Engineering Science & Technology
https://ojs.wiserpub.com/index.php/EST/

UNIVERSAL WISER
PUBLISHER

Research Article

RNN-Enhanced Takagi-Sugeno Fuzzy Control for Tesla Model 3 Car-Like WMR Dynamics

Mohamed W. A. Ramadan[1,2*], **Lobna Tarek Aboserre**[3]

[1]Mechatronics Engineering Department, Faculty of Engineering, German International University Berlin Campus, Berlin, 13507, Germany
[2]Mechatronics Engineering Department, Faculty of Engineering and Materials Science, German University in Cairo, New Cairo, 11835, Egypt
[3]Mechatronics Engineering Department, Faculty of Engineering, German International University, New Administrative Capital, Cairo, 4824208, Egypt
E-mail: mohamed.waled@student.guc.edu.eg

Received: 26 September 2025; **Revised:** 24 November 2025; **Accepted:** 15 December 2025

Abstract: This paper investigates the integration of machine learning to enhance the performance of fuzzy control systems, specifically for steering control for Car-Like Wheeled Mobile Robot (WMR) dynamics. A Takagi-Sugeno fuzzy controller is employed to effectively manage the dynamic behavior of a Tesla Model 3, successfully controlling its steering across multiple trials. The non-linear vehicle model consistently follows the desired trajectories dictated by the fuzzy controller. To further enhance performance, data consisting of control signals and errors generated by the fuzzy controller are collected for various steering angles, and used to train a Recurrent Neural Network (RNN) to emulate and replace the controller. The trained RNN is subsequently tested on previously unseen steering angles, demonstrating a rapid response and accurate control signal generation. Remarkably, the RNN successfully generalizes beyond its training range, providing reliable control for steering angles outside the range of the training data. This study highlights the potential of hybrid fuzzy-Artificial Neural Network (ANN) systems to enhance the adaptability and efficiency of control strategies in autonomous vehicle applications.

***Keywords*:** machine learning, fuzzy controller, Takagi-Sugeno, recurrent neural network, Tesla Model 3, car-like Wheeled Mobile Robot (WMR)

1. Introduction

The integration of machine learning techniques with fuzzy logic controllers has significantly advanced the control of autonomous systems. Fuzzy logic controllers, particularly the Takagi-Sugeno (T-S) model, are well known for their adaptability and computational efficiency in handling nonlinear dynamics. Recent research has increasingly focused on enhancing these controllers with data-driven learning methods, such as Artificial Neural Networks (ANNs) and Recurrent Neural Networks (RNNs), to further improve their performance and adaptability [1]-[4].

For instance, the Adaptive Neuro-Fuzzy Inference System (ANFIS) has been widely used to learn control laws directly from data, enabling the development of controllers that can adapt to dynamic environments. This approach combines the learning capability of neural networks with the interpretability of fuzzy logic systems, enabling effective

DOI: https://doi.org/10.37256/est.7120268738

handling of nonlinearities and uncertainties inherent in autonomous systems.

Moreover, integrating fuzzy logic with deep learning models has shown notable improvements in accuracy and generalization within control applications. In particular, incorporating Recurrent Neural Networks (RNNs) into fuzzy controllers has been explored to capture temporal dependencies, thereby enhancing dynamic system modeling and enabling more efficient, real-time adaptive control strategies. Recent studies, for instance, have proposed neuro-fuzzy controllers for autonomous ground vehicle navigation, demonstrating improved performance in complex and unpredictable environments.

In the realm of autonomous vehicle navigation, hybrid neuro-fuzzy approaches have been widely applied to optimize controller performance. The integration of fuzzy logic controllers with Internet of Things (IoT) technologies and genetic algorithms has enabled the development of more efficient and adaptive navigation systems capable of managing complex driving scenarios. Furthermore, the use of fuzzy logic to enhance situational awareness in Advanced Driver Assistance Systems (ADAS) has been actively explored, with the goal of improving both safety and operational efficiency in real-world driving conditions [5], [6].

Recent advancements have also introduced improved Takagi-Sugeno-Kang (TSK) fuzzy neural networks that incorporate balancing composite motion optimization techniques to enhance control performance in autonomous systems. In addition, the integration of Type-2 Fuzzy Logic Controllers with Digital Twin technology and neural networks has been proposed for advanced system management, highlighting the continuous evolution of hybrid control strategies.

This study aims to enhance the performance of a Takagi-Sugeno fuzzy controller by leveraging the capabilities of a Recurrent Neural Network (RNN). The target system for this research is a Tesla Model 3, modelled as a car-like Wheeled Mobile Robot (WMR). This representation simplifies the vehicle’s dynamics while preserving essential characteristics, making it an effective case study for advanced control system development.

The Takagi-Sugeno fuzzy controller is employed to regulate the vehicle’s steering dynamics. Control signals and corresponding errors generated by the fuzzy controller at steering angles of 30°, 60°, and 90° are collected to construct a training dataset for the RNN. The trained RNN is then evaluated on previously unseen steering angles—such as 45°, 100°, and 180°—to assess its ability to generalize beyond the training set [7]-[9].

In this study, the variable φ is defined as the vehicle’s orientation angle, which represents the primary state to be controlled. The controller is therefore designed to regulate the orientation φ so that it accurately tracks the desired reference angle. Although the steering actuator provides the physical mechanism for adjustment, the control objective remains centered on minimizing the orientation error rather than directly controlling the steering angle itself. The control signal generated by the Takagi-Sugeno fuzzy controller, and subsequently by the recurrent neural network, is thus applied to the steering actuator with the specific purpose of aligning the vehicle’s orientation with its target value. This clarification ensures consistency in terminology and accurately reflects the relationship between the controlled state and the actuation mechanism.

In the proposed framework, the Takagi-Sugeno (T-S) fuzzy controller utilizes both the orientation error φ and its derivative $\dot{\varphi}$ to ensure stable and responsive steering control. In contrast, the RNN is intentionally designed to operate using only the error φ as its input. This design choice reflects one of the enhancement objectives of the study: to develop a controller capable of reproducing the dynamic behavior of the T-S fuzzy controller while relying on fewer state variables. By learning the temporal relationships embedded within the training data, the RNN implicitly captures the effect of the error derivative, enabling it to generate control signals that emulate the underlying dynamics without explicitly accessing $\dot{\varphi}$. This reduction in state dependence demonstrates the RNN’s ability to provide an efficient and streamlined control strategy while maintaining high performance.

In the context of this work, the decision to employ a RNN rather than an ANN or other conventional machine-learning models is directly aligned with the dynamic nature of the control problem under investigation. The target system—the Tesla Model 3 modeled as a car-like WMR—exhibits highly time-dependent, nonlinear steering dynamics, making temporal modeling essential for accurate controller reproduction and improvement. While ANNs, Random Forests, k-Nearest Neighbors, or Polynomial Regression can approximate static input-output mappings, they inherently lack a mechanism to capture sequential dependencies and temporal evolution of control signals. In contrast, the RNN used in this study incorporates internal memory through recurrent feedback, enabling it to learn how the Takagi-Sugeno fuzzy controller behaves over time rather than merely fitting instantaneous points. This makes the RNN substantially more suitable for emulating and eventually replacing a dynamic controller. Moreover, recent literature has shown that RNN-based models consistently outperform static machine-learning architectures in dynamic control tasks, achieving improved stability, smoother transient behavior, and better generalization beyond the training range [10].

This paper presents a comprehensive analysis of the hybrid fuzzy-RNN control system, covering mathematical modeling, training methodologies, and performance evaluation. The results demonstrate the effectiveness of integrating RNNs with fuzzy controllers for autonomous vehicle applications, offering insights into their potential for adaptive and efficient control under dynamic and previously unobserved conditions.

2. Tesla Model 3 car-like WMR dynamics model

The Tesla Model 3, depicted in Figure 1a, serves as the physical basis for this study and is mathematically represented as a car-like WMR, illustrated in Figure 1b. The car-like WMR model provides a simplified yet effective abstraction for analyzing the steering dynamics and control strategies of vehicles with similar configurations. This section offers a comprehensive explanation of the mathematical model used for the car-like WMR, detailing its key components, state variables, and control signals.

The model defines multiple state variables representing the system's dynamics, such as the position, orientation, and steering angle, among others. Additionally, the control signals, including steering input and torque, are described in the context of their impact on the system's behavior. Particular emphasis is placed on identifying and focusing on the specific states that are central to the scope of this study, such as the steering angle and vehicle direction, as these directly influence the control objectives [11]-[14].

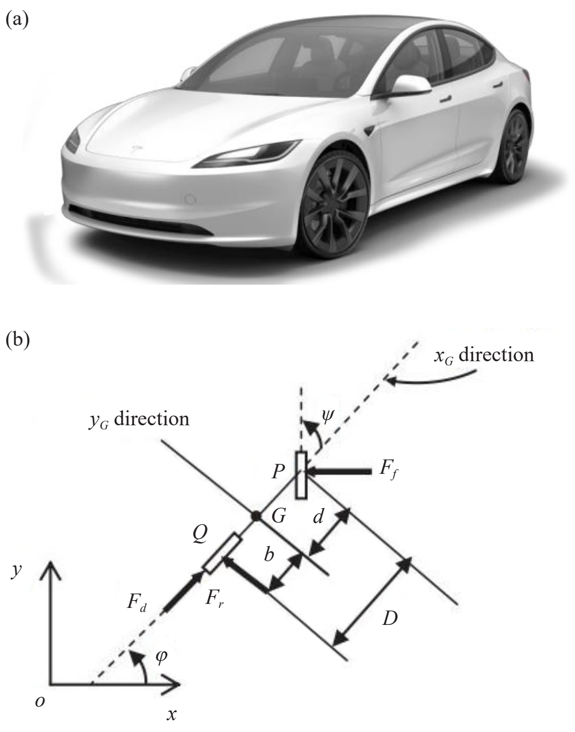


Figure 1. Tesla Model 3 car-like WMR schematic diagram: (a) Tesla Model 3 car [15], (b) Free diagram of the car-like WMR [8]

To ensure a realistic and practical framework, the parameters of the car-like WMR model are carefully chosen to match those of the Tesla Model 3, which is utilized as the case study in this research. This alignment ensures that the results and insights derived from the control and modeling experiments are directly applicable to real-world scenarios involving the Tesla Model 3 and similar vehicles [15], [16].

The Tesla Model 3, as depicted in Figure 1a, is modeled as a car-like WMR. This part extends the kinematic equations analysis by deriving the Newton-Euler dynamic model for a four-wheel, rear-drive, front-steering configuration, as depicted in the equivalent bicycle model in Figure 1b. x_G and y_G are the coordinates of the point G in the world coordinate frame. D is the addition of b and d; b is the length between point G and rear wheel and d is the length between point G and front wheel. φ represents the orientation angle of the vehicle's longitudinal axis, and ψ denotes the heading angle of the vehicle relative to the global coordinate system [8].

The dynamics of the WMR are influenced by several forces, including the driving force (F_d) and lateral slip forces (F_r and F_f), which act perpendicularly to their respective wheels. Before presenting the dynamic equations, we derive the nonholonomic constraints that apply to the Center of Gravity (COG), denoted as point G. The COG is located at a distance b from point Q, and d from point P. Using these distances, the nonholonomic constraints for points Q and P are expressed as,

$$\dot{x}_Q \sin(\varphi) - \dot{y}_Q \cos(\varphi) = 0 \tag{1}$$

$$\dot{x}_P \sin(\varphi + \psi) - \dot{y}_P \cos(\varphi + \psi) = 0 \tag{2}$$

Introducing the positions of Q and P in the global coordinate frame in terms of the COG coordinates (x_G, y_G), such as the derive the following relationships,

$$x_Q = x_G - b\cos(\varphi) \tag{3}$$

$$x_P = x_G + d\cos(\varphi) \tag{4}$$

$$y_Q = y_G - b\sin(\varphi) \tag{5}$$

$$y_P = y_G + d\sin(\varphi) \tag{6}$$

Substituting these into Equations (1), (2) and transforming the velocities of Q and P using the rotational transformation matrix, it obtained,

$$\dot{y}_G = b\dot{\varphi} \tag{7}$$

$$\ddot{\varphi} = \frac{\tan\psi}{D}\ddot{x}_G + \frac{1}{D\cos^2\psi}\dot{x}_G\dot{\psi} \tag{8}$$

From this analysis, the dynamic model is derived, resulting in the Newton-Euler equations that describe the motion of the system. These equations include the effects of F_d, F_r, F_f, and other forces acting on the system, given as,

$$m\left(\ddot{x}_G - \dot{y}_G\dot{\varphi}\right) = F_d - F_f \sin(\psi) \tag{9}$$

$$m\left(\ddot{y}_G + x_G\dot{\varphi}\right) = F_r + F_f \cos(\psi) \tag{10}$$

$$J\ddot{\varphi} = dF_f \cos\psi - bF_r \tag{11}$$

where:

- m: mass of the WMR.

- J: moment of inertia of the WMR about the COG.
- $F_d = \left(\frac{1}{r}\right)\tau_d$: driving force, with r the rear wheel radius, and τ_d the applied motor torque.
- F_f, F_r: front and rear lateral forces.
- T: time constant of the steering system.
- u_s: steering control input.
- τ_d: applied motor torque.

The system is further reduced into a state-space representation with the state vector,

$$x = \left[x_G, y_G, \varphi, \dot{x}_G, \psi\right]^T \tag{12}$$

and the governing equations are written in affine form as:

$$\dot{x} = f(x) + g_1(x)\tau_d + g_2(x)u_s \tag{13}$$

Where the drift term,

$$f(x) = \begin{bmatrix} \left[\cos(\varphi) - \left(\frac{b}{D}\right)\tan(\psi)\sin(\varphi)\right]\dot{x}_G \\ \left[\sin(\varphi) + \left(\frac{b}{D}\right)\tan(\psi)\cos(\varphi)\right]\dot{x}_G \\ \left(\frac{1}{d}\right)(\tan(\psi))\dot{x}_G \\ \left(\frac{1}{a}\right)\left[\left(mb^2 + J\right)(\tan(\psi))\dot{\psi}\dot{x}_G\right] \\ -(1/T)\psi \end{bmatrix} \tag{14}$$

and the input fields,

$$g_1(x) = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{ra}D^2\cos^2\psi & 0 \end{bmatrix}^T \tag{15}$$

$$g_2(x) = \begin{bmatrix} 0 & 0 & 0 & 0 & K/T \end{bmatrix}^T \tag{16}$$

Where,

$$a = \left(\cos^2\psi\right)\left(mD^2 + \left(mb^2 + J\right)\tan^2\psi\right) \tag{17}$$

where the drift term $f(x)$ and input fields $g_1(x)$, $g_2(x)$ are defined based on the derived dynamic equations. This formulation provides a comprehensive representation of the car-like WMR's dynamics, capturing both kinematic and dynamic behavior, making it suitable for advanced control and analysis. The gain parameter (K) is chosen to be 0.1 and the time constant (T) chosen to be 0.1 second.

Table 1 below presents the relevant parameters for the Tesla Model 3, which are critical for modeling its dynamics based on the Ackerman steering principle or similar vehicular dynamic studies. These parameters include mass, moment of inertia, axle-to-center-of-gravity distances, wheelbase, and center-of-gravity height, which together define the physical characteristics of the Tesla Model 3 necessary for constructing accurate dynamic and control models.

Table 1. Tesla Model 3 parameters relevant to ackerman dynamics modeling

Parameter	Value	Description
Mass (m)	1,610 kg	Total mass of the Tesla Model 3.
Moment of inertia	2,900 kg·m²	Estimated moment of inertia about the z-axis.
Front axle to COG distance (a)	1.2 m	Length between the front axle and the COG.
Rear axle to COG distance (b)	1.5 m	Length between the rear axle and the COG.
Total wheelbase ($D = a + b$)	2.7 m	Distance between the front and rear axles (total wheelbase).
Height of center of gravity	0.3 m	Approximate vertical distance of the center of gravity above the ground level.

3. Takagi-Sugeno fuzzy control

The Takagi-Sugeno (TS) fuzzy controller is a computationally efficient and highly adaptable fuzzy inference system that has found significant applications in the control of dynamic nonlinear systems. Unlike Mamdani fuzzy systems, which rely on two-dimensional defuzzification methods, the TS controller employs singleton output membership functions, resulting in a weighted average or sum for the defuzzification process. This efficiency makes the TS controller particularly suited for real-time control applications and complex dynamic systems [17]-[21].

In the TS fuzzy controller, each rule is defined by a combination of a rule antecedent (firing strength) and a rule consequent (output level). For a two-input TS system, the following are the essential components. Initially, rule consequent (z_i) where each rule produces an output level (z_i), which is either a constant or a linear function of the input variables. The TS fuzzy model, originally proposed by Takagi and Sugeno (in 1985), consists of an if-then rule base. The i^{th} rule is described as,

$$\text{If } z_1 \text{ is } Z_1^i \ldots \text{ and } z_p \text{ is } Z_p^i \text{ then } \dot{x} = A_i x + B_i u \tag{18}$$

where the vector z has p components, z_j, $j = 1, 2, ..., p$, and stands for the vector of *scheduling variables*, as their values determine the degree to which rules are active. In this case, z_1 and z_2 is defined to be the ϕ and its rate of change respectively [22]-[27].

Table 2. Sign combinations of premise variables in the Takagi-Sugeno fuzzy rules

Model	Z_1^i	Z_1^{i2}
1	Negativie	Negativie
2	Negativie	Zero
⋮	⋮	⋮
m	Positive	Positive

Z_1^i, …, Z_p^i are the corresponding sets of the premise variables (i = 1, 2, …, m). These sets where defined to be (negative, zero, positive) for each variable. Consequently, the TS system consists of $m = 2^p$ rules (local models). The rules are constructed in Table 2.

The second rule is firing strength where each rule's firing strength is computed using membership functions associated with the rule antecedents,

$$\omega_1^j = \frac{\overline{z}_j - z_j}{\overline{z}_j - \underline{z}_j}, \quad \omega_2^j = 1 - \omega_1^j \quad j = 1, 2, \ldots, p \tag{19}$$

Where $\underline{z}_j$, $\overline{z}_j$ are the minimum and maximum. The membership function of rule i is computed as the product of the weighting functions that correspond to the fuzzy sets in the rule,

$$\omega_i(z) = \prod_{j=1}^{p} \omega_{ij}(z_j) \tag{20}$$

Where $\omega_{ij}(z_j)$ is either Z_1^j or Z_2^j, depending on which weighting function is used in the rule. The resulting membership functions are normal, i.e., $\omega_i(z) \geq 0$ and $\sum_{i=1}^{m} \omega_i(z) = 1$. Finally, using the membership functions given the nonlinear system is exactly represented by the TS fuzzy model given by,

$$\dot{x} = \sum_{i=1}^{m} \omega_i(z)(A_i x + B_i u) \tag{21}$$

The Takagi-Sugeno (TS) fuzzy controller exhibits several key characteristics and advantages that make it a powerful tool for dynamic system control. One notable feature is its computational efficiency. The TS controller's defuzzification process is significantly faster compared to Mamdani systems due to its reliance on linear or constant outputs. This enables the controller to operate effectively in real-time applications.

Another advantage is its dynamic system adaptability. By interpolating between multiple linear models, the TS controller is exceptionally well-suited for handling nonlinear systems and varying operating conditions. This ability allows the controller to adapt seamlessly to changes in system behaviour, ensuring robust performance across a wide range of scenarios.

Additionally, the TS controller offers gain scheduling capability. It acts as an efficient gain scheduler by adapting control gains smoothly across the input space, which is particularly beneficial for systems with dynamic behaviour, such as vehicles or aircraft. This ensures smooth transitions and stability in operations, even under rapidly changing conditions.

The Takagi-Sugeno fuzzy controller is implemented to control the steering dynamics of a Tesla Model 3 vehicle in this study. The controller takes two primary inputs: the steering angle error (φ) and the rate of change of the steering angle error ($\dot{\varphi}$). The output of the controller is the steering control signal, which is used to adjust the vehicle's steering dynamics.

To ensure proper scaling and range normalization, the input signals are scaled down by a factor of 0.01, constraining their values within the range [-1, 1]. Conversely, the output control signal is scaled up by a factor of 100 to amplify the steering input beyond the controller's default output range of [-1, 1].

The controller employs three membership functions for each input variable to capture different operating conditions. These membership functions include 'neg', 'zero', and 'pos', defined as follows (Table 3).

Table 3 and Figure 2b are representing the membership functions for the Takagi-Sugeno fuzzy controller. Figure 2a illustrates Matlab structure of Takagi-Sugeno fuzzy controller. The reference steering angle (φ) is set to $\pi/3$, $\pi/6$, $\pi/2$, $5\pi/9$, and π radians, allowing the controller to dynamically adjust the steering input based on both the error and its rate of change. This ensures smooth and precise steering control under varying operating conditions, enhancing the vehicle's overall performance and stability.

Table 3. Membership functions for Takagi-Sugeno fuzzy controller inputs

Membership function	Range	Parameters
neg	[-1, 1]	[-2, -1, 0]
zero	[-1, 1]	[-1, 0, 1]
pos	[-1, 1]	[0, 1, 2]

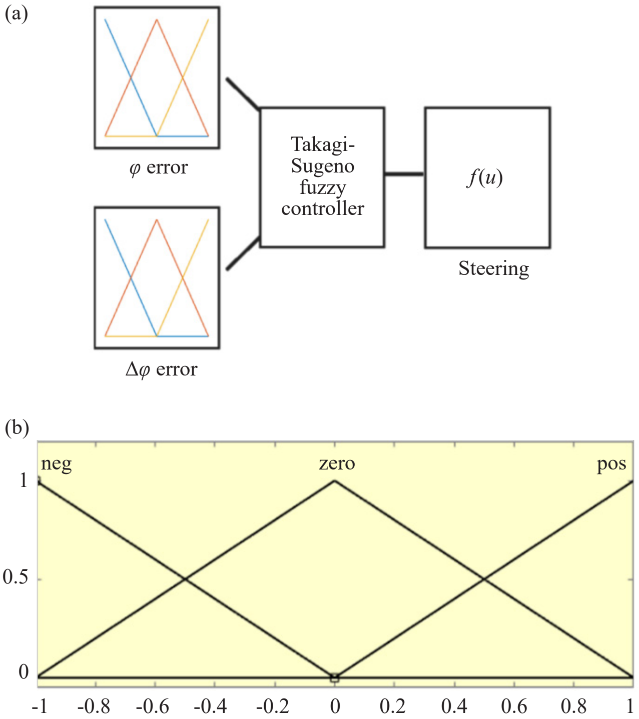


Figure 2. Takagi-Sugeno fuzzy controller; (a) Matlab structure of controller, and (b) Membership function plot

The Figure 3 shows the implementation of the Takagi-Sugeno fuzzy controller with Car-Like WMR for Tesla Model 3 and the controller internal structure.

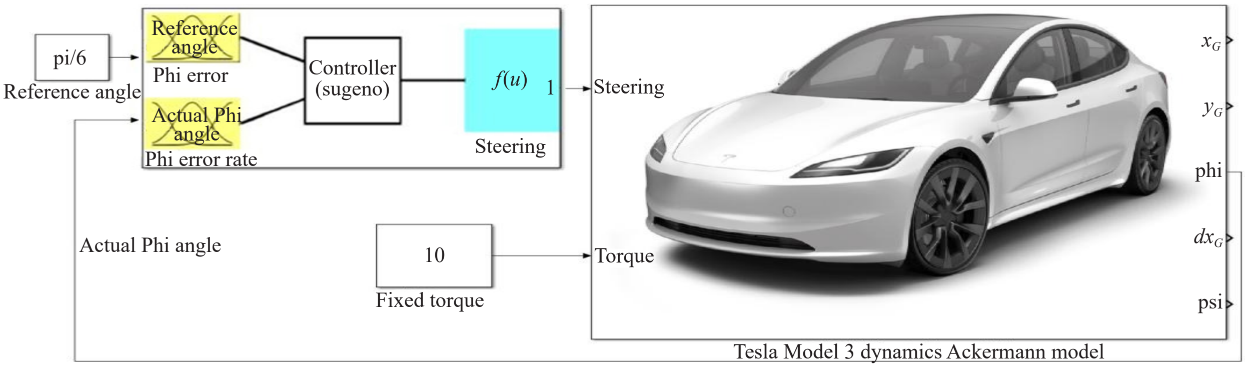


Figure 3. Implementation of Takagi-Sugeno fuzzy controller with car-like WMR for Tesla Model 3

4. Recurrent neural network control

In this study, a Recurrent Neural Network (RNN) is utilized as an alternative to the Takagi-Sugeno fuzzy controller to evaluate its ability to replicate and generalize the control behavior. The RNN is trained using only the input error (φ) of the steering angle and the corresponding output of the TS fuzzy controller. Training data is generated for specific steering angles of $\pi/3$, $\pi/6$, and $\pi/2$, representing different dynamic conditions [28]-[32].

The trained RNN is then tested on unseen data, including steering angles of $\pi/4$, $5\pi/9$, and π. Notably, the angles π and $5\pi/9$ fall outside the range of the training data, making them critical for evaluating the RNN's generalization capability. The results demonstrate that the RNN successfully generates control signals consistent with those of the TS controller for both within-range and out-of-range angles, highlighting its potential as a robust controller [33]-[38].

By focusing solely on the error input (φ) rather than its rate of change, the RNN simplifies the computational process while maintaining high accuracy. This approach underscores the adaptability and scalability of RNNs in dynamic control applications, particularly in scenarios requiring generalization beyond training data [39]-[45].

The recurrent neural network is a type of artificial neural network with internal feedback from the predicted previous output. This Long Short-Term Memory (LSTM) help the RNN to follow the dynamics of any physical system and for specifically a controller for this study. Figure 4 shows the Matlab RNN structure.

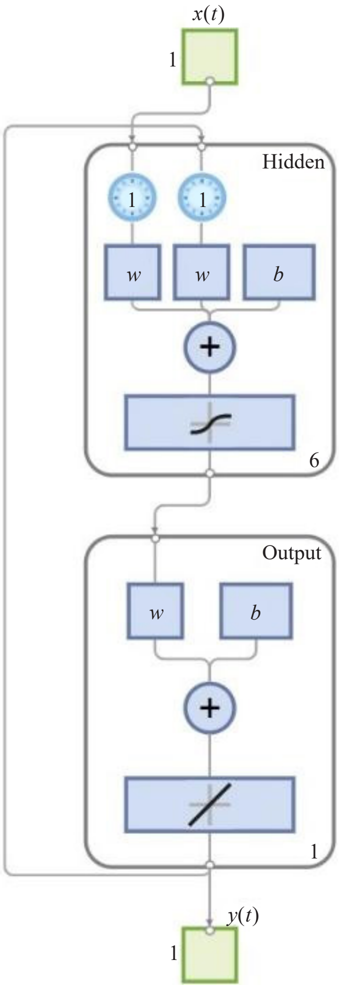


Figure 4. Matlab recurrent neural network structure

The training details of the recurrent neural network is presented at Table 4. The epochs is the iterations which taken to update the weights and bias of the RNN. This RNN consists of 1 hidden layer of 6 neurons and the activation function of it is Tanh function. The elapsed time is the real time taken for the training process (hours: minutes: seconds). The performance is calculated by determining the Mean Square Error (MSE) between the predicted control signal CS_P and the actual control signal CS_A from TS controller as shown below,

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(CS_P^2 - CS_A^2\right) \tag{22}$$

The gradient is the derivative of the loss function which is the MSE which also should be minimized. The Mu is the parameter of the computational power of Matlab.

Table 4. Recurrent neural network training details

Unit	Initial value	Stopped value	Target value
Epoch	0	379	1,000
Elapsed time	-	00: 00: 58	-
Performance	1,380	2.27	0
Gradient	2,270	1.17×10^5	10^{-7}
Mu	10^{-8}	10^8	10^{300}

The Training performance during the training process is calculated by determining the MSE for each epoch is shown at Figure 5.

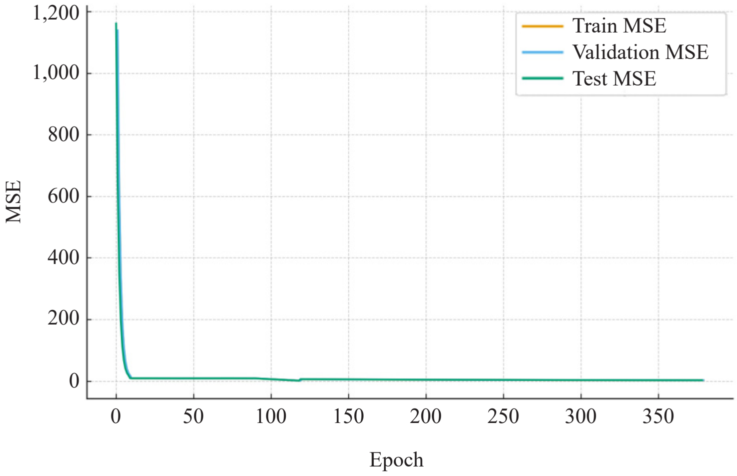


Figure 5. Training performance

The regression plot which illustrated at Figure 6 represents also how training process take a place. The regression plot shows the linear relation between the predicted and actual output. When the predicted output is almost equal to the actual output it creates a straight line.

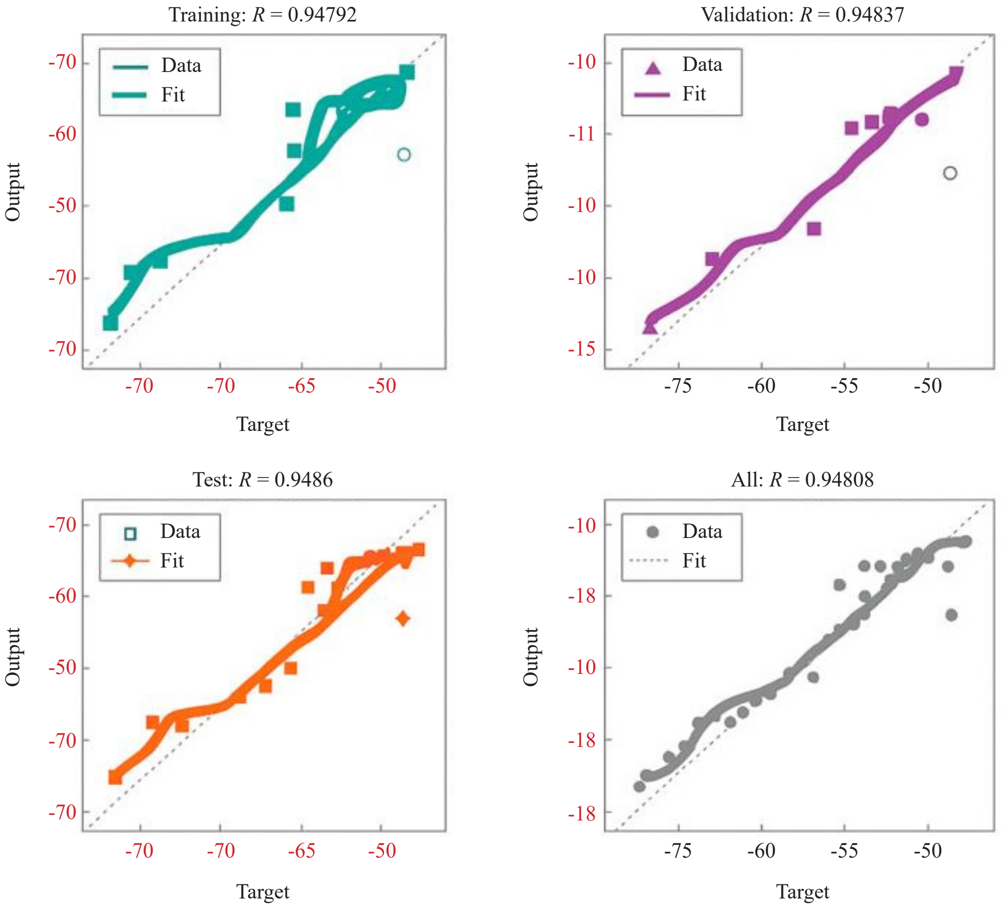


Figure 6. Regression plot

Figure 7, illustrates the control process for Tesla Model 3 car-like WMR by using the RNN as controller from phi (φ) error only.

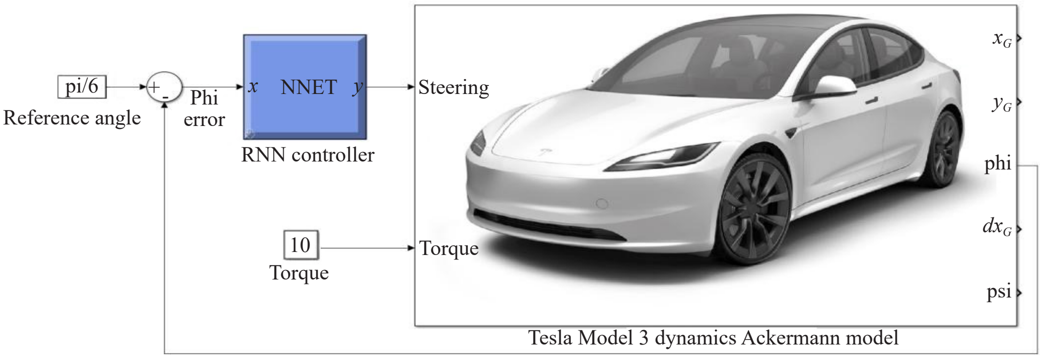
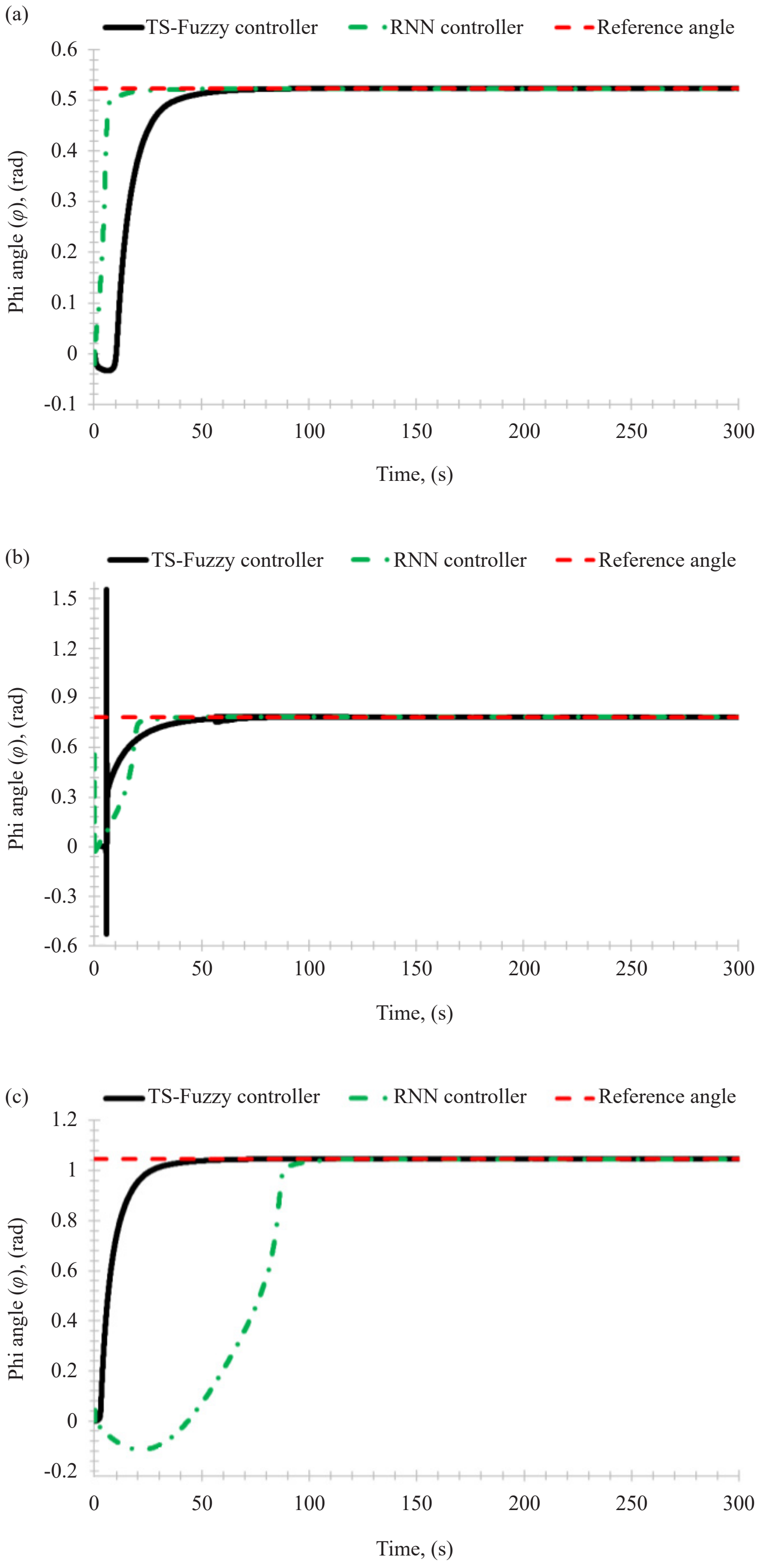


Figure 7. RNN controls Tesla Model 3 car-like WMR in Simulink

5. Results

The results section presents the system responses and error profiles for all scenarios involving different reference steering angles. It includes the error plots comparing the desired steering angle (φ) with the actual steering angle (φ). These results are presented for both the Takagi-Sugeno fuzzy controller and the RNN controller [46]-[51].



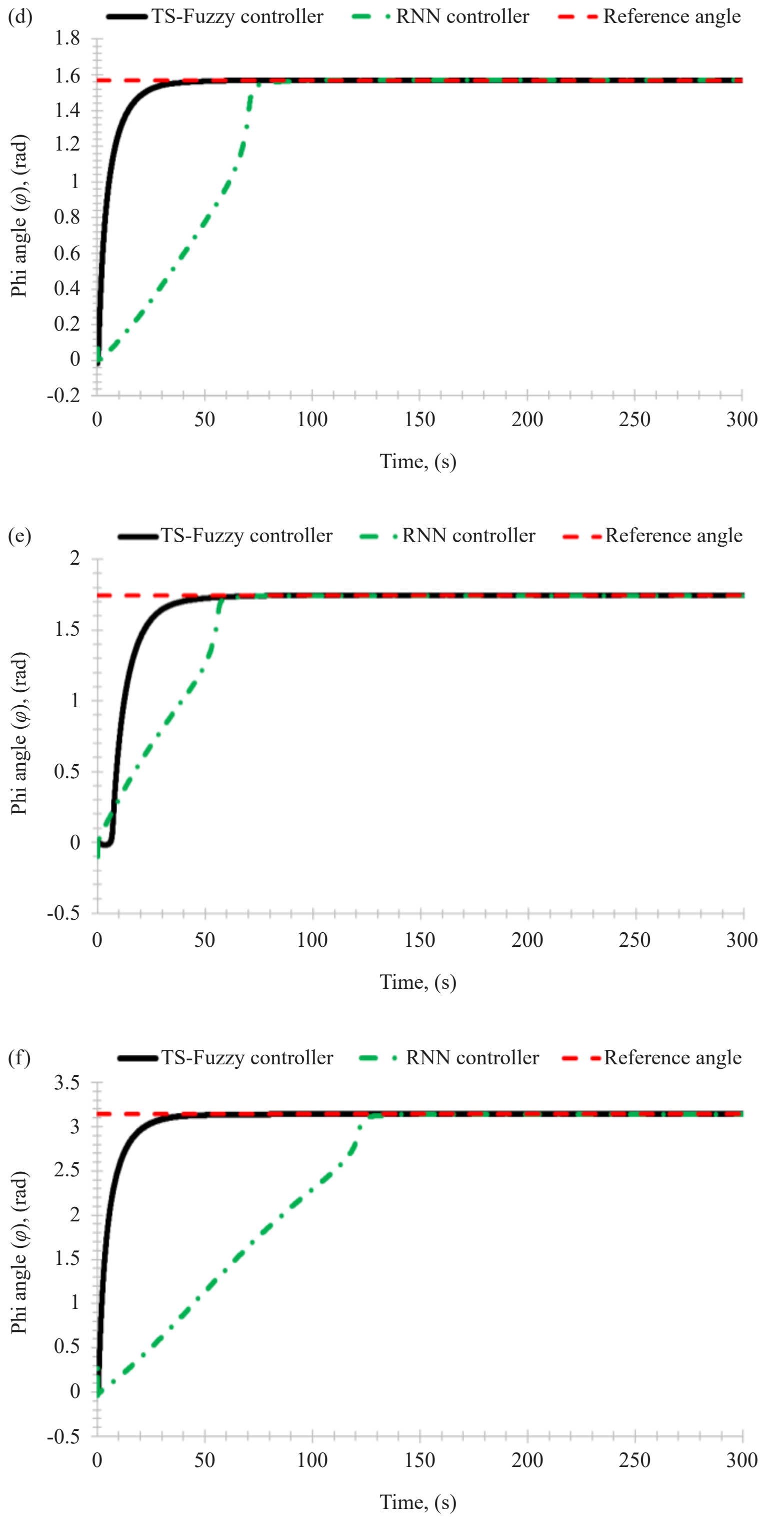
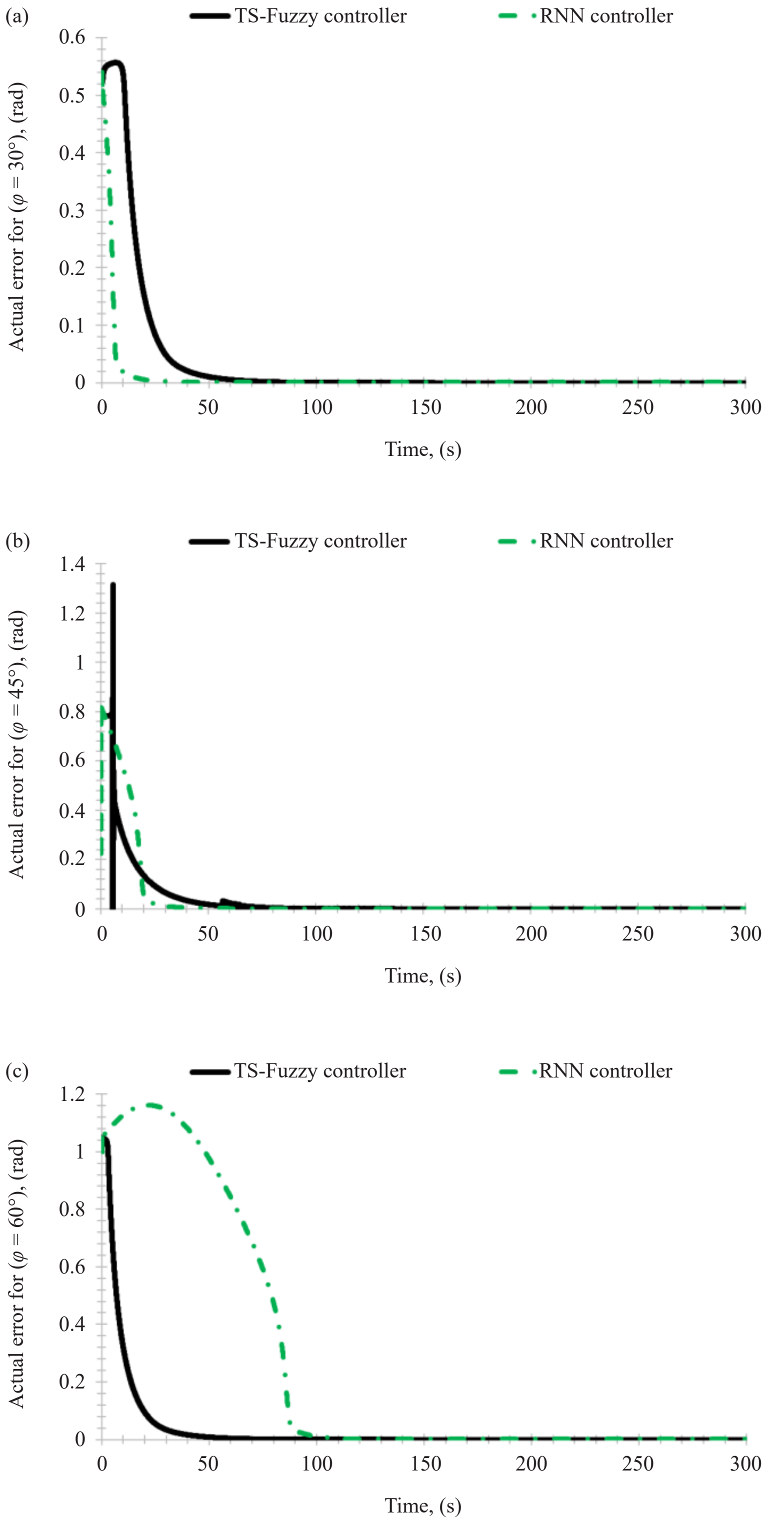


Figure 8. Responses of angle phi (φ) regards RNN controller vs TS-fuzzy controller comparison; (a) $\varphi = 30°$, (b) $\varphi = 45°$, (c) $\varphi = 60°$, (d) $\varphi = 90°$, (e) $\varphi = 100°$, (f) $\varphi = 180°$

Figure 8 illustrates the responses of the Takagi-Sugeno fuzzy controller and the RNN controller, demonstrating how effectively each method drives the steering angle φ toward its reference value across various scenarios. Figure 9 presents the corresponding error plots, highlighting the difference between the reference angle and the actual angle achieved by both controllers.



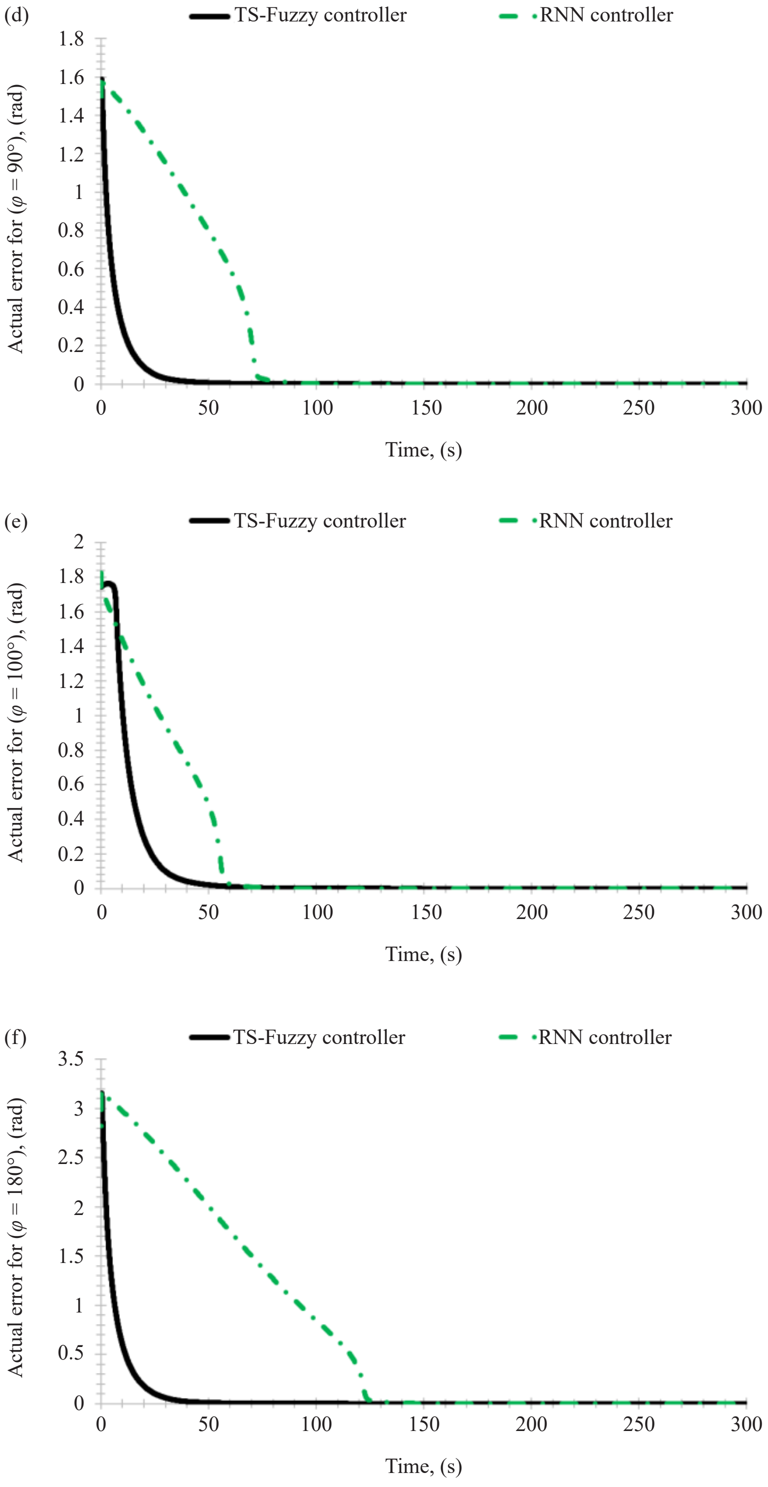


Figure 9. Actual error responses of angle phi (φ) regards RNN controller vs TS-fuzzy controller comparison; (a) φ = 30°, (b) φ = 45°, (c) φ = 60°, (d) φ = 90°, (e) φ = 100°, (f) φ = 180°

These results and figures show the contribution of RNN comparing to the TS-fuzzy controller. The RNN succeed to replace the fuzzy controller and make the mathematical model of the Tesla Model 3 WMR create smooth response for phi (φ). Both controllers react differently regarding for each scenario but for smoother response RNN controller react better than TS-fuzzy controller.

In this study, steering angles of 100° and 180° were included as part of an extended evaluation to examine the robustness and generalization capability of the RNN controller under extreme, non-physical scenarios. While such angles exceed the mechanical limits of real-world Ackermann steering systems, their use in simulation provides a controlled environment to test whether the RNN can maintain stability and avoid overshoot when exposed to highly nonlinear and abrupt reference changes. The results demonstrate that the RNN is capable of handling these extreme inputs smoothly—albeit with a slower response than the T-S fuzzy controller—while also exhibiting improved behavior in realistic operating ranges by reducing overshoot. This highlights the RNN's ability to generalize beyond the T-S training data and underscores its potential as a more adaptive controller in both typical and stress-test conditions [52]-[57].

During the RNN training process, the MSE decreased significantly, indicating that the network successfully learned the underlying mapping from the input error φ to the control signal. Although the gradient reached a relatively large value, this behavior reflects the intended design choice rather than numerical instability (See Table 4). The objective of the RNN in this study is not to achieve a perfect fit to the T-S fuzzy controller data, but rather to learn a control policy that generalizes beyond the training samples and avoids replicating the T-S controller's overshoot behavior. Allowing the training process to stop prior to full convergence prevents overfitting to the T-S control signals and enables the RNN to produce smoother responses. Thus, the observed gradient magnitude is consistent with the goal of maintaining a balance between learning accuracy and improved control performance.

The first two scenarios represent realistic steering conditions within the physical limits of an Ackermann steering mechanism, making them suitable for evaluating the practical performance enhancement achieved by the RNN controller. In these real-world scenarios, the RNN consistently demonstrated smoother and more stable behavior than the T-S fuzzy controller. For Figure 9a, with a reference angle of approximately 0.52 rad, the T-S controller produced a measurable overshoot of about 5.8% and settled within 25-30 s, whereas the RNN exhibited zero overshoot and reached the steady-state value much faster, settling in approximately 10-12 s. Similarly, for Figure 9b, with a reference of roughly 0.80 rad, the T-S controller produced a larger overshoot of around 10% and settled in 40-45 s, while the RNN again achieved zero overshoot and settled within 20-25 s. These results confirm that, under realistic operating conditions, the RNN not only eliminates overshoot but also provides faster and smoother convergence than the T-S fuzzy controller, demonstrating a clear enhancement in control performance.

The regression curve illustrates the training performance of the RNN in comparison with the output of the fuzzy controller (See Figure 6). Although the fuzzy controller exhibits significant overshoot and an unstable transient response—particularly at the initial phase—the RNN is trained to learn the general behavior of the system rather than to replicate these undesirable characteristics. The regression results show a close correspondence between the RNN and the fuzzy controller at the steady state, where both reach the desired angle. However, deviations appear in the early response, which is expected and aligned with the objective of the study: to demonstrate that the proposed RNN controller achieves superior stability and performance compared to the conventional fuzzy controller. Consequently, while the regression R-value may not reach very high levels and the RMSE may not be extremely low due to these intentional differences, the evaluation should be interpreted in the context of controller improvement rather than strict signal replication.

6. Conclusion

This study introduces a hybrid control framework that combines a T-S fuzzy controller with a RNN to improve the steering control performance of a Tesla Model 3 modeled as a car-like WMR. The T-S fuzzy controller provides reliable baseline steering behavior, and its control signals—paired with the corresponding orientation errors—are used to train the RNN. By training the RNN solely on the error signal, the approach aims not only to replicate the T-S controller but also to enhance its dynamic characteristics while reducing dependence on derivative state information. The trained RNN demonstrates strong generalization capabilities, accurately generating control signals for both in-range and out-of-range steering angles, including extreme values used for stress-testing scenarios. Across realistic operating conditions, the RNN consistently delivers smoother and more stable responses by avoiding the overshoot observed with the T-S controller. Furthermore, in practical, real-life steering scenarios—i.e., within the physical limits of Ackermann steering—the RNN responds faster and more efficiently than the T-S controller while maintaining superior smoothness. Although the T-S controller may appear faster in unrealistic high-angle cases, such scenarios fall outside physically feasible vehicle behavior; within actual vehicle limits, the RNN exhibits a clear advantage. These findings highlight the potential of RNN-based controllers as enhanced alternatives to conventional fuzzy controllers for autonomous vehicle applications. The hybrid T-S–RNN approach provides improved smoothness, reduced overshoot, faster real-world response, and lower state dependency, demonstrating its adaptability and robustness in dynamic nonlinear control tasks. Future work may extend this framework by incorporating quantitative performance metrics, additional system states, or more advanced learning architectures to further optimize performance under complex dynamic conditions.

Conflict of interest

The authors declare no competing financial interest.

References

[1] M. S. Rahman and M. H. Ali, "Adaptive neuro fuzzy inference system (ANFIS)-based control for solving the misalignment problem in vehicle-to-vehicle dynamic wireless charging systems," *Electronics*, vol. 14, no. 3, pp. 507, 2025.

[2] G. K. Munahy, "Fuzzy logic deep learning control system for detecting arabic tweets spam based on large language models," *Mars: Journal of Mechanical, Industrial, Electrical and Computer Engineering*, vol. 3, no. 1, pp. 239-249, 2025.

[3] Y. Zeng, Z. A. Hussein, M. H. Chyad, A. Farhadi, J. Yu, and H. Rahbarimagham, "Integrating type-2 fuzzy logic controllers with digital twin and neural networks for advanced hydropower system management," *Scientific Reports*, vol. 15, pp. 5140, 2025.

[4] S. Shaheen, T. Mahmood, M. A. Z. Raja, F. A. Al-Yarimi, M. B. Arain, and J. Hu, "Machine learning-based investigation of activation energy for bio-convective boundary layer flow inspired by challenges in aerospace heat transfer systems," *Engineering Applications of Artificial Intelligence*, vol. 160, pp. 111917, 2025.

[5] M. I. Khan, A. Zeeshan, M. B. Arain, A. S. Alqahtani, and M. Y. Malik, "Temporal stability and non-unique solution of reacting eyring-powell flows over shrinking wedges using neural networks," *Engineering Applications of Artificial Intelligence*, vol. 141, pp. 109828, 2025.

[6] M. B. Arain, D. Hussain, F. A. Al-Yarimi, S. Shaheen, and J. Hu, "Intelligent predictive networks of swimming of microorganism for tangent hyperbolic fluid flow with heat and mass transfer and Thompson and Troian slip conditions using the Levenberg-Marquardt algorithm," *Engineering Applications of Artificial Intelligence*, vol. 152, pp. 110749, 2025.

[7] P. M. P. Raj, S. K. Vittapu, M. Gupta, P. Chinnasamy, P. B. Ganganaik, and P. Gajbhiye, "Evolution of advanced manufacturing technologies: Impact on industry 4.0 and beyond," *AIP Conference Proceedings*, vol. 3342, pp. 020024, 2025.

[8] R. K. Ayyasamy, F. B. Shaikh, N. S. Che Lah, S. Kalhoro, P. Chinnasamy, and S. Krisnan, "Industry 4.0 digital technologies and information systems: Implications for manufacturing firms' innovation performance," in 2023 International Conference on Computer Communication and Informatics (ICCCI), 2023, pp. 1-8.

[9] I. Sapthami, V. V. Narasimha Raju, V. Vaithianathan, P. Chinnasamy, G. Kumaran, and R. K. Ayyasamy, "IoT-based alcohol detection and vehicle control system," in 2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI), 2024, pp. 1-7.

[10] A. Alzaydi, K. Abedalrhman, I. Alotaibi, and F. Alessa, "Advancing autonomous vehicle navigation through hybrid fuzzy-neural network training systems," *International Journal of Advanced Research in Artificial Intelligence*, vol. 12, no. 3, pp. 45-52, 2024.

[11] C. Pozna, R.-E. Precup, E. Horváth, and E. Petriu, "Hybrid particle filter-particle swarm optimization algorithm and application to fuzzy controlled servo systems," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4286-4297, 2022.

[12] M. Sheikhsamad and V. Puig, "Learning-based control of autonomous vehicles using an adaptive neuro-fuzzy inference system and the linear matrix inequality approach," *Sensors*, vol. 24, no. 8, pp. 2551, 2024.

[13] J. C. M. Tan, Q. Cao, and C. Quek, ‘‘FE-RNN: A fuzzy embedded recurrent neural network for improving interpretability of underlying neural network,’’ *Information Sciences*, vol. 663, pp. 120276, 2024.

[14] S. G. Tzafestas, *Introduction to Mobile Robot Control*, 1st ed., Amsterdam, Netherlands: Elsevier, 2014.

[15] B. B. V. L. Deepak, ‘‘Kinematic model of wheeled mobile robots,’’ *International Journal on Recent Trends in Engineering & Technology*, vol. 5, no. 4, pp. 5-10, 2011.

[16] H. M. Alwan, ‘‘Kinematics modeling and simulation of holonomic wheeled mobile robot with mecanum wheels,’’ *Journal of Mechanical Engineering Research and Developments*, vol. 43, no. 5, pp. 451-459, 2020.

[17] M. W. A. Ramadan, ‘‘Neural network representation of aircraft flight dynamics,’’ B. Sc. thesis, German University Cairo (GUC), Cairo, Egypt, 2022.

[18] A. Khadr, ‘‘Multibody dynamics modelling and simulation of a three-wheeled mobile robot using a robotic approach,’’ *International Journal of Intelligent Unmanned Systems*, vol. 12, no. 3, pp. 45-52, 2024.

[19] M. Trojnacki, ‘‘Dynamics model of a three-wheeled mobile robot taking into account slip of wheels,’’ in *Recent Advances in Systems, Control and Information Technology*, R. Szewczyk and M. Kaliczyńska, Eds. Springer, 2017, pp. 378-386.

[20] A. Santana, *Computational Dynamics*, 3rd ed., Wiley-Interscience, 2013.

[21] R. Featherstone, *Rigid Body Dynamics Algorithms*, Springer, 2008.

[22] National Transportation Safety Board (NTSB), *Vehicle Specifications: Tesla Model 3*. SW Washington, DC: NTSB, 2021.

[23] MathWorks, ‘‘Steering system-vehicle dynamics blockset,’’ *MathWorks*, 2023. [Online]. Available: https://www.mathworks.com/help/driving/ref/steeringsystem-vehicledynamics.html. [Accessed Feb. 8, 2025].

[24] L. Tarek, H. Schulte, and A. A. El-Badawy, ‘‘Takagi-Sugeno observer for tower crane system,’’ in Proceedings 30th Workshop on Computational Intelligence, 2020.

[25] M. Sugeno and G. T. Kang, ‘‘Structure identification of fuzzy model,’’ *Fuzzy Sets and Systems*, vol. 28, no. 1, pp. 15-33, 1988.

[26] R. E. Precup and H. Hellendoorn, ‘‘A survey on industrial applications of fuzzy control,’’ *Computers in Industry*, vol. 62, no. 3, pp. 213-226, 2011.

[27] S. R. J. Tafti, Y. Radparvar, H. Salarieh, and A. Alasty, ‘‘Chaos control using PID and lead compensator based on fuzzy gain scheduling technique,’’ in ASME International Mechanical Engineering Congress and Exposition (IMECE), 2007, pp. 773-779.

[28] Y. Bai, H. Zhuang, and D. Wang, *Advanced Fuzzy Logic Technologies in Industrial Applications*, 1st ed., Cham, Switzerland: Springer, 2006.

[29] L. Mazmanyan, ‘‘Fuzzy-model-based (FMB) control of a spacecraft with fuel sloshing dynamics,’’ Ph.D. dissertation, Santa Clara University, Santa Clara, CA, USA, 2018.

[30] N. Silva, F. F. Lopes, C. Valderrama, and M. A. C. Fernandes, ‘‘Proposal of a Takagi-Sugeno fuzzy-PI controller hardware,’’ *Sensors*, vol. 20, no. 7, pp. 1996, 2020.

[31] L. Rajabpour, M. Shasadeghi, and A. Barzegar, ‘‘Design of robust H_∞ fuzzy output feedback controller for affine nonlinear systems: Fuzzy Lyapunov function approach,’’ *arXiv*, 2018. [Online]. Available: https://arxiv.org/abs/1810.08759. [Accessed Jun. 20, 2025].

[32] I. Aldarraji, A. Kakei, A. G. Ismaeel, G. Tsaramirsis, F. Q. Khan, P. Randhawa, M. Alrammal, and S. Jan, ‘‘Takagi-Sugeno fuzzy modeling and control for effective robotic manipulator motion,’’ *Computers, Materials & Continua*, vol. 71, no. 1, pp. 1011-1024, 2022.

[33] P. Saha, S. Dash, and S. Mukhopadhyay, ‘‘Physics-incorporated convolutional recurrent neural networks for source identification and forecasting of dynamical systems,’’ *Neural Networks*, vol. 144, pp. 359-371, 2021.

[34] H. Hewamalage, C. Bergmeir, and K. Bandara, ‘‘Recurrent neural networks for time series forecasting: Current status and future directions,’’ *International Journal of Forecasting*, vol. 37, no. 1, pp. 388-427, 2021.

[35] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, ‘‘Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions,’’ *Journal of Big Data*, vol. 8, no. 1, pp. 53, 2021.

[36] S. K. Sharma and S. Sharma, ‘‘Time series forecasting using recurrent neural network and long short-term memory,’’ *International Journal of Computer Applications*, vol. 177, no. 30, pp. 1-5, 2020.

[37] A. Sherstinsky, ‘‘Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network,’’ *Physica D Nonlinear Phenomena*, vol. 404, pp. 132306, 2020.

[38] J. Lalu and B. Jose,‘‘A survey on time-series data prediction models using recurrent neural networks’’, in International Conference on Networks and Advancesin Computational Technologies, 2021, pp. 305-315.

[39] S. Hochreiter and J. Schmidhuber, ‘‘Long short-term memory,’’ *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[40] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer, 2012.

[41] Y. Bengio, P. Simard, and P. Frasconi, ‘‘Learning long-term dependencies with gradient descent is difficult,’’ *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157-166, 1994.

[42] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, ‘‘LSTM: A search space odyssey,’’ *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222-2232, 2017.

[43] M. W. A. Ramadan and A. R. Ali, ‘‘Artificial neural network control for stabilizing an inverted pendulum using bond graph modeling,’’ in 2025 International Telecommunications Conference (ITC-Egypt), 2025.

[44] A. R. Ali, M. W. A. Ramadan, and M. Helal, ‘‘Real-time detection of surface cracks in wood using deep learning-based image analysis for quality control,’’ in 2025 International Telecommunications Conference (ITC-Egypt), 2025.

[45] A. R. Ali, M. W. A. Ramadan, and Y. Moenes, ‘‘Design and development of an autonomous pipeline inspection robot with integrated control system,’’ in 2025 7th Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2025.

[46] A. R. Ali, M. W. A. Ramadan, and M. Helal, ‘‘Optimizing product design workflows through 3D scanning and reverse engineering technologies,’’ in 2025 7th Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2025.

[47] M. W. A. Ramadan, M. K. Al-Herbawi, and A. R. Ali, ‘‘A GUI-based tool for social anxiety assessment: A comparative study of machine learning models,’’ *SHS Web of Conferences*, vol. 224, pp. 01002, 2025.

[48] A. R. Ali, M. W. A. Ramadan, and P. Zometa, ‘‘Machine learning for EMG-based gesture recognition in brain-computer interfaces and humanoid robots,’’ *International Journal of Intelligent Robotics and Applications*, vol. 9, pp. 1789-1800, 2025.

[49] M. W. A. Ramadan, ‘‘Thermal sensor-based analysis for animal detection and classification utilizing machine learning,’’ in 2025 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA), 2025.

[50] A. R. Ali, M. W. A. Ramadan, S. Ahmed, M. Khafagy, and M. A. Elsalahawy, ‘‘Bond graph modeling and optimization of a bipedal robot leg for enhanced walking stability,’’ in 2025 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA), 2025.

[51] M. W. A. Ramadan, M. K. Al-Herbawi, and A. R. Ali, ‘‘Utilizing machine learning for automated depression severity classification: A comparative study of standard assessment tools,’’ in 2025 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA), 2025.

[52] A. R. Ali and M. W. A. Ramadan, ‘‘Artificial neural networks as digital twins for whispering gallery mode optical sensors in robotics applications,’’ *Photonic Sensors*, vol. 15, no. 2, pp. 1-18, 2025.

[53] A. R. Ali, M. W. A. Ramadan, S. Ahmed, and M. Khafagy, ‘‘Artificial neural network control of a bipedal robot using a bond graph model,’’ in 2024 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT), 2024, pp. 1-5.

[54] A. R. Ali, M. W. A. Ramadan, and M. H. Abdelhafeez, ‘‘Artificial neural networks for personalized lower limb prosthetic socket design: A proof of concept,’’ in 2024 6th Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2024, pp. 202-205.

[55] M. W. A. Ramadan, ‘‘Enhancing humanoid robot performance through neuro-robotic integration with artificial neural networks for EEG and EMG control,’’ M. S. thesis, German International University Berlin Campus (GIU), 2024.

[56] A. R. Ali, M. W. A. Ramadan, and A. T. Darwish, ‘‘Neural network-based classification of hand gestures using electromyography sensor data,’’ in 2023 International Telecommunications Conference (ITC-Egypt), 2023, pp. 231-236.

[57] P. Gautam, P. S. Agrawal, S. Sahai, S. S. Kelkar, and M. Reddy, ‘‘Designing variable ackerman steering geometry for formula student race car,’’ *International Journal of Analytical, Experimental and Finite Element Analysis*, vol. 8, no. 1, pp. 1-11, 2021.