# Adaptive Cruise Control for Electric Vehicles: A Performance Study Using CARLA

**Assem Meghawer[1*], Mohamed Fawzy El-Khatib[2], Mohamed I. Abu El-Sebah[3] and Yasser I. El-Shaer[1]**

[1]Mechanical Engineering Department, Arab Academy for Science and Technology and Maritime Transport (AASTMT), Smart-Village Branch, Cairo, Egypt
[2]Mechatronics and Robotics Engineering Department, Faculty of Engineering, Egyptian Russian University (ERU), Cairo 11829, Egypt
[3]Power Electronics and Energy Conversion Department, Electronics Research Institute, Cairo, 11843, Egypt

E-mail: assem.abdelhafez@student.aast.edu

**Abstract.** Adaptive cruise control (ACC) plays a crucial role in enhancing safety and efficiency in autonomous and electric vehicles by regulating vehicle speed and maintaining safe following distances. This study evaluates ACC performance under two distinct driving scenarios: steady-speed following, where the ego vehicle must track a lead vehicle maintaining a constant velocity, and stop-and-go traffic conditions, which require frequent acceleration and deceleration to adapt to dynamic traffic flow. By implementing a two-level proportional integral derivative (PID) control system within the CARLA simulation environment, we conduct a detailed assessment of key performance metrics, including following distance accuracy, control stability, and deviation from ISO-defined safe distances. In the steady-speed scenario, the system is expected to achieve smooth speed tracking with minimal deviations, ensuring a stable and predictable driving experience. In contrast, the stop-and-go scenario poses greater challenges, as the ego vehicle must respond rapidly to sudden changes in traffic conditions while maintaining control precision. The study explores how the controller adapts to these variations, analyzing its ability to minimize excessive braking, maintain appropriate acceleration, and optimize safety without compromising comfort. Our findings highlight the system's capability to maintain stability in steady-speed conditions while demonstrating adaptability in dynamic stop-and-go situations. This research underscores the importance of refining ACC strategies to accommodate diverse driving conditions, ultimately improving the performance and reliability of autonomous vehicle control systems in real-world environments.

## 1. Introduction

The rapid evolution of automotive technologies has positioned ACC as a cornerstone of modern autonomous vehicles (AVs) and electric vehicles (EVs). The integration of ACC into AVs and EVs has become increasingly significant [1-2]. In AVs, reliable ACC systems are essential for ensuring safe navigation through complex traffic environments, as highlighted by Paden et al. [3] and Tsugawa et al. [4-5]. For EVs, optimized ACC not only enhances safety but can also improves energy efficiency, which is a critical factor considering the importance of battery life in EV operation [6]. Studies have shown that fine-tuning control strategies in ACC can lead to measurable improvements in energy consumption, thereby extending the driving range of EVs [7].

ACC systems automatically adjust vehicle speed to maintain a safe following distance, thereby reducing collision risks and enhancing driving comfort. Much of the traditional research on ACC has focused on environmental challenges like sensor performance degradation in adverse weather conditions [7-8]. The influence of distinct driving modes on ACC behavior remains a largely underexplored area. With the increasing prevalence of AVs and EVs, there is a critical need for control systems that are not only safe and efficient but also adaptable to a range of driving styles. In AVs, reliable ACC is essential for managing the complexities of dynamic traffic environments, while in EVs, optimizing ACC performance can directly contribute to better energy management and extended battery life. A key aspect of ACC performance is maintaining a safe following distance. The ISO 15622:2009 standard defines performance requirements and test procedures for ACC systems, ensuring they maintain appropriate gaps to minimize collision risks [9]. Empirical studies have confirmed that adherence to these standards not only improves safety but also contributes to smoother traffic flow [9-11]. Additional investigations have explored how control strategies can be optimized to consistently meet or exceed these ISO guidelines, reinforcing the importance of integrating standardized safe distance calculations into ACC system design [9].

Recent developments in simulation technology have greatly facilitated ACC research. The CARLA simulator, introduced by Dosovitskiy et al. [12] and later expanded by Dosovitskiy and Ros [13], offers a high-fidelity, open-source environment that provides precise ground-truth data. This level of precision is crucial for validating control algorithms in a repeatable and controlled manner. While other simulation platforms such as PreScan and VISSIM have been used historically, CARLA's ability to replicate complex urban driving scenarios without sensor noise makes it particularly well suited for our study [13]. The CARLA simulator, which offers a state-of-the-art, open-source platform for the development and testing of autonomous driving technologies, providing a high-fidelity environment for evaluating control algorithms and vehicle behavior in diverse scenarios. Its ability to provide high-fidelity, ground-truth data that is free of the noise and variation that come with real-world sensor readings makes sure that evaluations of performance are accurate and can be repeated. This precision is crucial for systematically assessing control strategies under controlled yet diverse driving scenarios. Also, CARLA's adaptability lets you quickly make prototypes and test different control settings over and over again. This makes it a great place to see how different driving modes affect ACC performance [7,8,13].

ACC has evolved from early basic implementations to sophisticated control systems integral to modern autonomous vehicles. Rajamani's early work [1] laid the groundwork for ACC systems by establishing rules for how vehicles behave, and early experiments set the stage for automated speed control in traffic. Over the years, research has expanded to address various operational challenges, from sensor inaccuracies in adverse weather to the complexities of real-time decision-making in dynamic traffic conditions [1,3]. Vahidi and Eskandarian [15] provide a comprehensive overview of how early research in intelligent collision avoidance and autonomous driving spurred advances in ACC technology.

Traditional methods, like the PID controller, are used by many researchers. It is still one of the most popular methods because it is easy to use and works well in real-time control situations [7-8]. Huang et al. [17] demonstrated that PID controllers can maintain stable inter-vehicle spacing across various traffic scenarios, while additional studies have fine-tuned PID parameters to balance responsiveness and comfort.

Intelligent controllers were created to get around the problems that linear PID controllers have in non-linear traffic environments [18]. Fuzzy logic controllers that can adapt to uncertain situations have been studied [19]. Recent research, like that by Li et al. [20], shows that Model Predictive Control (MPC) is a strong alternative that improves control actions over a limited time frame. MPC methods take limitations and future system behavior into account, which makes them more reliable in situations where traffic changes quickly.

Researchers have developed hybrid control methods that integrate PID, fuzzy logic, and Model Predictive Control (MPC) to leverage the strengths of each approach while mitigating their individual limitations. This combination enhances control precision, adaptability, and robustness, allowing for improved system performance across diverse driving conditions [21]. Schindler [10] and later research

have shown that these hybrid methods can improve ACC performance by changing control parameters on the fly based on the current driving conditions.
The main aspects covered in this study are:

- ACC scenarios: The study evaluates ACC performance in two driving conditions: steady-speed following, where the ego vehicle maintains a constant velocity, and stop-and-go traffic, which requires frequent speed adjustments.
- Two-level PID control implementation: A hierarchical control system regulates acceleration and braking. The upper-level controller determines acceleration based on real-time inputs, while the Lower-level controller ensures smooth throttle and braking transitions.
- Safety and compliance: Following distances adhere to ISO 15622 standards, aligning with industry safety guidelines.
- CARLA Simulation Environment: The CARLA platform provides a high-fidelity, modular testing ground for ACC evaluation, enabling realistic vehicle behavior modeling, customizable traffic scenarios, and precise data collection for performance assessment.

The remainder of this paper is structured as follows: section 2 methodology, including the two-level PID control system, detailed vehicle modeling, simulation setup using CARLA, and data collection and performance metrics. section 3 results, and section 4 conclusion

## 2. Methodology

This section outlines the framework for evaluating ACC performance. The present approach integrates a hierarchical two-level PID control system, rigorous safe distance computations, detailed vehicle dynamics modeling, and a high-fidelity simulation environment using CARLA. As illustrated in Figure 1, the system workflow is structured into six key stages. The process begins with Scenario Setup, where the simulation parameters, including road conditions, vehicle behaviours, initial speeds, and sensor configurations, are defined using Python and visual studio (VS) Code. The two-level PID Controller regulates the ego vehicle's acceleration and braking, where the upper-level controller computes the desired acceleration based on safe distance calculations, and the lower-level controller converts this into throttle and brake commands for vehicle actuation.
Following this, vehicle dynamics & sensor data collection takes place within CARLA, where the control signals update the ego vehicle's state, and sensor data (e.g., speed, distance, and acceleration) is recorded in real time. A Feedback Loop ensures adaptive control by continuously feeding the measured vehicle states (speed, acceleration, and distance) back to the PID controller, allowing real-time adjustments. The recorded sensor data is then stored in an Excel-based dataset during the data logging stage, facilitating structured analysis. Finally, in the Performance Analysis stage, the logged data is processed using Python to evaluate key ACC performance metrics. This structured methodology ensures a systematic and data-driven evaluation of the ACC system, leveraging realistic vehicle dynamics modeling and high-fidelity simulation data.

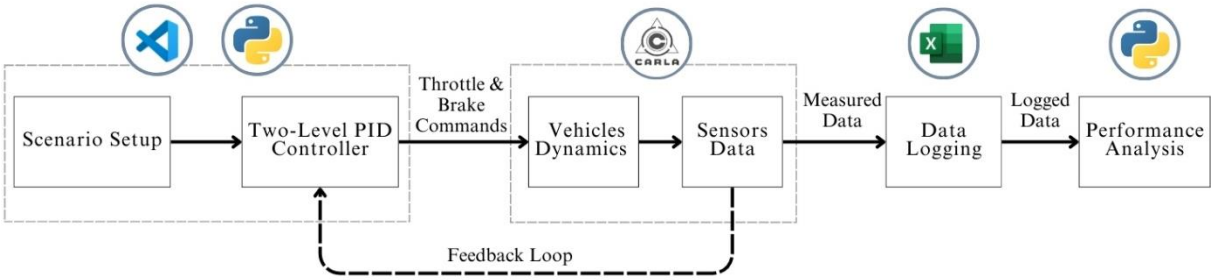


**Figure 1.** Overall System Framework for ACC Simulation

### *2.1. Two-level PID control system*

The ACC controller employs a hierarchical two-level structure to regulate both speed and following distance efficiently. It ensures real-time adjustments by modulating acceleration and braking based on

traffic conditions. By continuously monitoring inter-vehicle distance, speed, and acceleration, the controller minimizes abrupt maneuvers, improving stability. The upper-level controller processes kinematic data and external conditions to compute necessary acceleration changes. Using a PID control strategy, it dynamically regulates the safe following distance, ensuring smooth acceleration and deceleration. The controller continuously refines its parameters based on real-time deviations in inter-vehicle distance, speed variations, and traffic density, maintaining adaptive and responsive driving behavior. As illustrated in Figure 2, the two-level PID control system consists of two hierarchical levels to manage the ego vehicle's longitudinal dynamics. The system begins with the ISO safe distance equation, which calculates the required safe following distance based on the ego vehicle's speed and reaction time, following ISO 15622:2009 standards. The upper-level controller determines the distance error, representing the deviation between the real-time measured distance and the computed safe distance. The PID-based control strategy computes the desired acceleration to maintain an optimal following distance. The lower-level controller then translates this desired acceleration into precise throttle and braking commands. The system calculates an acceleration error by comparing the desired acceleration with the actual acceleration measured from the vehicle's dynamics. This ensures that the ego vehicle follows the computed reference acceleration while compensating for external factors such as road conditions and aerodynamic drag. The sensors data Module collects real-time vehicle speed, acceleration, and inter-vehicle distance information. This data is then used in a feedback loop, continuously refining control actions based on real-time driving conditions. The vehicle dynamics block processes the control signals and updates the vehicle's actual acceleration and speed. These outputs are continuously fed back into the controller, allowing for adaptive control adjustments.

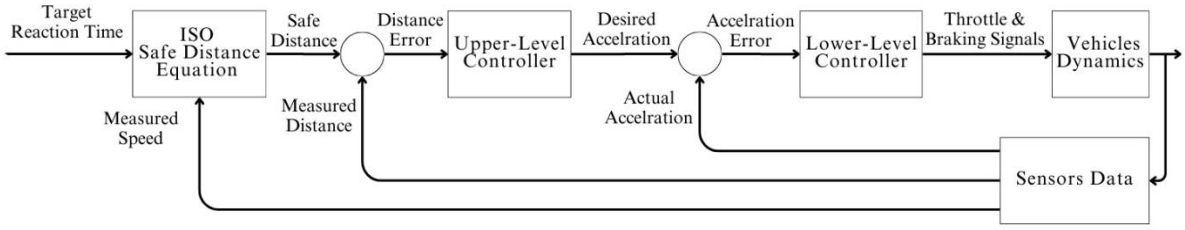


**Figure 2.** Two-level PID control system architecture

The upper-level controller calculates the distance error, which is the difference between the computed safe following distance and the real-time measured distance, as follows:

$$e(t) = d_{\text{safe}}(t) - d(t) \tag{1}$$

where: $d_{\text{safe}}(t)$ represents the computed safe following distance, and $d(t)$ is real-time measured distance to the preceding vehicle. The acceleration reference is then computed as:

$$a_{\text{ref}}(t) = K_{p1}\, e(t) + K_{i1} \int_0^t e(\tau)\, d\tau + K_{d1} \frac{de(t)}{dt} \tag{2}$$

where $K_{p1}$, $K_{i1}$, and $K_{d1}$ are the proportional, integral, and derivative gains, respectively. The lower-level controller is responsible for translating the desired acceleration into precise throttle and braking commands, ensuring the ego vehicle's actual acceleration accurately follows the reference acceleration. This controller continuously processes real-time data to compute control signals that adjust the vehicle's longitudinal dynamics. By compensating for external disturbances such as road grade and aerodynamic drag, it enhances the robustness of the ACC system in diverse driving conditions. The acceleration error is computed as:

$$\Delta a(t) = a_{\text{ref}}(t) - a(t) \tag{3}$$

where $a(t)$ is the actual acceleration. The control signal for throttle/braking is then determined as:

$$u(t) = K_{p2}\, \Delta a(t) + K_{i2} \int_0^t \Delta a(\tau)\, d\tau + K_{d2} \frac{d\, \Delta a(t)}{dt} \tag{4}$$

Where $u(t)$ is the control signal for throttle and braking; $K_{p2}$, $K_{i2}$, and $K_{d2}$ are the corresponding PID gains.

In a discrete-time implementation, the control law is formulated as:

$$u[k] = u[k-1] + K_p(e[k] - e[k-1]) + K_i\, e[k]\, \Delta t + K_d \frac{e[k] - 2e[k-1] + e[k-2]}{\Delta t} \quad (5)$$

where k is the discrete time index and Δt is the sampling interval.

The safe following distance is an essential parameter in the upper-level controller's decision-making process, ensuring that the ego vehicle maintains a safe gap from the lead vehicle while optimizing traffic flow. The calculation accounts for the reaction time of the driver or control system, the vehicle's deceleration capability, and a minimum safe margin to accommodate uncertainties. This approach ensures smooth vehicle operation under varying traffic conditions and enhances overall driving safety. Based on ISO 15622:2009 [9], the safe following distance is computed as follows:

$$d_{\text{safe}}(t) = v(t)\, t_{\text{reaction}} + \frac{v(t)^2}{2\, a_{\text{dec}}} + d_{\text{min}} \quad (6)$$

where: $v(t)$ is the vehicle's speed, $t_{\text{reaction}}$ is the driver's (or system's) reaction time, $a_{\text{dec}}$ is the maximum deceleration (braking capability), and $d_{\text{min}}$ is an additional safety margin.

*2.2. Detailed Vehicle Modeling*

A precise vehicle model is crucial for accurately simulating ACC performance, capturing the forces influencing vehicle motion. The longitudinal dynamics model used in this study incorporates aerodynamic drag, rolling resistance, and traction forces, which collectively govern vehicle acceleration and deceleration. The vehicle's motion is governed by Newton's Second Law:

$$m \frac{dv(t)}{dt} = F_{\text{traction}}(t) - F_{\text{drag}}(t) - F_{\text{roll}}(t) \quad (7)$$

where: $m$ is the vehicle mass, $F_{\text{traction}}(t)$ is the force generated by the motor, $F_{\text{drag}}(t)$ is the aerodynamic drag force, and $F_{\text{roll}}(t)$ is the rolling resistance force. The aerodynamic drag force is modelled as:

$$F_{\text{drag}}(t) = \frac{1}{2}\, \rho\, C_d\, A_f\, v(t)^2 \quad (8)$$

where: ρ is the air density, $C_d$ is the drag coefficient, and $A_f$ represents the vehicle's frontal area, influencing aerodynamic resistance. The rolling resistance force, caused by the deformation of tires on the road surface, is given by:

$$F_{\text{roll}}(t) = C_{\text{rr}}\, m\, g \quad (9)$$

with $C_{\text{rr}}$ representing the rolling resistance coefficient and $g$ is the gravitational acceleration and represents the vehicle mass. The traction force, generated by the vehicle's powertrain, is expressed as:

$$F_{\text{traction}}(t) = \eta\, u(t) \quad (10)$$

where η is a scaling factor incorporating drivetrain efficiency and torque conversion and represents the control input that modulates acceleration. This detailed vehicle model provides the necessary foundation for simulating realistic ACC behaviours, ensuring accurate controller evaluation across diverse driving conditions.

*2.3. Simulation Setup Using the CARLA Simulator*

CARLA simulator is utilized as the primary simulation platform due to its high-fidelity, open-source architecture, which enables realistic and controlled evaluation of ACC performance. CARLA replicates urban and highway driving scenarios with accurate road geometries, dynamic traffic interactions, and variable environmental conditions. It provides precise measurements of vehicle dynamics and inter-vehicle interactions, ensuring accurate performance evaluation. The simulated vehicle model used in the study is a Tesla Model 3, representing a modern autonomous electric vehicle, recording critical driving parameters, including speed $v(t)$, acceleration $a(t)$, throttle and brake inputs $u(t)$, and inter-vehicle distance $d(t)$. These data points provide an in-depth assessment of ACC behavior across different driving conditions. The simulation scenarios are designed to encompass a range of real-world conditions, including variations in traffic density, road layouts, and

environmental factors. ACC performance is evaluated under two distinct scenarios: case 1 (steady-speed following), where the ego vehicle follows a lead vehicle maintaining a constant velocity, and case 2 (Stop-and-Go Traffic), which involves frequent acceleration and deceleration to adapt to dynamic traffic conditions. These scenarios test the system's robustness and adaptability in maintaining safe following distances and stable control. to test the system's robustness and adaptability. This simulation framework allows for a controlled yet realistic evaluation of ACC strategies, ensuring reliable insights into system performance under diverse driving conditions.

### *2.4. Data Collection and Performance Metrics*

Data is captured from CARLA's high-frequency output, logged at a high sampling rate, and exported to an Excel sheet for post-processing. The dataset includes time-series data for key parameters such as vehicle speed $v(t)$, acceleration $a(t)$, throttle/brake command $u(t)$, inter-vehicle distance $d(t)$, and computed safe distance $d_{\text{safe}}(t)$ (from Equation (6)), ensuring comprehensive coverage of ACC performance metrics across different scenarios. Data processing was performed to extract key performance measures—mean, minimum, maximum, standard deviation, and variance. In addition, visualization techniques including radar charts, histograms, and time-series plots were employed to provide deeper insights into ACC behavior.

## 3. Results

### *3.1. Two-level PID Controller Performance Evaluation*

The performance of the two-level PID controller was evaluated under two distinct test scenarios using the CARLA simulator. The first scenario involved two vehicles initially moving at a constant speed of 50 km/h, while the second scenario analyzed the controller's behavior when both vehicles started from rest. The assessment primarily focused on maintaining a safe following distance and achieving smooth speed adaptation while responding to the lead vehicle's motion.

Case 1: Both Vehicles Initially Moving at 50 km/h

In this scenario, both the ego and lead vehicles were initially set to travel at 50 km/h. The two-level PID controller was tasked with adjusting the acceleration and deceleration to maintain a safe following distance. The distance-time plot (Figure 3) indicates that the actual following distance started at 30.0 m, while the ISO-defined safe distance was initially 33.795 m. As the controller adjusted the speed, the actual following distance gradually increased and stabilized at 35.026 m, aligning closely with the final ISO safe distance of 35.201 m. This demonstrates the controller's ability to effectively regulate the vehicle's spacing while ensuring safety compliance. As the ego vehicle approached stability, the ISO-defined safe distance fluctuated within the range of 33.5 m to 35.2 m due to minor variations in speed. The actual following distance remained responsive, tracking these variations with a lag of 0.2 m to 0.5 m. The speed-time plot (Figure 4) shows that the ego vehicle initially traveled at 48.668 km/h, matching the lead vehicle's speed. Initially, a slight dip in ego speed was observed due to early braking, bringing the speed momentarily down to 44.0 km/h before gradually stabilizing. The vehicle recovered and reached 50.021 km/h, closely following the lead vehicle's final speed of 49.992 km/h. These results validate the effectiveness of the two-level PID strategy in maintaining a stable car-following behavior even when vehicles start with a nonzero initial velocity.

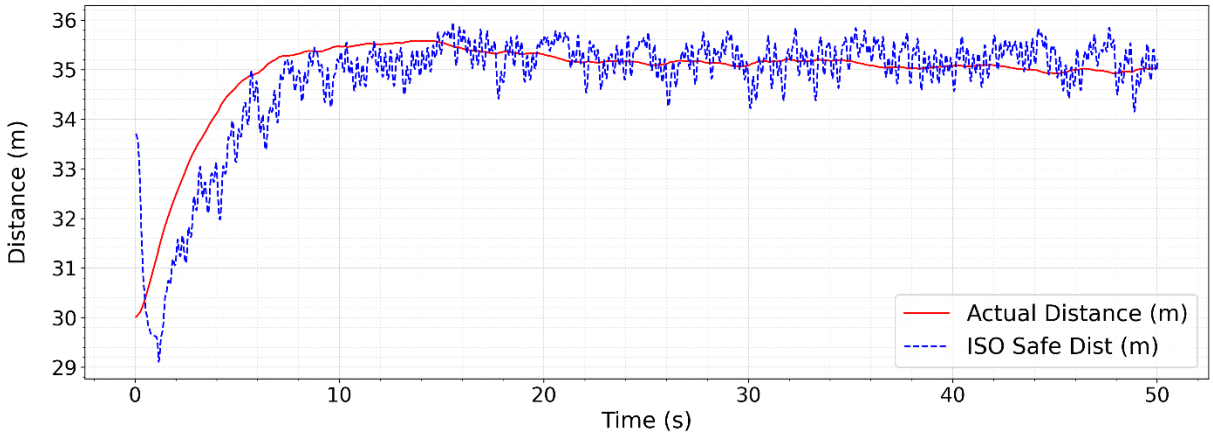


**Figure 3.** Distance-time plot for (case 1). Illustrates the evolution of the actual following distance and ISO-defined safe distance over time when both vehicles initially travel at 50 km/h

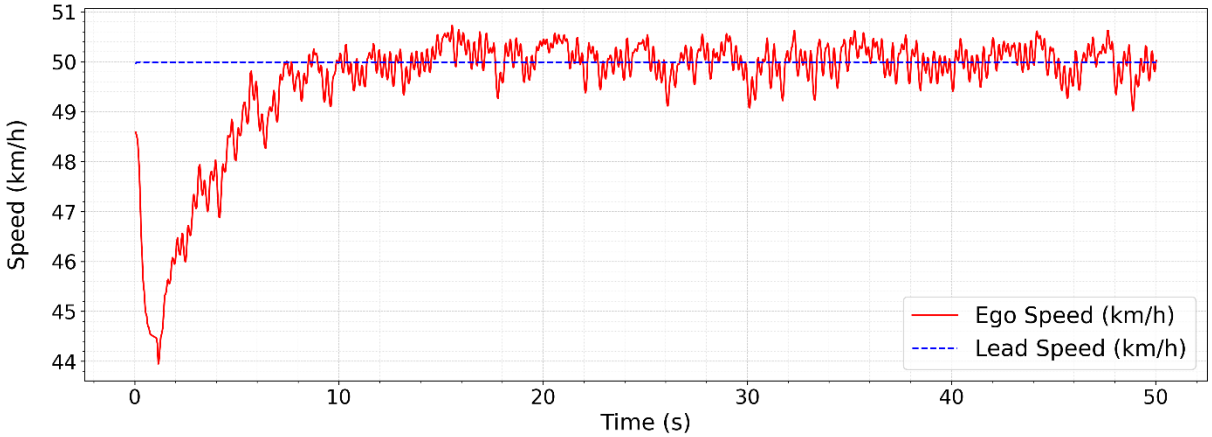


**Figure 4.** Speed-time plot for (case 1). Depicts the ego vehicle's speed response relative to the lead vehicle over time in case 1

Case 2: Both Vehicles Starting from Rest

In the second scenario, both vehicles were initially stationary, and the ego vehicle had to accelerate and adjust its speed while following the lead vehicle. The distance-time plot (Figure 5) illustrates that the following distance initially started at 10.0 m, which was above the ISO-defined safe distance of 2.0 m since at standstill the safe distance equals the minimum margin. As the vehicles accelerated, the controller successfully increased the gap, stabilizing it at 34.643 m, closely aligning with the final ISO-defined safe distance of 35.155 m. This indicates that the PID controller effectively maintained a gradual and stable increase in the gap, avoiding abrupt accelerations that could compromise passenger comfort. During the acceleration phase, the ISO-defined safe distance increased steadily from 2.0 m to 35.155 m, with occasional minor oscillations within ±0.5 m due to the lead vehicle's acceleration pattern. The actual following distance closely mirrored this trend with a minor delay of approximately 0.3 to 0.7 seconds, showing the controller's ability to adapt dynamically. The speed-time plot (Figure 6) highlights a controlled acceleration phase, where the ego vehicle initially started from 0.0 km/h and progressively increased its speed to 49.977 km/h. The lead vehicle, which also started from 0.0 km/h, reached a final speed of 49.08 km/h.

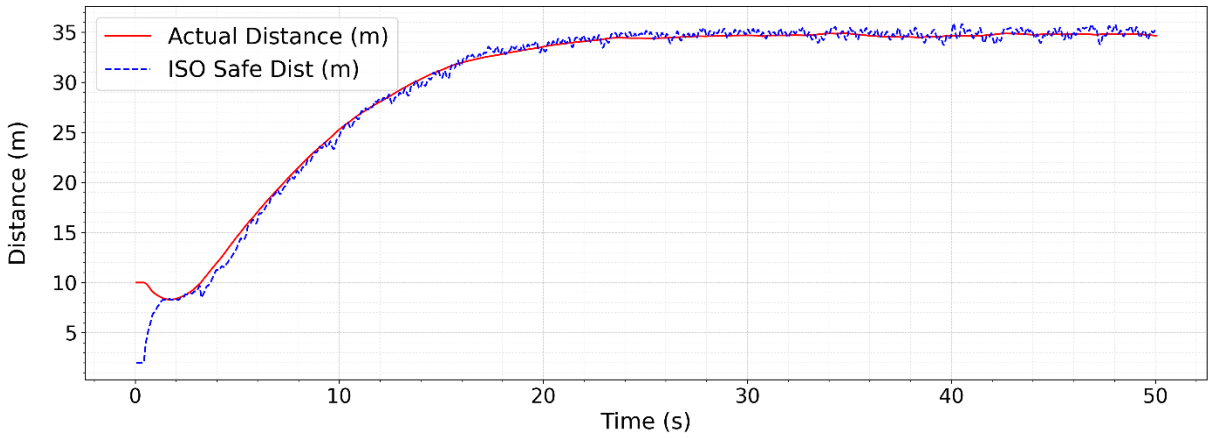


**Figure 5.** Distance-time plot for case 2. Shows the actual following distance and ISO-defined safe distance variations when both vehicles start from rest

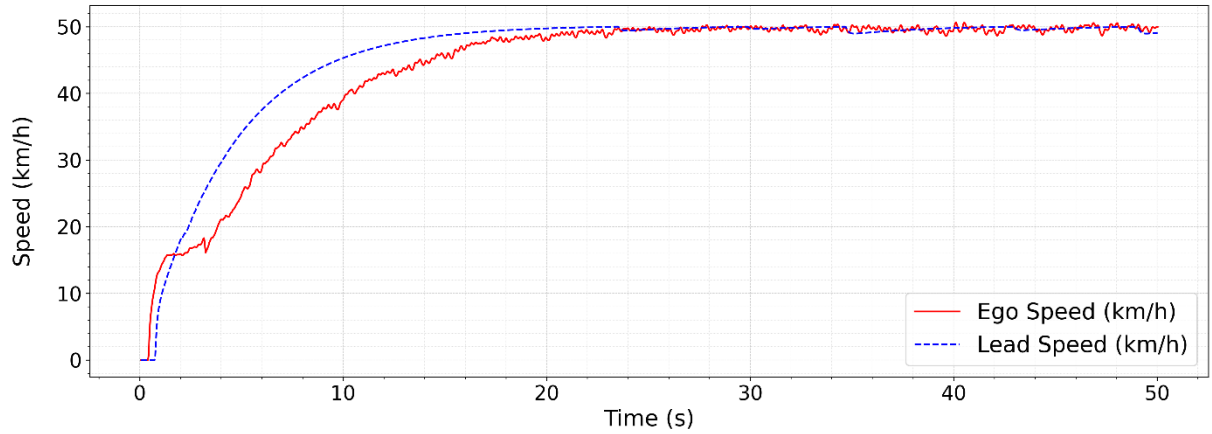


**Figure 6.** Speed-time plot for case 2. Represents the ego vehicle's speed adaptation as it follows the lead vehicle from a stationary start

The controller efficiently mitigated excessive acceleration spikes, maintaining a smooth acceleration ramp with minor deviations of ±1.5 km/h. Unlike conventional PID controllers that often struggle with aggressive responses in stop-and-go conditions, the two-level PID strategy demonstrated superior adaptability, reducing speed mismatches and unwanted jerks. The ISO-defined safe distance continued to fluctuate within a narrow range of 34.5 m to 35.2 m due to the minor variations in the lead vehicle's acceleration. The actual following distance effectively tracked these variations with a delay of approximately 0.3 to 0.6 seconds, ensuring a dynamic and adaptive response. By the end of the test, the ego vehicle maintained a near-perfect following profile with less than 1% deviation from the desired trajectory, demonstrating the robustness of the two-level PID controller under varying conditions.

*3.2. Safety Analysis of Following Distance*

The performance of the two-level PID controller was further analyzed in terms of safety by evaluating the actual following distance relative to the ISO-defined safe distance. The goal was to ensure that the ego vehicle consistently maintained an appropriate gap from the lead vehicle, minimizing the risk of unsafe proximity while optimizing traffic flow efficiency. The safe distance tracking plot (Figure 3) for case 1 illustrates that the actual following distance started at 30.0 m, while the ISO-defined safe distance was 33.795 m. The actual following distance initially deviated slightly, fluctuating between -3.8 m to 2.3 m relative to the ISO-defined safe distance. However, as the controller stabilized, the

following distance remained well-aligned, converging around 34.939 m on average with a standard deviation of 0.887 m. These results confirm that the two-level PID controller effectively reduces deviation from the required safe distance, ensuring that the ego vehicle maintains a safe yet efficient following distance even when minor variations in lead vehicle speed occur. For case 2, where both vehicles started from a stationary position, the safe distance tracking plot (Figure 5) indicates a more dynamic adjustment phase. Initially, the actual following distance was 10.0 m, while the ISO-defined safe distance was 2.0 m, leading to an overshoot. However, as acceleration progressed, the controller successfully adjusted the following distance, bringing it into alignment with the ISO-defined safe range. The final following distance stabilized at 29.941 m, closely following the ISO-defined requirement of 29.816 m, with a maximum deviation of 8.0 m and a minimum deviation of -1.2 m. The higher standard deviation of 7.97 m reflects the increased variability in tracking accuracy during the acceleration phase, which gradually smoothened over time. These findings indicate that the controller efficiently adapts to varying speed conditions, particularly in stop-and-go situations, ensuring a gradual and controlled increase in the following gap.

To evaluate how accurately the ego vehicle maintained the safe following distance, a histogram of distance error was generated for both test cases, showing the difference between the actual following distance and the ISO-defined safe distance. The histogram illustrates the distribution of distance errors, providing insight into how consistently the ego vehicle maintained the safe following distance over time. A well-centered distribution around zero indicates strong adherence to the safe following distance, while a wider spread suggests increased variations and necessary adjustments by the controller. For case 1, where both vehicles started at a constant speed of 50 km/h, the histogram (Figure 7) shows a high concentration of distance error values close to zero, confirming that the controller effectively maintained the safe following distance with minimal error. The deviations are mostly within a narrow range of -2.0 m to 1.5 m, with a mean deviation of 0.13 m and a standard deviation of 0.61 m. This indicates that the controller effectively reduced fluctuations, ensuring stable and predictable car-following behavior.

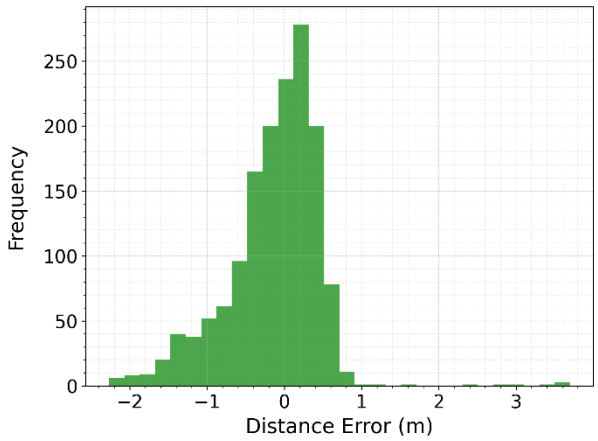


**Figure 7.** Histogram of distance error for case 1. The distribution shows that the controller maintained a stable following distance with minimal error, demonstrating strong adherence to the ISO-defined safe distance.

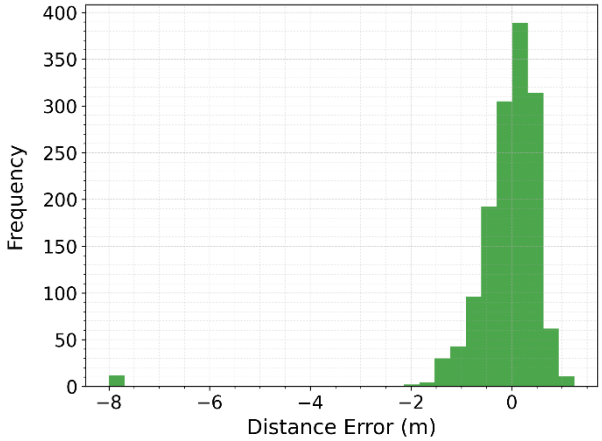


**Figure 8.** Histogram of distance error for case 2. The distribution indicates greater initial fluctuations in following distance due to acceleration, with a long tail towards negative errors before stabilizing.

The histogram demonstrates that the system rarely deviated significantly from the safe distance, highlighting the strength of the controller in steady-speed conditions. In contrast, case 2, where both vehicles started from rest, exhibits a wider distribution of distance error values (Figure 8), particularly in the early acceleration phase, indicating greater fluctuations before stabilizing. The histogram indicates a broader spread, particularly in the early acceleration phase, with deviations ranging from -

8.0 m to 0.5 m. The mean deviation was 1.12 m, and the standard deviation was 0.98 m, significantly higher than in case 1. The histogram shows a strong clustering of values around small deviations, but the long tail towards -8.0 m suggests the ego vehicle initially lagged behind the expected safe distance before successfully adjusting. These results indicate that, although the controller faced greater challenges in a stop-and-go scenario, it effectively reduced errors over time, demonstrating strong adaptability in dynamic driving conditions.

*3.3. Statistical Performance Analysis*

This section presents a statistical performance analysis to quantify the effectiveness of the two-level PID controller in regulating vehicle behaviour. To better understand the controller's efficiency, Table 1 presents key statistical metrics comparing both test cases, providing insights into the system's stability and adaptability. This table highlights the controller's strong performance in maintaining a stable following distance in case 1, while demonstrating adaptability in case 2 under stop-and-go conditions. To visually represent these metrics, Figure 9 presents a radar chart illustrating the normalized values of the key statistical metrics. The values were scaled between 0 and 1 to provide a comparative visualization of the controller's behavior across both cases. The radar chart provides a clear depiction of the relative differences between the two scenarios. case 1, represented by the blue plot, shows a more consistent and stable response, particularly in terms of maintaining a lower standard deviation of distance error and a minimal deviation from the ISO-defined safe distance. Meanwhile, case 2, shown in orange, exhibits a higher maximum deviation and a greater variance in following distance, which corresponds to the more dynamic nature of the stop-and-go scenario.

**Table 1.** Statistical comparison of following distance performance.

| Metric | Case 1 Value | Case 2 Value |
|---|---|---|
| Mean Actual Following Distance (m) | 34.94 | 29.94 |
| Mean ISO Safe Distance (m) | 34.81 | 29.81 |
| Maximum Deviation from Safe Distance (m) | 2.27 | 8.00 |
| Minimum Deviation from Safe Distance (m) | -3.79 | -1.24 |
| Standard Deviation of Distance Error (m) | 0.61 | 0.98 |

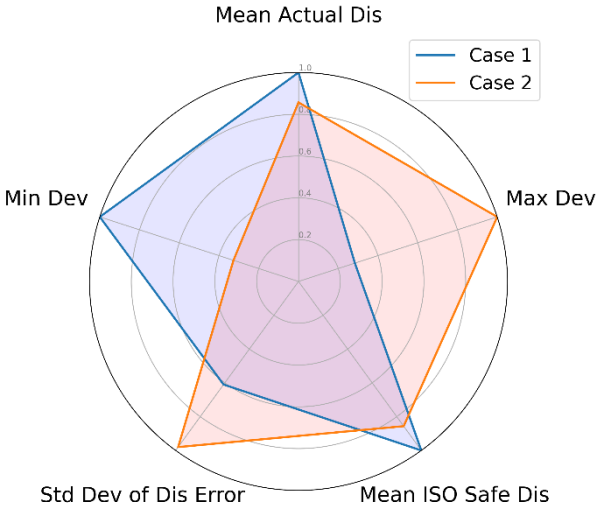


**Figure 9.** Normalized statistical performance metrics comparison

By analyzing the chart, we can observe that the Mean Actual Following Distance and Mean ISO Safe Distance are nearly identical for both cases, confirming that the controller effectively tracks the reference safe distance. However, the higher Maximum Deviation in case 2 highlights the increased challenge of maintaining smooth transitions in speed and spacing during acceleration phases. The Standard Deviation of Distance Error, although slightly higher in case 2, remains within an acceptable range, indicating that the controller successfully stabilizes the vehicle's response over time. in maintaining a stable following distance in case 1, while demonstrating adaptability in case 2 under stop-and-go conditions. The significantly lower standard deviation in case 1 (0.61 m) confirms better stability, whereas the slightly higher deviation in case 2 (0.98 m) reflects the adjustments made during acceleration while still keeping the vehicle close to the desired safe distance.

The statistical performance analysis reveals key differences between the two scenarios. case 1 exhibited higher stability, with deviations tightly controlled around the ISO-defined safe distance, as evidenced by a standard deviation of distance error of 0.61 m. In contrast, case 2 showed slightly greater variability, with a standard deviation of 0.98 m, which, while higher, remained within an acceptable range. The ego vehicle in case 1 experienced minimal fluctuations, confirming smoother control actions with minor adjustments, whereas higher variability was observed in case 2, particularly during the acceleration phase, where speed variations reached ±10 km/h. Additionally, more outliers were present in case 2, suggesting that rapid adjustments were required initially to regulate acceleration and establish a stable following distance. Despite these initial fluctuations, the controller effectively minimized deviation over time. Overall, the two-level PID controller demonstrated effective performance in both cases. In case 1, the system maintained a stable following distance with minimal deviation, ensuring strong tracking ability under steady-speed conditions. In case 2, although the initial acceleration phase introduced larger variations, the controller successfully adjusted the following distance over time, demonstrating adaptability in dynamic driving conditions. Ultimately, the vehicle remained within acceptable limits, balancing safety, responsiveness, and smooth operation throughout both scenarios.

## 4. Conclusion

This study evaluated the performance of a two-level PID-based (ACC) system using high-fidelity simulations in CARLA. The analysis focused on two driving scenarios: case 1 (steady-speed following) and case 2 (stop-and-go traffic), examining the controller's ability to maintain a safe following distance while ensuring smooth vehicle operation.

The results demonstrated that the ACC system successfully maintained stable and efficient control across both scenarios. case 1 exhibited higher stability, with minimal deviations from the ISO-defined safe following distance, confirming precise tracking and steady-state control. In contrast, case 2 introduced more dynamic variations due to frequent acceleration and deceleration, yet the controller effectively adjusted to ensure safety and responsiveness. Statistical analysis highlighted the system's capability to regulate inter-vehicle spacing efficiently. The mean actual following distance closely matched the ISO safe distance in both cases, validating the controller's accuracy in maintaining appropriate gaps. While maximum deviation was higher in case 2, the standard deviation of distance error remained within an acceptable range, signifying effective disturbance handling.

Despite its effectiveness, the proposed system has some areas for further refinement. The PID controller requires parameter tuning, which, while manageable, may involve some time to achieve optimal performance across different driving conditions. Additionally, the control system currently operates with fixed control parameters, and integrating adaptive or learning-based techniques in future work could further enhance its responsiveness in highly dynamic traffic scenarios.

In conclusion, the two-level PID ACC system proved to be robust in diverse driving conditions. The system demonstrated strong stability in steady-speed driving and adaptability in dynamic stop-and-go scenarios, making it a viable approach for improving vehicle automation and safety. Future research could explore further refinements, such as integrating AI-based predictive control strategies to enhance adaptation to varying traffic behaviors and further optimize vehicle efficiency.

## 5. References


[1] Rajamani R 2012 *Vehicle Dynamics and Control* 2nd ed (New York: Springer)

[2] El-Khatib M F, Sabry M N, El-Sebah M I A and Maged S A 2023 Hardware-in-the-loop testing of simple and intelligent MPPT control algorithm for an electric vehicle charging power by photovoltaic system *ISA Trans.* **137** 656–669

[3] Paden B, Čáp M, Yong J, Yershov D and Frazzoli E 2016 A survey of motion planning and control techniques for self-driving urban vehicles *IEEE Trans. Intell. Veh.* **1**(1) 33–55

[4] Tsugawa S, Jeschke S and Shladover S E 2016 A review of the state-of-the-art and future research needs of cooperative adaptive cruise control (CACC) *IEEE Trans. Intell. Veh.* **1**(1) 42–56

[5] El-Khatib M F, Khater F M, Hendawi E and El-Sebah M I A 2025 Simplified and intelligent controllers for multi-input multi-output processes *Eng. Appl. Artif. Intell.* **141** 109816

[6] Tran M, Banerjee S and Nguyen T T 2018 Electric vehicles: a review of the technology, infrastructure and market *Renew. Sust. Energ. Rev.* **89** 360–373

[7] Al-Hindawi R and Adas M 2024 Evaluation and optimization of adaptive cruise control in autonomous vehicles using the CARLA simulator: a study on performance under wet and dry weather conditions *Proc. IEEE Int. Conf. Adv. Syst. Emerg. Technol.* (IC_ASET 2024) pp 1–6

[8] Shrivastava S, Somthankar A, Pandya V and Patil M 2023 Implementation of a PID controller for autonomous vehicles with traffic light detection *Intelligent Computing and Networking: Proc. IC-ICN 2022* (Singapore: Springer Nature) pp 1–13

[9] ISO 15622:2009 Road vehicles—Adaptive cruise control systems—Performance requirements and test procedures (Geneva: International Organization for Standardization)

[10] Schindler G 2017 Adaptive cruise control *Encyclopedia of Automotive Engineering* ed J G Webster (Chichester: John Wiley & Sons)

[11] Wang B, Zhang Z and Li Y 2012 Impact of adaptive cruise control on road traffic flow: a simulation study using SUMO *IEEE Trans. Intell. Transp. Syst.* **13**(1) 140–148

[12] Dosovitskiy A, Ros G, Codevilla F, Lopez A and Koltun V 2017 CARLA: an open urban driving simulator *Proc. Conf. Robot Learning* (California, USA)

[13] Dosovitskiy A and Ros G 2019 Exploring the potential of CARLA for autonomous driving research *IEEE Trans. Veh. Technol.* **68**(2) 1590–1601

[14] Aner E A, Awad M I and Shehata O M 2024 Performance evaluation of PSO-PID and PSO-FLC for continuum robot's developed modeling and control *Sci. Rep.* **14**(1) 733

[15] Vahidi A and Eskandarian A 2003 Research advances in intelligent collision avoidance and autonomous driving *IEEE Trans. Intell. Transp. Syst.* **4**(2) 143–153

[16] Aner E A, Awad M I and Shehata O M 2023 Modeling and trajectory tracking control for a multi-section continuum manipulator *J. Intell. Robot. Syst.* **108**(3) 49

[17] Huang X, Li H and Du J 2018 Research on adaptive cruise control system of autonomous vehicles based on PID control *Proc. 37th Chinese Control Conf. (CCC)* (Wuhan, China) pp 8690–8695

[18] Gunasekaran P, Sivasubramanian R, Periyasamy K, Muthusamy S, Mishra O P, Ramamoorthi P, Sadasivuni K K and Geetha M 2024 Adaptive cruise control system with fractional order ANFIS PD+I controller: optimization and validation *J. Braz. Soc. Mech. Sci. Eng.* **46**(4) 184

[19] Zhang Y, Wang W and Wang X 2008 A fuzzy logic based adaptive cruise control system for improving vehicle safety in car following *IEEE Trans. Intell. Transp. Syst.* **9**(4) 714–724

[20] Li Y, Li K, Wang X, Wang Y and Liu J 2010 A model predictive control approach to adaptive cruise control *IEEE Trans. Control Syst. Technol.* **18**(4) 851–860

[21] Mao J, Yang L, Hu Y, Liu K and Du J 2021 Research on vehicle adaptive cruise control method based on fuzzy model predictive control *Machines* **9**(8) 160