İSTANBUL UNIVERSITY
CERRAHPAŞA

ORIGINAL ARTICLE
Electrica 2025; 25: 1-16

Comparative Analysis of Metaheuristic Algorithms in PID-Based Vehicle Cruise Control Systems

Furkan Heybetli[1], Yahya Danayiyen[1], Aleyna Begüm Taşdemir[2], Şule Şenyiğit[1]

[1]Department of Electrical and Electronics Engineering, Karadeniz Technical University Faculty of Engineering, Trabzon, Türkiye
[2]Department of Civil Engineering, Karadeniz Technical University Faculty of Engineering, Trabzon, Türkiye

Cite this article as: F. Heybetli, Y. Danayiyen, A. Begüm Taşdemir and Ş. Şenyiğit, "Comparative analysis of metaheuristic algorithms in PID-based vehicle cruise control systems," *Electrica*, 25, 0051, 2025. doi: 10.5152/electrica.2025.24051.

WHAT IS ALREADY KNOWN ON THIS TOPIC?

- *The performance of PID-based vehicle cruise control (VCC) systems heavily depends on the optimization algorithm used for tuning the controller parameters. Traditional PID tuning methods, such as Ziegler-Nichols, often suffer from slow response times, excessive overshoot, and limited stability, particularly in nonlinear and dynamic systems. Metaheuristic algorithms have been widely applied to improve PID tuning, providing more flexible and adaptive parameter selection, but their effectiveness varies depending on system conditions and disturbances.*

Corresponding author:
Yahya Danayiyen

E-mail:
yahya@ktu.edu.tr

Received: March 20, 2025
Revision Requested: March 31, 2025
Last Revision Received: March 20, 2025
Accepted: March 31, 2025
Publication Date: April 28, 2025

DOI: 10.5152/electr.2025.25051

ABSTRACT

This study presents a comparative analysis of five metaheuristic algorithms for optimizing Proportional Integral Derivative (PID) controller parameters in a vehicle cruise control system. The selected algorithms include the Rain Optimization Algorithm (ROA), Student Psychology-Based Optimization (SPBO), Sine Cosine Algorithm (SCA), Gradient-Based Optimization (GBO), and Grey Wolf Optimization (GWO). The PID parameters are optimized using four common performance criteria: Integral Time Absolute Error (ITAE), Integral Absolute Error (IAE), Integral Square Error (ISE), and Integral Time Square Error (ITSE). Unlike conventional tuning methods, metaheuristic approaches provide more flexible and adaptive parameter selection, enhancing the control system's robustness. The study analyses the algorithms under several internal and external disturbances, including road incline, passenger mass, and measurement noise. The findings demonstrate that no singular algorithm is universally superior; instead, efficacy is contingent upon the particular system conditions and control objectives. The Sine Conine Algorithm (SCA) has superior overall performance under typical settings, while GWO attains the quickest response time for slope fluctuations. Conversely, ROA reduces overshoot, rendering it appropriate for applications that necessitate stability. Additionally, the proposed methods are compared with the classical Ziegler-Nichols tuning approach, demonstrating the advantages of metaheuristic algorithms in terms of reduced overshoot, faster settling time, and improved robustness. The findings indicate that the SCA with the ITAE criterion yields superior performance under varying disturbance conditions. However, the fastest system response under different slope conditions is achieved with the parameter values determined by the GWO employing the ITSE criterion. The system model is designed in a MATLAB environment and the detailed comparative results are provided in figures and tables.
***Index Terms*—** Metaheuristic algorithms, PID optimization, rain optimization algorithm, vehicle cruise control

I. INTRODUCTION

The introduction of autonomous driving technology has brought about a significant improvement in the vehicle cruise control (VCC) system, which is a standard element in modern vehicle types such as gasoline, diesel, and electric vehicles. Modern cruise control systems have developed to include advanced sensors, actuators, and control algorithms, allowing vehicles to autonomously adjust speed and maintain safe distances from other vehicles on the road. It was originally designed as a mechanism to maintain a constant speed set by the driver [1]. In this context, cruise control systems stand out as a critical technology that allows drivers to effectively control vehicle speed, thereby improving driver safety and the safety of other road users. These systems are intelligent control systems that allow drivers to travel at a certain speed, and at the same time, maintain this speed by automatically intervening [2–5]. It is also desirable for the speed transitions to be smooth and for the speed to remain at the desired reference during normal travel. To perform a smart and safe cruise control operation in line with the desired objectives, one must design a suitable controller. Some advanced control methods, such as model predictive control [6–10], robust and optimal control [11, 12], fuzzy logic [13, 14], and learning-based [15, 16] control methods have been implemented in the literature. In addition to these advanced control methods, classical PID control still maintains its importance thanks to its easier applicability. However, achieving optimal performance from PID controllers often requires careful tuning

WHAT DOES THIS STUDY ADD ON THIS TOPIC?

- *This study presents a comparative analysis of five metaheuristic algorithms (ROA, SPBO, SCA, GBO, and GWO) for PID parameter optimization in a VCC system. The study evaluates the robustness of these algorithms under various internal (sensor noise) and external (road incline, passenger weight) disturbances, providing insights into their adaptability to real-world driving conditions. The results demonstrate that while no single algorithm is universally superior, the Sine Cosine Algorithm (SCA) achieves the best overall performance under normal conditions, while Grey Wolf Optimization (GWO) provides the fastest response time for slope variations. This study contributes to the field by offering a comprehensive comparison of optimization techniques, guiding the selection of suitable algorithms for robust and efficient vehicle speed control.*

of their parameters. Although manual tuning is straightforward to understand, it is rather arbitrary and may not necessarily produce the best controller performance, particularly for complex or nonlinear systems. The Ziegler-Nichols method has drawbacks, most notably a tendency to lead to increased overshoot, rise time, and settling time when the system is operating, especially when dealing with nonlinear systems that have a variety of parameters, a large amount of inertia, and a significant amount of delay. Due to the inherent nonlinearity of the controlled system, these constraints can impact system performance and result in poor responses at the output [17–21]. In recent years, metaheuristic algorithms have been successfully applied to adjust the PID controller parameters in various control systems, including industrial processes, robotics, and mechatronic systems. These algorithms search for optimal PID parameters by iteratively evaluating the performance of different parameter combinations using a fitness function [22]. With these developments, significant progress has been made in PID controller (and its derivatives) based vehicle speed control systems by applying metaheuristic algorithms [23–35].

In [23], the authors propose an optimal tuning method for a Proportional-Integral-Derivative-Accelerated controller for the VCC system using the modified bat algorithm (MBA), a powerful metaheuristic optimization technique. [24] proposes the design and implementation of an optimized PID controller for an adaptive cruise control system. Different objective functions, including ITE, ITAE, and ITSE, are employed for optimizing the controller using Particle Swarm Optimization (PSO) and teacher-learning-based optimization algorithms (TLBO). A linearized model of the vehicle system, including external disturbances such as road incline and air friction, is considered in the design of a mathematical model of the VCC system. A PSO-based optimizer for the third-order vehicle mathematical model is proposed in [25], aquila optimizer via chaotic local search, and modified opposition-based learning strategies (CmOBL-AO). A PIDD2 (proportional-integral-derivative plus second-order derivative) controller for a linearized model of the VCC system is proposed in [26], the weighted mean of vectors (INFO) algorithm-based $PI^{\lambda}DND^{2}N^{2}$ (a derivative of the PID controller) controller is proposed in [27] prioritizing road safety. For a simple vehicle model with only throttle pedal action, genetic algorithm (GA), memetic algorithm, and mesh adaptive direct search methods are used to optimize the PID controller in [30]. The performances are compared considering overshoot, settling time, and steady-state error. [31] explores the efficacy of a PID controller optimized by GA for automobile cruise control systems, considering the high non-linearity of the system. The study evaluates transient and steady-state performance metrics, revealing that the GA-tuned PID controller exhibits the shortest rise time. An Ant Lion Optimizer (ALO) tuned PID controller for the first time for an automobile VCC was introduced to enhance the robustness of the system. This emphasizes the exploration capabilities of the ALO algorithm in searching the potential regions of the parameter space, considering road incline and air resistance as external disturbances in [32].

Vehicle dynamics and parameters must be precisely understood for automobile VCC systems to operate at an accurate speed. Using data-driven techniques like system identification and machine learning algorithms, researchers have developed methods for parameter estimation and vehicle model acknowledgment. These methods make it possible to create precise vehicle models that can enhance the efficiency of speed control in a variety of vehicle types. Furthermore, to guarantee that the VCC system operates satisfactorily, the controller design process must consider both internal and external disturbances [36, 37].

Upon reviewing the literature, it becomes evident that linear vehicle dynamics are frequently employed in the construction of metaheuristic algorithm-based PID controllers. It has been found that research using these dynamics lacks an examination of how different types of disturbances affect controller performance [23–25, 29, 31]. While these controllers only have a restricted set of parameter values, it is considered that they are not robust, and that vehicle speed control is going to fail as intended when dealing with various disturbance types. Based on these findings, the performance of the PID controller in this study is optimized by considering both internal disturbances that represent errors, such as measurement and modeling, and external disturbances like passenger weight and road incline. The contribution of this study can be summarized as follows:

A key novelty of this study is the application of three relatively unexplored metaheuristic algorithms, Student PsychologyBased Optimization (SPBO), Rain Optimization Algorithm (ROA), and Sine Cosine Algorithm (SCA), for PID parameter tuning in VCC systems. To the best of the authors' knowledge, these algorithms have not been previously utilized in this context. While conventional methods such as GA and PSO have been extensively studied in PID tuning, this study investigates alternative optimization techniques that offer diverse search mechanisms and improved adaptability to nonlinear system dynamics. By integrating these novel algorithms into the control framework, the study provides new insights into their performance in handling real-world disturbances such as road incline variations, passenger weight fluctuations, and internal noise effects. The comparative analysis demonstrates that these approaches can achieve competitive, and in some cases superior, control performance in terms of rise time, settling time, and overshoot reduction. This contribution extends the applicability of metaheuristic optimization in VCC systems and opens new possibilities for further exploration in automotive control engineering.

The study demonstrates the robustness of the PID controllers optimized by these algorithms when subjected to external disturbances such as road incline and passenger weight, as well as internal disturbances like Gaussian and sinusoidal inputs. This aspect is crucial for real-world applications where such disturbances are common.

The findings indicate that the SCA with the ITAE criterion yields superior performance under varying disturbance conditions, while the GWO algorithm achieves the fastest system response under different slope conditions. This comparative analysis helps in understanding which optimization method is most effective for specific scenarios.

The system model is designed and implemented in the MATLAB environment, providing a practical framework for future research and development in VCC systems. The detailed comparative results presented in figures and tables enhance the clarity and applicability of the findings.

The rest of this paper is organized as follows: Section II covers the mathematical modeling of the VCC system and PID control. Section III describes the selected metaheuristic algorithms. Section IV details the implementation, performance evaluation, and comparative analysis. Section V discusses the results, including comparisons with classical PID tuning. Finally, Section VI presents conclusions and future research directions.

II. METHODS

A. Mathematical Modelling of Automobile Cruise Control System

By maintaining the speed of a vehicle at the driver-specified speed, the VCC system generates the appropriate amount of pedal input. The system suffers from external disturbances such as wind resistance and road interruptions. The VCC system determines the error rate brought on by these disturbances and sends the control signal to the appropriate actuator at the desired speed. Accelerator pedal pressure controls the speed. As a result, when the VCC is engaged, the pedal actuator functions as though the driver is pressing it. Fig. 1 shows the general block diagram of the VCC system that is considered for the study. The mathematical equations of the vehicle are given in (1), (2), and (3), respectively.

$$F_t = F_i + F_s + F_r + F_a \tag{1}$$

$$F_t = \frac{T_e . i_x . i_0 . \eta_d}{r_{wd}} \tag{2}$$

$$r_{wd} = 0.98 . r_{ws} \tag{3}$$

where F_t (traction force) is a positive force trying to move the vehicle forward. All other forces are resistive, negative forces that oppose the movement and try to slow the vehicle down. T_e [Nm] is the engine torque, i_x is the transmission gear ratio, i_0 is the final gear ratio, η_d is the driveline efficiency, r_{wd} [m] is the dynamic wheel radius, and r_{ws} is the static wheel radius. The dynamic wheel radius is the wheel radius when the vehicle is in motion. Because the tire undergoes slight compression while in motion, this value is marginally smaller than r_{ws}. The force that occurs during vehicle acceleration is the force of inertia (F_i) and is expressed in (4).

$$F_i = m_v . a_v \tag{4}$$

The force F_i is obtained by multiplying the total vehicle mass (kg) m_v by the vehicle acceleration a_v. The force that occurs when the vehicle is traveling on an upslope or downslope is the road slope force F_s and is expressed in (5).

$$F_s = m_v . g . \sin(a_s) \tag{5}$$

F_s is one of the resistance forces affecting the vehicle's motion. This force depends on the slope angle of the road and the mass of the vehicle. Here, g represents the acceleration due to gravity [m/s^2], and a_s represents the slope angle of the road. The road resistance force usually increases as the speed of the vehicle increases or as the gradient of the road increasesThat is, this force is a resistance that makes it difficult for the vehicle to move forward as defined in (6).

$$F_r = m_v . g . c_r . \cos(a_s) \tag{6}$$

Fr (road resistance force) represents the resistances that affect the movement of the vehicle as defined in (6). Where F_r is the road resistance forcewhiich represents the resistances that affect the movement of the vehicle. . Here, c_r represents the coefficient of road resistance. Aerodynamic resistance force (F_a) whis is defined as in (7), is the resistance force caused by the effect of air when a vehicle moves.

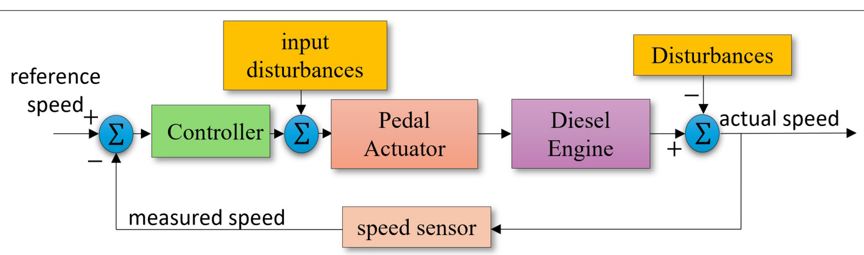


Fig. 1. General representation of the VCC system.

$$F_a = \frac{1}{2}.\rho.c_d.A.v^2 \quad (7)$$

The relationship between the speed of the vehicle (v) and the forces is given in (8).

$$v = \frac{1}{m_v}\int\left[F_t - \left(F_s + F_r + F_a\right)\right]dt \quad (8)$$

Where ρ is the air density at 20°C [kg/m^3], c_d is the air resistance coefficient, A is the vehicle cross-sectional area [m^2], and v is the vehicle speed [m/s]. To control the speed, a suitable controller must be designed. In this study, a classical PID controller is designed, and its performance is analyzed for variable reference speed.

B. PID Control

In this study, a PID controller is designed to control the speed of the vehicle. A PID controller is a control system that derives its name from the sum of proportional, integral, and derivative terms of the output. These terms are called k_p, k_i and k_d respectively. By optimally determining these variables, the dynamic response of a system is increased, overshoot is reduced, and steady-state error is eliminated [38]. The mathematical expression of the controller is given in (9) as follows.

$$u(t) = k_p e(t) + k_i \int_0^t e(t)dt + k_d \frac{de(t)}{dt} \quad (9)$$

Here, $e(t)$ represents the error term, k_p denotes the proportional gain, k_i represents the integral gain, and k_d signifies the derivative gain. Optimal determination of these parameters is essential for effective speed control. In this study, the SPBO, SCO, ROA, GBO, and GWO algorithms were employed to compute the parameter values of k_p, k_i, and k_d.

III. OPTIMIZATION ALGORITHMS

A. Rainwater Optimization Algorithm (ROA)

The ROA is a metaheuristic algorithm that determines the minimum points on the earth's surface, influenced by the natural flow of raindrops and precipitation formations [39]. When rain falls, raindrops reach the earth's surface. The weight of the rainfall causes it to accumulate over time and reach the lowest points on the land. However, some drops are absorbed by the soil or mixed with previous drops. On the other hand, the soil dissolves in water. These facts form the basis for the ROA, where each solution is modeled by a raindrop. The Rain Optimization Algorithm is an algorithm that mimics the natural behavior of raindrops falling from a peak (high position) to a valley (low position). Correct adjustment of the parameters allows this algorithm to identify both local and global extrema. In terms of finding the global minimum, the ROA appears to be superior to other algorithms. The movement of raindrops flowing downhill forms the basis of the optimization process.

Just like raindrops falling randomly to the ground, locations in the solution space can also be randomly selected based on the context. The radius of each raindrop is its most important characteristic and can decrease over time and expand when the raindrop merges with other raindrops. After the initial population of solutions is created, the radius of each droplet can be randomly assigned within a reasonable range. In each iteration, each droplet examines its surroundings depending on its size. A droplet connected to another droplet only examines the endpoint of the area it covers. In an n-dimensional space, each droplet contains n variables. Therefore, in the first stage, the upper and lower bounds of the first variable will be examined in a way that is determined by the droplet radius. In the next stage, by checking the two ends of the two variables, this process will continue until the last variable. At this stage, the cost of the first droplet is updated by moving downwards. This is not the final action of the droplet, and as the cost function decreases, it will continue to move downwards in the same direction. This movement causes the neighboring droplets to approach each other, resulting in an improvement in the results. When the droplet reaches the minimum point, its radius begins to decrease slowly, increasing the accuracy of the obtained solution. This action will be performed for all droplets, and then the costs and positions of all droplets will be determined. This entire process aims to identify the extremum points of the objective function.

The radius of each droplet is modified in two modes:

If two drops with radii r_1 and r_2 are close to each other and share a common point, they can be combined to form a larger drop with a radius R. The mathematical expression for this drop is given in (10).

$$R = \left(r_1^n + r_2^n\right)^{1/n} \quad (10)$$

If a droplet with a radius r_1 is not moving, it can be absorbed at a rate determined by α upon interaction with the ground. In this case, the droplet R is expressed as in (11).

$$R = \left(\alpha r_1^n\right)^{1/n} \quad (11)$$

In this equation, n represents the number of variables for each droplet. α represents the percentage of volume that can be absorbed in each iteration and takes a value between 0 and 100. Additionally, a minimum radius r_{min} is defined, and droplets smaller than this r_{min} value are lost. As the optimization process progresses, the droplet population decreases, and larger droplets expand their search areas. Larger droplets enhance their local search capabilities proportionally to their diameters. With each iteration, weaker droplets either vanish due to having smaller search areas or connect to stronger droplets with larger search areas. As a result, the initial population significantly decreases, and the speed of finding correct solutions increases.

B. Student Psychology Based Optimization Algorithm (SPBO)

The SPBO algorithm is based on the psychology of students who strive to become the best students in the class by obtaining the highest grades and improving their performance. A students' performance is evaluated based on the grades they receive in exams. The student who achieves the highest grade in the exam is generally considered the most successful student in the class and is rewarded accordingly. Typically, students in the class try to improve their performance in order to become the best students. This requires them to put in more effort for each subject. However, the effort put in by students for each subject depends on factors such as their ability, effectiveness, and interest in that subject. Therefore, the effort required for all students to improve their exam performance is not the same and can vary from student to student. The goal of becoming the best student demonstrates that the effort students put in is also linked to their psychology. In light of this information, the students in a class can be divided into four categories based on their subject-wise efforts: the best, good, average, and those who try to improve randomly [40].

C. Sine Cosine Algorithm (SCA)

The SCA is a population-based optimization algorithm. Initially, a random set of solutions is generated. These solutions are evaluated by an objective function and improved according to a certain set of rules. The SCA utilizes sine and cosine functions for updating the solutions. These functions calculate the next position by determining the current solution's location and direction of movement. Additionally, parameters are used to determine how much the solutions should move towards or away from the target, the impact of distance to the target, and to facilitate transitions between sine and cosine components. The main objective of SCA is to effectively utilize a region between two solutions in the search space. The cyclic pattern of sine and cosine functions allows solutions to be repositioned around each other, enabling efficient exploration of the defined region of solutions.

D. Gradient Based Optimizer (GBO)

GradientBased Optimization (GBO), as its name suggests, aims to find the best solution in an optimization problem using both gradientbased (i.e., derivative-based) and population-based (i.e., evaluating multiple solutions simultaneously) approaches. This algorithm creates a group of solution vectors (population) in a specific search space and ensures each vector is located somewhere. Then, with the gradientbased approach, it calculates the gradient of the objective function at the position of each solution vector, determining the best direction at that location. Subsequently, it updates the solution by taking a step in that direction. However, gradientbased approaches may not always be effective, especially in complex or irregular functions. In such cases, GBO is supported by population-based approaches. The population-based approach involves evaluating many different solutions simultaneously, allowing for a broad search and potentially discovering better solutions [41].

E. Grey Wolf Optimization (GWO)

Grey Wolf Optimization is a metaheuristic algorithm developed by drawing inspiration from the hunting strategies of grey wolves. Grey wolves, being apex predators at the top of the food chain, are typically pack animals. The leaders of the pack, known as alphas, consist of a male and a female who make decisions regarding hunting and resting places. At the second level of the hierarchy are the beta wolves, which assist the alphas and are considered potential candidates for leadership. At the lowest level are the omega wolves, which submit to the dominant wolves in the pack. The algorithm aims to solve optimization problems by updating solution candidates based on this hierarchical structure and hunting strategies while mimicking the behaviors of alpha wolves [42].

TABLE I. VEHICLE PARAMETERS

Parameter	Values
Maximum torque	450 Nm
Maximum power	340 HP
Vehicle mass (m_v)	1741 kg
Aerodynamic drag, Cd	0.36
Front area	2.42 m^2
Maximum speed	260 kph
Acceleration time from 0–100 kph	5.3 s

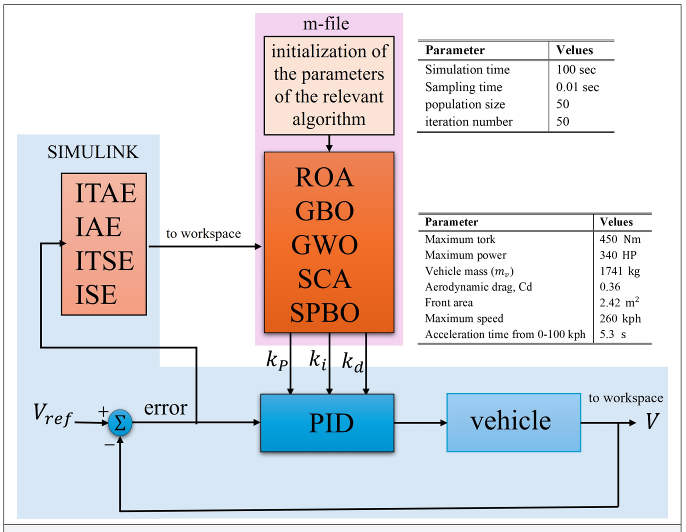


Fig. 2. General representation of the simulation model of the VCC system.

IV. IMPLEMENTATION AND DISCUSSION

In this section, the parameters of the designed PID controller (k_p, k_i, k_d) are optimized with ROA [43], SPBO [40], SCA [44], GBO [41] and GWO [42] algorithms in the MATLAB environment in order to perform a comparative analysis for the VCC system. The simulation studies were carried out using real internal combustion engine parameters, which are provided in Table I. To obtain the stabilizing PID gains, the upper and lower bounds of [k_p k_i k_d] are determined [45] as [0.1 0.1 0.1] and [100 10 0.1], respectively. For a fair comparison of the five algorithms, the population size and iteration number are chosen as 50 in each simulation. The general MATLAB model of the study is shown in Fig. 2.

In addition, in order to select the best controller parameters for the VCC system, the PID is optimized using the 4 different objective functions (performance index): ITAE (integral time absolute error) (12), IAE (integral absolute error) (13), ISE (integral square error) (14), and ITSE (integral time square error) (15), which are most commonly used in the literature. These performance indices are defined as follows:

$$ITAE = \int_0^\infty t\left|e(t)\right|dt, \quad (12)$$

$$IAE = \int_0^\infty \left|e(t)\right|dt, \quad (13)$$

$$ISE = \int_0^\infty e(t)^2\,dt, \quad (14)$$

TABLE II. OPTIMAL PID PARAMETERS OBTAINED WITH THE DIFFERENT ALGORITHMS AND FUNCTIONS (INTEGRAL TIME ABSOLUTE ERROR, INTEGRAL ABSOLUTE ERROR, INTEGRAL SQUARE ERROR AND INTEGRAL TIME SQUARE ERROR) WITHOUT DISTURBANCES

Algorithms	ITAE			IAE		
	k_p	k_i	k_d	k_p	k_i	k_d
ROA	0.44180	0.00010	0.00010	3.17490	0.00010	0.00010
SPBO	1.80050	0.00044	0.01763	4.41260	0.00034	0.00010
GBO	1.03920	0.00034	0.00018	3.75290	0.00022	0.00010
SCO	3.39930	0.00150	0.00560	3.24810	0.00036	0.00062
GWO	2.21210	0.00035	0.00012	4.75510	0.00035	0.00011
Algorithms	ISE			ITSE		
	k_p	k_i	k_d	k_p	k_i	k_d
ROA	1.72790	0.00010	0.00010	3.80400	0.00010	0.00010
SPBO	4.92730	0.01268	0.05640	4.93510	0.18200	0.00880
GBO	4.25070	0.01060	0.04360	4.12410	0.01150	0.04820
SCO	2.71500	0.01440	0.09380	2.71500	0.002700	0.06510
GWO	4.48910	0.01160	0.06450	5	0.0015	0.00100

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; IAE, integral absolute error; ISE, integral square error; ITAE, integral time absolute error; ITSE, integral time square error; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student PsychologyBased Optimization.

$$ITSE = \int_0^\infty te(t)^2\,dt, \tag{15}$$

where t is the time, $e(t)$ is the error between measured (V) and reference (V_{ref}) vehicle speed. During the comparative analysis, the efficacy of the algorithms in optimizing the system was evaluated across various magnitudes of external disturbances, such as road incline (R_i) and passenger weight (m_p), as well as internal disturbances including Gaussian (G_d) and sinusoidal (S_d) inputs applied to the mathematical model of the vehicle system. The evaluation of method performance is carried out by altering the reference speed V_{ref}, initially set at 90 km/h up to the 20th second, subsequently decreased to 60 km/h at the 20th second, then raised to 110 km/h at the 40th second, followed by a reduction to 80 km/h at the 60th second, and finally lowered to 50 km/h at the 80th second. This assessment is conducted based on key performance metrics including rise time (t_r), settling time (t_s), fall time (t_f), and percentage overshoot (M_o).

Initially, the parameters (k_p, k_i, and k_d) derived from employing the algorithms with different objective functions, excluding considerations of internal and external disturbances, are identified. This process facilitated the determination of the algorithm that exhibits superior optimization performance when coupled with specific objective functions. Table II shows the optimal parameter values obtained when different algorithms and different functions are used together. Disturbance effects are not considered in this analysis. Scenarios that take these effects into account are validated below.

The optimization process is carried out over four scenarios. The first scenario focused on the road incline, the second on passenger weight, the third on Gaussian and sine noise, and the fourth on combining parameters from the previous three using different values. Thus, both transient state, steady state, and robustness analysis are performed.

A. Scenario-I: Road Incline (R_i)

The behavior and performance of cruise control systems can be greatly affected by road inclines. Gravity gives the car more resistance when it is traveling uphill. The car may slow down if a conventional cruise control system finds it difficult to keep the pace attained. The cruise control system opens the throttle further to maintain speed despite the hill. In some circumstances, the engine may need to work harder, particularly on steeper inclines, which could result in increased fuel consumption and engine strain. Gravity helps the car drive faster when it is heading downhill; therefore, it might go faster than planned. Conventional systems might not be able to manage sudden increases in speedwell, forcing the driver to manually engage the brakes. Certain systems may momentarily deactivate cruise control. The optimization performances of the algorithms were looked at for various road incline conditions to make the proposed VCC system that was created here and one that was more robust. It was thus attempted to determine the optimal conditions under various inclination circumstances. For this purpose, R_i is set at 0°, 10°, and 30°, which are typical in urban and highway environments. According to international road design standards such as those set by the American Association of State Highway and Transportation Officials (AASHTO), the European Union Road Design Guidelines, the Chinese Highway Code (JTGB01-2014), and the regulations of the General Directorate of Highways in Turkey (KGM), highway inclines typically range between 4% and 6% (≈2.3°–3.4°), while urban roads may have inclines up to 8% (≈4.6°). The European Union and Chinese highway standards allow slightly higher inclines

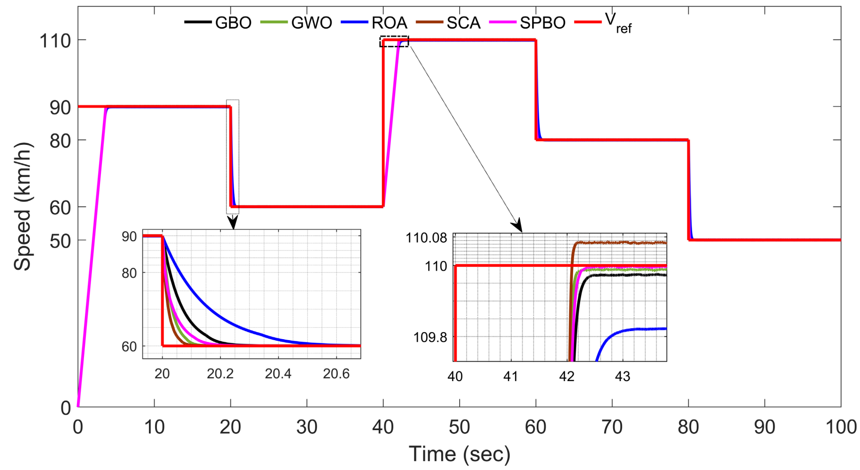


Fig. 3. Transient and steady state performance of the algorithms using ITAE for $R_i = 0°$.

in mountainous regions, with values reaching 10% (≈5.7°) in special cases. A 10° incline (~17.6% grade) represents a steep but realistic scenario, aligning with the upper limits of certain local roads and mountainous areas, where gradients can reach 12-15% (≈6.8°–8.5°) in challenging terrains, particularly in China and some European mountain roads. Although a 30° incline (~57.7% grade) is not commonly found in real-world roadways, it is included as a worst-case scenario for evaluating the robustness of the proposed system.

To evaluate these situations, the reference speed change is applied to calculate the PID coefficients and the resulting performance indicators are listed in Table III.In this scenario, the driver weight is set to 80 kg (no passenger), and Gaussian and sinusoidal disturbances are not considered. When the reference speed is reduced from 90 km/h to 60 km/h at the 20th second under zero road incline, the shortest reaction time is observed in the SCA algorithm, with a decrease of 0.05 seconds, while the longest reaction time is observed in the ROA algorithm, with a decrease of approximately 0.34 seconds. It is clear that the fall time of the SCA algorithm remained unchanged during the slope variation, whereas the ROA algorithm is the most affected one. Figs. 3–5 show the transient and steady-state behaviors of the algorithms. Note that the results given in Table II are calculated from these figures.

B. Scenario-II: Weight (m_p)

The total weight of the vehicle (m_v) increases with an increase in passenger weight (m_p). The extra weight puts more pressure on the cruise control system to precisely and frequently adjust the throttle in order to maintain a steady pace, particularly when accelerating or decelerating. Longer stopping distances result from more weight. To maintain safe stopping distances, VCC systems that interface with automatic braking and adaptive cruise control must take into consideration the increased momentum. Having more passengers alters the center of gravity of the car, which can impact handling. Stability control mechanisms that adapt to the weight distribution of the vehicle may be included in advanced control systems. While passenger weight does affect a cruise control system's operation, modern systems are designed to compensate for these variations.

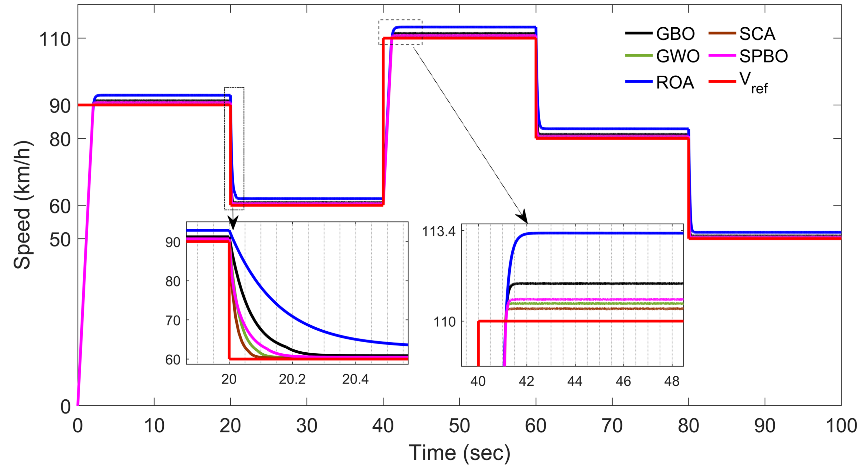


Fig. 4. Transient and steady state performance of the algorithms using ITAE for $R_i = 10°$.

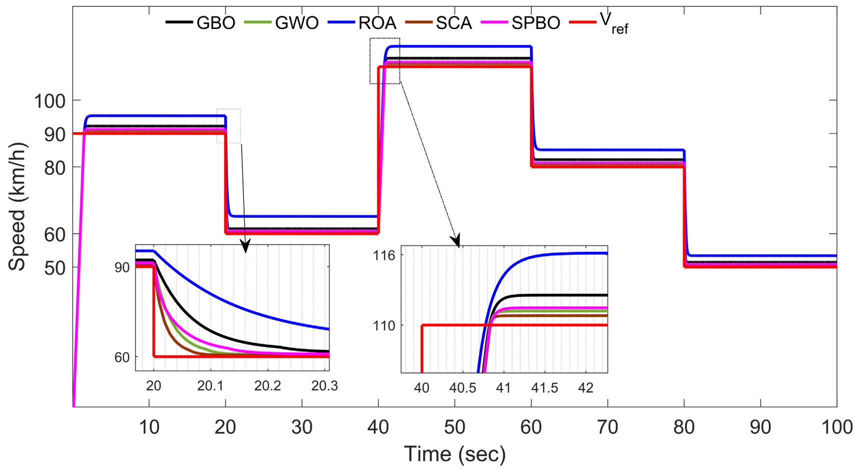


Fig. 5. Transient and steady state performance of the algorithms using ITAE for $R_i = 30°$.

The effects are most significant in terms of fuel efficiency, engine load, and braking performance. Advanced systems use various sensors and adaptive algorithms to ensure that the vehicle maintains a constant speed safely and efficiently, regardless of changes in passenger mass. This study investigates the efficiency of algorithms in response to fluctuations in vehicle weight and Table IV displays the performance measurements obtained by considering total passenger weights of 80 kg, 240 kg, and 400 kg, respectively. Other disturbances are taken to be zero in each scenario. It is observed that the algorithms exhibit negligible sensitivity to load changes. The SCA algorithm demonstrates superior performance under varying loads, achieving a fall time of 0.05 sec. and a settling time of 2.12 sec. at the maximum load. Although the rise time and overshoot are comparable across all algorithms, the ROA algorithm exhibits the slowest fall and settling times.

C. Scenario-III: Gauss and Sin

When crucial characteristics like vehicle speed, acceleration, and distance are measured, measurement disturbances are defined as noise or inaccuracies in the sensors. The feedback signals sent to the control system by these disturbances may contain inaccuracies that degrade the ability of systems to accurately control the speed. Inaccurate data can cause oscillations or overshooting of the desired speed setpoint, which can weaken a passenger's comfort and fuel efficiency. In VCC systems, external and internal disturbances significantly influence system stability and performance. Gaussian noise (G_d) and sinusoidal noise (S_d) are widely used as representative disturbance models to assess controller robustness against real-world uncertainties such as sensor measurement errors, road irregularities, and aerodynamic effects [46–51].

TABLE III. PERFORMANCE COMPARISON OF THE ALGORITHMS UNDER VARIOUS ROAD INCLINE (R_i) CONDITIONS (INTEGRAL TIME ABSOLUTE ERROR)

Algorithms	R_i (degrees)	t_f (sec)	t_r (sec)	M_o (%)	t_s (sec)
ROA	0	0.34	1.85	0.00	2.47
	10	0.66	0.97	3.57	1.73
	30	0.36	0.67	6.40	1.26
SPBO	0	0.09	1.85	0.00	2.16
	10	0.11	1.00	0.90	1.29
	30	0.12	0.74	1.59	0.96
GBO	0	0.15	1.85	0.00	2.21
	10	0.18	0.99	1.53	1.39
	30	0.22	0.73	2.73	1.02
SCA	0	0.05	1.85	0.01	2.09
	10	0.05	1.01	0.53	1.19
	30	0.05	0.75	0.87	0.88
GWO	0	0.07	1.85	0.00	2.12
	10	0.08	1.01	0.73	1.25
	30	0.08	0.75	1.30	0.91

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student PsychologyBased Optimization.

TABLE IV. PERFORMANCE COMPARISON OF THE ALGORITHMS IN RESPONSE TO PASSENGER MASS (M_P) CHANGE (INTEGRAL TIME ABSOLUTE ERROR)

Algorithms	m_p (kg)	$t_{f (sec)}$	t_r (sec)	M_o (%)	t_s (sec)
ROA	80	0.34	1.85	0.000	2.47
	240	0.37	1.85	0.000	2.47
	400	0.39	1.85	0.000	2.74
SPBO	80	0.09	1.85	0.000	2.16
	240	0.10	1.84	0.000	2.16
	400	0.11	1.84	0.000	2.24
GBO	80	0.15	1.85	0.000	2.21
	240	0.16	1.85	0.000	2.21
	400	0.17	1.84	0.000	2.32
SCA	80	0.05	1.85	0.001	2.09
	240	0.05	1.84	0.001	2.09
	400	0.05	1.84	0.001	2.12
GWO	80	0.07	1.85	0.000	2.12
	240	0.07	1.85	0.000	2.12
	400	0.08	1.84	0.000	2.16

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student Psychology-Based Optimization.

TABLE V. PERFORMANCE COMPARISON OF THE ALGORITHMS UNDER GAUSSIAN DISTURBANCE (G_D)

Algorithms	G_d	t_f (sec)	t_r (sec)	M_o (%)	t_s (sec)
ROA	0.05	0.34	1.85	0.11	2.52
	1	0.33	1.84	0.3	2.43
	5	0.34	1.84	0.4	2.25
SPBO	0.05	0.09	1.85	0.37	2.11
	1	0.09	1.83	0.58	2.07
	5	0.11	1.82	1.5	2.09
GBO	0.05	0.14	1.85	0.21	2.21
	1	0.14	1.85	0.4	2.16
	5	0.15	1.84	0.81	2.29
SCA	0.05	0.05	1.84	0.43	2.07
	1	0.05	1.84	1.1	2.04
	5	0.05	1.84	1.5	2.04
GWO	0.05	0.07	1.84	0.3	2.09
	1	0.14	1.84	0.35	2.11
	5	0.06	1.85	1.16	2.16

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student Psychology-Based Optimization.

Gaussian noise is frequently utilized to replicate random sensor errors and environmental variations, which are essential for maintaining precise feedback control [48]. Studies show that real-world vehicular sensors exhibit noise levels ranging from 0.05% to 5% of the measured signal, aligning with the values (0.05, 1, and 5) used in this study [48]. Sinusoidal noise is commonly utilized to represent periodic disruptions caused by road irregularities, engine vibrations, and aerodynamic variations [47]. Previous works have shown that such disturbances typically occur at frequencies between 0.1 Hz and 1 Hz, with amplitude variations up to 0.7% of the nominal speed, supporting the parameters (0.1, 0.4, and 0.7) selected in this research [46].

TABLE VI. PERFORMANCE COMPARISON OF THE ALGORITHMS UNDER SINUSOIDAL DISTURBANCE (S_D)

Algorithms	S_d	t_f (sec)	t_r (sec)	M_o (%)	t_s (sec)
ROA	0.1	0.34	1.44	0.000	2.14
	0.4	0.33	1.44	0.005	2.10
	0.7	0.32	1.44	0.009	2.09
SPBO	0.1	0.10	1.43	0.003	2.01
	0.4	0.09	1.43	0.008	2.01
	0.7	0.09	1.43	0.010	2.01
GBO	0.1	0.14	1.44	0.002	2.02
	0.4	0.14	1.43	0.006	2.02
	0.7	0.14	1.43	0.010	2.02
SCA	0.1	0.05	1.43	0.003	2.01
	0.4	0.05	1.43	0.008	2.01
	0.7	0.04	1.43	0.012	2.01
GWO	0.1	0.07	1.43	0.001	2.01
	0.4	0.07	1.43	0.007	2.01
	0.7	0.06	1.43	0.010	2.01

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student Psychology-Based Optimization.

TABLE VII. CHALLENGING CONDITIONS (A: $G_D=1$, $S_D=0.4$, $R_I=10°$, $M_P=240$ KG, B: $G_D=5$, $S_D=0.7$, $R_I=30°$, $M_P=400$ KG)

Algorithms	Scenario-IV	t_f (sec)	t_r (sec)	M_o (%)	t_s (sec))
ROA	a	0.26	0.97	4.375	1.61
	b	0.38	0.66	10	1.92
SPBO	a	0.11	1.01	2.25	1.15
	b	0.14	0.73	4.06	1.04
GBO	a	0.18	0.99	2.5	1.30
	b	0.25	0.72	4.375	1.21
SCA	a	0.05	1.00	1.625	1.14
	b	0.05	0.74	2.94	0.89
GWO	a	0.08	1.01	1.875	1.18
	b	0.08	0.75	3.125	0.99

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student Psychology-Based Optimization.

To systematically assess the influence of these disturbances on optimization-driven PID tuning, G_d and S_d were incorporated into the VCC system simulations. Tables V and VI show the performance measurements for these two cases. Firstly, assuming a flat road ($R_i=0$), no S_d, and an 80 kg passenger mass, G_d levels of 0.05, 1, and 5 are added to assess how well the algorithms performed. It is found that while the disturbances have an impact on the overshoot and settling times, they have no effect on the fall and rise times. With a fall time of 0.05 seconds, the SCA algorithm demonstrates the shortest fall time, whereas the ROA algorithm displays the slowest fall time, at 0.34 seconds. All the algorithms have almost the same rising times. The SCA algorithm yields the largest overshoot, while the ROA algorithm causes the lowest, at 0.4 percent. The SCA algorithm also has the fastest settling time, and the ROA algorithm has the slowest. Secondly, assuming a flat road ($R_i=0°$), no G_d with and an $m_p=80$ kg passenger mass, S_d levels of 0.1, 0.4, and 0.7 are used for assessment. The lowest M_o and the lowest t_f are obtained with ROA and GWO respectively, as seen in Table VI.

D. Scenario-IV: Challenging situations

In Scenario IV, the previous three scenarios are used together to produce a scenario with two separate and challenging situations. In scenario IV-a, $G_d=1$, $S_d=0.4$, $R_i=10°$, $m_p=240$ kg; in scenario IV-b, $G_d=5$, $S_d=0.7$, $R_i=30°$, $m_p=400$ kg are taken. Table VII shows that in both conditions (a and b), the SCA demonstrates the fastest fall and settling times and the lowest overshoot, while the ROA exhibits the slowest fall time and the highest overshoot. However, the ROA has the fastest rise time. It is observed that as the conditions become more challenging, the rise time decreases; on the other hand, the overshoot increases. Figs. 6 and 7 show the performances of all algorithms for Scenario IV-a and Scenario IV-b with ITAE, respectively.

When tuning control systems, it is crucial to evaluate their performance using various performance indices. These indices help measure how well the system responds to changes, disturbances, and errors. Four common performance criteria are ISE, ITSE, ITAE, and IAE. Each of these functions has distinct characteristics and advantages,

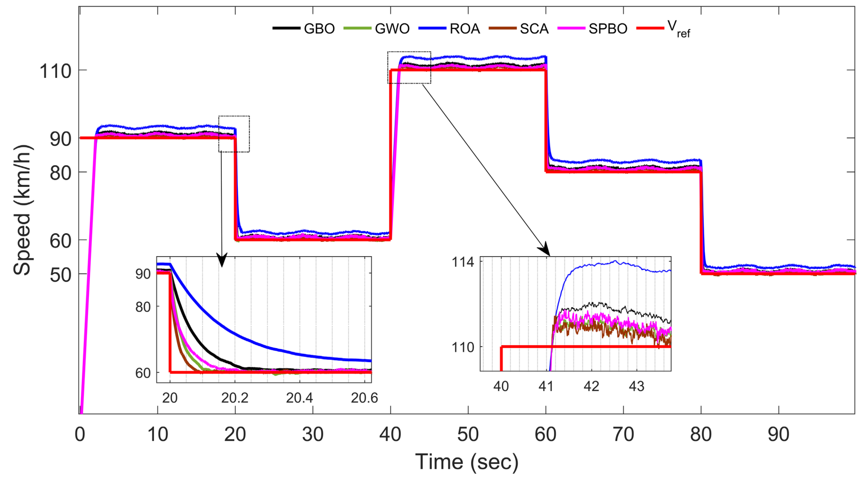


Fig. 6. Challenging situations: scenario IV-a.

and understanding their differences is essential for selecting the appropriate criterion for a given control application. ISE is useful for systems where larger errors are more critical and need to be minimized aggressively.

However, smaller errors are given less importance, which might result in slower settling times. IAE often leads to faster responses as the controller is sensitive to all errors. However, it may result in more aggressive control actions, potentially causing more wear and tear on the system components. ITSE can result in balanced performance for systems where both large and prolonged errors are critical. But it can be complex to tune due to the double emphasis on error magnitude and time weighting. As a result, ISE is suitable for systems where reducing large errors is more critical than minor ones. IAE is useful for systems where all errors should be minimized equally, leading to potentially quicker responses. ITAE is ideal for systems where prolonged errors are particularly detrimental, encouraging fast correction. ITSE is beneficial for systems needing a balance between reducing large errors and minimizing prolonged errors. In this study, a detailed verification is carried out using these cost functions to see the performance of the algorithm in the internal combustion engine system. This evaluation is carried out for R_i values used in scenario-I, and results are given in Table VIII. When ITSE is used, t_f is in GWO with 0.02 seconds; t_r is slightly slower in SPBO and unchanged in others; The minimum overshoot is in GWO, and the maximum is in SPBO; It is seen that t_s is shortest in GWO and ROA. When IAE is used, t_f is the fastest in GWO and SPBO with 0.02 seconds, and the algorithms are not affected by the incline change; t_r is the same in all and does not change with incline; M_o is minimum in GWO; It can be seen that t_s is almost the same in all of them, but the fastest is in SPBO. When ISE is used, t_f is the fastest in GWO and SPBO with 0.03 seconds and the algorithms are not affected by the incline change; t_r is almost the same in all and decreases with R_i; when $R_i = 0$, the overshoot is at least in ROA, and when there is a change in slope, M_o is minimum in GBO; It is seen that t_s is fastest in ROA when there is no incline and fastest in SPBO when $R_i \neq 0$. The graphical comparisons of the algorithms under $R_i = 30$ for ITSE, IAE, and ISE are represented in Figs. 8–10 respectively. The results for ITAE are already given in Table II.

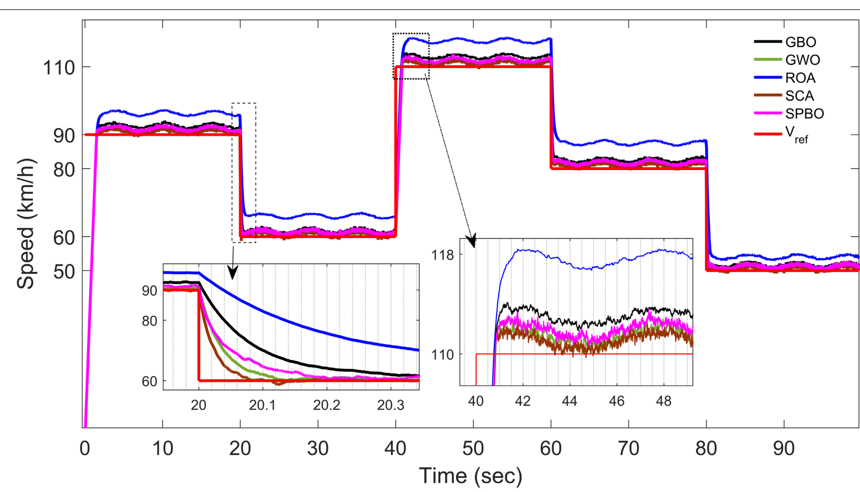


Fig. 7. Challenging situations: scenario IV-b.

TABLE VIII. PERFORMANCE OF THE ALGORITHMS UNDER DIFFERENT COST FUNCTIONS

Algorithms	Cost Functions	R_i	t_f (sec)	t_r (sec)	M_o (%)	t_s (sec)
ROA	ITSE	0	0.03	1.85	0.00	2.07
		10	0.03	1.01	0.30	1.15
		30	0.03	0.75	0.61	0.87
	IAE	0	0.04	1.85	0.10	2.08
		10	0.04	1.01	0.50	1.18
		30	0.04	0.75	0.74	0.88
	ISE	0	0.08	1.85	0.04	2.09
		10	0.09	1.01	1.01	1.20
		30	0.10	0.74	1.03	0.95
SPBO	ITSE	0	0.06	2.09	3.10	2.13
		10	0.04	1.15	1.82	1.18
		30	0.04	0.85	1.41	0.88
	IAE	0	0.02	1.85	0.1	2.062
		10	0.02	1.01	0.4	1.18
		30	0.02	0.75	0.63	0.86
	ISE	0	0.03	1.83	0.45	2.09
		10	0.03	1.01	0.64	1.17
		30	0.03	0.74	0.81	0.89
GBO	ITSE	0	0.04	1.84	0.55	2.1
		10	0.03	1.01	0.73	1.16
		30	0.04	0.75	0.77	0.92
	IAE	0	0.03	1.85	0.1	2.07
		10	0.03	1.01	0.42	1.17
		30	0.03	0.75	0.68	0.87
	ISE	0	0.03	1.83	0.45	2.11
		10	0.03	1.01	0.64	1.17
		30	0.04	0.75	0.72	0.94
SCA	ITSE	0	0.05	1.84	0.09	2.12
		10	0.05	1.01	0.45	2.14
		30	0.06	0.74	0.92	1.01
	IAE	0	0.04	1.85	0.1	2.07
		10	0.04	0.01	0.5	1.18
		30	0.04	0.75	0.74	0.88
	ISE	0	0.05	1.82	0.9	2.17
		10	0.06	1.01	1.02	1.27
		30	0.06	0.75	1.03	0.99
GWO	ITSE	0	0.02	1.85	0	2.08
		10	0.02	1.01	0.24	1.15
		30	0.02	0.75	0.52	0.86
	IAE	0	0.02	1.85	0.1	2.07
		10	0.02	0.01	0.37	1.18
		30	0.02	0.75	0.59	0.86
	ISE	0	0.03	1.82	0.45	2.1
		10	0.03	1.01	0.64	1.19
		30	0.04	0.75	0.9	0.95

GBO, GradientBased Optimization; GWO, Grey Wolf Optimization; IAE, integral absolute error; ISE, integral square error; ITSE, integral time square error; ROA, Rain Optimization Algorithm; SCA, Sine Cosine Algorithm; SPBO, Student PsychologyBased Optimization.

Traditional methods and metaheuristic algorithms offer different advantages and limitations in PID parameter tuning [52–54]. Classical methods, such as the Ziegler-Nichols approach, provide quick and practical solutions for certain systems but often result in high overshoots and long settling times. On the other hand, metaheuristic algorithms explore a broader parameter space, enabling more optimal and adaptive tuning, especially for nonlinear and dynamic systems. The key differences between these approaches in terms of optimization strategy, computational cost, adaptability, and robustness are summarized in Table IX.

This comparison given in Table IX provides a theoretical perspective on the strengths and weaknesses of both approaches. To further illustrate these differences in a practical application, Fig. 11 presents a performance evaluation of the Ziegler-Nichols method versus the metaheuristic optimization algorithms used in this study in a VCC system is presented. As observed in Fig. 11, the Ziegler-Nichols method exhibits a significantly higher overshoot compared to the metaheuristic-based tuning approaches. Additionally, the settling time is considerably longer, indicating a slower stabilization of the system after a reference speed change. Furthermore, the rise time is also elevated, suggesting a delayed response in reaching the desired speed. These findings highlight the limitations of the Ziegler-Nichols method in handling dynamic and nonlinear disturbances, whereas metaheuristic algorithms provide more optimized control parameters, ensuring faster and more stable system performance.

In this study, the effectiveness of different metaheuristic algorithms for PID parameter tuning in a VCC system was analyzed under various internal and external disturbances. However, certain limitations should be acknowledged. First, while the selected metaheuristic algorithms provide improved performance compared to traditional methods, numerous newly developed algorithms exist, and the field continues to evolve with the introduction of novel approaches. As a result, the selection of the most suitable algorithm remains an open research area, and future studies could explore the potential of emerging optimization techniques to further enhance control performance.

V. CONCLUSION

In this study, a VCC system was developed using a PID controller, considering internal (Gaussian (G_d) and sinusoidal (S_d)) and external (road incline (R_i) and passenger weight (m_p)) disturbances. The PID parameters were optimized via five different metaheuristic algorithms: ROA, SPBO, SCA, GBO, and GWO. The efficiency of these algorithms was evaluated through several scenarios that replicated real-world circumstances. The findings indicate that no singular algorithm is uniformly superior under every scenario. The optimal algorithm varies based on certain system specifications and performance metrics.

The SCA, enhanced by the ITAE criterion, showed improved effectiveness in adjusting to varying disturbance conditions, especially with a fast response and reduced overshoot. Under scenarios of slope variation, the GWO algorithm employing the ITSE criterion had the fastest response time, indicating its potential suitability for applications prioritizing rapid response. Likewise, although the ROA exhibited delayed adaptation in most situations, it yielded minimum overshoot, which may be useful in systems where the reduction of overshoot is essential.

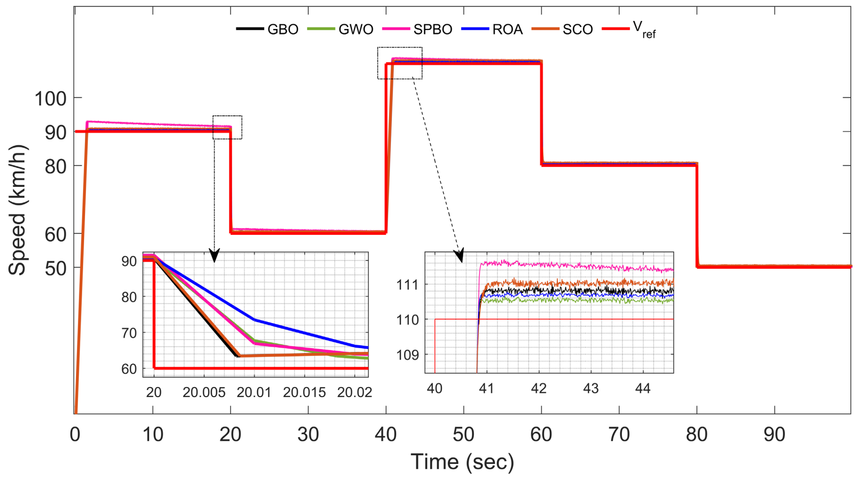


Fig. 8. Transient and steady state performance of the algorithms using integral time square error for $R_i = 30°$.

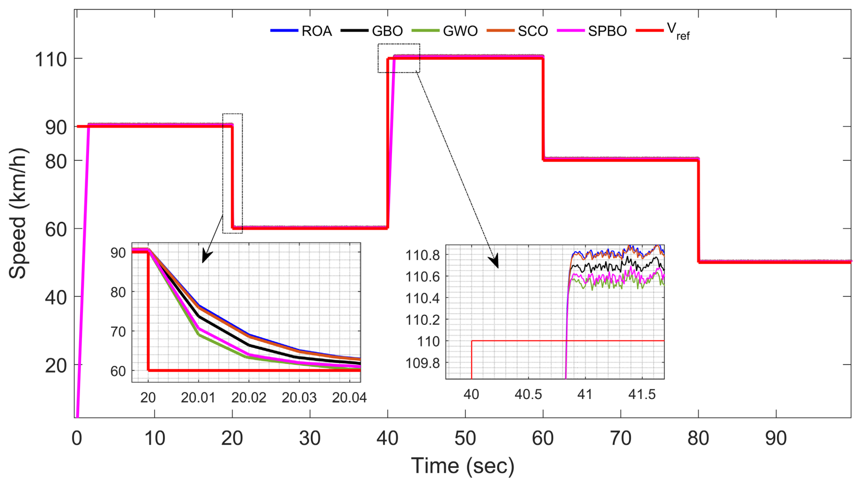


Fig. 9. Transient and steady state performance of the algorithms using integral absolute error for $R_i = 30°$.

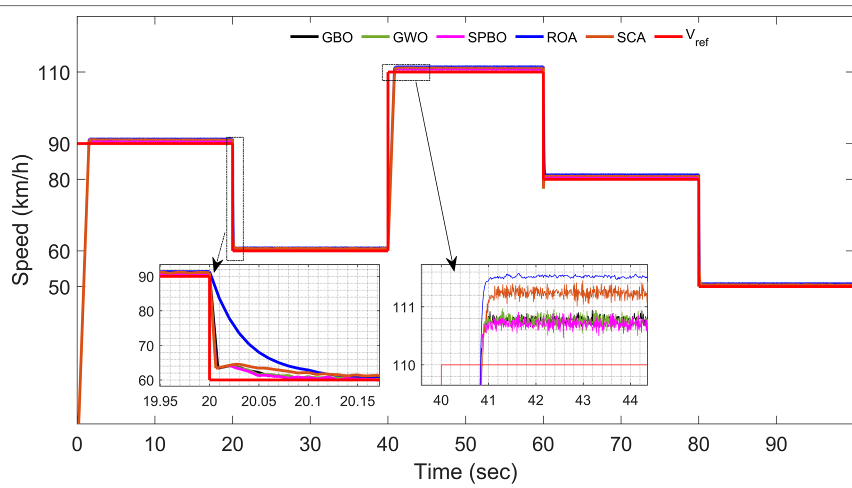


Fig. 10. Transient and steady state performance of the algorithms using integral square error for $R_i = 30°$.

TABLE IX. COMPARISON OF METAHEURISTIC ALGORITHMS AND CLASSICAL TUNING METHODS IN PID CONTROL.

Criteria	Metaheuristic Algorithms	Classical Methods (e.g., Ziegler–Nichols)
Optimization āpproach	Search-oriented, investigates an extensive solution landscape	Rule-based, relies on predefined tuning rules
Adaptability	Can handle nonlinear, time-varying, and complex systems	Limited adaptability, works best for linear systems
Computational cost	Higher due to iterative optimization	Lower, as it provides direct parameter estimation
Tuning accuracy	Provides more optimal and fine-tuned parameters	Often results in suboptimal tuning with high overshoot
Robustness	More robust against disturbances and uncertainties	Performance decreases under changing conditions
Real-Time applicability	May require significant computational resources, limiting real-time use	More suitable for real-time applications due to lower computational burden
Implementation complexity	Requires parameter selection for the optimization algorithm	Easier to implement with straightforward formulas
Scalability	Can be applied to a wide range of control problems	Limited applicability beyond simple PID tuning
Convergence time	May require multiple iterations to converge to optimal values	Provides quick parameter estimation
Generalization capability	Can be applied to various control systems with different dynamics	Works best for specific system structures

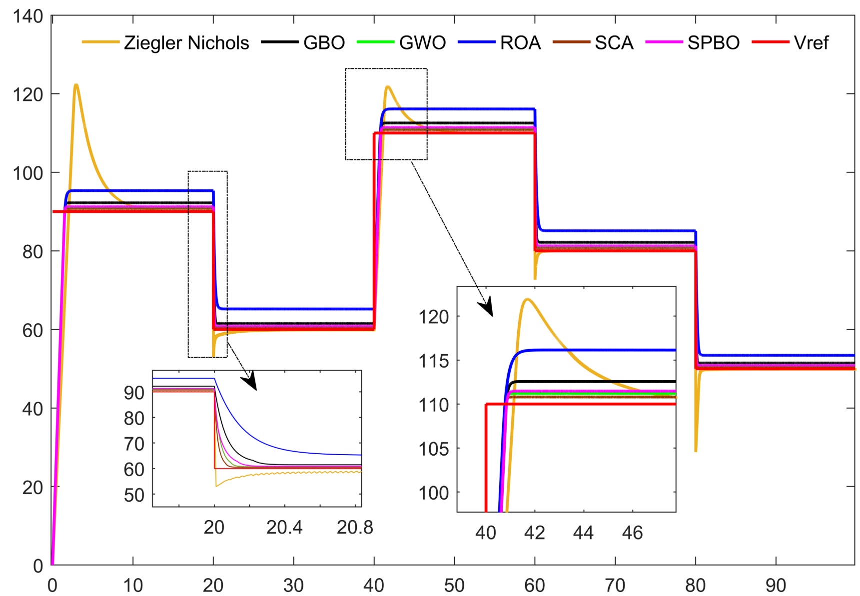


Fig. 11. Comparative performance of Ziegler–Nichols and the metaheuristic optimization algorithms in vehicle cruise control system.

Given these findings, it is essential to select the appropriate optimization algorithm based on the specific control objectives. For instance:

- If minimizing overshoot is the primary goal, ROA may be the best choice.
- If achieving the fastest response time is critical, GWO with ITSE should be preferred.
- If balanced performance across different disturbance conditions is required, SCA with ITAE may be a more suitable option.

A systematic decision framework for algorithm selection based on system requirements may improve the practical applicability of this research. Future research can focus on developing adaptive hybrid optimization strategies that dynamically switch among several algorithms according to real-time performance measures, hence enhancing the robustness and efficiency of PID tuning in VCC systems. Moreover, additional research could investigate statistical variability via multiple-run simulations. This analysis was beyond the scope of the research due to computational limitations. Future endeavors may encompass comprehensive statistical validation utilizing confidence intervals and significance tests (e.g., ANOVA, t-tests) to further evaluate algorithm adaptability.

This work illustrates that metaheuristic optimization methods significantly enhance the efficiency of PID-based VCC systems through adaptive and efficient parameter adjustments. The choice of the most appropriate algorithm must be determined by the particular operating conditions and performance criteria of the system.

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author.

Peer-review: Externally peer-reviewed.

Author Contributions: Concept –YD; Design –Y.D., F.H., A.B.T., Ş.Ş.; Supervision – Y.D.; Materials – F.H., Ş.Ş.; Data Collection and/or Processing – F.H.; Analysis and/or Interpretation – Y.D., F.H., A.B.T., Ş.Ş.; Literature Review – F.H., A.B.T., Ş.Ş.; Writing – F.H., A.B.T., Ş.Ş.; Critical Review – Y.D., F.H.

Declaration of Interests: The authors have no conflicts of interest to declare.

Funding: The authors declare that this study received no financial support.

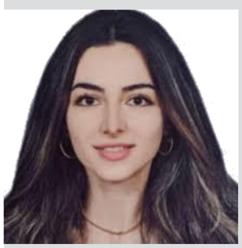
REFERENCES

1. Z. Wang, G. Wu, and M. J. Barth, "A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications," 2018 21st International Conference on Intelligent Transportation Systems (ITSC) [Internet]. New York: IEEE; pp. 2884–2891. Available: https://ieeexplore.ieee.org/document/8569947/.
2. B. Gao et al., A Review and Outlook on Predictive Cruise Control of Vehicles and Typical Applications under Cloud Control System. Machine Intelligence Research. *Chinese Academy of Sciences*, pp. 614–639, 2023.
3. L. Xiao, and F. Gao, "A comprehensive review of the development of adaptive cruise control systems," *Veh. Syst. Dyn.*, vol. 48, no. 10, pp. 1167–1192, 2010. [CrossRef]
4. Y. Liu, Q. Sun, C. Liu, H. Guo, and M. Yue, "Anti-disturbance adaptive cruise control method based on H-infinity robust controller with optimized safe velocity and following time headway," *Meas. Sci. Technol.*, vol. 36, no. 1, 016212. 2025. [CrossRef]
5. H. QI, "Robust responsibility-sensitive safety: Noise disturbed adaptive cruise control," *Syst. Control Lett.*, vol. 196, 106021, 2025. [CrossRef]
6. D. G. Nguyen, S. Park, J. Park, D. Kim, J. S. Eo, and K. Han, "An MPC approximation approach for adaptive cruise control with reduced computational complexity and low memory footprint," *IEEE Trans. Intell. Veh.*, vol. 9, No. 2, pp. 3154–3167, 2024. [CrossRef]
7. M. Shen, R. A. Dollar, T. G. Molnar, C. R. He, A. Vahidi, and G. Orosz, "Energy-efficient reactive and predictive connected cruise control," *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 944–957, 2024. [CrossRef]
8. B. Asadi, and A. Vahidi, "Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 3, pp. 707–714, 2011. [CrossRef]
9. B. Gao, K. Cai, T. Qu, Y. Hu, and H. Chen, "Personalized adaptive cruise control based on online driving style recognition technology and model predictive control," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12482–12496, 2020. [CrossRef]
10. H. Chu, L. Guo, B. Gao, H. Chen, N. Bian, and J. Zhou, "Predictive cruise control using high-definition map and real vehicle implementation," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 11377–11389, 2018. [CrossRef]
11. J. Sawant, U. Chaskar, and D. Ginoya, "Robust control of cooperative adaptive cruise control in the absence of information about preceding vehicle acceleration," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5589–5598, 2021. [CrossRef]
12. Z. Zhou, J. Zhu, and Y. Li, "Robust control of vehicle multi-target adaptive cruise based on model prediction," *Cogn. Comput. Syst.*, vol. 2, no. 4, pp. 254–261, 2020. [CrossRef]
13. R. Alika, E. M. Mellouli, and E. H. Tissir, "A modified sliding mode controller based on fuzzy logic to control the longitudinal dynamics of the autonomous vehicle," *Results Eng.*, vol. 22, 2024.
14. M. H. Shojaeefard, M. Mollajafari, S. Ebrahimi-Nejad, and S. Tayebi, "Weather-aware fuzzy adaptive cruise control: Dynamic reference signal design," *Comput. Electr. Eng.*, vol. 110, 2023.
15. S. K. Chada, D. Görges, A. Ebert, and R. Teutsch, "Deep learning-based vehicle speed prediction for ecological adaptive cruise control in urban and highway scenarios," *IFAC PapersOnLine*. Elsevier B.V., pp. 1107–1114, 2023.
16. Y. Wang, Z. Guo, J. Wu, and S. Fu, "Research on vehicle adaptive cruise control based on BP neural network working condition recognition," *J. Eng.*, vol. 2022, pp. 132–147, 2022.
17. Z. Li, "Review of PID control design and tuning methods," *J. Phys. Conf. Ser.* Institute of Physics, vol. 2649, no. 1, 2023. [CrossRef].
18. Ieng S, S. Ang, V. Huy, H. Cheng, S. Keo, and C. Choeung, "Dingo optimizer for power loss minimization using optimal power flow". GMSARN International Journal, 2025.
19. A. Rezoug, J. Iqbal, and A. Nemra, "Dual FOPID-neural network controller based on fast grey wolf optimizer: Application to two-inputs two-outputs helicopter," *Syst. Sci. Control Eng.*, vol. 13, no. 1, 2025. [CrossRef]
20. A. M. Jarkas, and M. A. N. Doss, "Optimized PI controller tuning for improved performance in BLDC motor speed control using heuristic adaptive lyrebird optimization algorithm," *Electr. Eng.*, 2025.
21. J. Li, X. Xie, L. Dai, and L. Xu, "Dynamic synergy optimization (DSO): An integrated approach of metaheuristic algorithms and PID control for real-time stability enhancement in refrigeration systems," *Int. J. Refrig.*, vol. 173, pp. 100–110, 2025.
22. Y. Danayiyen, K. Dinçer, and Y. Nuhoğlu, "A novel fractional order proportional integral-fractional order proportional derivative controller design based on symbiotic organisms search algorithm for speed control of a direct current motor" [Internet], *ELECTRICA*, 2024. [CrossRef]
23. S. Hlangnamthip, C. Thammarat, C. Sinsukudomchai, and D. Puangdownreong, "Optimal tuning of PIDA controller for vehicle cruise control system by modified bat algorithm" Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering, 2024, pp. 108–111.
24. S. Chaturvedi, and N. Kumar, "Design and implementation of an optimized PID controller for the adaptive cruise control system," *IETE J. Res.*, vol. 69, pp. 7084–7091, 2023.
25. N. Salem, R. Hassan, and L. Muthanna, "Enhancing cruise performance through PID controller tuned with particle swarm optimization technique". 6th International Conference on Intelligent Robotics and Control. Engineering, IRCE 2023. New York: Institute of Electrical and Electronics Engineers Inc., 2023, pp. 23–28.
26. S. Ekinci, D. Izci, L. Abualigah, and R. A. Zitar, "A modified oppositional chaotic local search strategy based Aquila optimizer to design an effective controller for vehicle cruise control system," *J. Bionic Eng.*, vol. 20, pp. 1828–1851, 2023.
27. S. Ekinci, D. Izci, L. Abualigah, A. G. Hussien, C. Le Thanh, and S. Khatir, "Revolutionizing vehicle cruise control: An elite opposition-based pattern search mechanism augmented INFO algorithm for enhanced controller design," *Int. J. Comp. Intell. Syst.*, vol. 16, 2023.
28. M. I. Badruddin, F. A. Bakri, M. S. Sulaiman, and Z. Saad, "Design and Analysis of PID controller for Fuel Economy Driving," *Proceedings* International Conference on Computer and Drone Applications, IConDA 2022. New York: Institute of Electrical and Electronics Engineers Inc.; 2022, pp. 79–84.
29. C. Zhou, N. Qiao, J. Mao, C. Zhao, and L. Yang, "Research on vehicle multi-objective adaptive cruise control system based on particle swarm optimization algorithm" 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, CYBER. New York: Institute of Electrical and Electronics Engineers Incorporated; 2021 IEEE, pp. 426–431.
30. J. E. Naranjo, F. Serradilla, and F. Nashashibi, "Speed control optimization for autonomous vehicles with metaheuristics," *Electronics*, vol. 9, no. 4, 2020. [CrossRef]
31. Y. S. Dawood, A. K. Mahmood, and M. A. Ibrahim, "Comparison of PID, GA and fuzzy logic controllers for cruise control system," *Int. J. Comput. Digit. Syst.*, vol. 7, pp. 311–319, 2018.
32. R. Pradhan, S. K. Majhi, J. K. Pradhan, and B. B. Pati, "Antlion optimizer tuned PID controller based on Bode ideal transfer function for automobile cruise control system," *J. Ind. Inf. Integr.*, vol. 9, pp. 45–52, 2018. [CrossRef]
33. R. Pradhan, S. K. Majhi, J. K. Pradhan, and B. B. Pati, "Performance evaluation of PID controller for an automobile cruise control system using ant lion optimizer," *Eng. J.*, vol. 21, no. 5, pp. 347–361, 2017. [CrossRef]
34. A. R. Abdulnabi, "PID controller design for cruise control system using particle swarm optimization," *Iraqi J. Comput. Inform.*, vol. 43, no. 2, pp. 30–35, 2017. [CrossRef]
35. D. Izci, S. Ekinci, M. Kayri, and E. Eker, "A novel enhanced metaheuristic algorithm for automobile cruise control system," *Electrica*, vol. 21, no. 3, pp. 283–297, 2021. [CrossRef]
36. J. Marzbanrad, and I. Tahbaz-zadeh Moghaddam, "Self-tuning control algorithm design for vehicle adaptive cruise control system through

real-time estimation of vehicle parameters and road grade," *Veh. Syst. Dyn.*, vol. 54, no. 9, pp. 1291–1316, 2016. [CrossRef]
37. Z. Yang, R. He, S. Zhang, and J. Wu, "Research on adaptive cruise control algorithm considering road conditions," *IET Intell. Transp. Syst.*, vol. 15, no. 12, pp. 1478–1493, 2021. [CrossRef]
38. S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammas, "Metaheuristic algorithms for PID controller parameters tuning: Review, approaches and open problems" review, *Heliyon*. Amsterdam: Elsevier Limited, vol. 8, no. 5, e09399, 2022. [CrossRef]
39. A. R. Moazzeni, and E. Khamehchi, "Rain optimization algorithm (ROA): A new metaheuristic method for drilling optimization solutions," *J. Petrol. Sci. Eng.*, vol. 195, 2020. [CrossRef]
40. B. Das, V. Mukherjee, and D. Das, "Student psychology based optimization algorithm: A new population based optimization algorithm for solving optimization problems," *Adv. Eng. Softw.*, vol. 146, 2020. [CrossRef]
41. I. Ahmadianfar, O. Bozorg-Haddad, and X. Chu, "Gradient-based optimizer: A new metaheuristic optimization algorithm," *Inf. Sci. (N Y)*, vol. 540, pp. 131–159, 2020. [CrossRef]
42. S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014. [CrossRef]
43. S. H. Aghay Kaboli, J. Selvaraj, and N. A. Rahim, "Rain-fall optimization algorithm: A population based algorithm for solving constrained optimization problems," *J. Comput. Sci.*, vol. 19, pp. 31–42, 2017. [CrossRef]
44. S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," *Knowl. Based Syst.*, vol. 96, pp. 120–133, 2016. [CrossRef]
45. A. Turan, "PID controller design with a new method based on proportional gain for cruise control system," *J. Radiat. Res. Appl. Sci.*, vol. 17, no. 1, 100810, 2024. [CrossRef]
46. C. T. Yilmaz, and H. I. Basturk, "Rejection of sinusoidal disturbances for known LTI systems in the presence of output delay," *Automatica*, vol. 92, pp. 41–48, 2018. [CrossRef]
47. S. Pigg, and M. Bodson, "Adaptive algorithms for the rejection of sinusoidal disturbances acting on unknown plants," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 4, pp. 822–836, 2010. [CrossRef]
48. A. Falsone, L. Deori, D. Ioli, S. Garatti, and M. Prandini, "Optimal steady-state disturbance compensation for constrained linear systems: The Gaussian noise case," *IEEE Trans. Automat. Contr.*, vol. 67, no. 12, pp. 6322–6332, 2022. [CrossRef]
49. M. Karahan, C. Kasnakoglu, and A. N. Akay, "Robust backstepping control of a quadrotor UAV under pink noise and sinusoidal disturbance," *Stud. Inform. Control*, vol. 32, no. 2, pp. 15–24, 2023. [CrossRef]
50. P. Yan, Z. Li, F. Huang, W. Wen, and L. T. Hsu, "Fault application in lidar/IMU integrated localization systems," *Navig. J. Inst. Navig.*, vol. 72, no. 1, 2025. [CrossRef]
51. H. Zhou, W. Chen, C. Shen, L. Cheng, and M. Xia, "Intelligent machine fault diagnosis with effective denoising using EEMD-ICA- FuzzyEn and CNN," *Int. J. Prod. Res.*, vol. 61, no. 23, pp. 8252–8264, 2023. [CrossRef]
52. E. R. Fernandez Cornejo, R. C. Diaz, and W. I. Alama, "PID Tuning based on Classical and Meta-heuristic Algorithms: A Performance Comparison," 2020 IEEE Engineering International Research Conference (EIRCON) [Internet]. New York: IEEE; 2020, pp. 1–4. Available: https://ieeexplore.ieee.org/document/9253750/.
53. G. Yılmaz, "Comparison of different methods for optimization of PID controller gain coefficients," *Kirklareli Univ. J. Eng. Sci.*, vol. 9, no. 2, pp. 254–264, 2023. [CrossRef]
54. O. Aguilar-Mejía, H. Minor-Popocatl, and R. Tapia-Olvera, "Comparison and ranking of metaheuristic techniques for optimization of PI controllers in a machine drive system," *Appl. Sci. (Switzerland)*, vol. 10, no. 18, 2020. [CrossRef]

Furkan Heybetli is currently a master's student in the Department of Electrical and Electronics Engineering at Karadeniz Technical University, Trabzon, Türkiye. He received B.S. degree in Electrical and Electronics Engineering from Karadeniz Technical University in 2017. His research interests include metaheuristic algorithms, control systems, and biomedical applications.

Yahya Danayiyen is currently a lecturer with the Department of Electrical and Electronics Engineering, Karadeniz Technical University, Trabzon, Turkey. He received the B.S., M.S., Ph.D. degrees in electrical and electronics engineering from the Karadeniz Technical University, in 2009, 2013 and 2021, respectively. In 2015, he was a visiting PhD researcher with the University of Southampton, Southampton, UK and from 2017 to 2018, with Seoul National University of Science and Technology, Seoul, South Korea. His research interests include model predictive control, power electronic converters and optimal control.



Aleyna Begüm Taşdemir is currently a research assistant with the Department of Civil Engineering, Karadeniz Technical University, Trabzon, Turkey. She received the B.S. and M.S. degrees in Civil Engineering from the Karadeniz Technical University, in 2020 and 2023, respectively. She is currently a PhD in Civil Engineering at Karadeniz Technical University. Her research interests include transportation engineering, traffic control and intelligent transportation systems.

Şule Şenyiğit is currently a master's student in the Department of Electrical and Electronics Engineering at Karadeniz Technical University, Trabzon, Türkiye. She received B.S. degree in Electrical and Electronics Engineering from Gümüşhane University in 2022, Gümüşhane, Türkiye. Her research interests include control systems, power systems (with a focus on photovoltaic systems), metaheuristic algorithms and artificial intelligence methods.