



Article

Comparison between Genetic Algorithms of Proportional–Integral–Derivative and Linear Quadratic Regulator Controllers, and Fuzzy Logic Controllers for Cruise Control System

Ali Mahmood ^{1,2} , Karrar Y.A. Al-bayati ^{1,3,*} and Róbert Szabolcsi ⁴

¹ Doctoral School on Safety and Security Sciences, Óbuda University, 1081 Budapest, Hungary; ali.mahmood@uni-obuda.hu

² Systems and Control Engineering Department, Nineveh University, Mosul 41001, Iraq

³ Electronic and Communication Engineering Department, University of Kufa, Najaf 54001, Iraq

⁴ Kandó Kálmán Faculty of Electrical Engineering, Óbuda University, 1034 Budapest, Hungary; szabolcsi.robert@uni-obuda.hu

* Correspondence: karrar.albayati@uni-obuda.hu

Abstract: One of the most significant and widely used features currently in autonomous vehicles is the cruise control system that not only deals with constant vehicle velocities but also aims to optimize the safety and comfortability of drivers and passengers. The accuracy and precision of system responses are responsible for cruise control system efficiency via control techniques and algorithms. This study presents the dynamic cruise control system model, then investigates a genetic algorithm of the proportional–integral–derivative (PID) controller with the linear quadratic regulator (LQR) based on four fitness functions, the mean squared error (MSE), the integral squared error (ISE), the integral time squared error (ITSE) and the integral time absolute error (ITAE). Then, the response of the two controllers, PID and LQR, with the genetic algorithm was compared to the response performance of the fuzzy and fuzzy integral (Fuzzy-I) controllers. The MATLAB 2024a program simulation was employed to represent the system time response of each proposed controller. The output simulation of these controllers shows that the type of system stability response was related to the type of controller implemented. The results show that the Fuzzy-I controller outperforms the other proposed controllers according to the least J_{min} function, which represents the minimum summation of the overshoot, settling time, and steady-state error of the cruise control system. This study demonstrates the effectiveness of driving accuracy, safety, and comfortability during acceleration and deceleration due to the smoothness and stability of the Fuzzy-I controller with a settling time of 5.232 s and when converging the steady-state error to zero.



Citation: Mahmood, A.; Al-bayati, K.Y.A.; Szabolcsi, R. Comparison between Genetic Algorithms of Proportional–Integral–Derivative and Linear Quadratic Regulator Controllers, and Fuzzy Logic Controllers for Cruise Control System. *World Electr. Veh. J.* **2024**, *15*, 351.

<https://doi.org/10.3390/wevj15080351>

Academic Editors: Yujie Wang
and Michael Fowler

Received: 24 June 2024

Revised: 21 July 2024

Accepted: 2 August 2024

Published: 5 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, autonomous vehicles have made great strides in progress and development in the field of transportation at the individual and societal levels, especially in terms of comfort, safety, energy consumption, and maneuvering accuracy [1–5]. One of the most significant contributing factors that played a prominent role in the stages of development of autonomous vehicles is cruise control systems, which not only originally contributed to the stability of the vehicle speed but were also incorporated into many features in the gradual evaluation of control in autonomous vehicles, such as modifying speed automatically to maintain a safe distance between vehicles to avoid collision [6–9]. To achieve the optimal and most accurate performance of vehicles, the cruise control system uses various control techniques and methods such as the proportional–integral–derivative (PID) control, fuzzy

logic control (FLC), the linear quadratic regulator (LQR), and model predictive control (MPC) [10–15].

Many studies related to the cruise control system have been published and utilized various control methods. Pradhan et al. proposed a PID controller to improve the stability of the cruise control system based on the Ant Lion algorithm to optimize the PID parameters [16]. They showed better performance compared to the other proposed control approaches. Similarly, Wu et al. proposed a PID controller with the chaotic ant swarm algorithm to enhance the time response for an automobile cruise system [17]. They demonstrated increased driving comfortability and safety as well as decreased fuel consumption. Singhal et al., in [18], applied the GNA (global neighborhood algorithm) to specify PID controller parameters to mitigate the system response error that also led to increased driving comfortability. Other researchers also adopted a PID controller by applying different scenarios to increase control parameter optimization such as particle swarm optimization, the genetic algorithm, Fractional Order PID (FOPID), and the model predictive controller (MPC) [19–22].

From another point of view, several publication studies focused on using a fuzzy logic controller for an autonomous vehicle control system. Abdurohman et al. utilized a fuzzy control system to select the optimal control parameters of a PID controller [23]. Their proposed method showed the safety, comfortability, and stability of the cruise control system by adjusting the accelerator pedal under different drive conditions. On the other hand, Mao et al. proposed a fuzzy model predictive control algorithm (Fuzzy-MPC) to achieve better performance than the traditional algorithm under different road conditions by considering the safety, comfortability, and stability of the cruise control system [24]. Likewise, Shojaeeefard et al. presented fuzzy logic control to avoid the collision between vehicles by modifying the vehicle speed and acceleration dynamically to meet the minimum requirement distance based on changing weather conditions [25]. From a medical standpoint, Yang et al. introduced fuzzy control to analyze driver comfortability according to the driver's heat rate (HRV) [26]. In order to decrease driver discomfort physiologically related to the acceleration of braking, they extracted electrocardiography (ECG) sensor data, and stabilized the cruise control system by applying fuzzy control.

In addition to the controller methods mentioned above, some researchers have used the linear quadratic regulator (LQR) for a more complex cruise control system. Varma et al. designed a LQR controller for autonomous vehicle path tracking in consideration to roll and yaw stability control, where most car accidents occur [27]. Li et al. applied a variable weight of LQR to minimize the fluctuation of longitudinal acceleration that keeps cars in line and improves driving safety comparing to constant weight [28]. From another point of view, Zhang et al. adopted the Fuzzy-LQR method to design a cruise control system to modify the accelerating speed during lane change to avoid vehicle collision and ensure driving comfort and safety [29]. In this paper, three different controllers for the cruise control system were presented with genetic algorithm (GA) optimization to obtain the appropriate control parameters. A performance analysis comparison of the PID, FLC, and LQR controllers was conducted and discussed in terms of using various fitness functions, which make the system more accurate and stable using the MATLAB simulation program during acceleration and deceleration (mean squared error (MSE), integral squared error (ISE), integral time squared error (ITSE), and integral time absolute error (ITAE)).

This paper is divided in to five sections as follows: the first section includes the background and a literature review in the introduction. In the second section, the linearization of the cruise control system model and mathematical expression are derived. In section three, the three proposed controllers are designed using a genetic algorithm and fitness functions. Then, the simulation and results section is presented using the MATLAB program to show the best performance among controllers on applying various fitness functions. Finally, a performance system analysis of the proposed controllers will be summarized in the conclusion.

2. Model of the Cruise Control System

The main concern of using the cruise control system is the stability of vehicle speed according to driver input instruction. Rather than using the accelerator or brake pedal to maintain a constant vehicle speed, the cruise control system utilizes the feedback signals from the speed sensor to converge or match the speed reference (v_{ref}) set by the driver due to modifications to the gate angle of engine throttle actuator (u), and then specifies the appropriate driving force of the engine (F_d). The block diagram of the cruise control system is represented by Figure 1 [30].

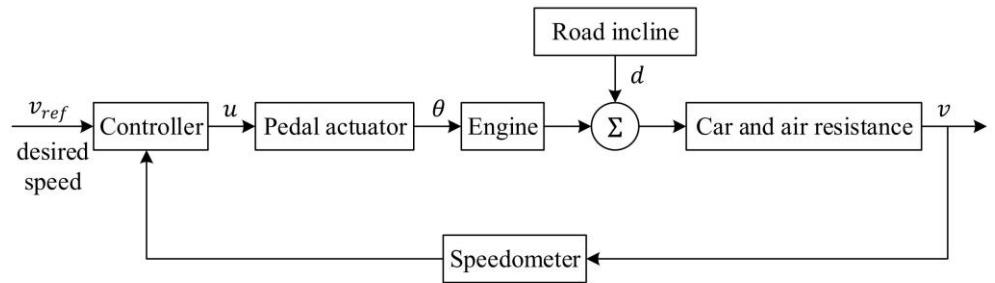


Figure 1. Block diagram of cruise control system.

By taking into consideration the aerodynamic drag force (F_a), the climbing resistance force (F_g) and, according to Newton's second law, the nonlinear longitudinal dynamic of the system can be represented by the following equation [31]:

$$F_d = M \frac{d}{dt} v + F_a + F_g \quad (1)$$

where M represents the overall passengers and vehicle mass, while $M \frac{d}{dt} v$ is the inertia force. All forces in Equation (1) can be found in the following expression [32]:

$$F_a = C_a (v + v_w)^2 \operatorname{sgn}(v + v_w) \quad (2)$$

$$F_g = Mg \sin \theta \quad (3)$$

$$F_d = \frac{C_1 e^{-\tau s}}{T_s + 1} \quad (F_{dmin} \leq F_d \leq F_{dmax}) \quad (4)$$

where C_a is the coefficient of aerodynamic drag, v and v_w are the actual vehicle and wind speed, respectively, g is the gravity acceleration, and θ is the incline angle of road. Equation (4) has the maximum and minimum limitations of brake and acceleration forces, respectively, where C_1 is the value added to the constant of the actuator force saturation, $e^{-\tau s}$ is the delay factor, τ is the reaction time of driver, and T_s is a time constant. Figure 2 shows the above three force equations in a block diagram of vehicle dynamics [31].

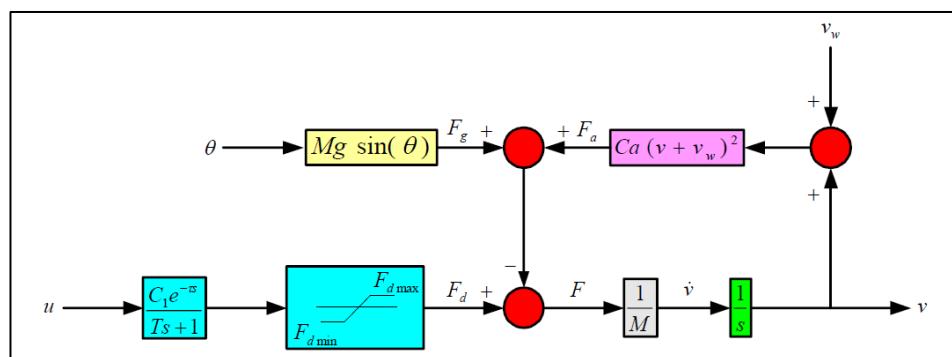


Figure 2. Block diagram of vehicle dynamics.

To simplify the control equations of the system shown in Figure 2, a value of zero will be set to all the initial condition due to disturbances that eliminated the wind and grading effects. Due to the simplification assumption, Equation (1) should be rearranged into the following equation:

$$\dot{v} = \frac{1}{M} (F_d - C_a v^2) \quad (5)$$

By applying inverse Laplace for Equation (4), Equation (1) is rewritten into the following equation:

$$F_d = \frac{1}{T} (C_1 u(t - \tau) - F_d) \quad (6)$$

To overcome the nonlinearity of Equation (5), Osman et al. differentiated both sides of the equation [31], whereas Izci et al. used Taylor's series expansion [30]. Thus, the transfer function of vehicle speed output related to the pedal actuator input of Equation (6) can be written as follows [30,31]:

$$\frac{\Delta V(s)}{\Delta U(s)} = \frac{\frac{C_1 e^{-\tau s}}{MT}}{\left(s + \frac{2C_a v}{M}\right)\left(s + \frac{1}{T}\right)} \quad (7)$$

By substituting the parameters and specification values in Table 1 [30,31], the linearized transfer function of the derived mathematical expression of the system can be found in Equation (8), while Equations (9) and (10) represent the state space of the system used by the proposed controller, as outlined in the next section.

$$G_p(s) = \frac{\Delta V(s)}{\Delta U(s)} = \frac{2.4767}{(s + 0.0476)(s + 1)(s + 5)} \quad (8)$$

$$\begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{v}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6.0476 & -5.2856 & -0.238 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 2.4767 \end{bmatrix} [\Delta u(t)] \quad (9)$$

$$y(t) = [1 \ 0 \ 0] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \quad (10)$$

Table 1. Parameter/specification.

| Parameter/Specification | Value |
|-------------------------|---------------------------|
| C_1 | 743 |
| C_a | 1.19 N (m/s) ² |
| M | 1599 kg |
| τ | 0.2 s |
| T | 1 s |
| F_{dmax} | 3500 N |
| F_{dmin} | -3500 N |
| g | 9.8 m/s ² |
| v | 30 Km/h |

3. Controller Design

In this section, the analysis and design of the presented controllers will be presented and discussed. This section is divided into two parts to ensure a comprehensive understanding of the methodologies employed. The first part will focus on the design and implementation of the proportional–integral–derivative (PID) and linear quadratic regulator (LQR) controllers. These controllers will be optimized using the genetic algorithm (GA) optimization method. This part will delve into the theoretical foundations of the PID and LQR controllers, followed by a detailed explanation of how the genetic algorithm is applied

to enhance their performance and efficiency. The second part will focus on the design and development of the fuzzy logic controller and the fuzzy integral controller. This section will cover the principles of fuzzy logic, explaining its application in control systems and how it can be integrated with traditional control methods to improve system responsiveness and robustness. The fuzzy integral controller will be introduced as an advanced technique that combines fuzzy logic with integral control to achieve superior control outcomes.

3.1. PID and LQR Controllers with the GA Optimization Method

The genetic algorithm (GA) method is considered one of the most prominent and efficacious optimization techniques utilized across numerous domains, including control system design. It has been extensively employed to ascertain the optimal values for the parameters of controllers, serving as a critical step in the controller design process. This method belongs to the category of evolutionary algorithms, which are inspired by the principles of natural selection and genetics. By emulating the process of natural selection, genetic algorithms iteratively evolve solutions to optimization problems, ultimately converging on the most optimal solutions.

In the context of controller design, the genetic algorithm plays an integral role in augmenting the performance and efficiency of control systems. It commences with the population of potential solutions, each represented as a set of chromosomes or individuals. These individuals are evaluated based on a predefined fitness function, which assesses the extent to which each solution satisfies the desired control objectives. The algorithm then applies genetic operators, such as selection, crossover, and mutation, to generate a new population of potential solutions. Through selection, the most proficient individuals are chosen to transmit their attributes to subsequent generations, analogous to the concept of survival of the fittest. Crossover enables the amalgamation of characteristics from disparate individuals, facilitating the creation of potentially superior offspring. Mutation introduces stochastic variations in the offspring's chromosomes, ensuring genetic diversity and aiding the algorithm in evading local optima. By iteratively applying these operators over multiple generations, the genetic algorithm incrementally refines the solutions, steering them toward optimal parameter values for the controllers. The outcome is a set of optimized controller parameters that enhance the overall performance, stability, and robustness of the control system.

To use GA, firstly, a population size that represents the potential solutions of the optimal problem should be selected; this population will be represented as chromosome structures, which are typically a binary string. This chromosome structure includes all the controller's elements that must be optimized. In this paper, it will be used for the PID and LQR controllers. For the PID controller, it is required to optimize the values of the K_p , K_i , and K_d , while for the LQR controller, it is required to optimize the Q and R matrices.

After forming the chromosome, the next step is the use of the fitness function in order to obtain the performance of each one. The next step is the crossover; this is between a pair of individuals, so the information between the two chromosomes will be exchanged to create totally new chromosomes. After crossover, mutation will occur randomly to only some of the individuals in order to increase the diversity in the population.

Figure 3 shows the flowchart of the GA.

In the domain of control system design utilizing genetic algorithms, the selection of an appropriate fitness function is of the utmost importance. The fitness function serves as a quantitative metric that evaluates the performance of a given set of controller parameters concerning the specified objectives. The choice of fitness function influences the optimization process's outcome, as different functions can produce different results. Thus, identifying the most suitable fitness function for the model in use is crucial for achieving optimal performance where each fitness function has its strengths and weaknesses, and their applicability is contingent upon the specific control objectives and evaluation criteria.

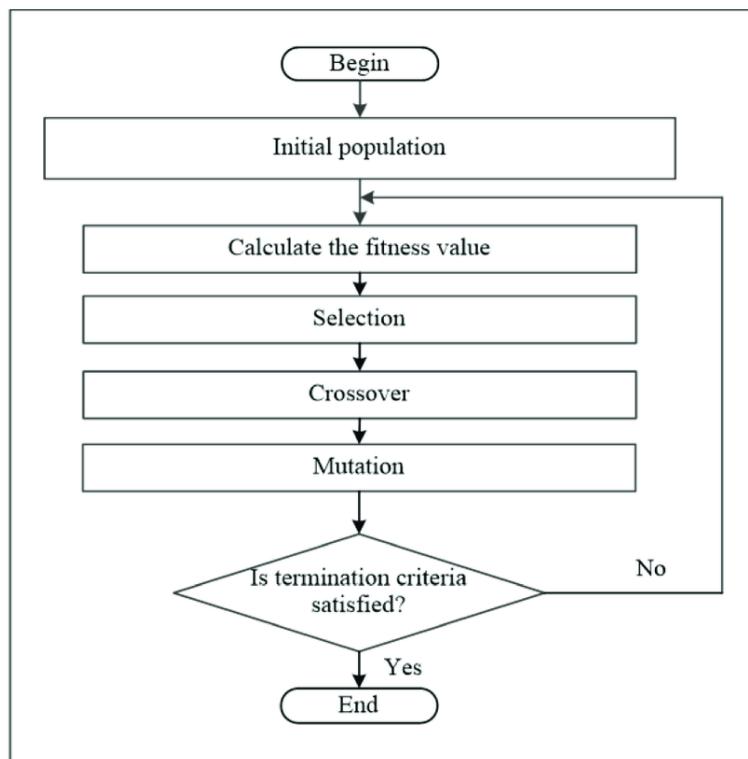


Figure 3. GA flowchart.

Commonly utilized fitness functions include mean squared error (MSE), integral squared error (ISE), integral time squared error (ITSE), and integral time absolute error (ITAE). Each of these functions is designed to address specific control evaluation scenarios by emphasizing performance aspects such as the transient response, steady-state response, or both.

The mean squared error (MSE) function is frequently employed due to its straightforward nature and efficacy in minimizing the average of the squared deviations between the desired and actual system outputs. This function is particularly beneficial when the primary objective is to reduce the overall deviation in the system's response from the desired trajectory. Integral squared error (ISE), on the other hand, integrates the square of the error over time, thereby placing emphasis on mitigating substantial deviations throughout the entire response period. This characteristic renders ISE suitable for systems minimizing both transient and steady-state errors. Integral time squared error (ITSE) further refines this approach by weighting errors that occur later in the response, making it advantageous for applications where the transient response is critically important and later-stage precision is essential. At the same time, the integral time absolute error (ITAE) emphasizes the minimization of time-weighted absolute errors, prioritizing rapid error correction and minimal overshoot. This makes ITAE particularly well-suited for scenarios where a balance between transient and steady-state performance is desirable.

The equations of these fitness functions are represented in Equations (11)–(14).

$$MSE = \frac{1}{N} \sum_{i=1}^N (e_i)^2 \quad (11)$$

$$ISE = \int_0^T (e(t))^2 dt \quad (12)$$

$$ITSE = \int_0^T t \cdot (e(t))^2 dt \quad (13)$$

$$ITAE = \int_0^T t \cdot |e(t)| dt \quad (14)$$

To use the GA with the PID controller, it is crucial to understand the parts of the PID controller and their effect on the response. It consists of three parts (proportional, integral, and derivative), and the sum of their effect forms the controller output. When increasing the proportional gain K_p , the response of the system will be faster, but it will increase the overshoot of the response while increasing the derivative gain K_d and reduce the overshoot and any sudden changes in the response. Finally, the use of the integral part will eliminate the steady-state error, even though adding it will have a negative effect on the transient response.

Equation (15) shows the representation of the control single of the PID controller, while Figure 4 shows the block diagram of the GA with the PID controller.

$$u(t) = K_p \cdot e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (15)$$

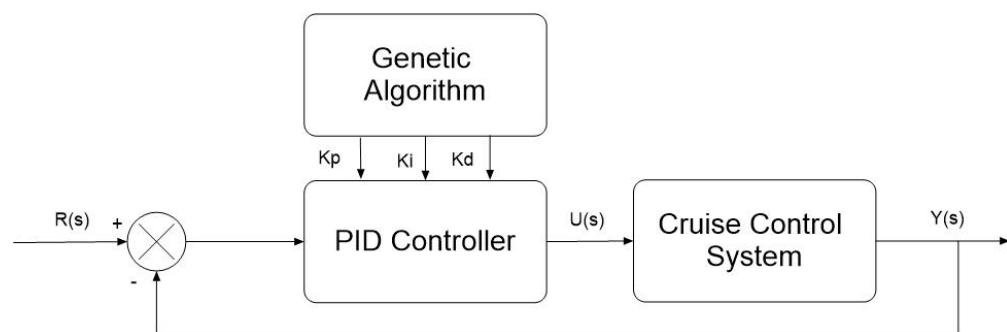


Figure 4. PID with GA.

For the LQR controller, the optimization of the weighting matrices (Q and R) is used to minimize the quadratic cost function where they play a crucial role in detecting the role of each state and control input in the controller, as the Q matrix is the state-cost weight matrix, while the R matrix is the control weight matrix.

Figure 5 shows the block diagram of the LQR controller which is a state feedback control technique that uses the quadratic cost function to determine the optimal control input to minimize the performance index of the system. The role of the feedback loop is to measure the state variables of the system to calculate the optimal control inputs.

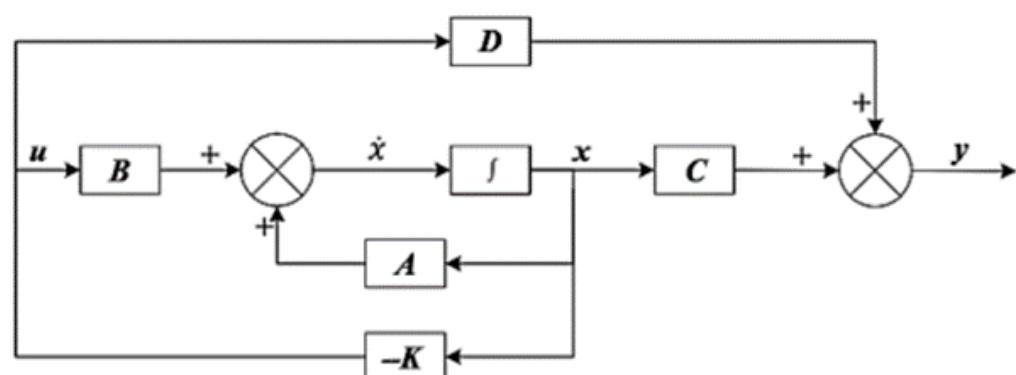


Figure 5. Block diagram of the LQR controller.

As the LQR controller used the state space representation shown in Equation (16), the quadratic cost function to be minimized is given by Equation (14).

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (16)$$

where $x(t)$ is the state vector, $u(t)$ is the control input vector, A is the state matrix, and B is the input matrix.

$$J = \int_0^\infty (x^T(t)Qx(t) + u^T(t)Ru(t))dt \quad (17)$$

where Q is the state cost matrix and R is the control cost matrix. By using the optimal values of the Q and R matrices, the controller gain K can be computed using Equations (18–20).

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (18)$$

where P is the solution to the continuous-time algebraic Riccati equation.

$$u(t) = -R^{-1}B^T Px(t) \quad (19)$$

$$K = R^{-1}B^T P \quad (20)$$

Finally, the block diagram represents the LQR controller with the GA is shown in Figure 6.

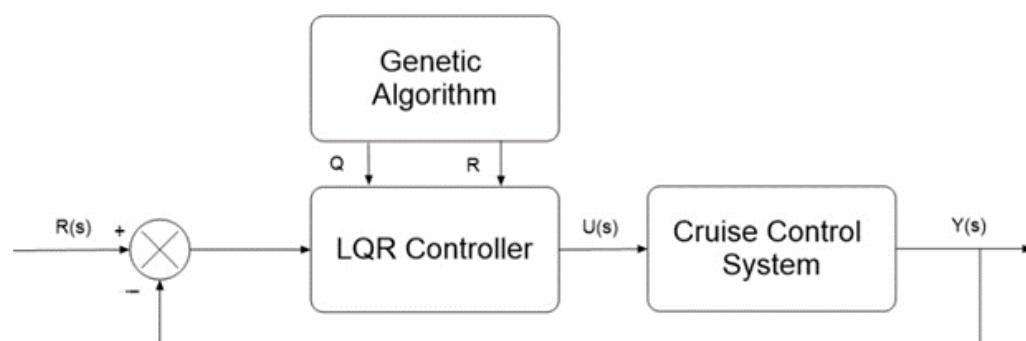


Figure 6. LQR controller with GA.

3.2. Fuzzy Logic Controller

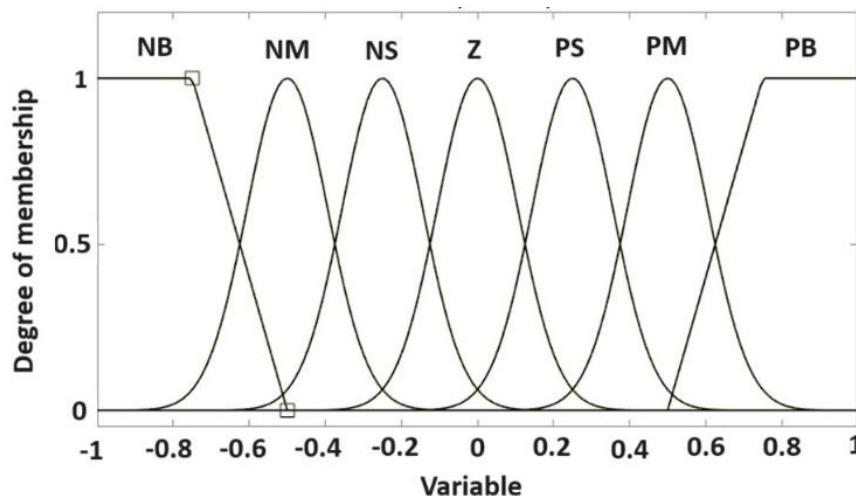
The fuzzy logic controller (FLC) is a knowledge-based controller that uses the fuzzy logic principle and utilizes linguistic variables and fuzzy rules to represent and control complex, nonlinear systems. For this reason, it has been widely applied, especially with systems that have uncertainty and nonlinearity. One drawback of the FLC is that its response still has steady-state errors, even though it diminishes the overshoot. To overcome this problem, an integral part (I) will be added to design a Fuzzy-I controller. The FLC has two inputs, proportional error PE and the change in error CE, and one output, which is the output of the controller U . Table 2 list the linguistic rules that will be used for the fuzzy and Fuzzy-I controllers where both inputs and the output have the following seven membership functions:

- Negative big (NB);
- Negative medium (NM);
- Negative small (NS);
- Zero (Z);
- Positive small (PS);
- Positive medium (PM);
- Positive big (PB).

Table 2. Fuzzy rules.

| PE\CE | NB | NM | NS | ZR | PS | PM | PB |
|-------|----|----|----|----|----|----|----|
| NB | NB | NB | NB | NB | NS | Z | PS |
| NM | NB | NB | NB | NM | NS | Z | PS |
| NS | NB | NB | NM | NS | Z | PS | PM |
| Z | NB | NM | NS | Z | PS | PM | PB |
| PS | NM | NS | Z | PS | PM | PB | PB |
| PM | NS | Z | PS | PM | PB | PB | PB |
| PB | Z | PS | PM | PB | PB | PB | PB |

The method that is used to determine the fuzzy characteristics of the output is called the fuzzy inference, which takes into account every rule in the rule base for the fuzzy inputs. The next step is finding the crisp control output using the defuzzification process. Figure 7 shows the membership functions of the inputs and the output. It is important to use trapezoidal functions for both ends of the inputs and the output to cover all extreme values while using gaussian functions for a smooth change in membership.

**Figure 7.** Input and output membership functions.

4. Simulation and Results

In this section, firstly, the PID and LQR controllers will be run individually for the four fitness functions to choose the controller with the best fitness function. Then, MATLAB Simulink will be used to compare the results for all controllers. After finding the PID controller gain values using the GA optimization method for the used fitness functions shown in Table 3, the first comparison is made between the PID controllers with different four fitness functions, as shown in Figure 8.

Table 3. PID controller parameters.

| Controller | Kp | Ki | Kd |
|------------|-----|-------|----|
| PID-MSE | 2.8 | 0.437 | 2 |
| PID-ISE | 2.8 | 1.247 | 2 |
| PID-ITSE | 2.8 | 0.4 | 2 |
| PID-ITAE | 2.8 | 0.493 | 2 |

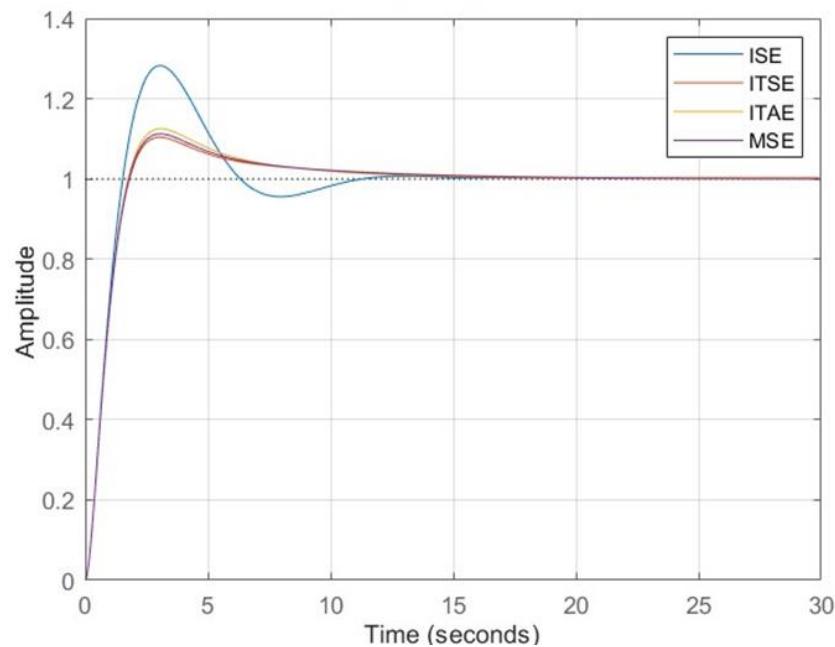


Figure 8. Time response of the PID controller for all fitness functions.

At the same time, the same test is conducted to the LQR; the values of the K matrix are shown in Table 4, and the step response is shown in Figure 9.

Table 4. LQR controller parameters.

| Controller | K11 | K22 | K33 |
|------------|--------|--------|--------|
| LQR-ISE | 0.0204 | 1.5362 | 1.4038 |
| LQR-ITSE | 0.1968 | 4.4657 | 3.5938 |
| LQR-ITAE | 0.0898 | 1.5708 | 1.4131 |
| LQR-MSE | 7.3238 | 6.9351 | 3.8548 |

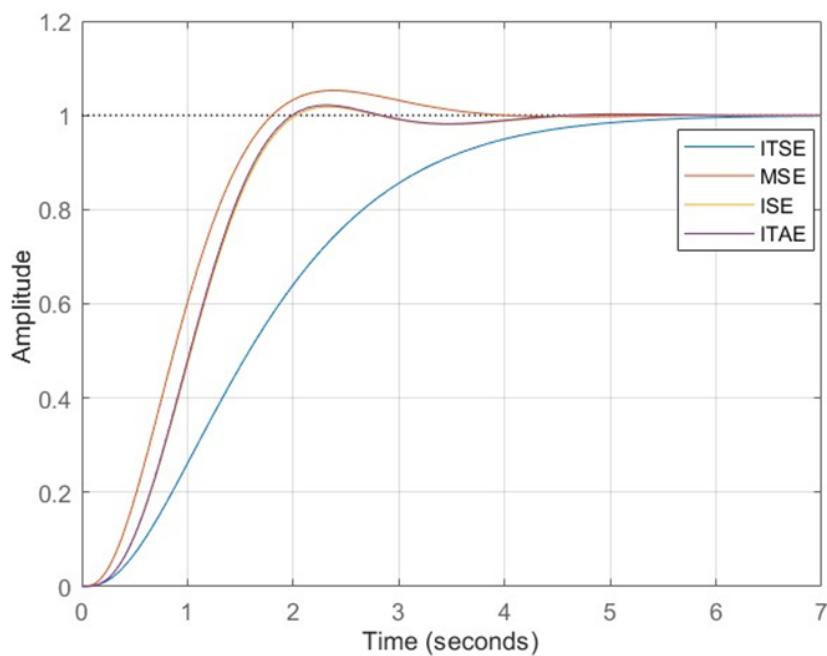


Figure 9. Time response of the LQR controller for all fitness functions.

As the response specifications for both controllers have very similar responses, the J_{min} equation was used to find the controller with the best fitness function by comparing the transient and steady-state response specifications, which are the overshoot percent (%OS), rise time (Tr), and settling time with a threshold of 2% (Ts), as shown in Table 5.

$$J_{min} = \sum \%OS + Tr + Ts \quad (21)$$

Table 5. Time response specifications of PID and LQR controllers for all fitness functions.

| Controller | %MP | Tr (s) | Ts (s) | Ess | J _{min} |
|------------|--------|--------|--------|-----|------------------|
| PID-MSE | 10.1 | 1.483 | 11.118 | 0 | 22.701 |
| PID-ISE | 27.564 | 1.072 | 10.859 | 0 | 39.495 |
| PID-ITSE | 9.2 | 1.471 | 11.407 | 0 | 22.078 |
| PID-ITAE | 11.3 | 1.502 | 10.759 | 0 | 23.561 |
| LQR-MSE | 3.646 | 1.232 | 4.316 | 0 | 9.194 |
| LQR-ISE | 0.496 | 2.033 | 5.498 | 0 | 8.027 |
| LQR-ITSE | 0.503 | 4.146 | 8.748 | 0 | 13.397 |
| LQR-ITAE | 0.495 | 1.961 | 5.419 | 0 | 7.875 |

As the PID-ITSE and LQR-ITAE have the minimum J_{min} value, they will be used in the MATLAB Simulink to be compared with the fuzzy and Fuzzy-I controllers. The block diagram for all controllers with the model of the cruise control system is shown in Figure 10.

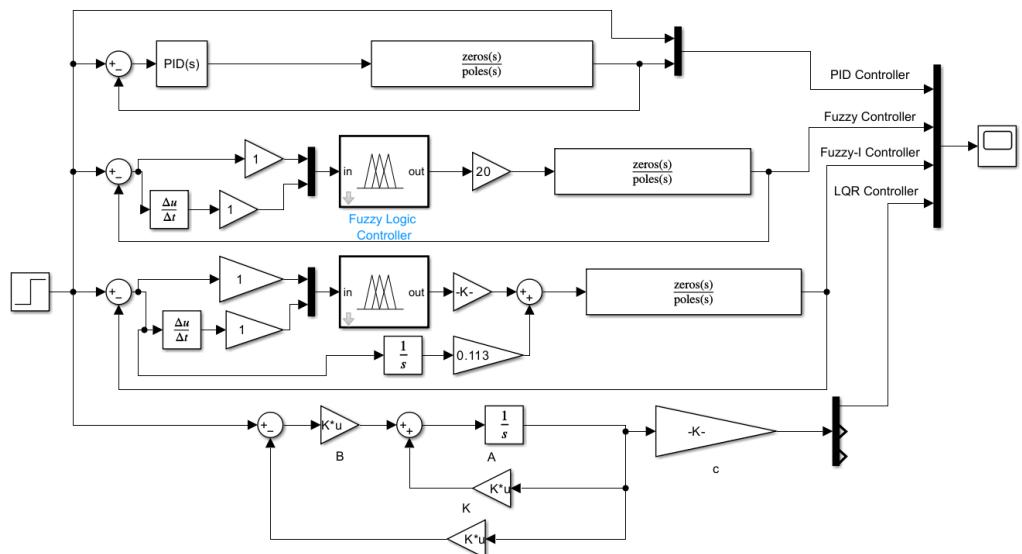


Figure 10. Block diagram of the cruise control system with controllers.

The time response of the PID-ITSE, LQR-ITAE, fuzzy, and Fuzzy-I controllers are shown in Figure 11.

Figure 12 shows the time response in the case of the acceleration of speed from 60 km/h (16.667 m/s) to 90 km/h (25 m/s) and the deceleration of speed back to 60 km/h.

To compare the time response specifications of the designed controllers, their values are shown in Table 6. It is important to use the J_{min} function to compare the responses.

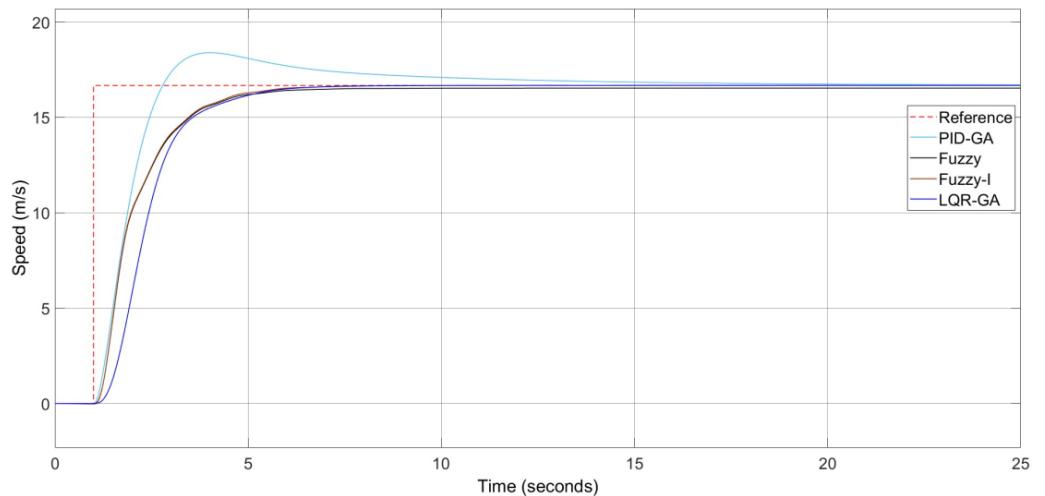


Figure 11. Step response of PID, LQR, fuzzy, and Fuzzy-I controllers.

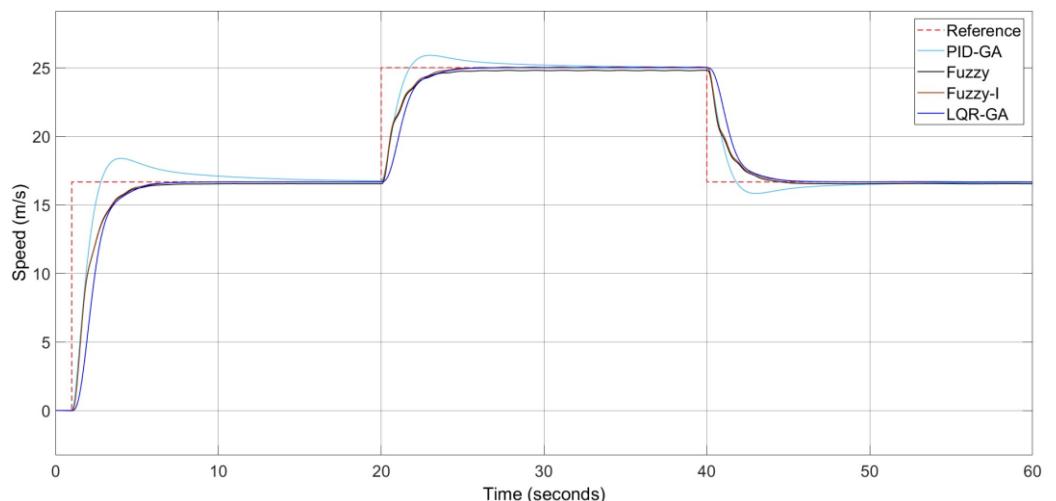


Figure 12. Step response of PID, LQR, fuzzy, and Fuzzy-I controllers with speed acceleration and deceleration.

Table 6. Controller time response specifications.

| Controller | %MP | Tr (s) | Ts (s) | Ess | Jmin |
|------------|-------|--------|--------|-------|--------|
| PID-GA | 9.2 | 1.471 | 11.407 | 0 | 22.078 |
| LQR-GA | 0.495 | 2.018 | 5.419 | 0 | 7.932 |
| Fuzzy | 0.476 | 2.131 | 5.926 | 0.011 | 8.544 |
| Fuzzy-I | 0.472 | 2.145 | 5.232 | 0 | 7.849 |

As shown in Table 6, the Fuzzy-I controller has the smallest J_{min} value, and it can be concluded that it is the best controller for the cruise control system. Moreover, the fuzzy controller has more advantages over the LQR controller, like its ability to handle nonlinearity, robustness, ease of implementation, flexibility, adaptability, and no requirements for precise modeling.

5. Conclusions

Based on our extensive research, we concluded that the Fuzzy-I controller has the best time response compared to the PID-GA, LQR, fuzzy, and Fuzzy-I controllers for cruise control systems. Before reaching this conclusion, firstly, the performance of the PID-GA and LQR-GA were tested separately using four different fitness functions, which were the MSE,

ISE, ITSE, and ITAE. Then, the best controller with the best fitness functions was chosen for a later comparison with the fuzzy and Fuzzy-I controllers, where the best controller for the PID controller was with the ITSE fitness function, while for the LQR controller, it was ITAE. Secondly, the responses of the PID-ITSE and LQR-ITAE were compared with the fuzzy and Fuzzy-I controllers. As it was difficult to determine which controller had the best response, the J_{min} function was used to find it; the Fuzzy-I has the smallest J_{min} , so it is the best controller for the cruise control system.

Author Contributions: The authors have the following contribution. A.M.: controller design, simulation and results, conclusion, writing—original draft preparation, software. K.Y.A.A.-b.: writing—original draft preparation, Abstract, Literature review, Introduction, Methodology, References. R.S.: supervision, investigation, reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Óbuda University, 1034 Budapest, Bécsi út 96/b, Hungary.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

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

References

- Mohajer, N.; Nahavandi, S.; Abdi, H.; Najdovski, Z. Enhancing Passenger Comfort in Autonomous Vehicles through Vehicle Handling Analysis and Optimization. *IEEE Intell. Transp. Syst. Mag.* **2021**, *13*, 156–173. [[CrossRef](#)]
- Nie, Z.; Farzaneh, H. Energy-efficient lane-change motion planning for personalized autonomous driving. *Appl. Energy* **2023**, *338*, 120926. [[CrossRef](#)]
- Muzahid, A.J.M.; Kamarulzaman, S.F.; Rahman, M.A.; Alenezi, A.H. Deep Reinforcement Learning-Based Driving Strategy for Avoidance of Chain Collisions and Its Safety Efficiency Analysis in Autonomous Vehicles. *IEEE Access* **2022**, *10*, 43303–43319. [[CrossRef](#)]
- Huang, C.; Huang, H.; Hang, P.; Gao, H.; Wu, J.; Huang, Z.; Lv, C. Personalized Trajectory Planning and Control of Lane-Change Maneuvers for Autonomous Driving. *IEEE Trans. Veh. Technol.* **2021**, *70*, 5511–5523. [[CrossRef](#)]
- Zsombok, I.; Zöldy, M. Modelling, Simulation and Validation of Hybrid Vehicle Fuel Consumption. *Acta Polytech. Hung.* **2023**, *20*, 61–74. [[CrossRef](#)]
- He, D.; He, W.; Song, X. Efficient predictive cruise control of autonomous vehicles with improving ride comfort and safety. *Meas. Control* **2020**, *53*, 18–28. [[CrossRef](#)]
- Liu, Z.; Yuan, Q.; Nie, G.; Tian, Y. A Multi-Objective Model Predictive Control for Vehicle Adaptive Cruise Control System Based on a New Safe Distance Model. *Int. J. Automot. Technol.* **2021**, *22*, 475–487. [[CrossRef](#)]
- Emirler, M.T.; Güvenç, L.; Güvenç, B.A. Design and Evaluation of Robust Cooperative Adaptive Cruise Control Systems in Parameter Space. *Int. J. Automot. Technol.* **2018**, *19*, 359–367. [[CrossRef](#)]
- Németh, B. Providing Guaranteed Performances for an Enhanced Cruise Control Using Robust LPV Method. *Acta Polytech. Hung.* **2023**, *20*, 133–152. [[CrossRef](#)]
- Li, S.E.; Guo, Q.; Xu, S.; Duan, J.; Lii, S.; Li, C.; Su, K. Performance Enhanced Predictive Control for Adaptive Cruise Control System Considering Road Elevation Information. *IEEE Trans. Intell. Veh.* **2017**, *2*, 150–160. [[CrossRef](#)]
- Turan, A. PID controller design with a new method based on proportional gain for cruise control system. *J. Radiat. Res. Appl. Sci.* **2024**, *17*, 100810. [[CrossRef](#)]
- Mahmood, A.; Almaged, M.; Alnema, Y.H.S.; Noaman, M.N. Adaptive Cruise Control of A Simscape Driveline Vehicle Model Using Fuzzy Logic Controller. *J. Eur. Syst. Autom.* **2023**, *56*, 743–749. [[CrossRef](#)]
- Mustafar, M.Z.C.; Bakar, S.A.A.; Zainuddin, Z.; Mansor, M.S.F.; Ismail, N.H.F.; Rani, M.R. Optimal Design of an Adaptive Cruise Control System for Driving Comfort and Fuel Economy. *J. Soc. Automot. Eng. Malays.* **2021**, *4*, 279–287. [[CrossRef](#)]
- Dawood, Y.S.; Mahmood, A.K.; Ibrahim, M.A. Comparison of PID, GA and fuzzy logic controllers for cruise control system. *Int. J. Comput. Digit. Syst.* **2018**, *7*, 311–319.
- Kovacs, A.; Vajk, I. Tuning Parameter-free Model Predictive Control with Nonlinear Internal Model Control Structure for Vehicle Lateral Control. *Acta Polytech. Hung.* **2023**, *20*, 185–204. [[CrossRef](#)]
- Pradhan, R.; Majhi, S.K.; Pradhan, J.K.; Pati, B.B. Performance evaluation of PID controller for an automobile cruise control system using ant lion optimizer. *Eng. J.* **2017**, *21*, 347–361. [[CrossRef](#)]
- Wu, X.; Qin, G.; Yu, H.; Gao, S.; Liu, L.; Xue, Y. Using improved chaotic ant swarm to tune PID controller on cooperative adaptive cruise control. *Optik* **2016**, *127*, 3445–3450. [[CrossRef](#)]
- Singhal, A.; Roy, R.; Mittal, D.; Dahiya, P. Optimal Tuning using Global Neighborhood Algorithm for Cruise Control System. In Proceedings of the 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 19–20 March 2021; pp. 336–339. [[CrossRef](#)]

19. Abdulnabi, A. PID Controller Design for Cruise Control System using Particle Swarm Optimization. *Iraqi J. Comput. Inform.* **2017**, *43*, 30–35. [[CrossRef](#)]
20. Rout, M.K.; Sain, D.; Swain, S.K.; Mishra, S.K. PID controller design for cruise control system using genetic algorithm. In Proceedings of the 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 3–5 March 2016; pp. 4170–4174. [[CrossRef](#)]
21. Pradhan, R.; Pati, B.B. Optimal FOPI Controller for an Automobile Cruise Control System. In Proceedings of the 2018 International Conference on Recent Innovations in Electrical, Electronics & Communication Engineering (ICRIECE), Bhubaneswar, India, 27–28 July 2018; pp. 1436–1440. [[CrossRef](#)]
22. Gulzar, M.M.; Sharif, B.; Sibtain, D.; Akbar, L.; Akhtar, A. Modelling and controller design of automotive cruise control system using hybrid model predictive controller. In Proceedings of the IEEE International Conference on Emerging Technologies (ICET) 2019, Peshawar, Pakistan, 2–3 December 2019. [[CrossRef](#)]
23. Abdurohman, A.; Kang, H.X.; Hidayat, T. Vehicle ACC Control Based on Fuzzy PID. *Int. J. Eng. Contin.* **2022**, *1*, 36–55. [[CrossRef](#)]
24. Mao, J.; Yang, L.; Hu, Y.; Liu, K.; Du, J. Research on vehicle adaptive cruise control method based on fuzzy model predictive control. *Machines* **2021**, *9*, 160. [[CrossRef](#)]
25. Shojaeebard, M.H.; Mollajafari, M.; Ebrahimi-Nejad, S.; Tayebi, S. Weather-aware fuzzy adaptive cruise control: Dynamic reference signal design. *Comput. Electr. Eng.* **2023**, *110*, 108903. [[CrossRef](#)]
26. Yang, Z.; Fu, W.H.; Zhang, Z.; Zhang, J.; Wang, L. Comfort optimization of adaptive cruise control based on heart rate variability and fuzzy control. *J. Phys. Conf. Ser.* **2021**, *2010*, 012176. [[CrossRef](#)]
27. Varma, B.; Swamy, N.; Mukherjee, S. Trajectory Tracking of Autonomous Vehicles using Different Control Techniques(PID vs LQR vs MPC). In Proceedings of the International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE 2020), Bengaluru, India, 9–10 October 2020; pp. 84–89. [[CrossRef](#)]
28. Li, X.; Xie, N.; Wang, J. A variable weight adaptive cruise control strategy based on lane change recognition of leading vehicle. *Automatika* **2022**, *63*, 555–571. [[CrossRef](#)]
29. Zhang, K.; Lu, Y.; Huang, X. Design of Adaptive Cruise Control Considering Mult-scene Application. In Proceedings of the 2021 China Automation Congress (CAC), Beijing, China, 22–24 October 2021; pp. 956–961. [[CrossRef](#)]
30. Izci, D.; Ekinci, S.; Kayri, M.; Eker, E. A novel improved arithmetic optimization algorithm for optimal design of PID controlled and Bode's ideal transfer function based automobile cruise control system. *Evol. Syst.* **2022**, *13*, 453–468. [[CrossRef](#)]
31. Osman, K.; Rahmat, M.F.; Ahmad, M.A. Modelling and controller design for a cruise control system. In Proceedings of the 2009 5th International Colloquium on Signal Processing & Its Applications, Kuala Lumpur, Malaysia, 6–8 March 2009; pp. 254–258. [[CrossRef](#)]
32. Frank, A.A.; Liu, S.J.; Liang, S.C. Longitudinal Control Concepts for Automated Automobiles and Trucks Operating on a Cooperative Highway. *SAE Tech. Pap.* 1989; 1308–1315. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.