Journal of Intelligent Systems and Control
https://www.acadlore.com/journals/JISC

Robust Neural Network-Based Trajectory Tracking Control for Mobile Vehicles

Hasan H. Juhi[1], Nihad M. Ameen[1], Sarab A. Mahmood[2], Yousra Abd Mohammed[1*], Ammar A. Yahya[1]

[1] Department of Communication Engineering, University of Technology- Iraq, 10066 Baghdad, Iraq
[2] Department of Electrical Engineering, University of Technology- Iraq, 10066 Baghdad, Iraq

* Correspondence: Yousra A. Mohammed (yousra.a.mohammed@uotechnology.edu.iq)

Received: 11-03-2024 **Revised:** 12-19-2024 **Accepted:** 12-25-2024

Citation: H. H. Juhi, N. M. Ameen, S. A. Mahmood, Y. A. Mohammed, and A. A. Yahya, "Robust neural network-based trajectory tracking control for mobile vehicles," *J. Intell Syst. Control*, vol. 3, no. 4, pp. 251–259, 2024. https://doi.org/10.56578/jisc030405.

Abstract: The ability of neural network-based control systems for trajectory tracking in wheeled mobile vehicles was evaluated in this study. A significant challenge often encountered is the deviation from the desired trajectory, particularly in high-speed motion. A robust control scheme, designed using the Nonlinear Auto-Regressive Moving Average-Level 2 (NARMA-L2) approach, was employed to enhance the tracking performance under dynamic conditions. The NARMA-L2 controller, a well-established technique for nonlinear systems, was utilized to improve the accuracy and robustness of trajectory tracking in the presence of external disturbances and noise. In heavy-duty mobile vehicles, such as agricultural machines, maintaining straight-line motion at high speeds is particularly susceptible to external load effects and system noise. The proposed control strategy integrates several stages, including system modeling, controller design, and the training of the neural network. To optimize the parameters of a proportional-integral-derivative (PID) controller, the Particle Swarm Optimization (PSO) algorithm was applied, ensuring precise regulation of the vehicle's speed. The controller generates a reference velocity, which is fed as a signal to control the motion of the left and right wheels, enabling effective steering and trajectory adherence. Simulation results demonstrate the effectiveness of the proposed controller in mitigating the impact of disturbances and load effects. The optimization of control parameters successfully minimizes the discrepancy between the left and right wheel positions, bringing them closer to zero. The robust parameter optimization approach, which was employed to counteract the influence of external loads, can significantly improve system performance under varying conditions.

Keywords: Differential steering mobile vehicle; Neural network controller; Particle Swarm Optimization (PSO); PI controller; Speed control

1 Introduction

Heavy-duty machines, such as excavators, cranes, and bulldozers, are characterized by complex nonlinear dynamics. The modeling of such systems can be achieved through system identification techniques [1], which are employed to accurately capture the system's behavior in the presence of varying loads, noise, and external disturbances [1]. The study of system parameters is critical for addressing these challenges, with a particular focus on improving performance metrics such as stability, error rate, and robustness to external perturbations [2, 3]. A tunable feedback proportional-integral (PI) controller and a NARMA-L2 controller were compared to show the linearization and neural network effect on system parameters. To ensure the robustness of the design, the neural network was trained with a wide range of operating conditions, including disturbances and uncertainty [4]. Previous research has studied the trade-off between standard accuracy and robust accuracy, revealing the fundamental tension between robustness and generalization obtained [5, 6]. To move the mobile vehicle in a straight line, a PI controller was used to produce the same speed for the left and right wheels. An optical encoder sensor was used to measure each wheel's velocity [7].

2 Research Methodology

2.1 PI Controller Parameter Alteration Mechanism

The proportional and integral control algorithm is most widely in motion control. The digital integrators produce error to develop a prototype design. PI controller. It is a closed-loop feedback control mechanism for gain adjustment, interacting with set point. It leads to achieving a balance between speeds of response to deviations and reducing errors in the long term. By setting the required value, the control unit can be adjusted [8]. The control system component with an equation:

$$U(t) = K_p e(t) + K_i \cdot \int e(t)dt \tag{1}$$

where, K_p is the proportional gain and K_i is the integral gain. The proportional component responds to the error, while the integral component accumulates past errors to eliminate steady state error.

2.2 Model Description

The mobile vehicle is represented in Figure 1, which illustrates the motion of the vehicle along a straight path and its turning angle during manoeuvres. The path of desired position is denoted by d to notify the straight line of the mobile vehicle motion and the angle of mobile vehicle turning is θ_f.

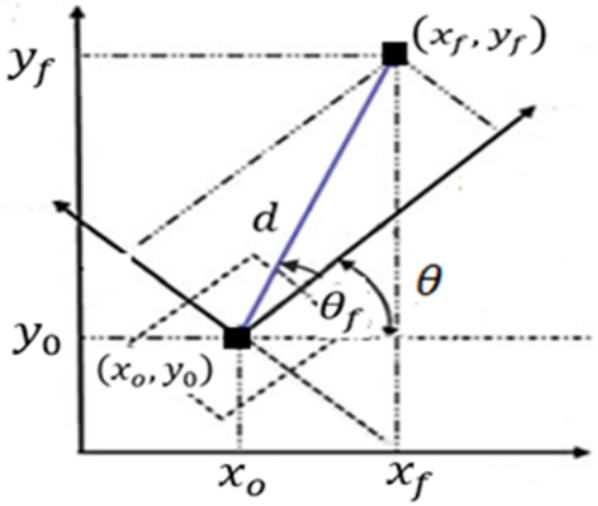

Figure 1. Motion of a differential steering mobile robot [8]

The analogue output of the sensor was digitized and integrated to measure the coinstantaneous speed difference $V_{d(n)}$. An error was obtained by subtracting the two encoder measurements, as shown in Eq. (2).

$$E_{(n)} = \left|V_{(1)} - V_{(2)}\right| \tag{2}$$

where, $E_{(n)}$ is the error of the right and left wheels. An error signal was supplied to the PI controller to generate the active signal controller to reduce the input of the source, aiming to equalize the mobile vehicle wheel's speed [9]. With the tunable PID controller employed, the suitable gain can be chosen to overcome unstable signal noise integrators.

$$\frac{U_{(s)}}{E_{(s)}} = \mathrm{K_p E_{(n)}} + \mathrm{K_i E_{(n-1)}} + \mathrm{K_d E_{(n)}/s} \tag{3}$$

Tuning the proportional and integral controller, by adjusting the parameters of system gain K_p and K_i. If the installation considers the time constant Δt, then $\omega = \frac{1}{\Delta t}$ and $v_R = r\omega_R, v_L = r\omega_L$, where r is the radius of wheel, ω_R, ω_L is an angular speed of the right and left wheels, measured by rad/sec.

To specify the motion of the mobile vehicle when it travels from a current known position (x_o, y_o, θ_o) to a new location (x_f, y_f, θ_f), the distance d and the slope θ of the straight line and final location can be calculated as follows [10]:

$$\mathrm{d} = \sqrt{(\mathrm{x_f} - \mathrm{x_0})^2 + (\mathrm{y_f} - \mathrm{y_0})^2} \tag{4}$$

The slope θ of the straight line and final location is given by:

$$\theta = \arctan \frac{(y_f - y_0)}{(x_f - x_0)} \tag{5}$$

For modeling the mobile vehicle in Matlab, two Direct Current (DC) motors with integrated gears and additional sensors were represented. The purpose of the control system is to provide a mobile vehicle for moving in a straight line, correcting the input voltage to two DC motors. Each speed of the sprockets was sensed by the sensor and counted in the counters. The analogue output of the sensor was digitized and integrated to measure the displacement of the wheels. The instantaneous speed difference by the wheels is called an error signal.

3 The Controller and Mobile Vehicle Parameters

The control system can be classified as a classical controller (e.g., PID controller) and a modern controller (e.g., neural network NARMA-L2 controller). The tuning of the PID controller was obtained by using PSO to satisfy the controller parameters [11]. That enables the system to generate data used for the training phase of the NARMA-L2 controller.

3.1 Particle Swarm Optimization (PSO)

The general form of the PSO method is given by a differential equation [11]. In this study, the PSO technique was employed to fine-tune the parameters of the PID controller, aiming to mitigate the noise present in the measured signal following integration [12, 13]. Figure 2 shows the tuning procedure, where e represents the error between the input and the output of the system. The integral time of squared error (ITSE) criterion was used to minimize the difference between the desired input signal and the output [14]. In tuning phase the parameters of differential is low and can be neglected in this work.

$$V_{i,j}^{(p+1)} = W * V_{i,j}^{(p)} + c1 * R_1 * \left(\text{Pbest}_i - X_{i,j}^{(p)}\right) + c2 * R_2 * \left(\text{Gbest}_i - X_{i,j}^{(p)}\right) \tag{6}$$

$$X_{i,j}^{(p+1)} = X_{i,j}^{(p)} + V_{i,j}^{(p+1)} \tag{7}$$

where, $i = 1, 2 \ldots m; j = 1, 2 \ldots$; m, L and p are the number of particles, variables, and iteration; $X_{i,j}^{(p)}, V_{i,j}^{(p)}$ are the position and velocity of particles; and R_1 and R_2 is an arbitrary number. $c_1 = 1.1, c_2 = 1.2$

$$W = 0.9 - (0.48) * iteration\, number \tag{8}$$

where, W is the weight factor. Maximum iteration number used is 100 iteration. Function of (ITSE) is given by,

$$FIT = \int te^2(t)dt \tag{9}$$

After running PSO program with n= 20 and 100 iterations, the obtained nonlinear controller gains for K_p and K_i are 160 and 100 respectively. After tuning the differential effect is very low and can be neglected in this work.

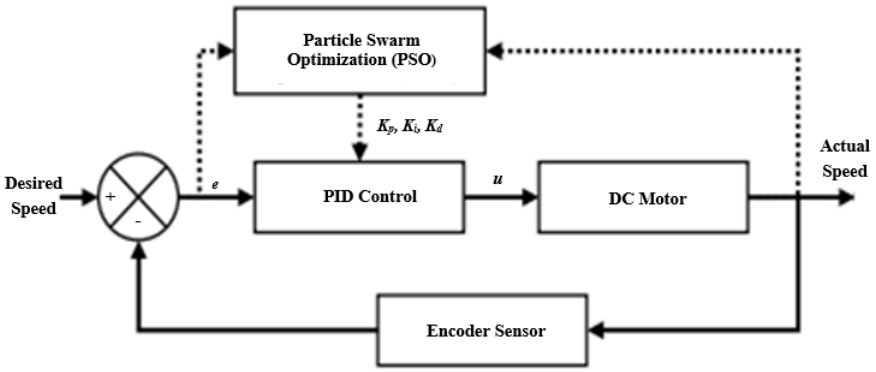

Figure 2. Parameter tuning of the PID controller

3.2 System and PI Controller Implementation

The vehicle hardware contains two optical encoders connected to the motors to measure the speed of the vehicle. The system and the controller implementation are presented in Figure 3. As shown in the figure, a discrete time integrator is located in the feedback path to amplify the speed measurement from the encoder. The speed and position are controlled by the PI control system.

By using PSO method and setting the PI controller parameters, the controller response can be used to generate a sufficient data used for NARMA-L2 controller training. The simulated result of PI controller, as shown in Figure 4, is used to train the NARMA-L2 controller.

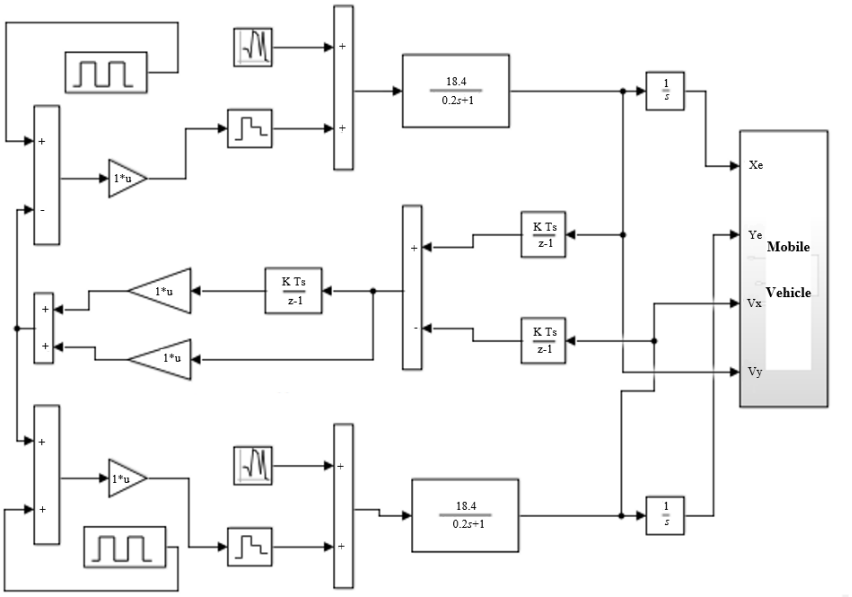

Figure 3. Design the system and the PI controller

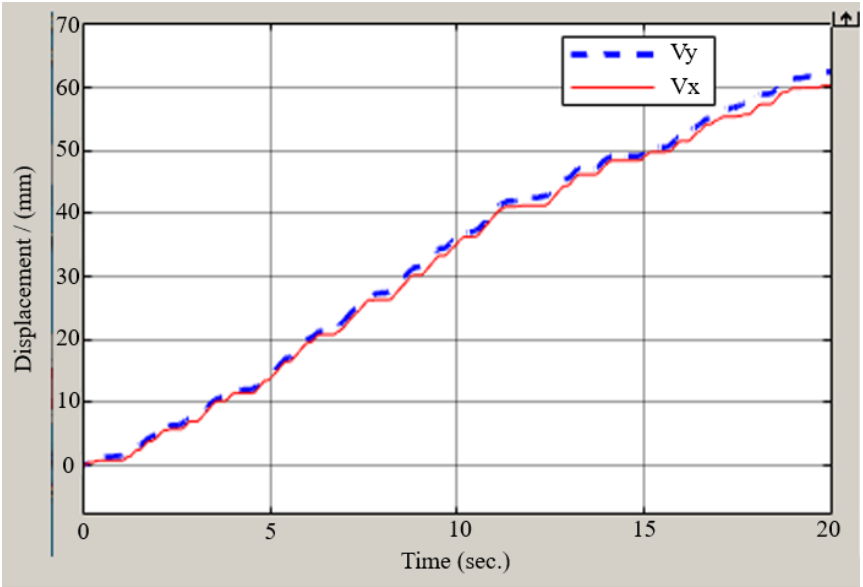

Figure 4. System response using the PI controller

3.3 NARMA-L2 Controller

The controller calculates the desired speed of the left and right wheels and sends commands to the robot. The speed regulator NARMA-L2 represents the construction of a neural network based on the inverse linearization controller for two degrees of freedom [15, 16]. The nonlinear NARMA-L2 neurocontroller equation was used to

follow the reference trajectory by equalizing the two output parameters, i.e., the reference output $yr(k+d)$ and the desired output $y(k+d)$ [17].

Most popular activation function of the networks neurons is (trainlm) function with 9 hidden layers. 100 training sample has used to train 1000 times of epochs is obtained.

4 Mobile Vehicle Analysis and Simulation Results

The simulation and analysis of the mobile vehicle are to determine the variables necessary for the operation to provide the desired results. Continuous and discrete time design frameworks for vehicle navigation were developed foremost with trajectory tracking. In order to follow specific paths, the system parameters were adjusted and the parameters of the PI controller were set, where K_p=160 and K_i=100, which are generally described as speed and position profiles. The proposed design uses the motion equation written as follows [18, 19]:

$$\begin{bmatrix} \dot{x}_f \\ \dot{y}_f \\ \dot{\theta}_f \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\cos\theta_f & \frac{1}{2}\cos\theta_f \\ \frac{1}{2}\sin\theta_f & \frac{1}{2}\sin\theta_f \\ \frac{1}{b} & \frac{-1}{b} \end{bmatrix} \cdot \begin{bmatrix} V_R \\ V_L \end{bmatrix} \tag{10}$$

where, V_R, V_L is the linear vehicle velocity (meter/second), b=1 is the width of the mobile vehicle (m), and θ_f is the angular speed of the mobile vehicle.

Figure 5 shows a simulated flowchart equivalent to the mobile vehicle, where H is the optical sensor, and F_1 and F_2 are the noise. Discrete Time Integrator (DTI) is used in feedback path in this design. The input to the system is desired speed in x and y position V_x and V_y.

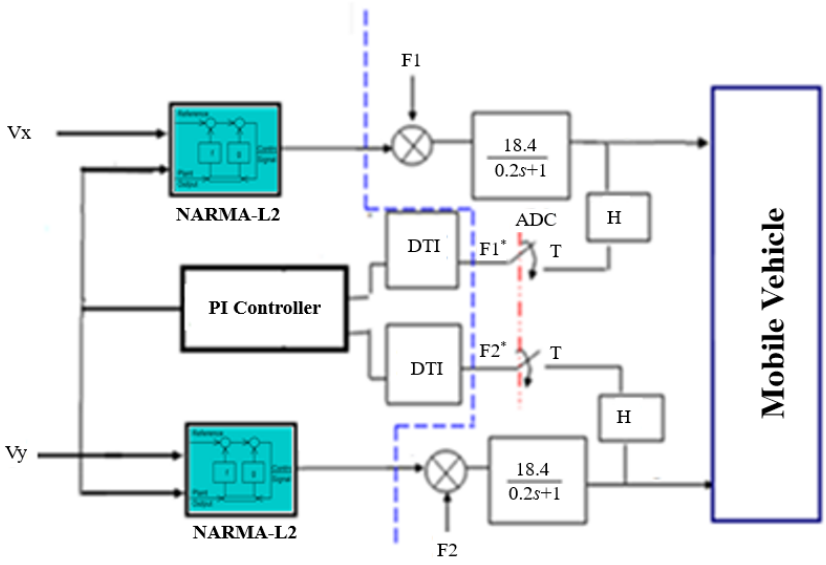

Figure 5. PI and NARMA-L2 controller design

Table 1 shows the training of NARMA-L2 with suitable values, which can yield a suitable response after 1,000 training epochs.

Table 1. Training of the NARMA-L2 parameters

Parameters	Parameter Values	Parameters	Parameter Values
Hidden layer size	9	Delayed plant input	3
Sample interval (sec)	0.005	Delayed plant output	2
Min interval (sec)	0.1	Max interval value (sec)	3
Training sample	100		
Training epochs	1,000		

In the system identification phase, a neural network model of the plant to be controlled was designed. In the controller design phase, the plant model identified was used. The vehicle uses a simple kinematic model of differentially steered wheels, as shown in Eq. (10) [20]. The Simulink model is shown in Figure 6.

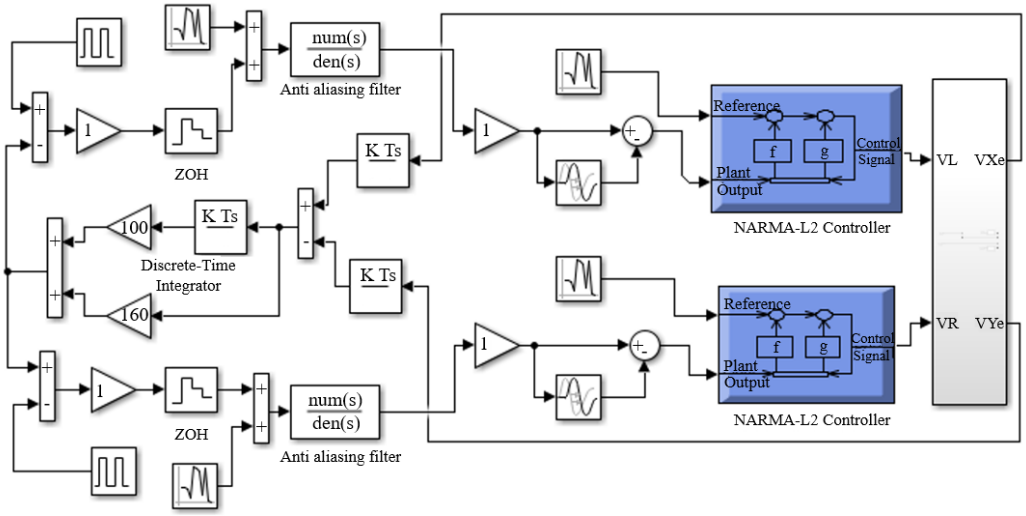

Figure 6. Model design in Matlab/Simulink

The trained network was validated on unseen data to ensure that it generalizes well. The controller was tested in the simulation before deploying it to the real model to follow a reference track (e.g., a track centerline or a track from a motion map) while taking into account vehicle dynamics and constraints. Lane keeping ensures that the vehicle stays within the track limits by adjusting the steering angle. The vehicle's speed was regulated, enabling smooth and safe navigation in complex environments. The optical encoder type was used to measure pulses of sprockets per second, with the time interval to measure the vehicle speed. Figure 7 shows the left and right outputs of the encoder, which represent the angular velocities of the two encoders in volts, with the band limiting achieved by using the analog low-pass filter.

Figure 8 shows the difference in speed between the two wheels and the displacement of the vehicle.

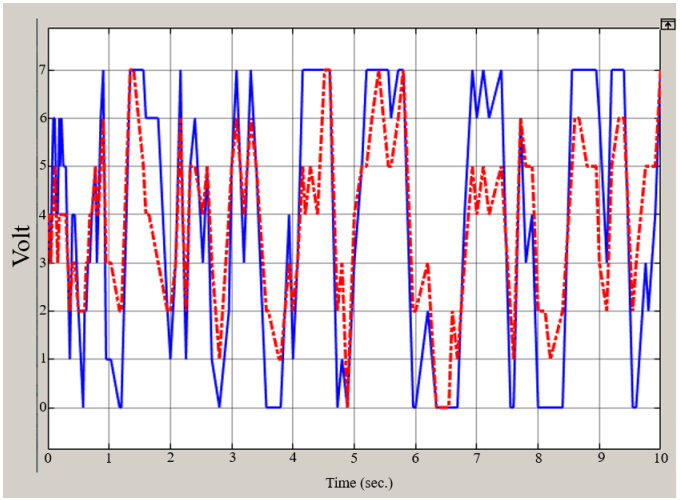

Figure 7. Left and right outputs of the encoders

The model was used in simulation to obtain a theoretical speed and position displacement of the mobile vehicle. The result is shown in Figure 9, which represents two components of speed measured in v_x and v_y.

After reaching the desired position, the reset controller can get a new position or line for the vehicle to follow. From the simulation results, it can be noticed that the output speed difference of the left and right wheels is minimized to zero after setting the PI controller parameters and applying the NARMA-L2 controlling effect. For the position at any moment, the mobile vehicle's displacement can be noticed from the graph of motion, as shown in Figure 10.

The system response in the centimeter distance can be shown in Figure 10. It is noticed that the vehicle can move to a desired location. The mobile vehicle is capable of moving in a precise position and its speed allows cooperative parameter control adjustments, enabling smoother traffic.

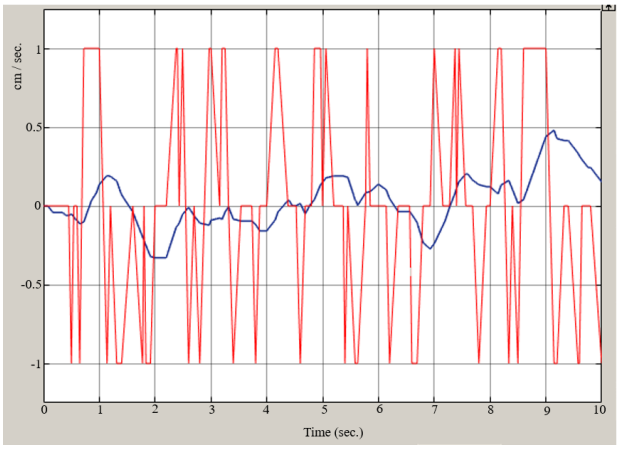

Figure 8. Speed difference between the two wheels of the vehicle

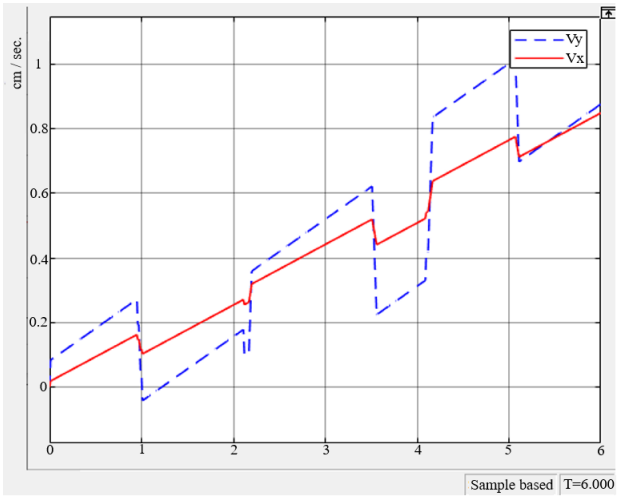

Figure 9. Speed of the mobile vehicle

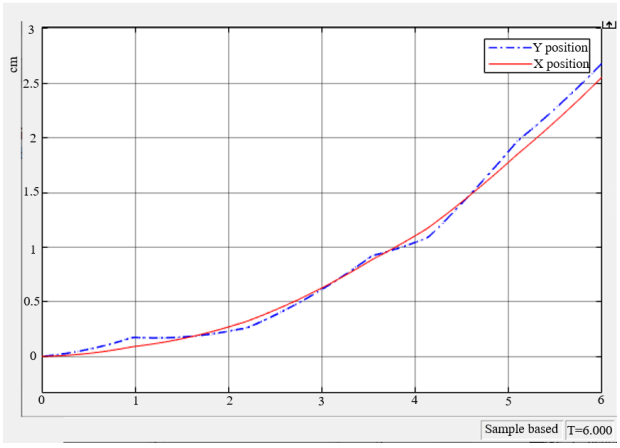

Figure 10. Displacement of the mobile vehicle

5 Conclusions

The speed and position of the mobile vehicle were effectively controlled through the integration of a neural network and a PI controller, enabling the vehicle to follow a desired trajectory. This trajectory was determined by a differential equation that models both the target and actual positions of the vehicle. The reference velocity, used to regulate the vehicle's motion, was fed into the control system. Left and right DC motors were used for steering the mobile vehicle. Lane keeping enabled the vehicle to stay within the track limits by adjusting the steering angle.

The vehicle's speed was regulated for smooth navigation in complex environments. Optical encoders were used to measure the speed of the vehicle. The simulation result is good in terms of the adopted variables. Furthermore, the Simulink model clearly indicated that disturbances and load effects could be effectively rejected, with the system achieving near-zero deviation in pursuit of the reference trajectory. That enables the mobile vehicle to follow a desired trajectory. The simulation result of the required input with noise shows that the control scheme is robust and the designed NARMA-L2 controller is effective.

A motion map could be used for tracking while taking into account vehicle dynamics and constraints. A practical microcontroller with an optical encoder board could be used to compare with the simulated model of the vehicle.

Author Contributions

Hasan H. Juhi, Nihad M. Ameen, Sarab A. Mahmood, Ammar A. Yahya, Writing – original draft, Visualization, Validation, Resources, Methodology, and Investigation. Yousra A. Mohammed: Writing – original draft and data curation. Hasan H. Juhi: Editing, and Validation.

Data Availability

The data used to support the research findings are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] B. Varga, S. Meier, S. Schwab, and S. Hohmann, "Model predictive control and trajectory optimization of large vehicle-manipulators," in *2019 IEEE International Conference on Mechatronics (ICM)*, Ilmenau, Germany, 2019, pp. 60–66. https://doi.org/10.1109/ICMECH.2019.8722886

[2] H. A. Azzawi, N. M. Ameen, and S. A. Gitaffa, "Comparative performance evaluation of swarm intelligence-based FOPID controllers for PMSM speed control," *J. Europ. Syst. Autom.*, vol. 56, no. 3, pp. 475–482, 2023. https://doi.org/10.18280/jesa.560315

[3] K. Yuan, H. Shu, Y. J. Huang, Y. B. Zhang, A. Khajepour, and L. Zhang, "Mixed local motion planning and tracking control framework for autonomous vehicles based on model predictive control," *IET Intell. Transp. Syst.*, vol. 13, no. 6, pp. 950–959, 2019. https://doi.org/10.1049/iet-its.2018.5387

[4] A. T. Humod and N. M. Ameen, "Robust nonlinear PD controller for ship steering autopilot system based on particle swarm optimization technique," *IAES Int. J. Artif. Intell.*, vol. 9, no. 4, pp. 662–669, 2020. https://doi.org/10.11591/ijai.v9.i4.pp662-669

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020. https://doi.org/10.1145/3422622

[6] M. R. Amini, V. Feofanov, L. Pauletto, E. Devijver, and Y. Maximov, "Self-training: A survey," *arXiv:2202.12040*, 2022. https://doi.org/10.48550/arXiv.2202.12040

[7] S. Sanchez and P. A. Bhounsule, "Design, modeling, and control of a differential drive rimless wheel that can move straight and turn," *Automation*, vol. 2, no. 3, pp. 98–115, 2021. https://doi.org/10.3390/automation2030006

[8] A. Barsan, "Position control of a mobile vehicle through PID controller," *Acta Univ. Cibiniensis, Tech. Ser.*, vol. 71, no. 1, pp. 14–20, 2019. https://doi.org/10.2478/aucts-2019-0004

[9] A. H. Issa, S. A. Mahmood, A. T. Humod, and N. M. Ameen, "Robustness enhancement study of augmented positive identification controller by a sigmoid function," *Int. J. Electr. Eng. Inform.*, vol. 12, no. 2, pp. 686–695, 2023. https://doi.org/10.11591/ijai.v12.i2.pp686-695

[10] D. K. Muhsen, A. T. Sadiq, and F. A. Raheem, "Memorized rapidly exploring random tree optimization (MRRTO): An enhanced algorithm for robot path planning," *Cybern. Inf. Technol.*, vol. 24, no. 1, pp. 190–204, 2024. https://doi.org/10.2478/cait-2024-0011

[11] S. A. Mahmood, A. T. Humod, A. H. Issa, and N. M. Ameen, "Robust AVR based on augmented pi controller for synchronous generator," *AIP Conf. Proc.*, vol. 2804, p. 030005, 2023. https://doi.org/10.1063/5.0154585

[12] E. S. Rahayu, A. Ma'arif, and A. Cakan, "Particle swarm optimization (PSO) tuning of PID control on DC motor," *Int. J. Veh. Control Syst.*, vol. 2, no. 2, pp. 435–447, 2022. https://doi.org/10.31763/ijrcs.v2i2.476

[13] S. Bharat, A. Ganguly, R. Chatterjee, B. Basak, D. K. Sheet, and A. Ganguly, "A review on tuning methods for PID controller," *Asian J. Converg. Technol.*, vol. 5, no. 1, pp. 1–4, 2019.

[14] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammas, "Metaheuristic algorithms for PID controller parameters tuning: Review, approaches and open problems," *Heliyon*, vol. 8, no. 5, p. e09399, 2022. https://doi.org/10.1016/j.heliyon.2022.e09399

[15] C. Jeyachandran and M. Rajaram, "Neural network based predictive, NARMA-L2 and neuro-fuzzy control for a CSTR process," *J. Eng. Appl. Sci.*, vol. 5, no. 3, pp. 30–42, 2011.

[16] Y. Kondratenko, K. Wang, O. Kozlov, A. Shevchenko, and A. Denysenko, "Neural network control of the mobile robotic platform's adhesion force," *CEUR Workshop Proc.*, vol. 3538, pp. 65–77, 2023.

[17] K. Srakaew, V. Sangveraphunsiri, S. Chantranuwathana, and R. Chancharoen, "Design of NARMA L2 neuro-controller for nonlinear dynamical system," in *29th International Conference on Modeling, Identification, and Control*, Innsbruck, Austria, 2010, pp. 210–215.

[18] A. S. Al-Araji, A. K. Ahmed, and K. E. Dagher, "A cognition path planning with a nonlinear controller design for wheeled mobile robot based on an intelligent algorithm," *J. Eng.*, vol. 25, no. 1, pp. 64–83, 2019. https://doi.org/10.31026/j.eng.2019.01.06

[19] K. Shojaei, A. Tarakameh, and A. M. Shahri, "Adaptive trajectory tracking of WMRs based on feedback linearization technique," in *2009 International Conference on Mechatronics and Automation*, Changchun, China, 2009, pp. 729–734. https://doi.org/10.1109/ICMA.2009.5246126

[20] A. F. Mohammed, N. Basil, R. B. Abdulmaged, H. M. Marhoon, H. M. Ridha, A. Ma'arif, and I. Suwarno, "Selection and evaluation of robotic arm based conveyor belts (RACBs) motions: NARMA (L2)-FO (ANFIS) PD-I based jaya optimization algorithm," *Int. J. Robot. Control Syst.*, vol. 4, no. 1, pp. 262–290, 2024. https://doi.org/10.31763/ijrcs.v4i1.1243