent nonlinear characteristics of the robot and large disturbances from the terrain. Consequently, PID often exhibits large tracking errors and lacks the ability to adapt flexibly to changes in the operating environment [2].

Sliding Mode Controller: SMC represents a significant advancement over PID due to its superior robustness [3]. Theoretically, SMC can ensure system stability even in the presence of uncertain disturbances [4] and has been developed with various adaptive structures [5]. However, the inherent drawback of traditional SMC is the high-frequency chattering phenomenon. This oscillating control signal not only wastes energy but can also cause wear and damage to the robot's actuators over time [6- 8].

SMC Combined with NDO: To address the chattering problem, an effective improvement direction, pioneered by Ohnishi et al. [9], is to integrate a Nonlinear Disturbance Observer. The NDO is capable of estimating external disturbances (such as terrain resistance) and providing this information for the controller to actively compensate [10]. In this way, the burden on the switching component in SMC is alleviated, helping to significantly minimize the chattering phenomenon [11, 12]. However, this method still has a limitation: the NDO primarily focuses on environmental disturbances. Consequently, the system must still rely on the high-gain switching component of the SMC to cope with internal uncertainties in the robot model, such as inaccurate

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

mass or moment of inertia parameters. This means that chattering is not completely eliminated, and accuracy is not yet optimal, although this method has been successfully applied to various systems such as unmanned aerial vehicles and mobile robots [13, 14].

Recently, to further enhance the robustness of SMC against highly complex and unstructured uncertainties, researchers have shifted attention towards integrating Neural Networks (NN) with disturbance observers NDO [15-17]. Works from 2022 to 2025 demonstrate that while NDOs are effective at estimating matched, slowly varying disturbances, Neural Networks excel at universal approximation of highly nonlinear, fast-varying, and mismatched uncertainties [15, 16]. However, most recent SMC-NN-NNO frameworks have primarily focused on robotic manipulators or unmanned aerial vehicles [17]. The specific application of a dual-observer structure (NDO combined with NNO) to address the coupled effects of highly stochastic terrain disturbances and internal parametric uncertainties in DDMRs remains underexplored.

In our previous work [14], an SMC-NDO framework was successfully developed to mitigate terrain-induced disturbances for mobile robots. While effective for compensating slowly varying external friction, the SMC-NDO approach exhibited critical limitations when facing internal parametric uncertainties (e.g., mass variations, unmodeled actuator dynamics). Relying solely on the NDO forced the system to still depend on the high-gain switching component of the SMC to cope with these internal errors, leading to residual high-frequency chattering and tracking degradation. Recognizing these unresolved limitations, this paper proposes a more advanced and comprehensive hybrid control architecture: the SMC combined with an NDO and an NNO (SMC-NDO-NNO).

This method intelligently divides tasks to achieve maximum efficiency. The NDO is responsible for estimating and compensating for external disturbances such as friction and terrain impacts. The Neural Network Observer focuses on identifying and suppressing internal uncertainties of the robot model. Thanks to the ability to learn and approximate complex nonlinear functions, the neural network can model factors that the NDO does not handle well [18], such as inaccurate dynamic parameters or high-order nonlinear effects. This approach offers superior benefits compared to previous methods: highest trajectory tracking accuracy, strong robustness against all types of disturbances and uncertainties, and most importantly, the near-complete elimination of chattering, helping to protect the robot and save energy.

The main contributions of this paper are summarized as follows:

A novel hybrid control architecture is proposed: A robust Sliding Mode Controller is integrated with a dual-observer structure comprising a Nonlinear Disturbance Observer (and a Radial Basis Function Neural Network Observer (RBF-NNO). This architecture is specifically designed to address trajectory tracking for DDMRs operating on complex, uneven terrains.

Strategic decomposition of disturbance estimation: Unlike our previous work [14] which relied on a single observer, this study introduces a strict task-division strategy. The NDO is utilized to estimate and compensate for slowly varying external terrain disturbances, while the adaptive RBF-NNO is dedicated to actively learning and identifying fast-varying internal model uncertainties. This effectively prevents the NDO from being overwhelmed by complex coupled noises.

Superior performance and chattering suppression: The proposed method guarantees asymptotic stability through Lyapunov analysis. Specifically, the simulation results confirm that the SMC-NDO-NNO reduces the trajectory tracking Mean Absolute Error (MAE) by approximately 40% compared to the SMC-NDO [14] and over 60% compared to the standalone SMC. Most importantly, it completely eliminates the high-frequency chattering phenomenon under parametric uncertainty, thereby protecting actuators and reducing energy consumption.

To explicitly highlight the methodological advancements and quantitative performance improvements of the proposed architecture over existing works, a detailed comparison is summarized in *table 1*.

Table 1. Methodological and Performance Comparison of Control Strategies

Feature	Standalone SMC	SMC-NDO (Previous Work [14])	Proposed SMC-NDO-NNO
Adaptability to Parametric Uncertainty	Low (highly sensitive, strong error oscillation)	Partial (still relies on high-gain SMC component)	High (adaptive online learning via NN)
Chattering	Severe	Improved, but persists under combined uncertainties	Near-complete elimination, generating smooth control signals
Tracking Accuracy	Moderate (0.0055 m)	Good (0.0035 m)	Superior (0.0021 m)
Longitudinal Error	Moderate (0.0035 m)	Good (0.0021 m)	Superior (0.0004 m)
Computational Complexity	Low	Medium	Medium-High

2. MODELING OF DIFFERENTIAL DRIVE MOBILE ROBOT

The kinematic and dynamic models of the differential drive mobile robot are established based on the fundamental principles detailed in [21, 22]. To design an effective trajectory tracking controller, it is a prerequisite to first establish a mathematical model characterizing the kinematics and

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

dynamics of the mobile robot. In this study, we focus on a DDMR with the structure illustrated in *figure 1* [18, 19].

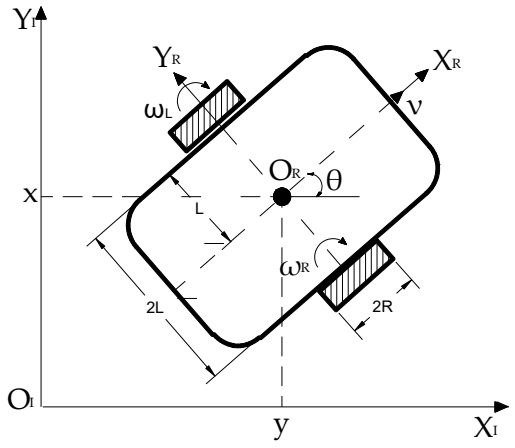

Figure 1. Structure of the differential drive mobile robot (DDMR)

where:(x, y, θ) denotes the position and orientation of the robot within the global coordinate system; (v, ω) represents the linear and angular velocities of the robot, respectively; (ω_L, ω_R) are the angular velocities of the left and right wheels, respectively; R is the radius of each wheel; 2L is the track width.

To characterize the dynamics of the robot, two coordinate systems are defined: The global coordinate and the robot-fixed coordinate System:

The global coordinate system$\{O_I, X_I, Y_I\}$: This frame is fixed at a specific origin within the robot's workspace. It serves to determine the absolute position of the robot at any given instant. The position of the robot's center is represented by coordinates(x, y) within this frame.

The robot-fixed coordinate system$\{O_R, X_R, Y_R\}$: This frame is attached to the robot chassis, with its origin$O_R$located at the robot's center. Its primary purpose is to describe the relative motion of the robot and define its heading. The X_R-axis aligns with the robot's forward direction, while the Y_R-axis is perpendicular to theX_R. The angleθrepresents the robot's orientation with respect to the global coordinate system.

2.1. Forward Kinematics

The forward kinematic model derives the robot's velocity vector$(\dot{x}, \dot{y}, \dot{\theta})$ from the angular velocities of the two wheels (ω_L, ω_R). The linear and angular velocities of the robot are calculated as follows [22]:

$$v = R\frac{\omega_R + \omega_L}{2}; \omega = R\frac{\omega_R - \omega_L}{2L} \quad (1)$$

By projecting these velocities onto the global coordinate system, we obtain:

$$\dot{x} = v\cos(\theta); \dot{y} = v\sin(\theta); \dot{\theta} = \omega \quad (2)$$

Combining the above equations yields the forward kinematic equation in matrix form

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\cos\theta & 0\\ \sin\theta & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v\\ \omega\end{bmatrix} \quad (3)$$

2.2. Inverse Kinematics

The inverse kinematic model determines the required angular velocities for the two wheels, denoted as$(\omega_{Rd}, \omega_{Ld})$ based on the desired linear and angular velocities of the robot(v_d, ω_d). From the velocity equations, we obtain:

$$\omega_{Rd} = \frac{1}{R}(v_d + L\omega_d); \omega_{Ld} = \frac{1}{R}(v_d - L\omega_d) \quad (4)$$

Equation (4) can be rewritten in matrix form as:

$$\begin{bmatrix}\omega_{Rd}\\ \omega_{Ld}\end{bmatrix} = \frac{1}{R}\begin{bmatrix}1 & L\\ 1 & -L\end{bmatrix}\begin{bmatrix}v_d\\ \omega_d\end{bmatrix} \quad (5)$$

The desired velocity values(v_d, ω_d) are generated by an outer-loop kinematic controller based on the position error between the robot and the reference trajectory.

2.3. Dynamic modeling

The dynamic model describes the relationship between the input torque and the actual angular velocities of the wheels. The dynamic model of the motors can be expressed as follows [22]:

$$J\dot{\omega}_{act} + B\omega_{act} + \tau_d = u \quad (6)$$

Where: $u = [u_R, u_L]^T$is the control torque vector; $\omega_{act} = [\omega_R, \omega_L]^T$is the actual angular velocity vector of the two wheels;$J$is the moment of inertia matrix of the motors and wheels;$B$is the viscous friction coefficient matrix; τ_d is the lumped disturbance vector, encompassing nonlinear friction, model uncertainties, and effects from uneven terrain.

The terrain disturbances are modeled considering the physical variations in rolling resistance and minor slippage. The disturbance torque can be physically approximated as $\tau_d = \mu(t)mgR\cos(\theta) + \tau_{unmodeled}$, where$\mu(t)$ represents the varying coefficient of rolling friction depending on the terrain type (e.g., flat, rough), and$\tau_{unmodeled}$captures high-frequency unmodeled dynamics. Since measuring exact terramechanics in real-time is impractical, this study bounds these effects within a stochastic mathematical function to simulate the worst-case varying friction profiles as shown in *figure 3*.

The above equation can be rewritten in state-space form:

$$\dot{\omega}_{act} = -J^{-1}B\omega_{act} + J^{-1}u - J^{-1}\tau_d \quad (7)$$

Let$f(\omega) = -J^{-1}B\omega_{act}$,$g = J^{-1}$,$d(t) = -J^{-1}\tau_d$be the lumped disturbance to be estimated. Consequently, *equation (7)* takes the form:

$$\dot{\omega}_{act} = f(\omega) + g.u + d(t) \quad (8)$$

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

3. DESIGN OF SLIDING MODE CONTROLLER AND OBSERVERS

Figure 2 depicts the heading control structure for the differential drive robot, employing the SMC controller integrated with the Nonlinear Disturbance Observer (NDO) and the Neural Network Observer (NNO). This is a cascaded control architecture combined with observers to achieve trajectory tracking for the mobile robot.

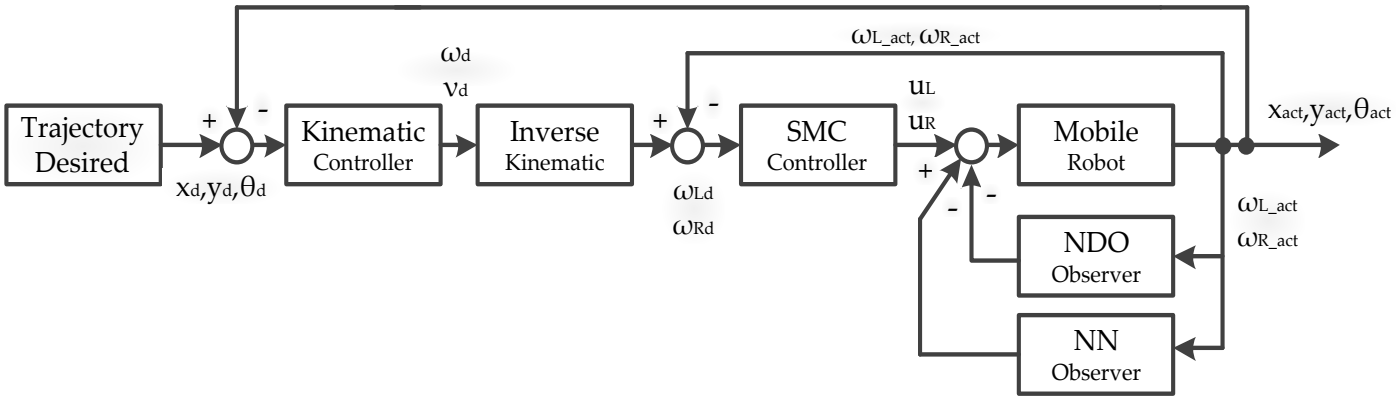

Figure 2. Heading control structure for the differential drive robot using SMC with NDO and NNO observers

3.1. Sliding Mode Controller Design

The objective of the SMC is to control the actual angular velocity ω_{act} to track the desired angular velocity obtained ω_d from the inverse kinematics. The tracking error$e(t)$ with respect to the setpoint is defined as:

$$e(t) = \omega_d(t) - \omega_{act}(t) \tag{9}$$

To simplify the design, the sliding surface is chosen as the tracking error itself:

$$s(t) = e(t) = \omega_d(t) - \omega_{act}(t) \tag{10}$$

When$s(t) = 0$, it implies$e\ (t) = 0$. In other words, when the system operates on the sliding surface, the control deviation becomes zero, and the system precisely achieves the desired state. The time derivative of the sliding surface is:

$$\dot{s} = \dot{\omega}_d - \dot{\omega}_{act} = \dot{\omega}_d - f(\omega) - g.u - d \tag{11}$$

The SMC law comprises two components: the equivalent control u_{eq} and the switching controlu_{sw}. The aggregate control signal u_{SMC} is further compensated by the signal u_x from the observers; thus

$$u_{SMC} = (u_{eq} + u_{sw}) + u_x \tag{12}$$

The equivalent control component$u_{eq}$is derived by setting ($\dot{s} = 0$) and assuming zero disturbance ($d = 0$). *Equation (11)* then becomes:

$$\dot{\omega}_d - f(\omega) - g.u_{eq} = 0 \tag{13}$$

Solving *equation (13)* yields:

$$u_{eq} = g^{-1}(\dot{\omega}_d - f(\omega)) \tag{14}$$

switching control component$u_{sw}$is designed to drive the system toward the sliding surface despite disturbances or model uncertainties. A common control law is:

$$u_{sw} = g^{-1}\begin{bmatrix} k_R & 0 \\ 0 & k_L \end{bmatrix}.sgn(s) \tag{15}$$

Where(k_R, k_L) are the control parameters for the right and left wheel motors.

To cancel the effect of environmental disturbances, the disturbance compensation signal is designed as:

$$u_x = g^{-1}\hat{d} \tag{16}$$

Where$\hat{d}$is the estimated disturbance value from the NDO and NNO observers. From *eq. (12) -(16)*, we obtain the overall control law:

$$u_{SMC} = g^{-1}\left(\dot{\omega}_d - f(\omega) + K.sgn(s) - \hat{d}\right) \tag{17}$$

3.2. Nonlinear Disturbance Observer (NDO) Design

The NDO is designed to estimate the lumped disturbance$d(t)$ based on the assumption that the disturbance varies slowly relative to the observer dynamics. The estimated disturbance$\hat{d}_{NDO}$is given by:

$$\hat{d}_{NDO} = z + p(\omega) \tag{18}$$

where$z$is the internal state vector of the observer, and$p(\omega)$ is a vector function to be designed. The dynamics of$z$are defined as follows:

$$\dot{z} = -L(\omega)(z + p(\omega) + f(\omega) + g.u) \tag{19}$$

Where$L(\omega)$ is the observer gain matrix. By selecting$L(\omega) = \partial(\omega)/\partial\omega$, the dynamics of the estimation error become:

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

$$\dot{\tilde{d}} = -L(\omega)\tilde{d} \quad (20)$$

Where($\tilde{d} = d - \hat{d}$) is the estimation error.

If $L(\omega)$ is a positive definite matrix, the estimation error$\tilde{d}$will asymptotically converge to zero. A simple and effective choice is to select $p(\omega) = \Gamma\omega$ as a constant positive diagonal matrix. In this case, $L(\omega) = \Gamma$. The implementation of the NDO is as follows:

$$\hat{d}_{NDO} = z + \Gamma\omega_{act} \quad (21)$$

Where:

$$\dot{z} = -\Gamma(\hat{d}_{NDO} + f(\omega_{act}) + g.u) \quad (22)$$

3.3. Neural Network Observer (NN Observer) Design

In control and system identification problems, Radial Basis Function (RBF) Neural Networks [23] are often preferred over other architectures like Multi-Layer Perceptron (MLP) due to their capability to approximate any continuous nonlinear function with arbitrary accuracy [24]. This makes them an ideal tool for modeling complex and undefined disturbance components of the robot. Additionally, the simple structure and fast convergence speed of RBF networks significantly reduce computational load, which is a critical factor in real-time control applications.

Therefore, based on the solid theoretical foundation of neural network function approximation [25, 26], this paper utilizes an RBF Neural Network to learn and approximate the nonlinear disturbance$d(t)$ without requiring its exact mathematical model. The disturbance$d$is approximated by an RBF network of the form [23]:

$$d = W^{*T}\Phi(x) + \varepsilon \quad (23)$$

Where$W^*$is the optimal weight matrix,$\Phi(x)$ is the vector of basis functions,$x$is the input vector of the network, and ($x = \omega_{act}$) is the approximation error. The estimated disturbance value from the NN is:

$$\hat{d}_{NN} = \widehat{W}^T\Phi(x) \quad (24)$$

where$\widehat{W}$is the estimated weight matrix. The update law for the weight matrix $\widehat{W}$is derived from Lyapunov stability analysis to ensure tracking error convergence. The selected update law is:

$$\dot{\widehat{W}} = -\gamma\Phi(x)s^T \quad (25)$$

where$\gamma > 0$is the learning rate matrix (positive definite) and$s$is the sliding surface vector from the SMC controller. This update law ensures that the weights are adjusted in a direction that minimizes the error on the sliding surface.

The basis function vector $\Phi(x)$ is a column vector where each element is the output of a neuron in the hidden layer.

Typically, the activation function for each RBF neuron is the Gaussian function [23], defined as:

$$\phi_j(x) = exp\left(-\frac{\|x-c_j\|^2}{2\sigma_j^2}\right) \quad (26)$$

Where: x is the input vector of the network (in this robot control problem,$x$corresponds to the actual angular velocity vector($x = x_{act}$); $\phi_j(x)$ is the output of the j^{th} neuron; c_j is the center vector of the j^{th} neuron. This is the point in the input space where the neuron$\phi_j$produces its maximum value (equal to 1). These centers are chosen to cover the operating workspace of the input;$\sigma_j$is the standard deviation (width) of the j^{th} neuron. It determines the range of influence of the neuron. If $\sigma_j$is small, the neuron only responds strongly when the input$x$is very close to the center c_j. Ifσ_j is large, the range of influence is broader.

If the network has$m$neurons in the hidden layer, the basis function vector will be a column vector $\Phi(x)$ formed by stacking the outputs of individual neurons:

$$\Phi(x) = \begin{pmatrix} \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_m(x) \end{pmatrix} = \begin{pmatrix} exp\left(-\frac{\|x-c_1\|^2}{2\sigma_1^2}\right) \\ exp\left(-\frac{\|x-c_2\|^2}{2\sigma_2^2}\right) \\ \vdots \\ exp\left(-\frac{\|x-c_m\|^2}{2\sigma_m^2}\right) \end{pmatrix} \quad (27)$$

In practice, the centers$c_j$and widths$\sigma_j$are usually pre-selected and kept fixed during operation. They can be uniformly distributed over the robot's workspace or determined using clustering algorithms (such as k-means) [27]. The learning process of the network focuses on updating the output layer weight matrix $\widehat{W}$to best approximate the disturbance$d(t)$.

3.4. Stability Proof of the SMC-NDO-NN System

The stability analysis of the system is conducted based on Lyapunov theory for nonlinear systems, following the methodology presented in [28, 29]. The objective of the proof is to demonstrate that the tracking error on the sliding surface converges to zero (or a very small neighborhood of zero), ensuring that the robot's actual velocity tracks the desired velocity [29]. To proceed with the proof, we introduce the following assumptions:

The disturbance and its derivative are bounded: that is, there exist positive constants D_0, D_1 such that $|d(t)| \leq D_0$ and $|\dot{d}(t)| \leq D_1$

The approximation error of the Neural Network is also bounded: there exists a positive constant ϵ_N such that the approximation error $\epsilon = d - W^{*T}\Phi(x)$.

Choose a Lyapunov function candidate of the form:

$$V = \frac{1}{2}s^Ts + \frac{1}{2}\mathrm{tr}(\widetilde{W}^T\Gamma^{-1}\widetilde{W}) \quad (28)$$

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

Where: $s = \omega_d - \omega_{act}$ is the sliding surface vector; $\widetilde{W} = \widehat{W} - W^*$ is the error of the Neural Network weight matrix; Γ is a positive definite diagonal matrix representing the learning rate.

Taking the time derivative of V:

$$\dot{V} = s^T\dot{s} + \text{tr}(\widetilde{W}^T\Gamma^{-1}\dot{\widetilde{W}}) \quad (29)$$

Since W^* is constant, we have $\dot{\widetilde{W}} = \dot{\widehat{W}}$. Substituting the Neural Network update law *eq. (25)* into the equation, we get:

$$\dot{V} = s^T\dot{s} - \text{tr}(\widetilde{W}^T\Phi(x)s^T) \quad (30)$$

Substituting the control law *eq. (17)* into the sliding surface derivative *equation (11),* we obtain:

$$\dot{s} = -K.\text{sgn}(s) - (d - \hat{d}) \quad (31)$$

Here, $\hat{d}$ is the aggregate estimated value from both NDO and NNO observers. To simplify, we focus on the disturbance compensation capability of the NN, *i.e.*, $\hat{d} = \hat{d}_{NN} = \widehat{W}^T\Phi(x)$. Thus:

$$d - \hat{d} = (W^{*T}\Phi(x) + \epsilon) - \widehat{W}^T\Phi(x) = -(\widehat{W} - W^*)^T\Phi(x) + \epsilon = -\widetilde{W}^T\Phi(x) + \epsilon \quad (32)$$

Substituting *eq. (32) into (31)*, we obtain the equation for $\dot{s}$:

$$\dot{s} = -K.\text{sgn}(s) + \widetilde{W}^T\Phi(x) - \epsilon \quad (33)$$

Substituting the expression of $\dot{s}$ into, we have:

$$\dot{V} = -s^TK\text{sgn}(s) + s^T\widetilde{W}^T\Phi(x) - s^T\epsilon - \text{tr}(\widetilde{W}^T\Phi(x)s^T) \quad (34)$$

Using the property of the trace operator where $x^TAy = \text{tr}(Ayx^T)$ for a scalar. We have:

$$s^T\widetilde{W}^T\Phi(x) = \text{tr}(\widetilde{W}^T\Phi(x)s^T) \quad (35)$$

Consequently, the expression for $\dot{V}$ becomes:

$$\dot{V} = -s^TK\text{sgn}(s) - s^T\epsilon \quad (36)$$

Let $A = -s^TK\text{sgn}(s), s = [s_R, s_L]^T$, K being diagonal matrices, we have:

$$A = -(k_Rs_R\,sgn(s_R) + k_Ls_L\,sgn(s_L)) \quad (37)$$

Since $x.sgn(x) = |x|$, the expression becomes:

$$A = -\sum_{i=R,L} k_i|s_i| \quad (38)$$

The second term in formula *eq. (36)* is a scalar. We utilize the basic inequality $s^T\varepsilon \leq \|s\|\|\varepsilon\|$. Thus, *equation (36)* becomes:

$$\dot{V} \leq \sum_{i=R,L} k_i|s_i| + |s^T\epsilon| \quad (39)$$

Let k_{min} be the minimum value among the gains in(k_R, k_L). According to the assumption, $\|\epsilon\| \leq \epsilon_N$ Therefore:

$$\dot{V} \leq -k1_{2_{N\,min}} \quad (40)$$

For $\dot{V} \leq 0$we need:

$$k1_{2_{N\,min}} \quad (41)$$

This holds true when$\|s\|$is sufficiently large. To ensure $\dot{V}$ is always negative definite, we simply need to select the switching gain$K$large enough such that:

$$kN_{min} \quad (42)$$

If this condition is satisfied,$\dot{V}$will be negative definite outside a small neighborhood of$s = 0$. This demonstrates that:

The system is stable in the Lyapunov sense.
All signals in the system are bounded.
The tracking error $s(t)$ converges to a small neighborhood of 0, referred to as uniformly ultimately bounded stability.

The size of this region depends on the approximation error of the Neural Network. By selecting the sliding mode controller gain large enough to overcome the upper bound of the approximation error, we can guarantee system stability and convergence of the tracking error to a very small region around the desired point, thereby proving the effectiveness and robustness of the designed controller.

4. SIMULATION RESULTS

4.1. Simulation Setup

A robot model with the parameters listed in *table 2* is utilized. The robot is controlled to navigate along a reference trajectory, traversing various terrain types, including flat, slightly rough, and very rough terrains. These terrains are characterized by disturbance signals from the ground acting upon the robot's wheels.

Table 2. Parameters of the simulated robot

Parameter	Value	Unit
Robot Mass	15	kg
Wheel Radius	0.03	m
Distance between wheels	0.5	m
Robot Moment of Inertia	0.27	$kg.m^2$
Wheel Moment of Inertia	0.0075	$kg.m^2$

To simulate the trajectory tracking process on mixed terrains, this study compares the proposed SMC controller integrated with the dual NDO-NNO observer against three other controllers: a PID controller, a standalone SMC (without observers), and an SMC combined with only an NDO (from our previous study [14]). The parameters for these controllers are provided in *table 3*.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

Table 3. Parameters of the comparative controllers

Controllers	Parameters
PID	$K_p = 5; K_i = 0.1; K_d = 0.01$
SMC	$\lambda = 25; K = 12; \Phi = 0.1$
SMC-NDO	$\lambda = 25; K = 12; \Phi = 0.1; L = 50$
SMC-NDO-NN	$\lambda = 25; K = 12; \Phi = 0.1; L = 50; n = 10; \gamma = 50; W_{nn} = 2$

The NNO is designed with 10 neurons in the hidden layer. The choice of 10 neurons in the hidden layer was determined through heuristic tuning to balance approximation accuracy and computational burden. A larger number of neurons increases computational complexity exponentially, which hinders real-time feasibility for mobile robot microcontrollers, whereas fewer neurons fail to approximate high-frequency uncertainty effectively. Learning rate matrix is set to$W_{nn} = 2$and the width of the radial basis functions is$\gamma = 50$. The training mode for the neural network is adaptive online learning, which occurs continuously throughout the simulation process. The neural network does not undergo a separate offline training phase prior to the simulation. Instead, it automatically updates its weights at each time step to learn and compensate for system uncertainties in real-time.

To bridge the gap between theoretical design and practical implementation, actuator saturation was explicitly incorporated into the simulation model. In physical DDMR platforms, DC motors are constrained by maximum supply voltages. Therefore, the control input U generated by (17) is bounded by a saturation function before being applied to the robot dynamics:$U_{applied} = \text{sat}(U, U_{max})$, where $U_{max} = 24V$. This ensures that the simulated control signals reflect realistic hardware capabilities and constraints.

4.2. Simulation Results

To evaluate the performance of the SMC-NDO-NNO controller, we conducted two simulation scenarios:

Scenario 1: Assumes that only external disturbances (terrain effects) exist, while the plant model parameters remain constant (no parametric uncertainty).

Scenario 2: Introduces internal parametric uncertainties within the plant model, combined with terrain disturbances, to assess the adaptability of the controllers.

In both simulation scenarios, the robot is controlled to move along a straight trajectory with a length of 15m and a velocity of$v = 1(m/s)$. The initial position and orientation of the robot are [0, 0.05, 0.05]. Along the trajectory, we generated three distinct terrain zones: flat, slightly rough, and very rough. These were created by applying a random noise signal to the robot's wheels using a stochastic mathematical function. The roughness level of the terrain corresponds to the amplitude of this noise signal; specifically, [0.1, 5, 100] correspond to flat, slightly rough, and very rough environments, respectively. The profile of the random terrain noise is illustrated in *figure 3*.

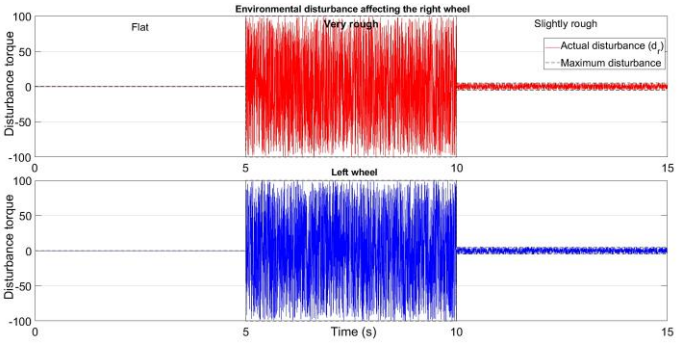

Figure 3. Amplitude of disturbance torque acting on robot wheels across different terrains in Scenario 1.

Figure 4 depicts the trajectory tracking performance of the robot using PID, SMC, SMC-NDO, and SMC-NDO-NNO controllers.

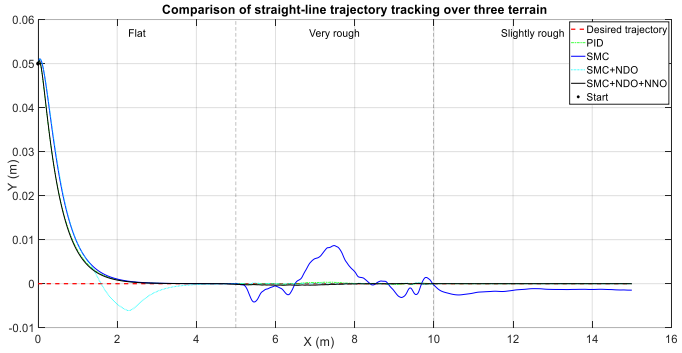

Figure 4. Comparison of trajectory tracking capabilities of different controllers in Scenario 1

The simulation results indicate that the PID controller, being linear, faces significant challenges in handling the robot's nonlinear characteristics and large terrain disturbances, resulting in inaccurate tracking. The SMC controller performs significantly better than PID, demonstrating its robustness against disturbances; however, certain errors and slight deviations persist, particularly when traversing rough terrain zones. The SMC-NDO controller shows improvement over the standalone SMC. Although better, the path does not perfectly coincide with the reference trajectory, and tracking errors remain. Clearly, the SMC-NDO-NNO controller is the optimal performer. The robot's path (black line) nearly perfectly overlaps with the reference trajectory throughout the entire movement.

The tracking error plot in *figure 5* further confirms that the SMC-NDO-NNO controller maintains an error line that is virtually a straight line at zero. This demonstrates its highly accurate noise rejection and error elimination capabilities, even under the harshest conditions.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

Figure 5. Comparison of tracking errors using different controllers

Figure 6 presents the longitudinal error(e_x) lateral error(e_y), and heading error(e_θ).

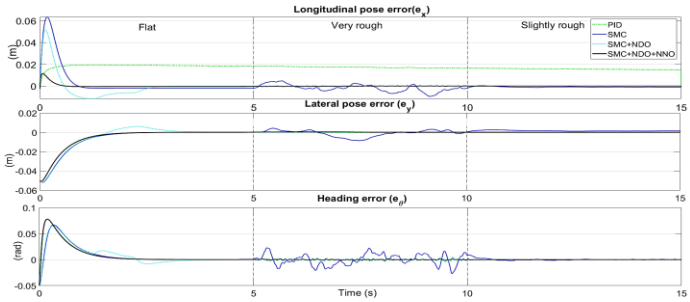

Figure 6. Pose errors of the robot moving on different terrains

The SMC controller exhibits severe heading oscillation in the very rough terrain zone (5-10s). In contrast, the SMC-NDO-NNO controller maintains a stable heading error of nearly zero.

The lateral error (e_y) is a direct consequence of the heading error. When the robot's heading oscillates, it moves in a zig-zag pattern, generating lateral error. This is clearly illustrated by the rippled blue line during the 5-10s interval. Conversely, since the SMC-NDO-NNO maintains a stable heading ($e_\theta \approx 0$), the robot does not drift sideways. Consequently, its lateral error is also nearly zero (the black line is straight). Similar to the longitudinal error, the SMC-NDO-NNO (black) remains the best performer with the flattest and most stable lateral error profile.

To ensure statistical confidence and address the stochastic nature of the terrain disturbances, a multi-run robustness analysis was conducted. The simulation was executed for 50 independent iterations, with the random seed for terrain noise and parametric uncertainty varied in each run. *Table 4* presents the MAE alongside the Standard Deviation (SD, σ).

Table 4. Statistical Tracking Performance under Multi-run Analysis (50 iterations)

Controllers	*Deviation (m)*	*Standard Deviation (σ)*	*e_x (m)*	*e_y (m)*	*e_θ (m)*
PID	0.0187	0.0045	0.0171	0.0020	0.0037
SMC	0.0052	0.0018	0.0035	0.0036	0.0066
SMC+NDO	0.0032	0.0009	0.0021	0.0026	0.0046
SMC-NDO-NN	0.0026	0.0002	0.0004	0.0019	0.0036

The proposed SMC-NDO-NNO not only achieves the lowest mean deviation (0.0021 m) but also exhibits an exceptionally small standard deviation (σ=*0.0002*). This minimal variance statistically proves that the dual-observer controller maintains highly consistent and robust tracking performance, regardless of the unpredictable variations in the simulated operating environment.

Figure 7 displays the disturbance estimation values of the NDO and the combined NDO+NNO observers.

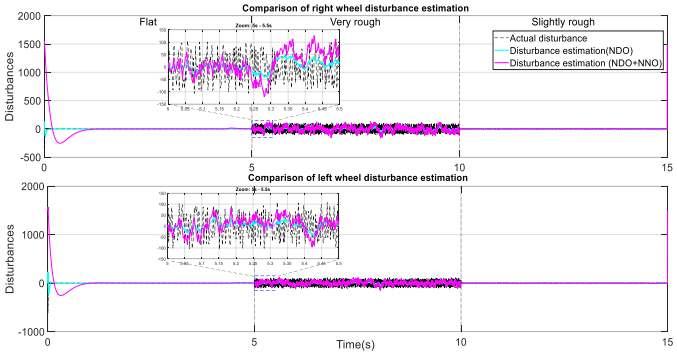

Figure 7. Disturbance estimation of NDO and NDO+NNO observers

The NDO observer (cyan line) captures only the general trend or low-frequency components of the disturbance. In the zoomed-in region of the graph, the cyan line fails to track sharp peaks and high-frequency oscillations. This is because the NDO is designed based on the assumption of slowly varying disturbances, rendering it unable to react promptly to complex and rapidly changing disturbances.

The NDO-NNO observer (purple line) demonstrates distinct superiority. By calculating the MAE index for estimation, results indicate that the average estimation error of the NDO-NNO is 15.9876, which is three times lower than that of the standalone NDO (48.3152). The purple line tracks the actual black disturbance line almost perfectly. It not only captures the general trend but also reconstructs high-frequency oscillatory components. This validates the success of the task decomposition strategy: while the NDO handles basic components, the Neural Network, with its ability to learn and approximate complex nonlinear functions, successfully identifies high-frequency and uncertain noise components that the NDO misses.

Figure 8 illustrates the control voltage signal U_r sent to the right wheel motor over time. This signal determines the wheel's speed and torque. A smooth and stable control signal is ideal as it enhances motor efficiency, saves energy, and reduces mechanical wear. Evaluating the control voltage, we observe that the SMC-NDO (dashed green line) is more stable than PID but is heavily affected by the chattering phenomenon. The SMC-NDO-NNO signal (solid black line) remains superior, providing a control signal that is both stable and smooth, effectively eliminating both large oscillations and chattering.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

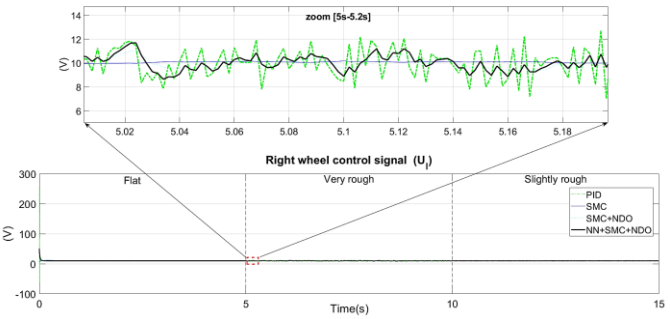

Figure 8. Right wheel control signal in Simulation Scenario 1

In sliding mode control, the objective is to drive the sliding surface value$\sigma$to 0 as quickly as possible and maintain it there. The convergence speed is the rate at which the graph ofσapproaches the zero line. The sliding surface plots for the right and left wheels are shown in *figure 9*. The sliding surface of the basic SMC fluctuates significantly around a large negative value (approx. -15 in the top graph and -25 in the bottom graph). This indicates that the basic SMC cannot effectively handle large disturbances, keeping s far from 0, leading to large tracking errors. During the initial phase (0-5s), the SMC-NDO shows faster sliding surface convergence than the SMC-NDO-NNO because, in this first interval, the NNO weights are not yet optimized. Consequently, the NNO requires a period to collect error data and update weights. During this time, the NDO's contribution might be imprecise, potentially slowing down the overall system convergence. However, the superior advantage of the NNO becomes evident during the rough terrain phase (5-10s). At this point, disturbances and uncertainties become too complex for the NDO to handle alone. In contrast, the NNO, having had time to optimize its parameters, successfully approximates complex nonlinear functions, helping the system maintain highly accurate stability around 0.

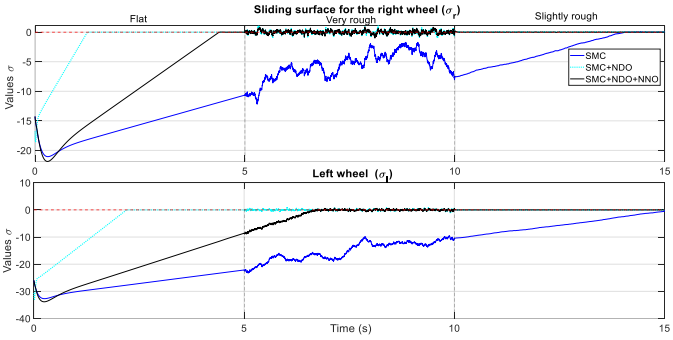

Figure 9. Comparison of sliding surface convergence speed of SMC, SMC-NDO, and SMC-NDO-NNO

In Simulation Scenario 2, we introduce parametric uncertainty into the model to test the Neural Network's capability. To add uncertainty, we randomly vary the robot's actual physical parameters over time by altering the electromechanical resistance coefficient a_{actual}. The value represents all factors naturally slowing down the motor (both electrical and mechanical friction). A largera_{actual}implies the motor tends to stop faster without control voltage. Terrain disturbances are also added alongside uncertainty, as shown in *figure 10*.

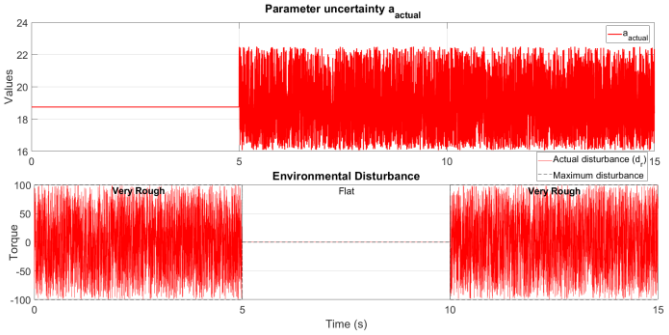

Figure 10. Environment and electromechanical resistance coefficient in Simulation Scenario 2

The phases are (0-5s) environmental disturbance only; (5-10s) uncertainty only; (10-15s) both environmental disturbance and uncertainty. The initial position, reference trajectory, and velocity remain the same as in Scenario 1.

The graph in *figure 11* presents the trajectory tracking error over time, comparing the performance of PID, SMC, SMC-NDO, and SMC-NDO-NNO.

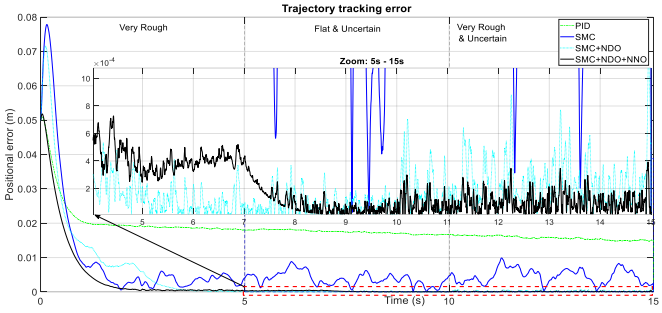

Figure 11. Trajectory deviation over time in Scenario 2

Phase 1 (0-5s): Only environmental disturbance exists. Similar to Scenario 1, the SMC-NDO-NNO demonstrates the highest accuracy and disturbance rejection.

Phase 2 (5-10s): Only parametric uncertainty exists. This phase tests the controller's ability to handle internal model errors. SMC (blue) exhibits spiking errors with very large peaks, indicating near instability against uncertainty. SMC-NDO (cyan) improves upon SMC but still oscillates strongly with continuous chattering. This implies the NDO is effective only for external disturbances, not internal uncertainties. SMC-NDO-NNO (black) maintains an extremely low and stable error ($\approx 4 \times 10^4$m). The black line is much smoother, proving the Neural Network effectively learned and compensated for internal uncertainties.

Phase 3 (10-15s): Both uncertainty and terrain disturbances exist, causing difficulties for PID and SMC. SMC-NDO suffers from severe chattering, with amplitudes much higher than in Phase 2. SMC-NDO-NNO remains the best performer, with oscillation amplitudes comparable to Phase 2.

Figure 12 presents the component error plots (longitudinal, lateral, heading) for Scenario 2. Under harsh conditions with

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

both noise and uncertainty (5-15s), PID, SMC, and SMC-NDO reveal distinct weaknesses. PID and SMC show large and continuous heading errors, indicating difficulty in maintaining direction. Both SMC-NDO and SMC-NDO-NNO maintain errors near zero. However, SMC-NDO exhibits higher overshoot, especially at the beginning of the characteristic. Thus, SMC-NDO-NNO is the most comprehensive and effective controller, suppressing errors well across all three dimensions: longitudinal, lateral, and heading.

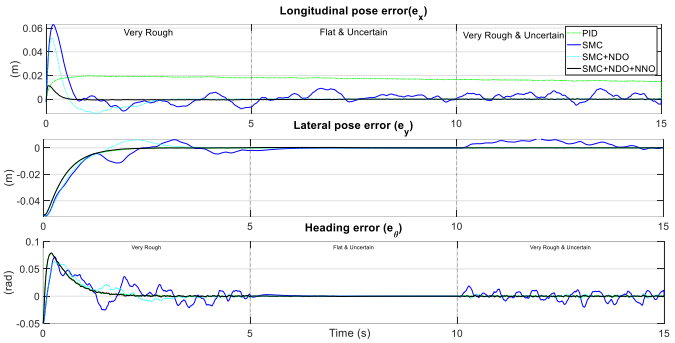

Figure 12. Longitudinal, lateral, and heading errors in Scenario 2

Figure 13 illustrates the right wheel control signal in Scenario 2. Observing the zoomed window, we see the SMC-NDO (dashed green) generates a control signal with large amplitude oscillations that change continuously and abruptly when uncertainty is added, ranging widely from below 8V to nearly 14V. Although the NDO improved tracking performance, it could not eliminate the high-frequency chattering inherent in SMC when continuously compensating for combined noise and uncertainty. For the SMC-NDO-NNO (solid black), the signal exhibits significantly smaller oscillation amplitudes compared to SMC-NDO. The signal varies but is smoother and confined to a narrower range (approx. 9V to 12V). The oscillation frequency is lower, and peaks are less sharp, indicating a more gradual and soft voltage change. This is because instead of reacting abruptly to errors like traditional SMC, the NNO learns and approximates uncertainty components, generating a more precise and less oscillatory compensation signal.

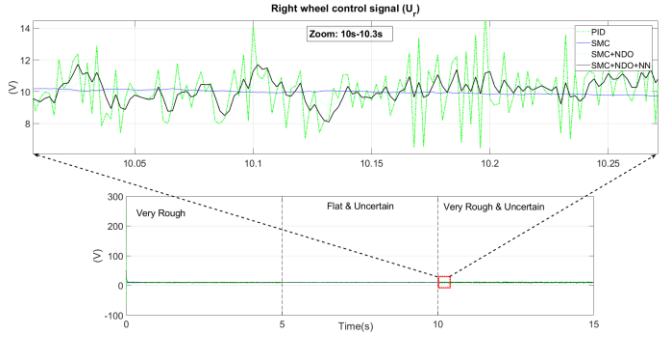

Figure 13. Right wheel control signal in Simulation Scenario 2

In *figure 14*, we compare the two disturbance estimation lines with the actual disturbance (dashed black). The actual disturbance has very large amplitude and high frequency, reflecting the complexity of combined rough terrain and uncertainty. The SMC-NDO (cyan) is smoother with much smaller amplitude, showing that the NDO estimates only the average component of the disturbance while missing complex, fast-changing components. The SMC-NDO-NNO (pink) is capable of accurately reconstructing both the large amplitude and high-frequency components of the actual noise. The pink line almost closely matches the peaks and valleys of the actual black noise line.

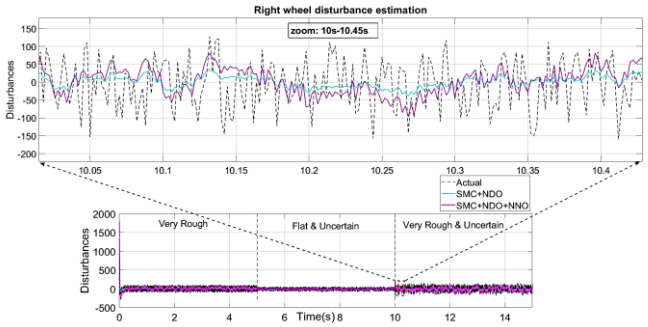

Figure 14. Disturbance estimation values of NDO and NDO combined with NNO

Thus, simulation results demonstrate that the SMC-NDO-NNO controller exhibits significantly improved performance compared to the other controllers (PID, SMC, SMC-NDO), especially in complex environments with noise and uncertainty.

In summary, the comparative simulation across two complex scenarios distinctly highlights the behavioral differences among the tested controllers. The linear PID controller predictably exhibited large errors and no adaptability to nonlinear terrain effects. The standalone SMC, while theoretically robust, proved unstable under complex parametric uncertainties, suffering from strong error oscillations and severe chattering. Introducing the NDO (SMC-NDO) improved the handling of external terrain disturbances; however, it failed to resolve internal parametric uncertainties, allowing chattering to persist. Ultimately, the proposed SMC-NDO-NNO architecture demonstrated superiority by effectively decoupling the disturbance estimation tasks. It provided precise trajectory tracking, near-perfect estimation of combined disturbances and uncertainties, and successful chattering elimination, thereby guaranteeing robust system stability under all tested harsh conditions.

While the proposed architecture demonstrates significant theoretical and simulated advantages, several practical implementation aspects and limitations must be acknowledged. First, deploying an online-learning NN alongside an NDO requires a microcontroller with sufficient computational bandwidth to ensure a high sampling frequency, as computational delays can degrade the sliding mode performance. Second, in a physical DDMR, sensor noise from encoders or IMUs will introduce measurement inaccuracies that were idealized in this simulation. Due to current laboratory hardware constraints, experimental validation is limited at this stage. However, the extended multi-run statistical analysis (*table 4*) incorporating actuator saturation and severe stochastic uncertainties serves as a rigorous preliminary validation. Developing a physical prototype to test the algorithm against

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

real-world sensor noise and processing latencies is the immediate next step in our future work.

5. CONCLUSION

This paper has successfully presented the design, simulation, and evaluation of a hybrid control structure combining Sliding Mode Control, Nonlinear Disturbance Observer, and Neural Network Observer to address the trajectory tracking problem for autonomous differential drive mobile robots in complex environments. The core innovation of the proposed method lies in the intelligent task decomposition, where the NDO focuses on handling external environmental disturbances, while the NNO takes charge of identifying and compensating for internal parametric uncertainties within the robot model.

Through two comprehensive simulation scenarios, encompassing operations on rough terrains and varying model parametric uncertainties, the SMC-NDO-NNO controller has demonstrated significantly improved performance compared to classical controllers such as PID, SMC, and even the improved SMC-NDO version.

Regarding accuracy: The proposed controller achieved the lowest MAE for trajectory tracking, confirming its capability to track the reference trajectory with highly accurate precision even under the harshest conditions.

Regarding disturbance and uncertainty rejection capability: The success of the controller stems from the near-perfect lumped disturbance estimation of the combined NDO and NNO observer system. The Neural Network demonstrated impressive effectiveness in learning and reconstructing high-frequency noise and uncertain components that the standalone NDO could not handle.

Regarding control signal quality: One of the most significant achievements is the near-complete elimination of the chattering phenomenon. This not only ensures stable system operation but also generates a smooth control signal, helping to protect actuators, reduce mechanical wear, and save energy.

In conclusion, the simulation results strongly affirm that the integration of the Neural Network for identifying internal uncertainties represents a breakthrough, thoroughly resolving the inherent drawbacks of SMC and NDO controllers. This method provides a robust, precise, and reliable control solution, opening up wide application potential for mobile robots operating in challenging real-world environments. Future research directions may focus on validating the controller on a physical robot platform to evaluate performance under real-world operating conditions, as well as investigating optimization methods.

Author Contributions: "Conceptualization, Nguyen Van Tien. and Do Khac Tiep; methodology, Nguyen Van Tien, Do Khac Tiep; formal analysis, Nguyen Van Tien; investigation, Nguyen Van Tien; resources, Nguyen Van Tien; data curation, Nguyen Van Tien, Do Khac Tiep; Do Khac Tiep, Nguyen Van Tien, Do Khac Tiep.; Nguyen Van Tien; writing—review and editing. All authors have read and agreed to the published version of the manuscript".

Funding: "This research was funded by the Vietnam Maritime University".

Conflicts of Interest: The authors declare no conflict of interest.

Ethical Approval: The material is the author's original work, which has not been previously published.

REFERENCES

[1] Kanayama, Y., Kimura, Y., Miyazaki, F., & Noguchi, T. A stable tracking control method for an autonomous mobile robot. Proceedings of the IEEE International Conference on Robotics and Automation, 384-389, 1990.

[2] Fierro, R., & Lewis, F. L. Control of a nonholonomic mobile robot: a survey. Proceedings of the 36th IEEE Conference on Decision and Control, 1, 1-6, 1997.

[3] V. I. Utkin, Sliding Modes in Control and Optimization. Springer-Verlag, 1992.

[4] Corradini, M. L., & Orlando, G. A robust tracking controller for mobile robots. Journal of Dynamic Systems, Measurement, and Control, 2007.

[5] J. J. E. Slotine and J. A. Coetsee, "Adaptive sliding controller synthesis for non-linear systems," International Journal of Control, vol. 43, no. 6, pp. 1631-1651, 1986.

[6] Slotine, J. J. E., & Li, W. Applied nonlinear control. Prentice Hall, 1991.

[7] H. Lee and V. I. Utkin, "Chattering suppression in sliding mode control systems," Mechanism and Machine Theory, vol. 42, no. 12, pp. 1657-1667, 2007.

[8] Yan-Jun Liu, Lei Liu, Dapeng Li, Shaocheng Tong, C. L. Philip Chen, Adaptive Neural Network Control for a Class of Nonlinear Systems with Function Constraints on States, IEEE Transactions on Neural Networks and Learning Systems, vol.34 Issue: 6, pp.2732-2741, June 2023.

[9] Ohnishi, M. Shibata, and T. Murakami, Motion control for advanced mechatronics, IEEE/ASME Transactions on Mechatronics, vol. 1, no. 1, pp. 56-67, Mar. 1996.

[10] W. H. Chen, Disturbance Observer Based Control for Nonlinear Systems, IEEE/ASME Transactions on Mechatronics, vol. 9, no. 4, pp. 706-710, Dec. 2004.

[11] Chen, W. H., Yang, J., Guo, L., & Li, S., Disturbance-observer-based control and related methods—an overview. IEEE Transactions on Industrial Electronics, 63(2), 1083-1095, 2016.

[12] W. H. Chen, D. J. Ballance, P. J. Gawthrop, and J. O'Reilly, A nonlinear disturbance observer for robotic manipulators, IEEE Transactions on Industrial Electronics, vol. 47, no. 4, pp. 932-938, 2000.

[13] A. Das, F. L. Lewis, and K. Subbarao, Backstepping approach for controlling a quadrotor using nonlinear disturbance observers, Journal of Intelligent & Robotic Systems, vol. 70, pp. 311-326, 2013.

[14] Do Khac Tiep, Nguyen Van Tien, Design of Sliding Mode Controller Combined with Nonlinear Disturbance Observer for Trajectory Tracking of Mobile Robots in Mixed Terrain, nternational Journal of Electrical and Electronics Research (IJEER), Volume 13, Issue 3, pp. 547-556, 2025.

[15] Tianli Li, Gang Zhang, Tan Zhang, Jing Pan, Adaptive Neural Network Tracking Control of Robotic Manipulators Based on Disturbance Observer, Processes, vol. 12, no. 3, p. 499, 2024.

[16] Bolun Huang, Qi Yang, Disturbance Observer-Based Double-Loop Sliding-Mode Control for Trajectory Tracking of Work-Class ROVs, Journal of Marine Science and Engineering, vol. 10, no. 5, p. 601, 2022.

[17] Liqun Cheng, Wanzhong Chen, Liguo Tian, Ying Xie, A Finite-Time Sliding-Mode Controller Based on the Disturbance Observer and Neural Network for Hysteretic Systems with Application in Piezoelectric Actuators, Sensors, vol. 23, no. 14, p. 6246, 2023.

[18] Ge, S. S., & Wang, C., Adaptive NN control of uncertain nonlinear systems. IEEE Transactions on Neural Networks, 13(6), 1464-1479, 2002.

[19] Chun-Liang Hou, Shie-Jue Lee, A self-constructed radial basis function neural network and its applications, IEEE International Conference, vol. 5, pp. 3623-3628, 2000.

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 14, Issue 1| Pages 242-253|e-ISSN: 2347-470X

[20] Liu, P., Yu, H. & Cang, S. Adaptive neural network tracking control for underactuated systems with matched and mismatched disturbances, Nonlinear Dyn 98, pp. 1447–1464, 2019.

[21] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, Introduction to Autonomous Mobile Robots, 2nd ed. MIT Press, 2011.

[22] Robins Mathew, Somashekhar S. Hiremat, Trajectory Tracking and Control of Differential Drive Robot for Predefined Regular Geometrical Path, Procedia Technology, Volume 25, 2016, pp. 1273-1274, 2016.

[23] Yang, Y.; Wang, P.; Gao, X. A Novel Radial Basis Function Neural Network with High Generalization Performance for Nonlinear Process Modelling. Processes 2022.

[24] J. Park and I. W. Sandberg, Universal Approximation Using Radial-Basis-Function Networks, Neural Computation, vol. 3, no. 2, pp. 246-257, 1991.

[25] T. Poggio and F. Girosi, Networks for approximation and learning, Proceedings of the IEEE, vol. 78, no. 9, pp. 1481-1497, Sep. 1990.

[26] S. Haykin, Neural Networks and Learning Machines, 3rd ed. Pearson, 2009.

[27] J. Moody and C. J. Darken, Fast learning in networks of locally-tuned processing units, Neural Computation, vol. 1, no. 2, pp. 281-294, 1989.

[28] F. L. Lewis, C. T. Abdallah, and D. M. Dawson, Neural Network Control of Robot Manipulators and Non-Linear Systems. Taylor & Francis, 1993.

[29] H. K. Khalil, Nonlinear Systems, 3rd ed. Prentice Hall, 2002.