International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

Application of an Adaptive Neuro-Fuzzy Inference System (ANFIS) to Adaptive Cruise Control for Four-Wheeled Vehicles

[1]Muhammad Tsaqif Wicaksono, [2]*Wahyudi

[1,2]Department of Electrical Engineering, Faculty of Engineering, Diponegoro University, Jl. Prof. H. Soedarto, SH, Tembalang-Semarang 50275, Indonesia
*Corresponding Author's E-mail: wahyuditinom@elektro.undip.ac.id

***Abstract* - Adaptive Cruise Control (ACC) is an active safety feature that automatically regulates vehicle speed to maintain a safe distance from a preceding vehicle. Conventional ACC systems often lack adaptability to nonlinear and changing traffic conditions. This paper applies an Adaptive Neuro-Fuzzy Inference System (ANFIS) as the decision-making controller of an ACC system for four-wheeled vehicles in a single-vehicle-following scenario. The controller uses three inputs, namely relative distance, relative velocity and host-vehicle speed, each described by three linguistic sets, and produces a normalized throttle–brake command (TBC) ranging from −1 to 1. A rule base of 27 IF–THEN rules was formulated, and a dataset of 250 samples was generated procedurally in Python from these rules. A Sugeno-type ANFIS was trained in the MATLAB Neuro-Fuzzy Designer using the hybrid learning method for 250 epochs, reaching a training error of 0.197. The trained model was then tested in Simulink under five driving scenarios. The ANFIS outputs followed the trend and magnitude of the expected commands: two of the five outputs (−0.021 and −0.499) fell within the expected ranges, while the remaining three deviated by 0.03–0.16 from the nearest bound of the expected range, with the controller producing slightly softer braking in the critical close-approach case (−0.838 against approximately −1.0). These results indicate that ANFIS can reproduce the intended ACC control logic and provide a basis for further optimization, closed-loop vehicle-dynamics validation and real-time testing.**

Keywords: Adaptive cruise control, ANFIS, fuzzy logic, Sugeno fuzzy inference system, throttle–brake control, MATLAB/Simulink, intelligent vehicle control.

I. INTRODUCTION

Advances in automotive technology have enabled the development and integration of semi-autonomous systems, among which Adaptive Cruise Control (ACC) plays a crucial role. ACC systems automatically regulate vehicle speed to maintain a safe following distance from the preceding vehicle, reducing driver workload and enhancing overall road safety [1], [2]. However, conventional ACC systems often perform poorly under varying road and traffic conditions, mainly because they lack adaptability to dynamic environments [3], [4].

To address these limitations, researchers have increasingly focused on integrating intelligent control strategies. One such approach is the Adaptive Neuro-Fuzzy Inference System (ANFIS), which combines the reasoning capability of fuzzy logic with the learning ability of neural networks. This hybrid architecture allows ANFIS to model nonlinear relationships and adjust control actions in real time based on input variations [1], [3], [5]. Ambuj and Machavaram [3], for instance, showed that ANFIS can improve decision-making reliability in autonomous vehicle navigation by adaptively tuning control parameters in response to sensor feedback.

In this context, ANFIS is a promising method for improving ACC functionality, particularly in scenarios where a vehicle must maintain appropriate spacing and respond intelligently to changes in the motion of the vehicle ahead. This study explores and simulates the behavior of an ANFIS-based ACC system designed to respond to a leading vehicle. The system considers three primary inputs: host-vehicle speed, relative velocity with respect to the vehicle in front, and the distance between the two vehicles.

To support this investigation, a procedural dataset was generated in Python to emulate various traffic scenarios and sensor input conditions. This simulation-based approach enables detailed analysis of how ANFIS responds to different lead-vehicle behaviors without physical testing. By focusing on a single-vehicle-following scenario, this research highlights ANFIS's potential to provide smooth, intelligent speed adjustments that follow the real-time dynamics of a preceding

© 2026 ICJES All Rights Reserved

International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

vehicle, laying the groundwork for further enhancements and real-world applications of adaptive cruise systems.

The remainder of this paper is organized as follows. Section II reviews related work, Section III defines the problem, Section IV describes the methodology, Section V presents and discusses the results, Section VI concludes the paper, and Section VII outlines the future scope.

II. LITERATURE SURVEY

Researchers have applied neuro-fuzzy techniques to longitudinal vehicle control for more than a decade. Ghaffari *et al.* [6] designed ANFIS models to predict a vehicle's future state in stop-and-go traffic, showing that ANFIS can capture car-following behavior from data. For cruise control, Al-Saoudi *et al.* [1] compared PID, fuzzy logic, ANFIS and model predictive controllers, while Choi and Chang [2] used ANFIS to improve ACC performance on sloped roads. Gunasekaran *et al.* [7] proposed a fractional-order ANFIS PD+I controller for ACC and reported its optimization and validation.

Other studies have focused on ACC robustness and safety. Mehraban *et al.* [4] combined fuzzy ACC with model predictive control to respond to dynamic traffic conditions, Shojaeefard *et al.* [9] developed a weather-aware fuzzy ACC with a dynamic reference signal, and Qi [8] addressed responsibility-sensitive safety for ACC under noise disturbances. Yu and Yeo [10] characterized commercial ACC systems by response time, string stability, and asymmetric behavior, highlighting practical performance gaps in current products.

Beyond longitudinal control, neuro-fuzzy methods have been used widely for autonomous and unmanned systems, including ANFIS-based auto-adaptive MPC for path tracking [11], fuzzy obstacle avoidance for unmanned systems [12], ANFIS-derived inference rules for user adoption of autonomous vehicles [13], learning-based control combining ANFIS with linear matrix inequalities [14], optimization-based fuzzy controllers for path tracking [5], and ANFIS-based sliding mode control for UAV trajectory tracking [15]. Most of these works rely on complex hybrid architectures or detailed plant models. A transparent ANFIS controller that maps three directly measurable quantities, namely relative distance, relative velocity and host speed, to a single normalized throttle–brake command, trained from a rule-based procedural dataset, has received less attention and is the focus of this study.

III. PROBLEM DEFINITION

A conventional ACC must decide, at every instant, whether the host vehicle should accelerate, hold its speed or brake, based on its own speed and the motion of the vehicle ahead. Because this relationship is nonlinear and depends on the combination of distance, closing rate and speed, fixed-gain controllers are difficult to tune for all situations. The problem addressed in this paper is therefore to design and train an ANFIS that maps three inputs (relative distance, relative velocity and own speed) to a normalized throttle–brake command in the range −1 (full braking) to 1 (full acceleration), and to evaluate whether the trained model reproduces the intended control logic.

The scope is limited to a single-vehicle-following scenario evaluated in simulation. Predictive models, multi-vehicle (string) interactions, road-gradient effects and closed-loop vehicle dynamics are outside the scope of this study.

IV. METHODOLOGY

The research methodology consists of the systematic stages undertaken to design and implement an ACC system based on ANFIS: (1) system requirements analysis, (2) design of the ANFIS model, and (3) training and testing of the ANFIS. Fig. 1 illustrates the research workflow.

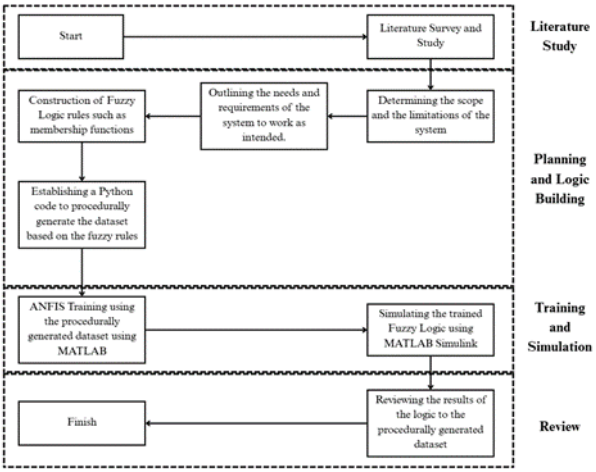


Figure 1: Research workflow

4.1 System Requirements Analysis

As the first step in designing the rule base and dataset, a system requirements analysis was conducted to identify the key parameters that influence ACC performance, i.e., the ACC system inputs. The primary objective of the ACC system is to maintain a safe following distance while preserving a consistent speed in accordance with traffic conditions. The inputs identified are:

© 2026 ICJES All Rights Reserved

International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

- current (own) vehicle speed;
- inter-vehicle distance, i.e., the distance to the vehicle in front;
- relative velocity with respect to the leading vehicle.

In addition to these inputs, the system considers a user-defined setpoint representing the desired cruising speed. The system output is the acceleration or deceleration command, which it then transmits to the throttle or braking system. The functional requirements of the system include:

- real-time acquisition of vehicle data from radar sensors and internal vehicle systems;
- interpretation of the data using ANFIS to generate control decisions;
- transmission of control signals to the vehicle actuators;
- the ability to adapt dynamically to changing environmental conditions.

4.2 Design of the ANFIS Model

The ANFIS design for the ACC system begins with selecting the model structure and formulating the fuzzy rule base. The model integrates the strength of fuzzy logic in handling system uncertainty with the learning capability of artificial neural networks. Its structure was developed by taking into account the input and output variables relevant to ACC: relative distance, relative velocity and own vehicle speed as inputs, which are critical for controlling vehicle speed and maintaining a safe following distance, and the throttle–brake control command as output. Fig. 2 illustrates the resulting system flowchart.

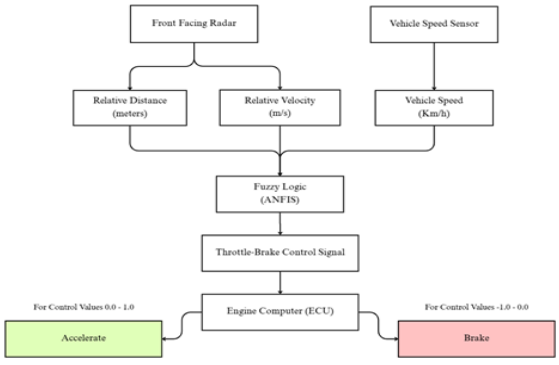


Figure 2: System flowchart

4.3 Fuzzy Variables, Membership Functions and Rule Base

ANFIS is used as the decision-making method of the ACC system. The controller has three input variables, namely relative distance, relative velocity and vehicle speed, and one output variable, the throttle or brake control command. Each input variable is divided into three linguistic sets, using trapezoidal membership functions for the extreme categories and triangular membership functions for the intermediate category.

The vehicle speed comes from the speed sensor installed in the host vehicle's gearbox, while a radar sensor at the front of the vehicle measures the relative distance to the vehicle ahead. The relative velocity is derived from the rate of change of the distance between the host vehicle and the preceding vehicle. A decreasing distance indicates that the host vehicle is approaching the vehicle in front, whereas an increasing distance indicates that the preceding vehicle is moving away. The relative velocity is formulated in (1).

$$V_{rel} = \frac{d_2 - d_1}{t_2 - t_1} = \frac{\Delta d}{\Delta t} \qquad (1)$$

where d_1 and d_2 are the measured distances at times t_1 and t_2. A negative value ($d_1 > d_2$) indicates that the host vehicle is approaching the vehicle ahead, whereas a positive value indicates that the vehicle in front is moving away. Table 1 lists the membership functions and fuzzy sets used in the system.

Table 1: Membership functions and fuzzy sets

Fuzzy variable	Fuzzy set	Membership function parameters	Unit
Relative distance (RD)	Close	trap (0, 0, 10, 30)	m
	Medium	tri (10, 30, 50)	m
	Far	trap (30, 50, 70, 70)	m
Relative velocity (RV)	Closing Fast	trap (−20, −20, −10, 0)	m/s
	Stable	tri (−10, 0, 10)	m/s
	Opening	trap (0, 10, 20, 20)	m/s
Own speed (OS)	Low	trap (0, 0, 40, 70)	km/h
	Medium	tri (40, 70, 100)	km/h
	High	trap (70, 100, 140, 140)	km/h
Throttle–brake command (TBC)	Strong Braking (BS)	(−1.0, −1.0, −0.7)	normalized (−1 to 1)
	Light Braking (BD)	(−0.7, −0.5, −0.2)	
	Neutral (TZ)	(−0.2, 0.0, 0.2)	
	Light Acceleration (TM)	(0.2, 0.5, 0.7)	
	Strong Acceleration (TS)	(0.7, 1.0, 1.0)	

trap = trapezoidal (a, b, c, d); tri = triangular (a, b, c).

© 2026 ICJES All Rights Reserved

International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

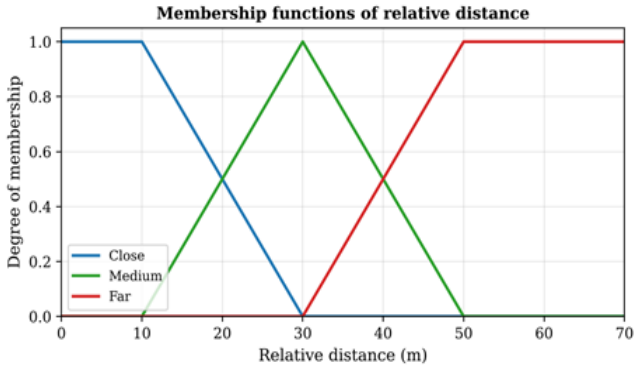


Figure 3: Membership functions of relative distance

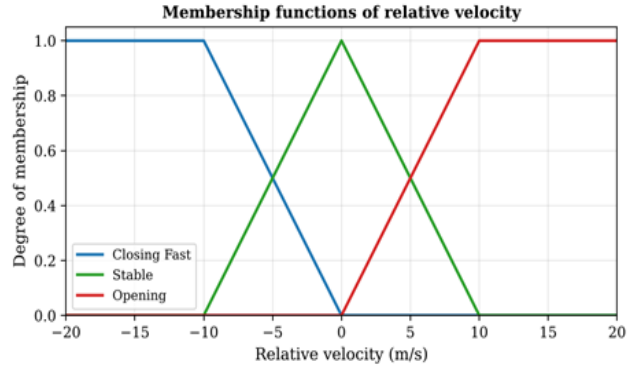


Figure 4: Membership functions of relative velocity

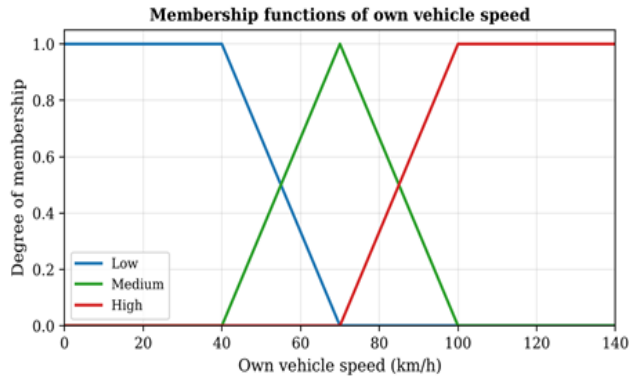


Figure 5: Membership functions of own vehicle speed

Based on the definitions in Table 1, Figs. 3–5, a rule base was constructed. The rule base is a collection of IF–THEN rules that associate every combination of the three input variables with the corresponding control action executed by the engine control unit (ECU) through throttle and brake regulation. A total of 27 rules were formulated to cover all possible input combinations, as presented in Table 2.

Table 2: Fuzzy rule base

Rule No.	Distance (IF)	Rel. velocity (AND)	Own speed (AND)	Output (THEN)
1	Close	Closing Fast	Low	BS
2	Close	Closing Fast	Medium	BS
3	Close	Closing Fast	High	BS
4	Close	Stable	Low	BD
5	Close	Stable	Medium	BD
6	Close	Stable	High	BD
7	Close	Opening	Low	TZ
8	Close	Opening	Medium	TZ
9	Close	Opening	High	BD
10	Medium	Closing Fast	Low	BD
11	Medium	Closing Fast	Medium	BD
12	Medium	Closing Fast	High	BS
13	Medium	Stable	Low	TZ
14	Medium	Stable	Medium	TZ
15	Medium	Stable	High	BD
16	Medium	Opening	Low	TM
17	Medium	Opening	Medium	TM
18	Medium	Opening	High	TZ
19	Far	Closing Fast	Low	TZ
20	Far	Closing Fast	Medium	BD
21	Far	Closing Fast	High	BD
22	Far	Stable	Low	TM
23	Far	Stable	Medium	TZ
24	Far	Stable	High	BD
25	Far	Opening	Low	TS
26	Far	Opening	Medium	TM
27	Far	Opening	High	TZ

BS = strong braking; BD = light braking; TZ = neutral (zero throttle); TM = light acceleration; TS = strong acceleration.

4.4 ANFIS Training and Testing

The ANFIS model training and testing for the ACC system were carried out systematically in several stages. The training dataset was generated procedurally in Python from the fuzzy rule base and contains the input variables (inter-vehicle distance, relative velocity and own vehicle speed) together with the corresponding expected output. Generating data that cover the entire input space was intended to reduce the risk of overfitting, that is, the tendency of a model to memorize the training data without generalizing to new, unseen data.

The ANFIS model was trained using a hybrid learning method that combines the backpropagation algorithm with the least-squares method. Backpropagation updates the nonlinear

© 2026 ICJES All Rights Reserved

International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

parameters of the fuzzy membership functions, while the least-squares method optimizes the model's linear consequent parameters. Training was performed in MATLAB using the Neuro-Fuzzy Designer (anfisedit), which allows the ANFIS model to be designed, trained and tested visually, and the trained model was subsequently tested in Simulink.

V. RESULTS AND DISCUSSION

The ANFIS training was conducted in MATLAB using the Neuro-Fuzzy Designer and the Fuzzy Logic Designer, and the trained model was then tested in Simulink to evaluate the fuzzy system's performance. Fig. 6 shows the fuzzy logic design used for training. The model uses three inputs and a single output representing the throttle or brake command, implemented as a Sugeno-type fuzzy inference system (FIS).

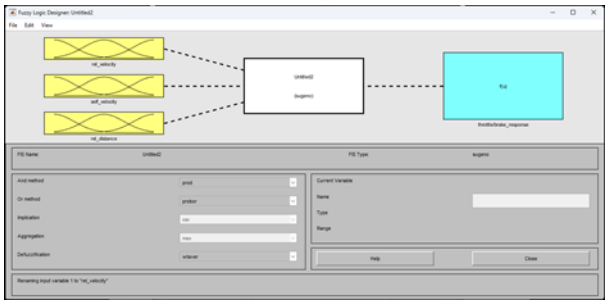


Figure 6: Sugeno-type fuzzy logic design in MATLAB

After the fuzzy logic design, a dataset of 250 samples, each representing the expected output for the corresponding input conditions, was used to train the system. In the Neuro-Fuzzy Designer, the FIS structure was generated by grid partitioning with three membership functions for each input, producing the structure shown in Fig. 7.

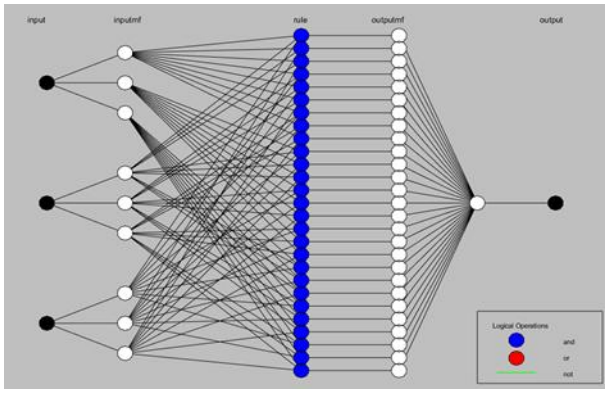


Figure 7: ANFIS structure

Training was run for 250 epochs with an error tolerance of 0.01. Fig. 8 shows the training result: the training error decreased steadily and reached a minimum of 0.197 at the 250th epoch.

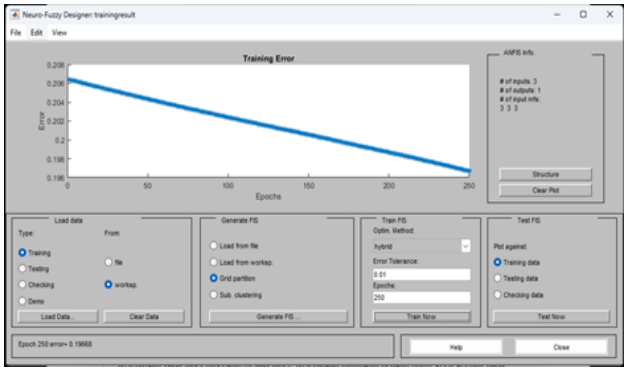


Figure 8: Training error of the ANFIS in MATLAB

To evaluate the trained model, a test was conducted in Simulink using the simplified setup shown in Fig. 9. Three constant blocks representing relative distance, relative velocity and vehicle speed were combined by a Mux block and fed to the Fuzzy Logic Controller block, and the output was shown on a display block on the right-hand side of the model.

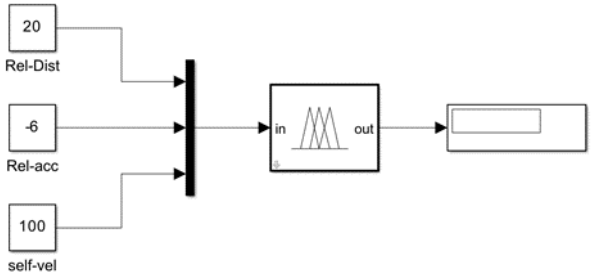


Figure 9: Simulink model used for testing

To assess the controller under different conditions, five test cases representing different driving situations were used. Table 3 presents the results.

Table 3: Simulation results in MATLAB Simulink

No.	RD (m)	RV (m/s)	OS (km/h)	Expected TBC	ANFIS TBC
1	20	−6	100	≈ −1.0	−0.8378
2	50	0	80	−0.2 to 0	−0.0214
3	90	5	120	0.8 to 1.0	0.7000
4	35	−3	60	−0.6 to −0.3	−0.4990
5	110	−1	40	0.2 to 0.5	0.5300

RD = relative distance; RV = relative velocity; OS = own vehicle speed; TBC = throttle–brake command.

Table 3 shows that the trained ANFIS model generates TBC values that closely approximate the expected values in most test cases. In the first case, with a relative distance of 20 m, a relative velocity of −6 m/s (the host vehicle is rapidly approaching the lead vehicle) and a speed of 100 km/h, the expected TBC is approximately −1.0 (strong deceleration).

© 2026 ICJES All Rights Reserved

International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

ANFIS produced −0.8378, a slightly less intense braking response than the ideal target, although it still lies within the range of significant braking.

In the second case, with a relative distance of 50 m, a relative velocity of 0 m/s and a speed of 80 km/h, the expected TBC lies between −0.2 and 0.0 (slight deceleration or speed maintenance). The ANFIS output of −0.0214 falls within this range, showing the model can maintain speed under steady following conditions.

In the third case, with a relative distance of 90 m, a relative velocity of 5 m/s (the lead vehicle is pulling away) and a speed of 120 km/h, the expected TBC is between 0.8 and 1.0 (acceleration). ANFIS produced 0.7, a slightly more conservative acceleration response than expected. The fourth case, with a relative distance of 35 m, a relative velocity of −3 m/s and a speed of 60 km/h, has an expected range of −0.6 to −0.3 (moderate deceleration); the ANFIS output of −0.499 falls within this range. In the fifth case, with a relative distance of 110 m, a relative velocity of −1 m/s and a speed of 40 km/h, the expected TBC is between 0.2 and 0.5 (light acceleration); ANFIS yielded 0.53, slightly above the upper limit, indicating a marginally more aggressive acceleration.

Overall, the trained ANFIS model replicates the desired ACC control logic: its outputs follow the trend and magnitude of the expected TBC values. The deviations observed in the first, third and fifth cases are relatively small (0.16, 0.10 and 0.03 from the nearest bound of the expected range) and may be acceptable depending on system tolerances and safety requirements. Notably, the third and fifth cases use relative distances (90 m and 110 m) beyond the 0–70 m universe of discourse defined in Table 1, so part of these deviations may reflect extrapolation outside the training domain. Other contributing factors include the shape of the membership functions, the number of fuzzy rules and the quality and quantity of the training data. Further optimization of the ANFIS parameters or the inclusion of more representative training data could improve accuracy, particularly the braking intensity in critical close-approach situations.

VI. CONCLUSION

This study shows that an ANFIS trained from a rule-based procedural dataset can generate relevant longitudinal control decisions for an ACC system, approximating the expected throttle–brake commands across various test scenarios. The model tracked both the trend and the magnitude of the ideal control responses, indicating that the hybrid learning process captured the intended control logic. The minor deviations identified can be attributed to the configuration of the membership functions, the number of fuzzy rules and the characteristics of the training data.

The combination of adaptive learning and fuzzy reasoning makes ANFIS a promising solution for handling the nonlinearities of real-world driving within intelligent automotive control systems, particularly ACC. The main limitations of this work are that the evaluation was carried out in an open-loop simulation with a small number of test cases and without a vehicle dynamics model.

VII. FUTURE SCOPE

Future work will (i) optimize the membership function parameters and the rule base, (ii) extend the universe of discourse and validate the model with a larger and more representative dataset, (iii) integrate the controller with a closed-loop longitudinal vehicle-dynamics model, including road-gradient scenarios, and compare it with conventional PID and MPC controllers, and (iv) conduct hardware-in-the-loop and real-time testing to improve the accuracy and reliability of the system.

REFERENCES

[1] A. F. Al-Saoudi, K. M. Al-Aubidy, and A. J. Al-Mahasneh, "Comparison of PID, fuzzy logic, ANFIS and model predictive controllers for cruise control system," in *Proc. 2024 21st Int. Multi-Conf. Systems, Signals & Devices (SSD)*, 2024, pp. 263–265, doi: 10.1109/SSD61670.2024.10548200.

[2] K.-H. Choi and H.-J. Chang, "Improvement of adaptive cruise control system performance on sloped roads based on adaptive neuro-fuzzy inference system," *IEEE Access*, vol. 13, pp. 60519–60531, 2025, doi: 10.1109/ACCESS.2025.3557089.

[3] Ambuj and R. Machavaram, "Intelligent path planning for autonomous ground vehicles in dynamic environments utilizing adaptive neuro-fuzzy control," *Eng. Appl. Artif. Intell.*, vol. 144, Art. no. 110119, 2025, doi: 10.1016/j.engappai.2025.110119.

[4] Z. Mehraban *et al.*, "Fuzzy adaptive cruise control with model predictive control responding to dynamic traffic conditions for automated driving," *Eng. Appl. Artif. Intell.*, vol. 136, Art. no. 109008, 2024, doi: 10.1016/j.engappai.2024.109008.

[5] I. Aliskan, "The optimization-based fuzzy logic controllers for autonomous ground vehicle path tracking," *Eng. Appl. Artif. Intell.*, vol. 151, Art. no. 110642, 2025, doi: 10.1016/j.engappai.2025.110642.

[6] A. Ghaffari, A. Khodayari, and F. Alimardani, "Predicting the future state of a vehicle in a stop&go

© 2026 ICJES All Rights Reserved

International Current Journal of Engineering and Science
Volume 5, Issue 11, pp 31-37, November 2026
ISSN (Online): 2583-7257
Article DOI: doi.org/10.47001/ICJES/2026.511005

behavior based on ANFIS models design," in *Proc. 2012 6th IEEE Int. Conf. Intelligent Systems (IS)*, 2012, pp. 368–373, doi: 10.1109/IS.2012.6335244.

[7] P. Gunasekaran *et al.*, "Adaptive cruise control system with fractional order ANFIS PD+I controller: Optimization and validation," *J. Braz. Soc. Mech. Sci. Eng.*, vol. 46, no. 4, 2024, doi: 10.1007/s40430-024-04699-z.

[8] H. Qi, "Robust responsibility-sensitive safety: Noise disturbed adaptive cruise control," *Syst. Control Lett.*, vol. 196, Art. no. 106021, 2025, doi: 10.1016/j.sysconle.2025.106021.

[9] M. H. Shojaeefard *et al.*, "Weather-aware fuzzy adaptive cruise control: Dynamic reference signal design," *Comput. Electr. Eng.*, vol. 110, Art. no. 108903, 2023, doi: 10.1016/j.compeleceng.2023.108903.

[10] H. Yu and H. Yeo, "Dynamic characteristics of commercial adaptive cruise control across driving situations: Response time, string stability, and asymmetric behavior," *Transp. Res. Part C Emerg. Technol.*, vol. 170, Art. no. 104931, 2025, doi: 10.1016/j.trc.2024.104931.

[11] Y. Kebbati, N. Ait-Oufroukh, V. Vigneron, and D. Ichalal, "Neural network and ANFIS based auto-adaptive MPC for path tracking in autonomous vehicles," in *Proc. 2021 IEEE Int. Conf. Networking, Sensing and Control (ICNSC)*, Xiamen, China, 2021, pp. 1–6, doi: 10.1109/ICNSC52481.2021.9702227.

[12] R. Palaniappan, S. K. Nataraj, Z. Ismail, and N. M. Noaman, "Fuzzy logic based obstacle avoidance algorithm for unmanned systems," in *Proc. 2023 IEEE 8th Int. Conf. Engineering Technologies and Applied Sciences (ICETAS)*, Bahrain, 2023, pp. 1–6, doi: 10.1109/ICETAS59148.2023.10346317.

[13] C.-Y. Huang *et al.*, "An ANFIS based derivations of inference rules for users' adoptions of autonomous vehicles," in *Proc. 2020 Int. Conf. Fuzzy Theory and Its Applications (iFUZZY)*, Hsinchu, Taiwan, 2020, pp. 1–5, doi: 10.1109/iFUZZY50310.2020.9297811.

[14] 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, Art. no. 2551, Apr. 2024, doi: 10.3390/s24082551.

[15] F. T. Yareshe, N. W. Madebo, C. M. Abdissa, and L. N. Lemma, "Trajectory tracking of fixed-wing UAV using ANFIS-based sliding mode controller," *IEEE Access*, vol. 13, pp. 61986–61999, 2025, doi: 10.1109/ACCESS.2025.3557472.

Citation of this Article:

Muhammad Tsaqif Wicaksono, & Wahyudi. (2026). Application of an Adaptive Neuro-Fuzzy Inference System (ANFIS) to Adaptive Cruise Control for Four-Wheeled Vehicles. *International Current Journal of Engineering and Science (ICJES)*, 5(11), 31-37. Article DOI: https://doi.org/10.47001/ICJES/2026.511005

****

© 2026 ICJES All Rights Reserved