International Journal of Innovative Computing, Information and Control
Volume **22**, Number **2**, April **2026**
ICIC International ©2026 ISSN 1349-4198
pp. 373–391

OPTIMIZATION OF ADAPTIVE CRUISE CONTROL IN CURVED ROADS USING ARCHIVED GENETIC ALGORITHM AND CROW SEARCH ALGORITHM

Dhidik Prastiyanto*, Esa Apriaskar, Feddy Setio Pribadi
Imam Khoirul Akbar, Ilya Amelia and Akhyar Abdillah Manaf

Department of Electrical Engineering
Universitas Negeri Semarang
Building E11, UNNES Sekaran Campus, Gunungpati, Semarang 50229, Indonesia
{ esa.apriaskar; feddy.setio }@mail.unnes.ac.id
{ imamkakbar; ilyame9; akhyarabdillah313 }@students.unnes.ac.id
*Corresponding author: dhidik.prastiyanto@mail.unnes.ac.id

Received July 2025; revised November 2025

Abstract. *Road accidents cause fatalities and economic losses. Studies aim to reduce them using ADAS, including ACC, which ensures a safe distance. This research aims to contribute to ADAS development by addressing the optimization challenges of control algorithms for ACC parameters on curved roads with various curvatures of the roads. The road environment consists of a one-lane road with curvature radii 380 and 760 meters. Some algorithms are developed in this research for ACC with steering control on different road curvature, utilizing Crow Search Algorithm (CSA) and Genetic Algorithm (GA) with Integral Absolute Error (IAE) as the objective function to minimize errors for both control systems. The optimized parameters are G_V, G_X, and G_{VX} for ACC and G_Y, k_p and k_i for steering control. This research develops the advancement of the two algorithms in the form of Archived CSA (ACSA) and GA (AGA), which have faster computational time. The archived methods demonstrate significant improvement by reducing computational time by 77-83% while maintaining competitive performance against original. Although OCSA produces the lowest average IAE values, namely 2.1851 for wide curve and 5.2330 for sharp curve, the archived methods stand out for their faster convergence. These results confirm that improved vehicle control in various road curvature enhances driver safety.*
Keywords: Adaptive cruise control, Road curvature radius, Crow search algorithm, Genetic algorithm

1. **Introduction.** The volume of road traffic has been steadily increasing, which has consequently heightened the risk of crashes and traffic oscillations. Traffic accidents represent a major global concern, causing millions of fatalities and severe injuries each year. These incidents have far-reaching consequences, impacting not only the direct victims but also society at large, both economically and socially. Beyond safety considerations, previous research has emphasized the importance of understanding transportation system dynamics to improve stability and passenger comfort [1]. Building upon this understanding, it becomes essential to develop more reliable vehicle control systems that can minimize human error and improve overall driving safety. To reduce the likelihood of such accidents, the development of Automated Vehicle (AV) offers a promising solution. AV offers various features designed to enhance driver safety and comfort [2]. Advancements in AV technology have the potential to alleviate the burden on drivers, thereby minimizing the risk of driver negligence [3]. One example of this technology is Adaptive Cruise Control (ACC),

DOI: 10.24507/ijicic.22.02.373

which automatically regulates vehicle velocity to maintain a safe distance from the preceding vehicle. However, while ACC performs effectively on straight road, its adaptability and control precision on curved road condition remain an ongoing research challenge that calls for further optimization.

Recent research on AV equipped with ACC has increasingly focused on their dynamic performance on curved roads. These studies address several aspects, including modeling AV speed profiles on curved roads [4,5] and optimizing fuel efficiency [6,7]. In addition to such studies, stability and control analyses of car-following models in connected and autonomous environments have provided valuable insights into cooperative vehicle behavior and adaptive speed regulation under varying traffic conditions [8]. However, most of them still focus on specific control goals rather than the overall adaptability of ACC to diverse road geometries. Given the critical importance of maintaining safety while AV navigate corners, recent work has shifted toward the advancement of control methods within ACC systems [9]. Nevertheless, existing approaches often remain limited by analytically derived parameters that may not effectively adapt to dynamic road conditions.

The introduction of a curvature gain in analytically regulated ACC control has been shown to enable AVs to traverse curved road smoothly, preventing sudden speed increases that could lead to accident [10]. Similarly, Model Predictive Control (MPC) has demonstrated superior vehicle performance on curved roads compared to Proportional-Integral-Derivative (PID) control [11]. The implementation of MPC in other studies also yielded favorable results under curved road conditions [10-13]. A different approach employed a linear quadratic controller to ensure safe vehicle [16]. However, the control parameters in these studies [8-14] were still determined analytically, which limits the controller's adaptability and accuracy under rapidly changing curvature and dynamic vehicle interactions.

The use of AI to determine control parameters has been explored by comparing fuzzy logic with PID controllers, where fuzzy logic generally demonstrated better performance [17]. Similarly, fuzzy-based decision-making frameworks have been effectively employed to handle uncertainty and optimize complex multi-criteria problems, demonstrating the potential of fuzzy logic in control system optimization [18]. However, its fuzzy variables still required manual calibration [19], indicating that further improvement is possible. This supports the notion that AI utilization in ACC can yield positive impacts. Neural Networks (NNs) have also been employed to enhance the performance of MPC and PID controllers for ACC on curved roads [10]. Nevertheless, the accuracy of NN-based controllers depends heavily on the training duration, which can be challenging to execute over a wide range of processes due to the substantial computational resources required [20]. Other studies have utilized metaheuristic algorithms such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), respectively [18-20]. Unfortunately, these studies focused solely on straight-road scenarios, which neglect the complex lateral dynamics and varying centrifugal forces that characterize real-world curved roads. As a result, their optimized controllers may perform well on straight segments but degrade in stability and responsiveness on curves.

To better understand this issue, it is essential to consider the influence of road geometry. Road geometry is a critical component of the driving environment, as it directly influences vehicle performance and the complexity of driving tasks [24]. On curves with smaller radius, the required speed reduction and steering precision impose significant control challenges, while larger radius allows smoother maneuvering and higher speeds. This difference in curvature directly affects the control effort and stability margin of ACC systems, creating varying optimization requirements. These factors collectively restrict the overall performance of ACC systems. However, none of the existing studies have examined the application of metaheuristic optimization methods under varying road-curvature radii

[5,6,10-14]. Therefore, this study introduces an optimization framework for ACC systems that explicitly considers different road curvature radii, a factor that has been overlooked in previous optimization studies. Unlike prior studies limited to single-radius or straight road conditions, this work compares four metaheuristic algorithms – Original Crow Search Algorithm (OCSA), Original Genetic Algorithm (OGA), Archived Crow Search Algorithm (ACSA), and Archived Genetic Algorithm (AGA) – to systematically evaluate their effectiveness on two curved-road scenarios. This approach provides a more realistic and comprehensive evaluation of ACC performance under varying curvature conditions.

The structure of the paper is as follows. Section 1 presents the background, research objectives, and contributions. Section 2 explains the system model and control optimization using OCSA, OGA, ACSA, and AGA. Section 3 describes the methodology for implementing and tuning the control system. Section 4 discusses the simulation results and algorithm performance. Section 5 concludes the study with key findings, limitations, and future directions.

2. System Model and Environment.

2.1. Vehicle model.

To simulate vehicle movement on curved roads, a mathematical model is developed that incorporates key dynamic properties, including mass, mass distribution, inertia, suspension, and braking systems. The vehicle's motion is described by a set of differential equations that incorporate the main external forces, including gravitational effects, tire-road friction, and aerodynamic resistance. These equations are then solved numerically. Parameters like the vehicle mass and friction coefficients play a critical role in shaping the resulting motion. The vehicle dynamics model depicted in Equation (1) is a mathematical representation of the vehicle's lateral and longitudinal dynamics. These equations incorporate various aspects such as lateral velocity (V_y), yaw angle (ψ), yaw angle rate (ψ'), longitudinal velocity (V_x), and longitudinal acceleration (V_x'). The first matrix on the left-hand side illustrates the interaction among these parameters, incorporating the front tire cornering stiffness coefficient (C_f), rear tire cornering stiffness coefficient (C_r), vehicle mass (m), and the distances from the center of gravity to the front (I_z) and rear (l_r) axles. This matrix reflects how changes in lateral velocity, yaw angle, and yaw rate influence the vehicle's dynamics. Furthermore, the steering input vector (δ) indicating the steering angle of the front wheels, and the longitudinal input vector (u) typically derived from the throttle or brakes, are present, along with matrices depicting the influence of these inputs on vehicle dynamics. The nonlinearity vector involving the product of lateral velocity and yaw rate ($V_y\psi'$) is also included to capture nonlinear effects in vehicle dynamics.

$$\frac{d}{dt}\begin{bmatrix} V_y \\ \psi \\ \psi' \\ V_x \\ V_x' \end{bmatrix} = \begin{bmatrix} -\frac{2C_f+2C_r}{mV_x} & 0 & -V_x-\frac{2C_fl_f-2C_rl_r}{mV_x} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -\frac{2C_fl_f-2C_rl_r}{I_zV_x} & 0 & -\frac{2C_fl_f^2+2C_rl_r^2}{I_zV_x} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix}\begin{bmatrix} V_y \\ \psi \\ \psi' \\ V_x \\ V_x' \end{bmatrix}$$

$$+\begin{bmatrix} \frac{2C_f}{m} \\ 0 \\ \frac{2C_f l_f}{I_z} \\ 0 \\ 0 \end{bmatrix} \delta + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ V_y \psi' \\ 0 \end{bmatrix} \quad (1)$$

Although vehicle dynamics are inherently nonlinear, mainly due to variations in tire cornering stiffness, lateral load transfer, changes in tire road friction, and aerodynamic coupling. This study employs a simplified linearized model around small slip angles and moderate speeds. These nonlinearities become significant during sharp turns or high lateral acceleration, where tire forces saturate and effective cornering stiffness decreases, potentially causing understeer or oversteer. However, incorporating such nonlinear effects would lead to high model complexity and computational cost, which can hinder the convergence of metaheuristic-based optimization processes such as GA and CSA. The selected curvature radii of 760 m and 380 m represent both wide and sharp turns, respectively, allowing the evaluation of the controller adaptability and robustness under different levels of curvature.

2.2. **Adaptive cruise control.** Based on Figure 1, ACC system tries to maintain vehicle speed based on information (speed and position) from the vehicle in front by emitting a signal to detect the position of the vehicle in front of it. Once detected, ACC then controls the actuators (gas and brake pedals) to reach a safe distance from the vehicle in front [22]. ACC uses an optimization method to increase vehicle control efficiency in order to make this happen. The equation used to find safe distance between the cars can be seen in Equation (2), in which the variables are (a) Default spacing (D_d) is minimum distance in meters between ego and lead cars when ego car is stationary; (b) Time gap (T_{gap}) represents the time interval between ego car and lead car, expressed in seconds; (c) Relative distance (D_{ego}) is a different distance between ego and lead cars in meters.

$$D_{safe} = D_d + T_{gap} D_{ego} \quad (2)$$

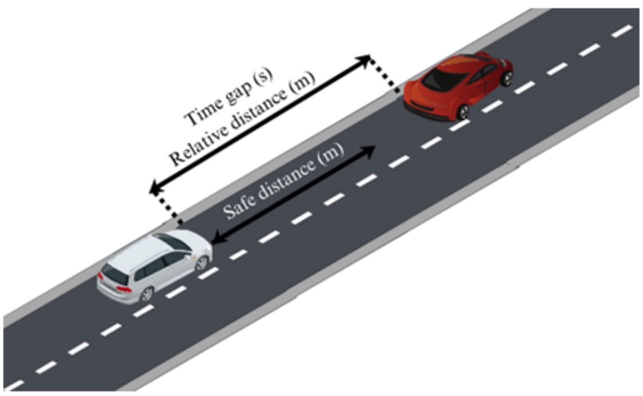


FIGURE 1. Illustration of ACC

In Figure 1, the vehicle equipped with ACC and steering control is designated as the ego car, while the vehicle ahead is referred to as the lead car. The ego car utilizes steering control to maintain its position along the centerline of the road. Conversely, the lead car, which does not have steering capabilities, remains within the confines of the curved lane.

This distinction highlights the enhanced maneuverability of the ego car compared to the lead car, emphasizing the role of steering control in navigating complex road conditions.

In traditional ACC systems, three key parameters significantly impact the performance of the system. These parameters play a crucial role in influencing two specific conditions related to acceleration. By adjusting these parameters, the system can effectively manage the acceleration behaviors of the ego car, ensuring it responds appropriately to changes in speed and distance from the lead car. This adaptability is essential for maintaining safe and efficient driving in various traffic situations.

Equation (3) uses the gain parameter G_V, which serves to adjust the vehicle's response to the difference between the specified maximum speed (V_{set}) and the longitudinal velocity (V_x).

$$u_1 = (V_{set} - V_x)G_V \tag{3}$$

Simultaneously, Equation (4) uses two gain parameters, namely G_{VX} and G_X. The gain G_{VX} functions to regulate how quickly the vehicle responds to the difference in relative speed between the ego car and the lead car (V_R). While G_X regulates the response to the difference between the desired safe distance (D_{safe}) and the actual distance (D_R). The formula is

$$u_2 = V_R G_{VX} - (D_{safe} - D_R)G_X \tag{4}$$

$$e_d = D_{safe} - D_R \tag{5}$$

When the system is in speed control mode (when D_R greater than D_{safe}), the control signal applied to the ego vehicle acceleration (a_E), is determined by selecting the minimum value between two control inputs, u_1 and u_2, as represented in Equation (6). This ensures that ego car adjusts its speed appropriately based on both velocity and distance considerations, selecting the safer of the two options.

$$a_E = \min(u_1, u_2) \tag{6}$$

On the other hand, in distance control mode (when D_R is less than D_{safe}), the control strategy only relies on the second input u_2, which is primarily focused on maintaining a safe following distance as shown in Equation (7):

$$a_E = u_2 \tag{7}$$

In the ACC testing, sharp and wide curves present different challenges for vehicle control systems. As seen in Figure 2, a sharp curve with a radius of curvature 380 meters requires the ACC system to adjust speed more aggressively to navigate the tight bend safely, while a wide curve with a radius of curvature 760 meters allows for smoother, less sudden adjustments. On smaller radius of curvature, ACC must reduce speed to avoid accidents or equalize speed with vehicles ahead, whereas in larger radius of curvature,

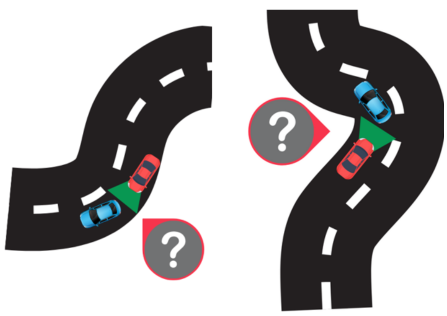


Figure 2. Curved road with different curvature

the system can maintain higher speeds without compromising safety [25]. Changes in the radius also impact vehicle stability and driving comfort; sharper curves demand a quick and accurate response to prevent slippage or loss of control, while wider curves allow for a more stable and comfortable driving experience. Testing ACC on both curve types provides essential insights into its ability to adapt to diverse road geometries, ensuring safety and efficiency in varying conditions.

2.3. **Steering control.** Steering is a system in a vehicle that allows the driver to control the direction of movement of the vehicle. In AV, especially in the steering control system when a vehicle drives a curved road, the role of steering control becomes very significant to ensure that the vehicle can navigate the curve safely and in accordance with applicable traffic rules. This system ensures the vehicle stays on the right track, increasing passenger safety and comfort. To achieve this goal, the system regulates the front-wheel steering angle. The control action is directed to drive the lateral position deviation (e_1) towards zero as indicated in Equation (8) and depicted in Figure 3. At the same time, the controller seeks to reduce the yaw-angle deviation (e_2), as shown in Equation (9). In essence, the steering command is continuously modified so the vehicle remains aligned with the intended trajectory and stays within its lane.

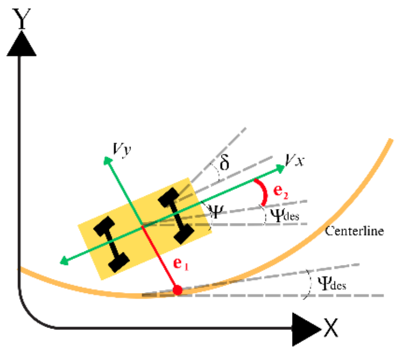


FIGURE 3. Vehicle dynamic model for steering control

First, e_2 is defined as the difference between the current vehicle orientation angle (ψ) and the desired orientation angle (ψ_{des}), as described in Equation (8):

$$e_2 = \psi - \psi_{des} \tag{8}$$

This error occurs when the vehicle direction is not in accordance with the desired trajectory direction. To correct this error, the steering control will adjust the steering angle. The vehicle orientation can be aligned again with the path it should be. Second, e_1 describes how far the lateral position of the vehicle deviates from the centerline on the lateral axis. Equation (9) shows that this lateral displacement error is influenced by V_x, yaw error (e_2), and lateral velocity (V_y).

$$e_1' = V_x \times e_2 + V_y \tag{9}$$

Therefore, given the importance of minimizing errors, a PI (Proportional-Integral) controller is applied to overcome possible steady-state errors. The PI controller is able to eliminate residual errors by taking account of the accumulation of errors over time. In situations that show constant errors, PI control will help adjust the system response to ensure that the vehicle remains on the desired path. Thus, more accurate and responsive control can be achieved, improving driving safety and comfort. The PI controller has a general equation as written in (10).

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau \tag{10}$$

The controller is responsible for correcting the deviation of the vehicle's lateral position from the desired path. When an error is detected, the controller computes the control signal $u(t)$ based on a proportional combination k_p of the current error $(e(t))$ and the integral k_i of the integral from 0 to t. This control mechanism enables timely adjustments to the vehicle's steering and speed, allowing the system to maintain the desired trajectory with improved lateral stability. The algorithm can be employed to fine-tune k_p and k_i for optimal performance, ensuring the minimization of yaw angle and lateral displacement errors.

ACC technology has effectively incorporated automation functions that assist vehicles in maintaining a safe following distance from the vehicles ahead. However, ACC primarily emphasizes speed and distance management relative to the lead vehicle, often overlooking steering control, especially when navigating curves. This creates challenges in achieving stable control between the ACC system and steering, as both involve numerous interrelated parameters. As seen in Figure 4, metaheuristic algorithms, such as GA and CSA, efficiently explore large and complex solution spaces to find optimal or near-optimal solutions. In control systems, metaheuristics can be used to set complex control parameters like PID controller gains, optimize navigation paths, or coordinate multi-agent control in autonomous vehicle systems. By employing metaheuristic methods, control systems can more effectively adapt to changing environments and dynamic operating conditions, thereby enhancing performance, reliability, and efficiency. These methods also address key limitations of conventional optimization techniques, which may become trapped in local optima or require gradient information that is difficult or impractical to obtain.

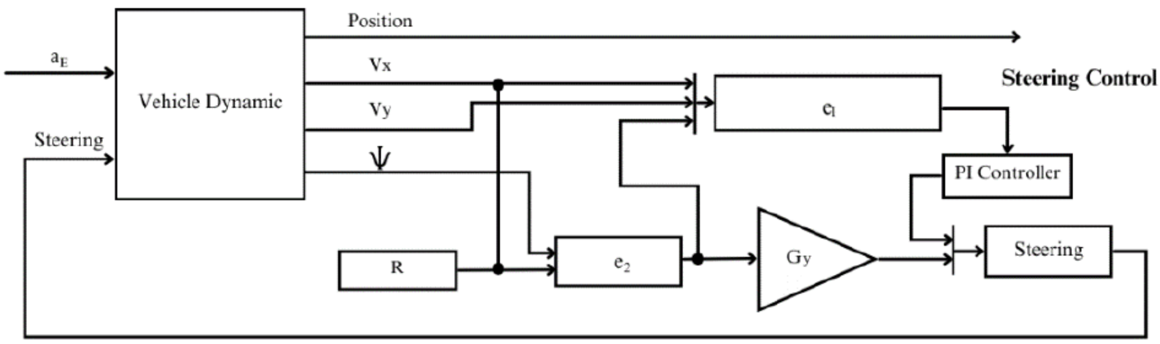


Figure 4. Block diagram of steering control

In this paper, the overall objective function was defined as the sum of IAE ACC and steering control, which includes distance error, lateral displacement error and yaw angle error. These terms were directly summed without weighting, as each error was expressed in comparable scales and holds equal significance in maintaining both distance and speed stability. The unweighted formulation also avoids introducing bias toward any specific control objective, allowing the optimization process to fairly evaluate the global performance of the controller.

3. Methods.

3.1. **Crow search algorithm.**
CSA is simulating the behavior of crows in putting and taking food. CSA mimics the behavior of a flock of crows to explore the search space, enabling it to discover optimal solutions and escape local optima [26]. This means that

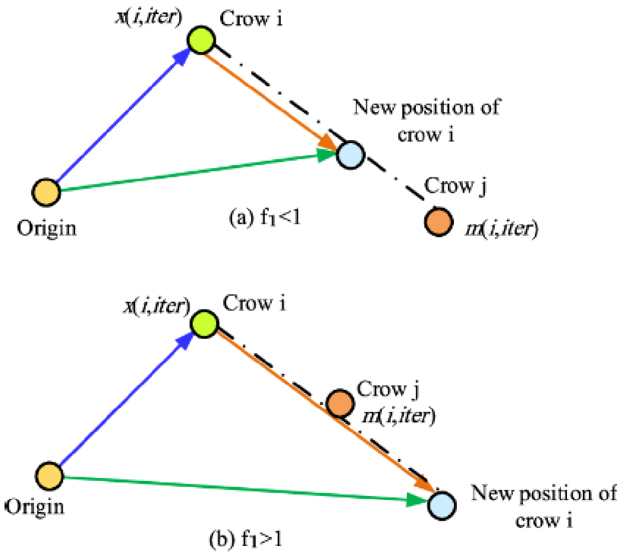


FIGURE 5. Two differences to the position of crow i if the f_l is (a) less than 1 and (b) more than 1

search optimization results will focus more on local improvements by searching in promising locations. The best position is obtained if the parameter f_l (Figure 5 shows how the parameter f_l works) and awareness probability (AP) are appropriate so that the crow can move far and can calculate the probability of moving to a new position. With just two parameters, CSA gives users more control over the search process.

The CSA begins by initializing the position of n crows in the search space. Each crow represents a solution, and their position corresponds to a specific point in the search space. This initialization step involves assigning random positions to the crows. After this, the fitness of each crow's position is evaluated using IAE, where the fitness function determines how well each position solves the optimization problem.

The memory (mem) stores each crow's previous best position, and the corresponding fitness memory (fit_{mem}) stores the fitness values of those positions. The algorithm then enters an iterative loop, running until the maximum number of iterations until $iter_{\max}$ is reached.

Within each iteration, for each crow (i), a random crow (j) is chosen for i to follow. AP is defined which determines whether crow i will follow crow j or move to a random location. If a random number (r_j) is greater than or equal to the AP, crow i updates its position based on crow j memory using

$$x_n^{i,iter+1} = x^{i,iter} + r_i \times f_l^{i,iter} \times \left(m^{j,iter} + x^{i,iter}\right) \tag{11}$$

where r_i is a random factor, and $f_l^{i,iter}$ is the flight length of the crow at iteration ($iter$). If r_j is less than AP, crow i moves to a random position within the search space.

After updating all crows' positions, the feasibility of the new positions is checked to ensure they are valid. Then, the fitness of the new positions is calculated, and memory updates are performed, where each crow's memory is updated if the new position offers a better solution than the previous one. Finally, mem and fit_{mem} are updated for the next iteration. The process continues until the maximum number of iterations is reached.

3.2. **Genetic algorithm.** After GA begins by initializing a population of size n, where each individual represents a potential solution to the problem. These individuals consist of chromosomes that contain parameters for the ACC and steering control. Next, the fitness of each individual is evaluated using an IAE, to measure the value of the solution.

Algorithm 1: Original crow search algorithm

Input: flock size, f_l, AP, maximum iteration
Ouput: Best position (solution) found

Initialize the position of n crows in the search space
Evaluate the position of the crows
  $mem = x \rightarrow$ memory initialization
  $fit_{mem} = f_t \rightarrow$ fitness memory
  for $iter < iter_{\max}$ **do**
    for $i = 1 : n$
      Choose a random crow to follow
      Define value of awareness probability
      if $r_j \geq AP^{j,iter}$
        $x_n^{i,iter+1} = x^{i,iter} + r_i \times f_l^{i,iter} \times (m^{j,iter} + x^{i,iter})$
      else
        $x_n^{i,iter+1} =$ a random position of search space
      end if
    end for
    Check the feasibility of new positions
    Determine the fitness of crows
    Memory updates for each crows
    $mem = x_n \rightarrow$ memory initialization
    $fit_{mem} = f_t \rightarrow$ fitness memory
  end for

The population memory is stored as *mem*, while the corresponding fitness values are saved in fit_{mem}.

For each iteration, the algorithm selects two individuals from *mem* to serve as $parent_1$ and $parent_2$. These parents are chosen based on their fitness values. If a random number is less than a specified crossover coefficient (c), crossover is performed by generating two children (c_1 and c_2) using a combination of the parent's chromosomes. The combination is determined by random number (α). If crossover does not occur, c_1 and c_2 are simply copies of $parent_1$ and $parent_2$. Mutation is then applied, with a random number determining whether c_1 and c_2 will be altered by introducing random values m_1 and m_2. If no mutation occurs, the children remain unchanged.

The fitness of the new solutions is evaluated, and the population memory *mem* is updated by replacing the least fit individuals with the newly generated offspring. This process ensures that the population evolves over time, gradually moving towards an optimal solution. The iteration continues until a maximum number of iterations $iter_{\max}$ is reached.

3.3. **Archived-based algorithm.** Archived-based metaheuristic optimization is an approach to optimization that uses archives to store and manage solutions found during the search process. This archive serves to maintain the diversity of solutions and avoid premature convergence, which is often a problem in traditional optimization methods [27]. In the process, potential solutions are generated and evaluated based on certain criteria, and then compared with existing solutions in the archive. Better solutions or those that increase the diversity of solutions in the archive will be added, while less effective or redundant solutions will be removed to maintain the size and quality of the archive. In contrast to original metaheuristics that may only focus on one current best solution, archive-based methods maintain a variety of good solutions, which allows exploration of

Algorithm 2: Original genetic algorithm

Input: Population size, probability of mutation, probability of crossover, maximum iteration

Ouput: Best individual (solution) found

Initialize the n population
Evaluate fitness of population
$mem = x \rightarrow$ population memory
$fit_{mem} = f_t \rightarrow$ fitness memory
for $iter < iter_{\max}$ **do**
    Select from mem to be $parent_1$ and $parent_2$
    if random number $< c$, $\alpha =$ random number
        $c_1 = \alpha * p_1 + (1 - \alpha) * p_2$
        $c_2 = \alpha * p_2 + (1 - \alpha) * p_1$
    else
        $c_1 = p_1$
        $c_2 = p_2$
    if random number $< m$
        $m_1 =$ rand;
        $m_2 =$ rand;
    else
        $m_1 = c_1$;
        $m_2 = c_2$;
    Check fitness of new solution
    Update mem the population memory
    Delete the worst in a based on fitness
    end if
end for

a broader and more robust solution space. This approach is particularly useful in Multi objective optimization, where Pareto optimal solutions can be distributed and managed more effectively, ensuring that the final solution is not only optimal in one aspect but also takes account of multiple different objectives.

In this research, the archive size was set to four, as smaller populations failed to maintain sufficient diversity, while larger ones increased computation time without notable performance gains. Thus, four represents the minimal stable population ensuring convergence and efficiency. This design choice aligns with [28], demonstrating that maintaining a compact and properly updated archived enhances convergence speed, as underlined in Algorithm 3. This can minimize computation so that it is expected to be faster in obtaining optimal individuals. Similarly, in the Archived Genetic Algorithm (AGA), optimization is carried out using only the four best individuals in each iteration, as underlined in Algorithm 4. If no archived selection is applied both algorithm operate in their original forms, namely OCSA and OGA.

4. Experiment Result.

4.1. Performance evaluation of the algorithm.

In optimization using metaheuristic algorithm, one indication that algorithm is performing well is if the objective function (IAE) value decreases periodically until it reaches its lowest point. To evaluate the algorithm, testing was carried out 20 times. The change in IAE values for each repetition in each category can be seen in Figure 6. The ACSA shows an outstanding reduction in

Algorithm 3: Archived genetic algorithm

Input: Population size, probability of mutation, probability of crossover, maximum iteration

Ouput: Best individual (solution) found

Initialize the n population
Evaluate fitness of population
Choose four best fitness
$a_{mem} = x \rightarrow$ archived memory
$fit_{mem} = f_t \rightarrow$ fitness memory
while $iter < iter_{\max}$ **do**
for $iter < iter_{\max}$ **do**
  Select from mem to be $parent_1$ and $parent_2$
  if random number $< c$, $\alpha =$ random number
    $c_1 = \alpha * p_1 + (1 - \alpha) * p_2$
    $c_2 = \alpha * p_2 + (1 - \alpha) * p_1$
  else
    $c_1 = p_1$
    $c_2 = p_2$
  if random number $< m$
    $m_1 =$ rand;
    $m_2 =$ rand;
  else
    $m_1 = c_1$;
    $m_2 = c_2$;
    Check finess of new solution
    Update a_{mem} the archived memory
    Delete the worst in a based on fitness
  end if
end for
end while

IAE values, coming to its lowest point within the 40th iteration. This accomplishment is underscored by two notable changes in IAE during the 6th and 40th iterations. In contrast, OCSA exhibits more fluctuations in IAE values and requires a longer time to achieve its optimal IAE. It needed a longer iteration to accomplish its ideal IAE, which happens within the 48th iteration. Third is OGA, and this method has better start than others. The value of IAE reduces in the 17th, 37th, 45th and make significant drop in the 73rd iteration.

In terms of iteration needed to get the lowest IAE value, AGA is the worst with 96th iteration. However, in terms of decreased value, each iteration AGA is the best. This suggests that AGA gradual and steady adjustments allow for a more balanced exploration of the search space, which is advantageous for finding the global optimum, despite requiring more iterations. In contrast, OCSA's more fluctuating behavior enables faster convergence but with a higher risk of getting trapped in local optima, making it more suitable for scenarios where quicker results are prioritized over global optimality.

As illustrated in Table 1, repetition test that is carried out indicates that ACSA and AGA are successfully reducing computational time. The archived algorithms significantly increase computational speed in reaching the 100th iteration. For wide curves, the speed improvement is 83.6% from OCSA to ACSA and 74.79% from OGA to AGA, while for

Algorithm 4: Archived crow search algorithm

Input: flock size, f_l, AP, maximum iteration
Ouput: Best position (solution) found

Initialize the position of n crows in the search space
Evaluate the position of the crows
$mem = x \rightarrow$ memory initialization
$fit_{mem} = f_t \rightarrow$ fitness memory
while $iter < iter_{\max}$ **do**
  for $i = 1 : n$
    Choose a random crow to follow
    Define value of awareness probability
    if $r_j \geq AP^{j,iter}$
      $x_n^{i,iter+1} = x^{i,iter} + r_i \times f_l^{i,iter} \times (m^{j,iter} + x^{i,iter})$
    else
      $x_n^{i,iter+1}$ = a random position of search space
    end if
  end for
  Check the feasibility of new positions
  Determine the fitness of crows
  Memory updates for each crow
  Memory update for archived flock
  $mem = x_n \rightarrow$ memory new crow
  $fit_{mem} = f_t \rightarrow$ fitness memory
  Sort fit_{mem}, mem, x, and f_t from best to worst
  x_n, x, f_t, mem, and fit_{mem} take only the first 4 rows.
end while

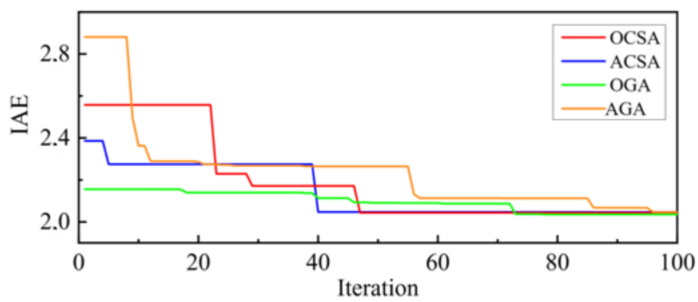


FIGURE 6. IAE changes after tuning

sharp curves, the improvements are 81.8% and 77.09%, respectively. Computational time is one of the indicators to evaluate the performance of algorithm. Besides the computational times, the parameters k_p, k_i, G_V, G_X and G_{VX} are also important as they directly affect the accuracy and stability of both systems.

Analyzing the results in Table 2, the Baseline values represent the system's performance using unoptimized parameters, where the controller configuration was set with $k_p = 0.2$, $k_i = 0.1$, $G_Y = 0.2$, $G_V = 0.5$, $G_X = 0.2$, and $G_{VX} = 0.4$. These initial parameters had not yet undergone any optimization using the algorithm. The results are divided into two categories: wide curves and sharp curves, representing different driving scenarios.

For both types of curves, it becomes evident that after optimization, the IAE values improve significantly. Specifically, the best generation for the OCSA achieves slightly better

Table 1. Computational time in minutes

Road	Category	ACSA	OCSA	OGA	AGA
Wide curve	Fastest	14	79	68	17
	Slowest	20	81	70	20
	Average	17.5	80.5	69.2	18.7
Sharp curve	Fastest	14	78	67	17
	Slowest	15	80	68	18
	Average	14.4	79.1	67.2	17.5

Table 2. IAE change after optimization

Category	Wide curve			
	Steering	ACC	Addition	Mean
Baseline	0.3591	18.5967	18.9558	
OCSA	0.0483	1.9674	2.0806	2.1851
ACSA	0.0737	1.9952	2.1375	2.2788
OGA	0.0539	1.9823	2.1021	2.2128
AGA	0.0882	2.0044	2.1278	2.2311
Category	Sharp curve			
	Steering	ACC	Addition	Mean
Baseline	0.3591	18.5967	18.9558	
OCSA	0.0802	5.0250	5.1271	5.2330
ACSA	0.0759	5.0212	5.1536	5.2394
OGA	0.0668	5.0244	5.1422	5.2102
AGA	0.0694	5.0198	5.1388	5.2126

IAE values than the OGA in both wide and sharp curves. In wide curves, OCSA shows superior performance compared to the other methods, though it also has the longest computational time. A similar trend is observed for sharp curves, where OCSA still performs the best.

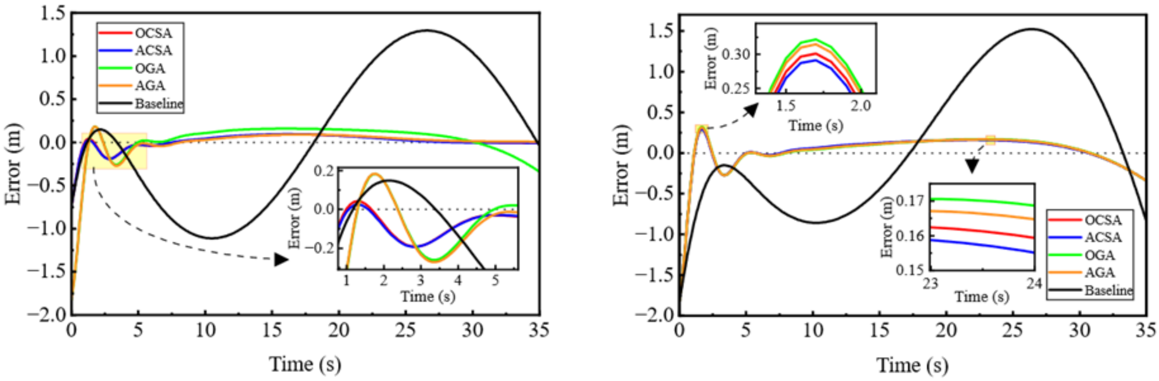
When comparing the coincidence level between the original and archived algorithms, OCSA and ACSA on the wide curve yielded mean IAE values of 2.1851 and 2.2788, with 95% confidence intervals of (2.10-2.27) and (2.19-2.37), respectively. On the sharp curve, the mean IAE values were 5.2330 for OCSA and 5.2394 for ACSA, with confidence intervals of (5.15-5.32) and (5.15-5.33). The close proximity of these intervals indicates a high coincidence level, confirming that the archived algorithm maintains equivalent control accuracy while improving computational efficiency. Similarly, OGA and AGA on the wide curve produced mean IAE values of 2.2128 and 2.2311, with confidence intervals of (2.13-2.30) and (2.14-2.32), and on the sharp curve 5.2102 and 5.2126, with confidence intervals of (5.12-5.30) and (5.12-5.31).

When comparing the two archived algorithms, GA performs slightly better than CSA, especially in terms of computational efficiency. The archived method in GA allows the algorithm to run more effectively and faster than when using CSA. Nevertheless, the consistency across the four methods in both ACC and steering control, for both sharp and wide curves, strongly suggests that the optimization process is robust and effective across different driving conditions.

The standard deviation of the optimized parameters for both the ACC and steering control systems remains consistently low, often approaching zero, as illustrated in Table 3. This demonstrates that the optimized parameters converge towards a stable and optimal

Table 3. Standard deviation of optimized parameter

Category	Sharp curve					
	G_V	G_X	G_{VX}	k_p	k_i	G_Y
OCSA	0.136	0.014	0.013	0.224	0.210	0.025
ACSA	0.269	0.290	0.057	0.238	0.278	0.211
OGA	0.265	0.067	0.014	0.145	0.122	0.198
AGA	0.202	0.029	0.010	0.069	0.152	0.234
Category	**Wide curve**					
	G_V	G_X	G_{VX}	k_p	k_i	G_Y
OCSA	0.186	0.172	0.157	0.216	0.201	0.155
ACSA	0.260	0.245	0.231	0.290	0.275	0.276
OGA	0.113	0.098	0.083	0.142	0.127	0.121
AGA	0.039	0.024	0.010	0.068	0.054	0.225



(a) (b)

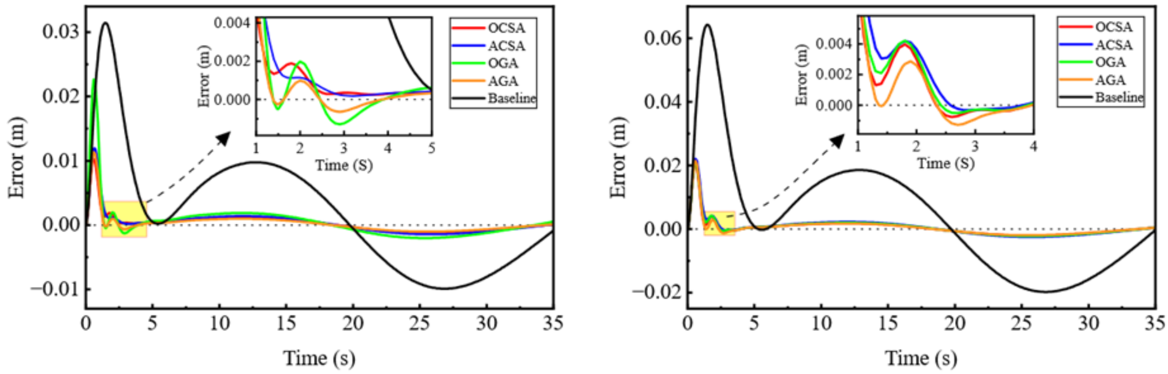
Figure 7. Compared distance error in (a) wide curve and (b) sharp curve

solution. The lower standard deviation values reflect reduced variability, leading to a more robust optimization process, and increasing precision and confidence in the model's overall performance. The stability of these parameters is key to ensuring the reliability and effectiveness of the control mechanisms, resulting in a more robust and dependable system.

Performance Evaluation of ACC. Figure 7 illustrates the distance error, defined as the difference between D_R and D_{safe} in the ACC system over time. The Baseline model (black line) shows the highest overshoot and the longest stabilization time, indicating less optimal control performance. In contrast, the optimized algorithms (OCSA, ACSA, OGA, and AGA) show smaller overshoots and faster convergence toward zero error, indicating improved control stability and tracking accuracy. A magnified view of the first five seconds in Figure 7(a) highlights more pronounced oscillations in the Baseline and AGA responses, reaching up to 0.18 m, while OCSA and ACSA achieve quicker error correction of approximately 0.04 m and exhibit smoother responses. Within the same interval, the Baseline reaches its maximum error of 0.15 m. In Figure 7(b), during the first 5 seconds, particularly around 1.7 s, OGA records the highest error of 0.32 m, whereas ACSA achieves the lowest error of 0.29 m. All algorithms show a transient peak before stabilizing, indicating that the control systems effectively compensate for the initial deviation.

TABLE 4. Settling time and steady-state error of the distance error

Road	Category	ACSA	OCSA	OGA	AGA	Baseline
Wide curve	S_t	3.77	3.75	3.45	3.63	29.98
	E_{ss}	0.01	0.005	0.19	0.01	0.39
Sharp curve	S_t	3.99	3.99	4.04	4.03	29.98
	E_{ss}	0.180	0.180	0.1783	0.1788	0.39



(a) (b)

FIGURE 8. Compared lateral displacement error in (a) wide curve and (b) sharp curve

TABLE 5. Settling time and steady-state error of the displacement error

Road	Category	ACSA	OCSA	OGA	AGA	Baseline
Wide curve	S_t	1.7	2.1	1.19	1.5	32.38
	E_{ss}	0.00028	0.00027	0.00032	0.00018	0.00272
Sharp curve	S_t	1.29	1.09	1.19	1.09	33.67
	E_{ss}	0.00040	0.00033	0.00035	0.00028	0.00636

As summarized in Table 4, both the steady-state error (E_{ss}) and settling time (S_t) analyses show that the adaptive algorithms substantially outperform the Baseline model under both curve conditions. In the wide curve scenario, OCSA reduces S_t by approximately 87.5% (from 29.98 s to 3.75 s) and E_{ss} by more than 98%, confirming its superior stability and fast dynamic response. Under the sharp curve condition, OGA achieves an 86.5% reduction in S_t and a 54% decrease in E_{ss}, reaching $E_{ss} = 0.1783$ m and $S_t = 4.04$ s compared to the Baseline ($E_{ss} = 0.39$ m, $S_t = 29.98$ s). Overall, OCSA demonstrates the most effective control performance in wide curve conditions, while OGA offers higher stability and tracking precision in sharp curve conditions, both providing significantly improved steady-state accuracy and convergence speed relative to the baseline.

Figure 8 demonstrates the significant superiority of all adaptive control algorithms over the Baseline algorithm in reducing the lateral displacement error and enhancing the transient response performance. As shown in Table 5, the AGA achieves the lowest steady-state error (E_{ss}) values in both scenarios, 0.00018 m for the wide curve and 0.00028 m for the sharp curve. Quantitatively, AGA achieves up to 93.4% and 95.6% improvements in steady-state accuracy for wide and sharp curve conditions, respectively. Moreover, the proposed algorithms exhibit considerably shorter S_t than the Baseline, which requires 32.38 s in the wide curve and 33.67 s in the sharp curve. In contrast, AGA reaches steady state within 1.5 s and 1.09 s, respectively, confirming the robustness and efficiency of the adaptive control strategies in enhancing both stability and responsiveness.

5. **Conclusion.** This article presents encouraging outcomes from archived optimization algorithms in enhancing ACC with steering control on curved roads. While the original algorithms (OCSA and OGA) deliver slightly lower IAE values, only about 4% and 1%, respectively. The archived versions (ACSA and AGA) achieve a substantial improvement in computational efficiency, reducing processing time by 77-83% without notable performance degradation.

Although these advancements, there are still considerable challenges in transitioning from simulations to real-world hardware. The implementation requires special attention to vehicle hardware and driver safety to guarantee optimal performance and safety in different driving environments. Future work will focus on implementing the archived algorithms in embedded automotive controllers, validating their performance under sensor noise and communication delays, and expanding to multi-vehicle cooperative ACC with real-time safety constraints.

Acknowledgment. This work is supported by Directorate of Research, Technology and Community Service, Directorate General of Higher Education. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] P. Weerapong, M. Katahira, K. Hashikura, M. A. S. Kamal, I. Murakami and K. Yamada, Modal analysis for evaluating the transmission of vertical vibrations in a wheelchair-occupant model with foam-based seat cushion, *International Journal of Innovative Computing, Information and Control*, vol.19, no.6, pp.1933-1952, DOI: 10.24507/ijicic.19.06.1933, 2023.

[2] X. Li, Trade-off between safety, mobility and stability in automated vehicle following control: An analytical method, *Transportation Research Part B: Methodological*, vol.166, pp.1-18, DOI: 10.1016/j.trb.2022.09.003, 2022.

[3] F. Jiménez, J. E. Naranjo, J. J. Anaya, F. García, A. Ponz and J. M. Armingol, Advanced driver assistance system for road environments to improve safety and efficiency, *Transportation Research Procedia*, vol.14, pp.2245-2254, DOI: 10.1016/j.trpro.2016.05.240, 2016.

[4] D. Zhang, Q. Xiao, J. Wang and K. Li, Driver curve speed model and its application to ACC speed control in curved roads, *International Journal of Automotive Technology*, vol.14, no.2, pp.241-247, DOI: 10.1007/s12239-013-0027-x, 2013.

[5] F. Ding and H. Jin, On the optimal speed profile for eco-driving on curved roads, *IEEE Transactions on Intelligent Transportation Systems*, vol.19, no.12, pp.4000-4010, DOI: 10.1109/TITS.2018.2795602, 2018.

[6] B. McAuliffe, A. Raeesi, M. Lammert, P. Smith, M. Hoffman and D. Bevly, Impact of mixed traffic on the energy savings of a truck platoon, *SAE Int. J. Adv. Curr. Pract. Mobil.*, vol.2, no.3, DOI: 10.4271/2020-01-0679, 2020.

[7] A. Weißmann, D. Görges and X. Lin, Energy-optimal adaptive cruise control combining model predictive control and dynamic programming, *Control Eng. Pract.*, vol.72, pp.125-137, DOI: 10.1016/j.conengprac.2017.12.001, 2018.

[8] W. Du, Stability analysis and control of a car-following model in a connected and autonomous environment, *International Journal of Innovative Computing, Information and Control*, vol.19, no.3, pp.671-686, DOI: 10.24507/ijicic.19.03.671, 2023.

[9] L. Yu and R. Wang, Researches on adaptive cruise control system: A state of the art review, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol.236, nos.2-3, pp.211-240, DOI: 10.1177/09544070211019254, 2022.

[10] J. Ma, K. Qian and Z. Gong, Optimization of ACC system spacing policy on curved highway, *AIP Conference Proceedings*, 020169, DOI: 10.1063/1.4982534, 2017.

[11] M. E. Rodriguez, *Adaptive Cruise Control Using Novel Sensor Fusion Configurations*, Master Thesis, California State Polytechnic University, Pomona, 2022.

[12] S. Cheng, L. Li, M. Mei, Y. Nie and L. Zhao, Multiple-objective adaptive cruise control system integrated with DYC, *IEEE Transactions on Vehicular Technology*, vol.68, no.5, pp.4550-4559, DOI: 10.1109/TVT.2019.2905858, 2019.
[13] H. Wang, Y. Sun, Z. Gao and L. Chen, Extension coordinated multi-objective adaptive cruise control integrated with direct yaw moment control, *Actuators*, vol.10, no.11, 295, DOI: 10.3390/act10110295, 2021.
[14] M. G. Plessen, D. Bernardini, H. Esen and A. Bemporad, Spatial-based predictive control and geometric corridor planning for adaptive cruise control coupled with obstacle avoidance, *IEEE Transactions on Control Systems Technology*, vol.26, no.1, pp.38-50, DOI: 10.1109/TCST.2017.2664722, 2018.
[15] K. Shin, J. Choi and K. Huh, Adaptive cruise controller design without transitional strategy, *International Journal of Automotive Technology*, vol.21, no.3, pp.675-683, DOI: 10.1007/s12239-020-0065-0, 2020.
[16] A. F. Idriz, A. S. A. Rachman and S. Baldi, Integration of auto-steering with adaptive cruise control for improved cornering behaviour, *IET Intelligent Transport Systems*, vol.11, no.10, pp.667-675, DOI: 10.1049/iet-its.2017.0089, 2017.
[17] K. Alomari, R. Mendoza, S. Sundermann, D. Goehring and R. Rojas, Fuzzy logic-based adaptive cruise control for autonomous model car, *Proc. of the International Conference on Robotics, Computer Vision and Intelligent Systems*, pp.121-130, DOI: 10.5220/0010175101210130, 2020.
[18] P. Wang, Y. Lin and Z. Wang, An integrated multi criteria group decision-making model applying fuzzy TOPSIS-CRITIC method with unknown weight information, *International Journal of Innovative Computing, Information and Control*, vol.18, no.3, pp.815-836, DOI: 10.24507/ijicic.18.03.815, 2022.
[19] N. C. Basjaruddin, K. Kuspriyanto, D. Saefudin and I. K. Nugraha, Developing adaptive cruise control based on fuzzy logic using hardware simulation, *International Journal of Electrical and Computer Engineering (IJECE)*, vol.4, no.6, DOI: 10.11591/ijece.v4i6.6734, 2014.
[20] P. Mahadika, A. Subiantoro and B. Kusumoputro, Neural network predictive control approach design for adaptive cruise control, *International Journal of Technology*, vol.11, no.7, 1451, DOI: 10.14716/ijtech.v11i7.4592, 2020.
[21] D. Prastiyanto, E. Apriaskar, R. F. Ibrahim, A. Rumanda, A. A. Manaf and I. Amelia, Adaptive cruise control tuned by genetic algorithm for safe distance of automated vehicle, *IOP Conf. Ser. Earth Environ. Sci.*, vol.1203, no.1, 012026, DOI: 10.1088/1755-1315/1203/1/012026, 2023.
[22] E. Özkaya, H. Arslan and O. T. Şen, Particle swarm optimization method based controller tuning for adaptive cruise control application, *Gazi University Journal of Science*, vol.34, no.2, pp.517-527, DOI: 10.35378/gujs.762103, 2021.
[23] N. Wiesner, J. Sheppard and B. Haberman, Autonomous vehicle control using particle swarm optimization in a mixed control environment, *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp.2877-2884, DOI: 10.1109/SSCI47803.2020.9308567, 2020.
[24] S. Tak, S. Kim, H. Yu and D. Lee, Analysis of relationship between road geometry and automated driving safety for automated vehicle-based mobility service, *Sustainability*, vol.14, no.4, 2336, DOI: 10.3390/su14042336, 2022.
[25] Z. Yang, R. He, S. Zhang and J. Wu, Research on adaptive cruise control algorithm considering road conditions, *IET Intelligent Transport Systems*, vol.15, no.12, pp.1478-1493, DOI: 10.1049/itr2.12108, 2021.
[26] A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, *Comput. Struct.*, vol.169, pp.1-12, DOI: 10.1016/j.compstruc.2016.03.001, 2016.
[27] D. Prastiyanto et al., The adaptive cruise control for curved roads using archived crow search algorithm, *International Journal of Transport Development and Integration*, vol.8, no.1, pp.179-188, DOI: 10.18280/IJTDI.080117, 2024.
[28] C. Bian, S. Ren, M. Li and C. Qian, An archive can bring provable speed-ups in multi-objective evolutionary algorithms, *International Joint Conference on Artificial Intelligence (IJCAI)*, pp.6905-6913, 2024.

Author Biography

Dhidik Prastiyanto is an associate professor in Signal and System at the Department of Electrical Engineering, Universitas Negeri Semarang (UNNES), Indonesia. He obtained his Bachelor's and Master's degrees in Electrical Engineering from Gadjah Mada University, Indonesia, in 2000 and 2004, respectively, and completed his Doctoral degree in Electrical Engineering and Information Technology at the Karlsruhe Institute of Technology (KIT), Germany, in 2015. He has published his works in numerous international journals and conference proceedings across the fields of control systems, optimization, and applied electromagnetics. His research interests include signals and systems, intelligent control, metaheuristic optimization, microwave heating systems, and advanced automotive control applications.

Esa Apriaskar is a lecturer at the Faculty of Engineering, Universitas Negeri Semarang, Indonesia. He completed his Bachelor's degree at Diponegoro University, Indonesia, in 2013 and his Master's degree at Bandung Institute of Technology, Indonesia, in 2017, both in the field of Electrical Engineering. He is currently pursuing his doctoral studies in Automatic Control and Systems Engineering at the University of Sheffield, UK. His expertise lies in control engineering and intelligent systems, and he is actively involved in various research activities in these fields. He has contributed to several collaborative projects focusing on intelligent control and system optimization and continues to expand his work through international research engagement.

Feddy Setio Pribadi is a lecturer at the Faculty of Engineering, Universitas Negeri Semarang, Indonesia. He completed his Bachelor's degree at Universitas Negeri Semarang, Indonesia, in 2002 and obtained his Master's and Doctoral degrees from Gadjah Mada University, Indonesia, in 2007 and 2019, respectively, both in the field of Informatics. His research interests include artificial intelligence, machine learning, and intelligent information systems. He has been involved in various academic and research activities related to AI applications in education, data analysis, and smart computing technologies. He has also contributed to several collaborative research initiatives focusing on the development of intelligent computational models and their application in real-world engineering problems.

Imam Khoirul Akbar received his Bachelor's degree in Electrical Engineering from Universitas Negeri Semarang, Indonesia, in 2024. He is a member of the Control System Research Group, where he has been involved in research activities related to robotics, autonomous vehicles, control systems, and artificial intelligence. His strong interest in robotics led him to serve as the leader of the university's robotics student club, where he actively contributed to project development and competition preparation. He is currently employed at a tire manufacturing company in Indonesia, where he applies his engineering background to supporting industrial automation and control-related operations.

Ilya Amelia completed her undergraduate degree in Electrical Engineering at Universitas Negeri Semarang, Indonesia, in 2024, where she conducted research in the area of control systems. As a member of the Control System Research Group, she has been involved in studies exploring the application of metaheuristic algorithms – such as the crow search algorithm and particle swarm optimization – for adaptive cruise control and intelligent vehicle technologies. Her research interests include control systems, intelligent vehicle systems, and the development of optimization-based approaches for advanced automotive applications. She is currently working as a freelancer in the field of control system application in Cirebon, Indonesia.

Akhyar Abdillah Manaf received his Bachelor's degree in Electrical Engineering from Universitas Negeri Semarang, Indonesia, in 2024. He is a member of the Control System Research Group, where he has participated in various academic and research activities related to intelligent systems and modern control applications. His research interests include artificial intelligence, control systems, and the development of computational approaches for engineering problem-solving. He is currently employed as a qualification engineer at PT TEC Indonesia, where he contributes to system evaluation, testing, and quality assurance processes within the company's engineering operations.