Performance Analysis of Fractionalized Order PID Controller-based on Metaheuristic Optimization Algorithms for Vehicle Cruise Control Systems

Abdelhakim Idir, Abderrahim Zemmit, Khatir Khettab, Mokhtar Nesri, Sifelislam Guedida, Laurent Canale

Performance Analysis of Fractionalized Order PID Controller-based on Metaheuristic Optimization Algorithms for Vehicle Cruise Control Systems

Abdelhakim Idir [1, 2], Abderrahim Zemmit[1], Khatir Khettab [1], Mokhtar Nesri [3], Sifelislam Guedida [4], Laurent Canale [5]

[1]*Electrical Engineering Department, University of M'sila 28000, Road Bourdj Bou Arreiridj, M'sila 28000 Algeria.*

[2]*Applied Automation Laboratory, F.H.C, University of Boumerdes, 35000 Boumerdes, Algeria*

[3]*Ecole Supérieur Ali Chabati, Reghaia Algiers, Algeria*

[4]*Ecole Militaire Polytechnique, UER ELT, 16111 Algiers, Algeria*

[5]*CNRS, LAPLACE Laboratory, UMR 5213 Toulouse, France*

Abstract

Recently, automotive manufacturers have prioritized cruise control systems and controllers, recognizing them as essential components requiring precise and adaptable designs to keep up with technological advancements. The motion of vehicles is inherently complex and variable, leading to significant non-linearity within the cruise control system (CCS). Due to this non-linearity, conventional PID controllers often perform sub optimally under varying conditions. This study introduces a fractionalized-order PID (FrOPID) controller, which incorporates an additional parameter to enhance the performance of conventional PID controllers. A comparative analysis is conducted between classical PID controllers and FrOPID controllers optimized using three metaheuristic algorithms: Harris Hawks Optimization (HHO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO). The evaluation is carried out using a linearized model of the vehicle cruise control system (VCCS). The results demonstrate that fractionalized-order PID controllers significantly outperform conventional PID controllers, particularly in terms of rise time and settling time. Among the proposed designs, the integration of HHO and FrOPID proves to be the most effective in achieving a balance between responsiveness and stability, exhibiting exceptional robustness and adaptability to variations in vehicle mass and environmental conditions. This highlights the effectiveness of fractionalized-order controllers in managing the dynamic behavior of vehicles.

Keywords*: Vehicle cruise control system (VCCS), Fractionalized order PID controller, Optimization methods, PID controller, Robustness analysis.*

1 Introduction

Minimizing fuel usage and pollutant emissions, particularly carbon dioxide and other harmful substances, presents a significant challenge for the transportation industry, especially within the automotive sector. This industry faces issues related to both oil

shortages and environmental concerns [1]. In response, car manufacturers and policymakers are exploring various solutions, including advanced engine technologies, smart vehicles, and alternative energy sources [2]. One such innovation is cruise control systems, which have proven effective in optimizing fuel efficiency, reducing driver fatigue, lowering accident risks, and improving traffic flow [3][4].

Control engineering focuses on designing systems to regulate the behavior of processes and devices, with the Proportional-Integral-Derivative (PID) controller being one of the most widely used tools in this field. The simplicity, reliability, efficiency, and robustness of PID controllers contribute to their broad application in industries worldwide. Despite being developed in the 1890s, PID controllers remain prevalent, with nearly ninety

percent of process industries employing them as their primary control mechanism [5][6][7][8].

2 Literature Review

Recent research has demonstrated the advantages of Fractional-Order PID (FOPID) controllers over traditional PID controllers in various applications. Shafiee et al. [9] showed that optimized FOPID controllers offer enhanced performance, which was further supported by studies from Idir et al. [10], Mishra et al. [11], and others [12–17]. These studies highlight the flexibility and improved control of systems with nonlinear dynamics or time-varying parameters achieved through FOPID controllers.

In the automotive sector, the primary objective of cruise control systems is to offer drivers a more comfortable driving experience while improving fuel efficiency and ensuring compliance with speed limits and regulations. Research has focused on maintaining constant vehicle speed with optimal ease of control, while balancing safety, fuel efficiency, and comfort. Several control techniques, such as traditional PID [18], PID with reference models [19], PIDA [20], FOPID [21], and ANFIS-based control [22], have been proposed for cruise control systems. Among these, PID remains a popular choice due to its simplicity.

Metaheuristic algorithms play a vital role in optimizing PID controllers for vehicle cruise control systems, particularly for handling complex nonlinearities. Particle Swarm Optimization (PSO) has been widely applied in this context, with Abdulnabi (2017) demonstrating its effectiveness in enhancing system stability and dynamic response [23]. Harris Hawks Optimization (HHO), inspired by the cooperative hunting strategies of Harris hawks, has also shown strong optimization capabilities for PID and Fractional Order PID controllers, as illustrated by Izci and Ekinci (2021) [24]. Similarly, the Genetic Algorithm (GA), inspired by natural evolution, has been successfully used for PID controller optimization in cruise control systems [25].

Other emerging algorithms include the Red Panda Optimization (RPO) [26], Ant Lion Optimizer (ALO) [27], and Gorilla Troops Optimization (GTO) [28], all of which have demonstrated their potential in optimizing PID controllers for improved system performance.

This paper proposes the design of PID and Fractionalized PID (FrOPID) controllers for vehicle cruise control, optimized using HHO, GA, and PSO algorithms. The objective is to enhance the system's overall performance and efficiency, with simulation results demonstrating that FrOPID controllers outperform traditional PID controllers, particularly when combined with HHO in terms of both rise time and settling time, especially under varying vehicle mass conditions and tire friction.

The paper is organized as follows: Section 2 introduce the literature review. Section 3 details the mathematical modeling of vehicle cruise control systems. Section 4 presents the controller designs. Section 5 describes the various metaheuristic optimization methods. Section 6 introduces the proposed fractionalized PID controller design, optimized using the Integral Time Absolute Error (ITAE) criterion. In Section 7, comprehensive computer simulations demonstrate the superiority of the proposed algorithm through rigorous comparisons with previous studies. Finally, conclusions are drawn in Section 8.

3 Vehicle Cruise Control System (VCCS)

The cruise control systems face challenges in maintaining driver-set speed due to inclines and wind resistance, which can be disrupted by gravitational forces and wind resistance, as illustrated in a schematic diagram given in figure 1.

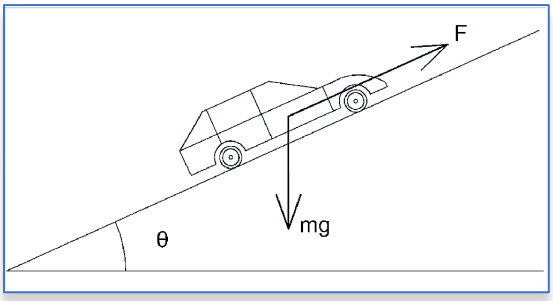

Figure 1. VCCS Model.

A vehicle cruise control system (VCCS) regulates the velocity of the vehicle by employing a predetermined speed as a reference (v_{ref}). Consequently, the vehicle's velocity (v) is sustained by adjusting the engine throttle input (u).

The linearized model that establishes the relationship between the output velocity (V) and the control input (U) can be derived as follows [9]:

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

where C_1 and C_a denote the actuator constant and aerodynamic drag coefficient, respectively, whereas τ and

T are the driver's reaction and observation times, respectively. The vehicle mass is represented by M. The block diagram of an VCCS and the model parameters used for this study can be found in [9].

Table 1 lists the factors and numbers for the automobile cruise system simulated in this paper [9,10]:

Table 1. VCCS parameters.

Parameter	Value
C_1	743
C_a	1.19 N/(m/s)^2
M	1500 Kg
τ	0.2 s
T	1 s
F_{dmax}	3500 N
F_{dmin}	-3500 N
g	9.81 m/s^2

Given the operating point $v = 30\ km/h$ and referring to Table 1, we can determine the the plant transfer function $G(s)$.

$$G(s) = \frac{\Delta V(s)}{\Delta U(s)} = \frac{2.4767}{(s + 0.0476)(s + 1)(s + 5)} \quad (2)$$

4 Controllers Design

4.1 Proportional Integral Derivative (PID) controller

Traditional PID controllers are widely utilized in process industries due to their straightforward design, robustness, and easily comprehensible regulatory processes. Despite the fluctuating dynamic behavior of process plants, conventional PID controllers can deliver excellent control performance.

The transfer function of classical PID controller is represented by equation (3).

$$G_c(s) = K_p + K_i s^{-1} + K_d s \quad (3)$$

Where K_p, K_i and K_d, are the proportional, integral, and derivative gains, respectively.

Then,

$$G_{IOPID}(s) = K_p(1 + \frac{1}{T_i s} + T_d s) \quad (4)$$

Where

T_i : is the integral time constant

T_d: is the derivative time constant.

4.2 Fractionalized Order PID Controller

The transfer function of a traditional PID controller is represented by the following equation:

$$G_c(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \quad (5)$$

The enhancement of fractionalization in the control system element alters the PID control rule, resulting in the fractionalization of the integral operator $1/s$ [15],[17],[29]:

$$\frac{1}{s} = \frac{1}{s^\alpha} \cdot \frac{1}{s^{1-\alpha}} \quad (6)$$

The fractionalization of the classical PID controller to be developed is represented by [15],[30]:

$$G_c(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) = \frac{1}{s}\left(\frac{K_p T_i T_d s^2 + K_p T_i s + K_p}{T_i}\right)$$

$$= \frac{1}{s^\alpha s^{1-\alpha}}\left(\frac{K_p T_d s^2 + K_p T_i s + K_p}{T_i}\right) \quad (7)$$

Where, $0 < \alpha < 1$.

5 Metaheuristic Optimization Algorithms

This section provides a concise summary of the heuristic methods proposed for classical PID and fractionalized order PID:

5.1 Harris Hawks Optimization (HHO) Algorithm

The Harris Hawks Optimization algorithm is inspired by the hunting behavior of Harris hawks. These hawks exhibit cooperative and surprise-based tactics to capture prey, mimicking various dynamic chasing strategies. The HHO algorithm uses these strategies to find an optimal solution to a problem. Below the different steps of the HHO algorithm:

Step 1: Initialization
Objective: Define the objective function $f(x)$ to be minimized (or maximized).

Initialize population: Generate an initial population of N hawks randomly within the defined search space. Each hawk represents a candidate solution and is denoted as x_i, where $i = 1, 2, \ldots, N$.

Set parameters: Define the maximum number of iterations (MaxIter) and other relevant parameters like the number of hawks and search space bounds.

Step 2: Evaluate Fitness

Evaluate the fitness of each hawk based on the objective function $f(x)$. The best fitness among all hawks represents the current best position of the prey, denoted as X_{rabbit} .

Step 3: Hunting Strategies Based on Energy E

Escape energy of prey E: The energy of the prey decreases as iterations progress, defined as:

$$E = 2E_0\left(1 - \frac{t}{T}\right) \quad (8)$$

where E_0 is a random number between -1 and 1, t is the current iteration, and T is the maximum number of iterations. The value of E decides the type of hunting strategy.

Step 4: Update Hawks' Positions

The hawks perform different movements based on the energy E. There are two main cases:

Case 1: $|E| >= 1$ (Exploration phase)

When the prey is energetic and still escaping, the hawks explore the search space using random movements. In this phase, the hawks' positions are updated as:

$$X(t+1) = \begin{cases} X_{\text{rand}}(t) - r_1|X_{\text{rand}}(t) - 2r_2X(t)| & q \geq 0.5 \\ X_{\text{rabit}}(t) - X_m(t) - r_3\left(L_b + r_4(U_b - L_b)\right) & q < 0.5 \end{cases} \quad (9)$$

where $X(t+1)$ represents the Hawks' position in the subsequent iteration, $X_{\text{rabbit}}(t)$ indicates the rabbit's position, $X(t)$ denotes the vector indicating the current position of the hawks, (r_1, r_2, r_3, r_4) denote random numbers within the range of(0,1), and (L_b, U_b) represent the variables of lower and upper bounds.

Case 2: $|E| < 1$ (Exploitation phase)

The final phase of the HHO algorithm consists of four distinct strategies, each contingent upon the energy level of the prey and the likelihood of its escape. When considering $r < 0.5$ as indicative of the prey's successful escape chance and $r \geqslant 0.5$ as representing an unsuccessful escape attempt: - In cases where $r \geqslant 0.5$ and $|E| \geq 0.5$, a soft besiege strategy will be executed, characterized by Equations (10) and (11).

$$X(t+1) = \Delta X(t) - E|JX_{\text{rabbit}}(t) - X(t)|, \quad (10)$$

$$\Delta X(t) = X_{\text{rabbit}}(t) - X(t), \quad (11)$$

where ΔX(t) represents the disparity between rabbit's location and current location at iteration t, while J denotes the magnitude of the rabbit's random jump.

- For $r \geqslant 0.5$ and $|E \leq 0.5|$, A severe besiege will be undertaken, as stated by Eq.(12).

$$X(t+1)X_{\text{rabbit}} - E|\Delta X(t)| \quad (12)$$

For $r < 0.5$ and $|E| \geq 0.5$, A mild besiege with gradual quick drive will be executed, as stated by Eqs.(13) and (14).

$$Y_1 = X_{\text{rabbit}} - E|JX_{\text{rabbit}} - X(t)|, \quad (13)$$

$$Z_1 = Y_1 + S \times LF(D), \quad (14)$$

where D is the dimension of the problem, S is a $1xD$ random vector, and LF is the levy flight function.

Therefore, Eq. (15) fulfills the position update.

$$X(t+1) = \begin{cases} Y_1, & \text{if } F(Y_1) < F\big(X(t)\big) \\ Z_1, & \text{if } F(Z_1) < F\big(X(t)\big) \end{cases} \quad (15)$$

- If both r and $|E|$ have values lower than a certain threshold, a hard besiege with increasing quick drive will be used, as described by Equations (16)-(18).

$$X(t+1) = \begin{cases} Y_2, & \text{if } F(Y_2) < F\big(X(t)\big) \\ Z_2, & \text{if } F(Z_2) < F\big(X(t)\big) \end{cases} \quad (16)$$

Y_2 and Z_2 are obtained using (17) and (18), respectively.

$$Y_2 = X_{\text{rabbit}} - E|JX_{\text{rabbit}} - X(t)|, \quad (17)$$

$$Z_2 = Y_2 + S \times LF(D). \quad (18)$$

Step 5: Evaluate New Solutions

Evaluate the fitness of each hawk after updating their positions. If a hawk's new position provides a better solution, update the current best position. X_{rabbit} .

Step 6: Check Stopping Criterion

If the maximum number of iterations MaxIter is reached or the desired solution is found, stop the algorithm.

Step 7: Return the Best Solution

Return the position X_{rabbit} as the optimal solution found by the algorithm.

For illustration, the overall flowchart of the HHO algorithm is presented in Figure. 2.

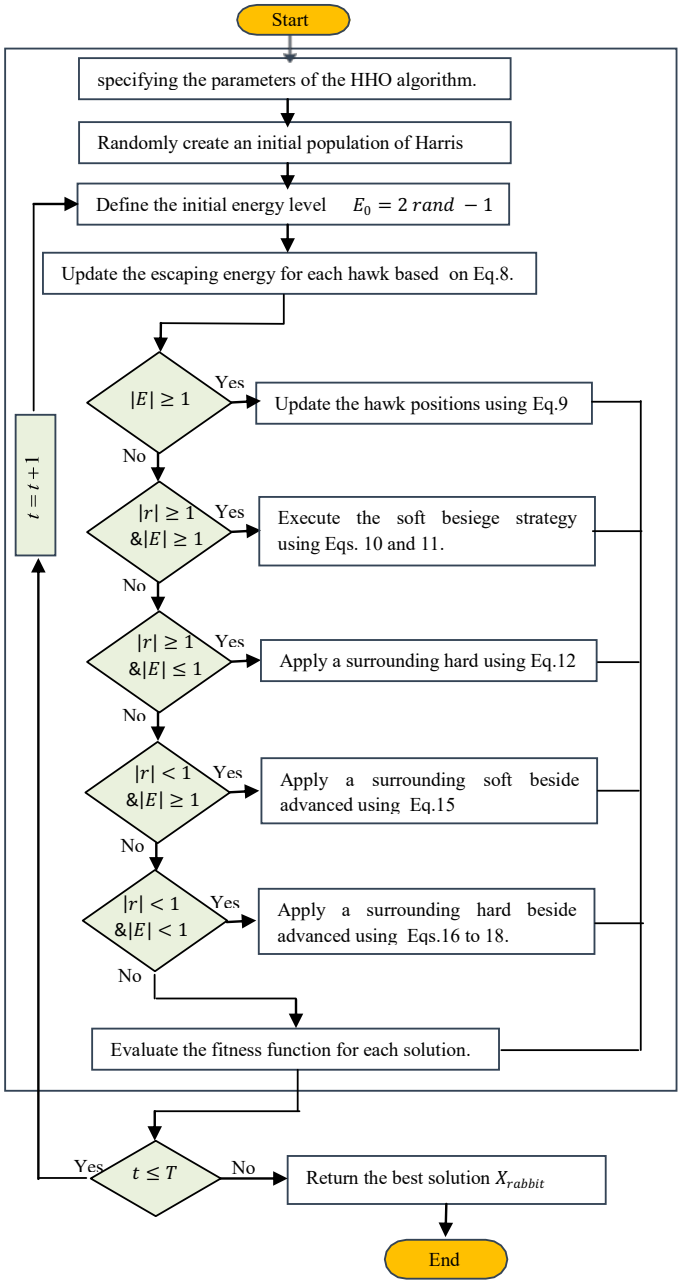

Figure 2. Flowchart of the HHO algorithm

5.2 Genetic Algorithm (GA)

The Genetic Algorithm is a bio-inspired optimization technique based on the principles of natural selection and genetics. It aims to find the optimal solution to a problem by evolving a population of candidate solutions over several generations. Below is a detailed explanation of each step:

Step 1: Initialization

- ***Objective***: Define the objective function $f(x)$ to be optimized.
- ***Generate the initial population***: Create an initial population of N individuals (candidate solutions), each represented by a chromosome (a set of encoded parameters). The chromosome can be a binary string, real numbers, or any other representation suitable for the problem.
- **Set parameters**: Define the population size (N), number of generations $(MaxGen)$, crossover probability (p_c), mutation probability (p_m), and other relevant parameters.

Step 2: Evaluate Fitness

- Evaluate the fitness of each individual in the population using the objective function $f(x)$.
- The fitness value indicates how good each candidate solution is.

Step 3: Selection

- Select parent individuals from the current population based on their fitness. This process is usually stochastic, with better solutions having a higher chance of being selected.

Step 4: Crossover (Recombination)

- Perform crossover between selected parent individuals to generate new offspring (children).

Step 5: Mutation

- Apply mutation to offspring with a probability p_m. Mutation introduces random changes to individual genes in the chromosome to maintain genetic diversity and prevent premature convergence.

Step 6: Evaluate New Population

- Calculate the fitness of the new offspring population.
- Combine the offspring with the current population if needed, depending on the chosen GA strategy.

Step 7: Replacement

- Select individuals for the next generation based on fitness.

Step 8: Check Stopping Criterion

- If the maximum number of generations $(MaxGen)$ is reached or if the improvement in fitness is below a certain threshold, stop the algorithm.

Step 9: Return the Best Solution

Return the best individual from the final population as the optimal solution.

5.3 Particle Swarm Optimization (PSO) Algorithm

The Particle Swarm Optimization (PSO) algorithm is a widely utilized optimization method derived from the social behaviors shown by birds in flocks or fish in schools. It identifies the ideal answer by progressively enhancing a population of possible solutions referred to as particles. These particles explore the search space by updating their positions and velocities according to their own experiences and those of their neighbors.

Below is a detailed sequential analysis of the PSO algorithm:

Step 1: Initialization

Objective: Define the objective function $f(x)$ to be minimized (or maximized). Randomly initialize the position and velocity of each particle within the search space. Each particle i has a position x_i and velocity v_i. Set parameters such as the number of particles N, maximum iterations (MaxIter), inertia weight (w), cognitive coefficient (c_1), and social coefficient (c_2).

Step 2: Evaluate Fitness

Evaluate the fitness of each particle using the objective function $f(x)$. Identify the personal best position $(pbest_i)$ for each particle and the global best position $(gbest_i)$ among all particles.

Step 3: Update Velocities

Update the velocity of each particle using the equation:

$$v_i^{k+1} = w_i v_i^k + c_1 r_1 \left(pbest_i - x_i^k\right) + c_2 r_2 \left(gbest_i - x_i^k\right) \quad (19)$$

where r_1 and r_2 are random values between 0 and 1. w_i is the inertia weight that controls the influence of the previous velocity. While c_1 and c_2 are cognitive and social coefficient respectively.

Step 4: Update Positions

Update the position of each particle using the equation:

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (20)$$

Step 5: Evaluate New Fitness

- Evaluate the fitness of each particle at its updated position.
- If a particle's new position yields a better fitness, update its personal best $(pbest_i)$.
- Update the global best $(gbest)$ if necessary.

Step 6: Check Stopping Criterion

Stop the algorithm if the maximum number of iterations is reached or the change in fitness values is below a predefined threshold.

Step 7: Return the Best Solution

Return the global best position $(gbest)$ as the optimal solution.

6. Proposed Design Procedure and FrOPID Controlled VCCS

Figure 3 describes a vehicle cruise control system using a PID and FrOPID feedback loop. In this system, $G(s)$ and $G_{FrOPID}(s)$ represent the plant and controller models, respectively.

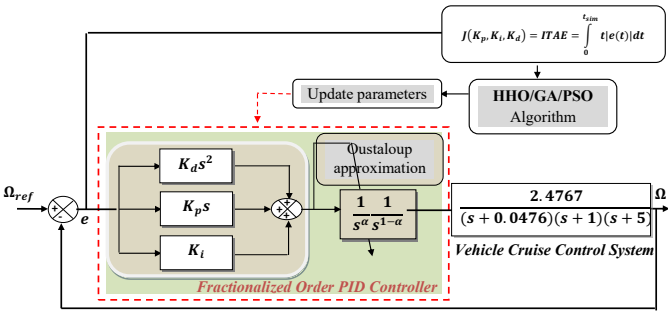

Figure 3. Proposed HHO/GA/PSO based FrOPID Controller.

The models apply the ITAE objective function to enhance key performance indicators. The controller manages the output Ω (speed), based on the input Ω_{ref} (reference speed), ensuring that the system remains stable and performs as desired, even in the presence of external disturbances $D(s)$.

The design of a traditional PID controller is represented by equation 4 as described in section 4.1, where:

$$G_{PID}(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s\right) \quad (21)$$

The enhancement of classical PID controller is represented by equation 22 as discribed in section 3.2,

$$G_{FrOPID}(s) = \frac{1}{s^\alpha s^{1-\alpha}} \left(\frac{K_p T_i T_d s^2 + K_p T_i s + K_p}{T_i}\right) \quad (22)$$

Where, $0 < \alpha < 1$.

In this study, ITAE is chosen as the performance criterion for different optimization methods. Calculation of the error $e(t)$ involves determining the difference between the reference model and the actual model. A

smaller error value indicates closer approximation to the desired controller parameters.

$$J(K_p, K_i, K_d) = ITAE = \int_0^{t_{sim}} t|e(t)|dt \quad (23)$$

The letter J denotes the performance criteria, indicating the degree of resemblance between the controlled object and the reference model, while $e(t)$ represents the error signal. Here, $e(t)$ corresponds to the disparity between the reference speed and the actual speed $(v_{ref}(t) - v(t))$. The simulation time (t_{sim}) was set to 10 seconds for this investigation.

7. Numerical Simulation Results and Discussion

This section presents the simulation results of the proposed architecture, which is based on the linearizing feedback and the FPID controller. The simulation process was carried out using MATLAB, a powerful tool for modeling and simulating control systems. Table 2 lists the parameters of the proposed HHO, GA and PSO algorithms.

The classical PID and the proposed Fractionalized Order PID (FrOPID) controllers for vehicle cruise control are optimized by three metaheuristic algorithms—Harris Hawks Optimization (HHO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO). The controller's performance is compared against that of a classical PID controller optimized by various contemporary methodologies.

The unity feedback closed-loop transfer function of the fractionalized-order PID controller, optimized using HHO, GA, and PSO algorithms, incorporates an integrator with a fractional order $\alpha = 0.5$. This fractional order is approximated using the Oustaloup technique. The approximation parameters are: are $\omega_b = 0.01$ rad/s, $\omega_h = 1000$ rad/s and $N = 5$ (filter order).

Table 2. Optimization Parameters for HHO, GA, and PSO Algorithms

Algorithm	Parameter	Value
HHO	Population Size	50
	Maximum Iterations	40
	Lower bounds $[K_p, K_i, K_d]$	[0.01;0.0
	Upper bounds $[K_p, K_i, K_d]$	[5;5; 5]
	Time of simulation	5s
GA	Population Size	40
	Crossover Probability	0.8
	Mutation Probability	0.125
	Maximum Iterations	25
	Lower bounds $[K_p, K_i, K_d]$	[0.01;0.0
	Upper bounds $[K_p, K_i, K_d]$	[5;5; 5]
	Time of simulation	5s
PSO	Population Size	50
	Maximum Iterations	40
	Acceleration Constants (c1, c2)	2
	Lower bounds $[K_p, K_i, K_d]$	[0.01;0.0
	Upper bounds $[K_p, K_i, K_d]$	[5;5; 5]

The parameters and corresponding values for these approaches are detailed in Table 3.

Table 3. Proposed controller's gain parameters and other controllers compared.

Controllers	K_p	K_i	K_d	α
HHO/FrOPID [Proposed]	4.1132	0.1714	4.2564	0.5
GA/FrOPID [Proposed]	3.5907	0.1630	3.3021	0.5
PSO/FrOPID [Proposed]	3.9578	0.1798	3.8583	0.5
HHO/PID	4.1132	0.1714	4.2564	1
GA/PID	3.5907	0.1630	3.3021	1
PSO/PID	3.9578	0.1798	3.8583	1

7.1 Transient and frequency stability analysis

Figure 4 presents a comparison of step responses among different controllers, focusing on their performance over time. It includes proposed fractionalized-order PID (FrOPID) controllers optimized using three metaheuristic algorithms—Harris Hawks Optimization (HHO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO)—as well as traditional integer-order PID controllers optimized with the same algorithms.

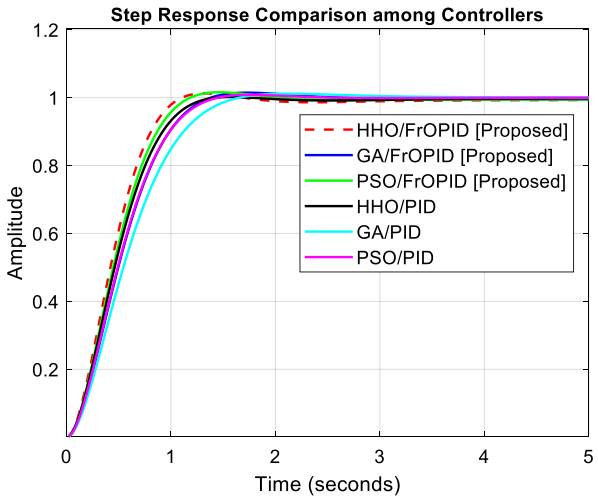

Figure 4. Step response of various optimizer-based controller schemes for ACCS

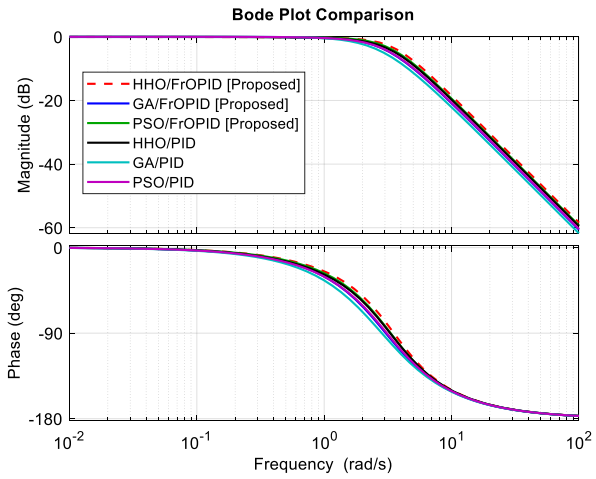

Figure 5. Bod plots of various optimizer-based controller schemes for VCCS

As can be seen from figure 4, the results show that the proposed FrOPID controllers consistently outperform their traditional counterparts. Specifically, the HHO/FrOPID controller, represented by a red dashed line, achieves a faster rise to the target value and settles more quickly, indicating a better response with minimal overshoot. This suggests a higher level of control stability and responsiveness. Meanwhile, the GA/FrOPID and PSO/FrOPID controllers also demonstrate improvements, though their rise and settling times are slightly longer compared to the HHO/FrOPID configuration.

In contrast, the traditional PID controllers, represented by solid lines, exhibit slower rise times and more pronounced overshoots. This comparison highlights the effectiveness of using fractionalized-order controllers optimized with advanced algorithms, particularly HHO, to enhance system performance. Overall, the HHO/FrOPID combination proves to be the most efficient in achieving a balance between responsiveness and stability, showcasing the robustness and adaptability of fractionalized PID controllers in dynamic environments.

Figure 5 presents a Bode plot comparison between proposed fractionalized-order PID (FrOPID) controllers and traditional PID controllers, each optimized using Harris Hawks Optimization (HHO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO). The Bode plot consists of two subplots: one showing the system's magnitude response (in dB) across a range of frequencies, and the other depicting the corresponding phase response (in degrees).

In the magnitude plot, the proposed FrOPID controllers maintain a higher gain at lower frequencies and show a smoother decline in gain as frequency increases. Among these, the HHO/FrOPID controller stands out with the most stable gain response across the frequency range, reflecting greater robustness to low-frequency disturbances. In contrast, the traditional PID controllers exhibit a steeper decline in magnitude at higher frequencies, indicating relatively less effective performance in maintaining stability at different frequency ranges.

In the phase plot, the proposed controllers demonstrate a more gradual and consistent phase shift across the frequency spectrum. The HHO/FrOPID controller, in particular, shows minimal phase deviations, which indicates better overall stability and response to dynamic changes in the system. Meanwhile, traditional PID controllers experience more abrupt drops in phase at certain frequencies, suggesting greater susceptibility to disturbances and reduced frequency stability.

These characteristics indicate that the proposed fractionalized-order PID controllers provide improved stability margins and more consistent performance across a wide frequency range compared to traditional PID controllers. This consistency in both magnitude and phase responses highlights the robustness and adaptability of the fractionalized-order designs, especially when optimized with the HHO algorithm. The comparison supports the conclusion that fractionalized-order controllers are more effective in handling system dynamics and uncertainties, with the HHO/FrOPID configuration being particularly advantageous.

To demonstrate the enhanced efficacy of the suggested technique, we have provided comparative numerical data for the VCCS in both the transient and frequency domains. The results are shown in Tables 4 and 5.

Table 4 compares the transient response metrics; rise time (T_r), settling time (T_s), and overshoot ($OS\%$) for different controller types, specifically fractionalized-order PID controllers (FrOPID) optimized by HHO, GA, and PSO, and traditional PID controllers. While Table 5 presents frequency response metrics; gain margin (G_m), phase margin (φ_m), and bandwidth (B_W) across the same set of controllers.

Table 4. Performance comparison of transient responses

Controllers	$\boldsymbol{T_r(s)}$	$\boldsymbol{T_s(s)}$	$\boldsymbol{OS}$ (%)
HHO/FrOPID	0.65467	0.98394	1.8203
GA/FrOPID [Proposed]	0.80665	1.8292	2.0597
PSO/FrOPID	0.70573	1.5615	2.1153
HHO/PID	0.76319	1. 1962	0.17565
GA/PID	0.93938	1. 4559	1. 1456
PSO/PID	0.82074	1.272	0.82093

Table 5. Performance comparison of frequency responses

Controllers	$\boldsymbol{G_m(dB)}$	$\boldsymbol{\varphi_m(°)}$	$\boldsymbol{B_w(Hz)}$
HHO/FrOPID	Inf	Inf	3.3323
GA/FrOPID [Proposed]	Inf	Inf	2.6398
PSO/FrOPID [Proposed]	Inf	Inf	3.0549
HHO/PID	Inf	-180	2.8663
GA/PID	Inf	-180	2.2657
PSO/PID	Inf	-180	**2.6264**

As can be seen from Tables 4 and 5, the fractionalized order PID controllers (especially HHO/FrOPID) outperform traditional PID controllers in both transient and frequency responses, demonstrating faster rise times and higher bandwidth. This suggests that fractionalized-order controllers are better suited for systems requiring both fast response and high stability, while traditional PID controllers may still be preferable in applications where minimizing overshoot is critical.

Table 6 effectively compares three controllers (HHO/FrOPID, GA/FrOPID, and PSO/FrOPID) across different performance criteria (rise time, settling time, and overshoot) for integrator orders (α) ranging from 0.1 to 0.5.

Table 6. Comparative Transient Response Results for Different Controllers and Integrator Orders α)

Criterion	$\boldsymbol{\alpha}$	**HHO/FrOPID**	**GA/FrOPID**	**PSO/FrOPID**
Rise time [s]	0.1	0.71961	0.88694	0.77530
	0.2	0.69029	0.85087	0.74403
	0.3	0.67029	0.82609	0.72225
	0.4	0.65845	0.81144	0.70992
	0.5	0.65467	1.45590	1.14560
Settling time [s]	0.1	1.10620	1.36030	1.18530
	0.2	1.04970	1.29520	1.12750
	0.3	1.01240	1.25100	1.08880
	0.4	0.99079	1.79260	1.53930
	0.5	0.98394	1.82920	1.56150
Overshoott [%]	0.1	0.76674	1.43020	1.24420
	0.2	1.20670	1.68500	1.60290
	0.3	1.54050	1.88570	1.87980
	0.4	1.75130	2.01520	2.05500
	0.5	1.82030	2.05970	2.11530

As can be seen from table 6, the The HHO/FrOPID demonstrates the lowest rising time and settling time, signifying a more rapid reaction in comparison to the GA/FrOPID and PSO/FrOPID. Nonetheless, GA/FrOPID demonstrates elevated overshoot values, perhaps resulting in diminished stability. PSO/FrOPID achieves a compromise between rising time and overshoot; nonetheless, it is generally surpassed by HHO/FrOPID in most instances. These results underscore the efficacy of HHO/FrOPID for applications necessitating both rapidity and stability.

7.2 Robustness analysis

7.2.1 Effect of changing vehicle mass

In this subsection, parameter variations are introduced by adjusting the vehicle mass to simulate the effects of fuel consumption. The transfer functions are updated based on the new vehicle mass, and both the nominal and varied-mass systems are compared using step responses and transient response analysis.

Figure 6 shows the comparison between nominal vehicle mass and varied vehicle mass for six different controller types, including both traditional PID and fractionaliazed order PID controllers optimized by HHO, GA, and PSO. The nominal mass system responses are

represented by dashed lines while the systems with varied mass are shown in solid lines.

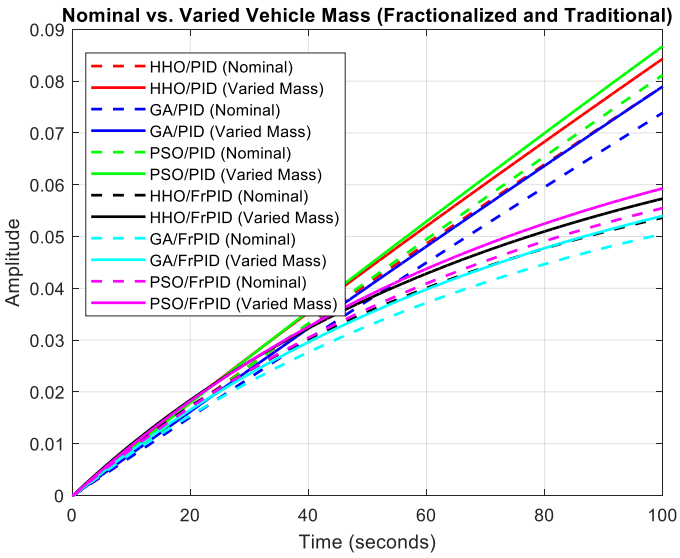

Figure 6. Comparison between nominal vehicle mass and varied vehicle mass

Figure 6 shows that all controllers exhibit a similar rising pattern, with fractional-order PID controllers (FrOPID) achieving faster rise times compared to traditional PID controllers. As shown in Figure 6, the fractional-order versions of the controllers demonstrate slightly better performance, with faster rise and settling times than their traditional PID counterparts, indicating improved handling of parameter variations, such as mass changes.

Table 7 presents the transient response metrics (RiseTime (T_r), SettlingTime (T_s), Overshoot (OS)) for nominal versus varied vehicle mass, clearly illustrating the controllers' performance under both conditions.

- **For Rise Time**: Fractionalized Order PID controllers achieve significantly faster rise times, around 175-178 seconds, compared to traditional PID controllers, which exceed 2400 seconds for varied mass. The HHO/FrOPID controller, with a rise time of 175.46 seconds for varied mass, stands out as the fastest among all controllers.

- **For Settling Time**: Similarly, fractional PID controllers settle much quicker, within the range of 313-318 seconds, while traditional PID controllers require over 4000 seconds to settle. Once again, HHO/FrOPID exhibits the best settling time performance.

- **For Overshoot:** All controllers demonstrate zero overshoot, indicating stable performance without oscillations in both nominal and varied mass scenarios.

Table 7. Performance comparison of frequency responses

Controllers	$T_r(s)$	$T_s(s)$	OS (%)
HHO/PID	2721.2	4843.5	0.0000
HHO/PID	2540	4520.9	0.0000
GA/PID	2856.7	5087	0.0000
GA/PID	2666.3	4747.8	0.0000
PSO/PID	2589 . 7	4611.6	0.0000
PSO/PID	2417.1	4304.1	0.0000
HHO/FrOPID	176.39	315.11	0.0000
HHO/FrOPID	175.46	313.47	0.0000
GA/FrOPID	178.57	318.22	0.0000
GA/FrOPID	177.66	316.6	0.0000
PSO/FrOPID	177.33	315.93	0.0000
PSO/FrOPID	176.35	314.17	0.0000

In summary, fractionalized-order PID controllers (FrOPID) outperform traditional PID controllers in terms of both rise time and settling time, with HHO/FrOPID being the most efficient, particularly under mass variations. This underscores the robustness and adaptability of fractional PID controllers to changes in vehicle mass.

7.2.2 Effect of Tire Friction

To simulate how a vehicle cruise control system is affected by tire friction, a k_f friction coefficient is added, then the transfer function would be updated by adding this coefficient to the damping terms. Assuming the damping term is $s + b$, where b is the original damping, we modify this to $s + b + k_f$, where k_f is the friction effect ($k_f = 0.1$).

Figure 7 compares the performance of various controllers in the presence of tire friction, showing how quickly they reach steady-state after a disturbance.

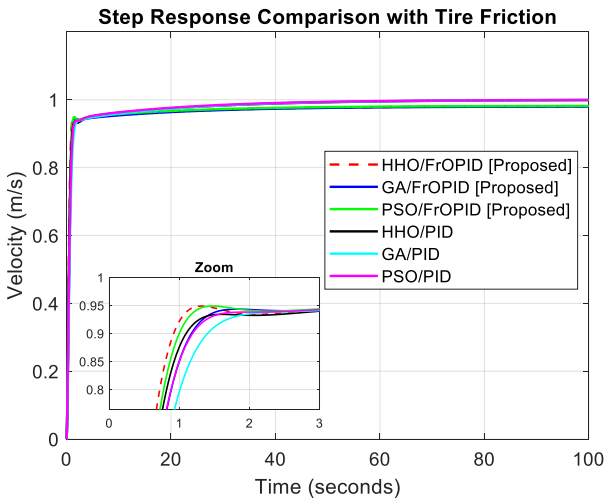

Figure 7. Response to tire friction

The proposed fractional PID controllers, like HHO/FrOPID and PSO/FrOPID, respond faster than traditional PID controllers, with HHO/FrPID having the shortest rise time and PSO/FrOPID showing the quickest settling time. All controllers avoid overshoot, ensuring stability.

Table 8 presents the transient response metrics under tire friction, highlighting the performance differences between various controllers in terms of rise time, settling time, and overshoot.

Table 8. Performance comparison of transient responses

Controllers	$T_r(s)$	$T_s(s)$	OS (%)
HHO/FrOPID [Proposed]	0.72735	14.877	0.0000
GA/FrOPID [Proposed]	0.93062	16.448	0.0000
PSO/FrOPID [Proposed]	0.79483	14.348	0.0000
HHO/PID	0.9381	25.863	0.0000
GA/PID	1.2189	26.076	0.0000
PSO/PID	1.0218	23.78	0.0000

As can be seen from table 8,

- **For Rise Time**: The time it takes for the system to go from 0% to 90% of the final value. The proposed HHO/FrOPID controller has the fastest rise time at 0.727 seconds, while the traditional PID controllers have longer rise times, with GA/PID controller being the slowest at 1.219 seconds.

- **For Settling Time**: The time it takes for the system to stabilize within a small range around the final value. The proposed PSO/FrOPID controller shows the shortest settling time of 14.348 seconds, with the traditional controllers settling much later, with GA/PID controller taking 26.076 seconds.

- **For Overshoot**: None of the controllers exhibit any overshoot, meaning the system does not exceed the desired value, indicating a well-damped response across all controllers.

In summary, Overall, the fractional PID controllers, especially those optimized by HHO and PSO, provide superior dynamic response and quicker stabilization in systems with tire friction compared to traditional PID controllers.

8. Conclusion

This paper demonstrates the superiority of fractionalized-order PID (FrOPID) controllers over traditional PID controllers in handling the complex dynamics of vehicle cruise control systems. By introducing an additional tuning parameter, the FrOPID controller significantly enhances adaptability and performance, particularly under diverse operational conditions. Through a comparative analysis employing three metaheuristic optimization algorithms—Harris Hawks Optimization (HHO), Genetic Algorithm (GA), and Particle Swarm Optimization (PSO)—the study identifies the HHO-optimized FrOPID as the most effective approach, achieving an ideal balance between responsiveness and stability. These findings underscore the promise of fractionalized-order controller as robust solutions for modern cruise control applications, capable of effectively managing non-linearity and variability within automotive systems.

Future studies will concentrate on experimentally validating the proposed FrOPID approach in real-world vehicle systems to confirm its practical effectiveness. Furthermore, investigating hybrid optimization methodologies to refine the tuning process may boost performance and reliability. Implementing this methodology in other intricate dynamic systems, such as robotics or industrial automation, would illustrate the adaptability of FrOPID controllers across other fields. Moreover, adding predictive models based on machine learning to controller design is also a great way to make control techniques that are smart and adaptable, which means that FrOPID can be used in more situations that are dynamic and unpredictable.

Competing Interest Statement

The authors declare no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data Availability Statement

No data or additional materials were utilized for the research described in the article.

References

[1] Z. Nie and H. Farzaneh, "Adaptive cruise control for eco-driving based on model predictive control algorithm," Applied Sciences, vol. 10, no. 15, pp. 5271, 2020.

[2] Chen,S.Xiong,Q.Chen,Y.Zhang,J.Yuetal.,"Eco-Driving:Ascientometric and bibliometric analysis," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 12, pp. 22716–22736, 2022.

[3] D. Izci and S. Ekinci, "A novel hybrid ASO-NM algorithm and its application to automobile cruise control system," in 2nd Int. Conf. on Artificial Intelligence: Advances and Applications, Singapore, pp. 333–343, 2022.

[4] K.Osman,M.F.RahmatandM.A.Ahmad,"Modelling and controller design for a cruise control system," in 2009 5th Int. Colloquium on Signal Processing & its Applications, Kuala Lumpur, Malaysia, pp. 254–258, 2009.

[5] Ang, K. H., Chong, G., & Li, Y. (2005). PID control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*, 13(4), 559-576.

[6] Silva, G. J., Datta, A., & Bhattacharyya, S. P. (2021). PID Controllers: History, theory, tuning, and application to modern process control. *Springer Nature*.

[7] Shah, P., & Patel, R. (2019). Comparative analysis of PID controller tuning methods for industrial processes. *International Journal of Industrial Electronics and Electrical Engineering*, 7(2), 51-57.

[8] Bansal, H.O., Sharma, R., Shreeraman, P., 2012. PID controller tuning techniques: a re-view. J. Control Eng. Technol.2, 168–176.

[9] Shafiee, M., Sajadinia, M., Zamani, A. A., & Jafari, M. (2024). Enhancing the transient stability of interconnected power systems by designing an adaptive fuzzy-based fractional order PID controller. *Energy Reports*, *11*, 394-411.

[10] Idir, A., Canale, L., Bensafia, Y., & Khettab, K. (2022). Design and robust performance analysis of low-order approximation of fractional PID controller based on an IABC algorithm for an automatic voltage regulator system. *Energies*, *15*(23), 8973.

[11] Mishra, A. K., Mishra, P., & Mathur, H. D. (2022). Enhancing the performance of a deregulated nonlinear integrated power system utilizing a redox flow battery with a self-tuning fractional-order fuzzy controller. *ISA transactions*, *121*, 284-305.

[12] Idir, A., Canale, L., Tadjer, S. A., & Chekired, F. (2022, June). High order approximation of fractional PID controller based on grey wolf optimization for DC motor. In *2022 IEEE International Conference on Environment and Electrical Engineering and 2022 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)* (pp. 1-6). IEEE.

[13] Abualigah, L., Ekinci, S., Izci, D., & Zitar, R. A. (2023). Modified Elite Opposition-Based Artificial Hummingbird Algorithm for Designing FOPID Controlled Cruise Control System. *Intelligent Automation & Soft Computing*, *38*(2).

[14] Bensafia, Y., Idir, A., Khettab, K., Akhtar, M. S., & Zahra, S. (2022). Novel robust control using a fractional adaptive PID regulator for an unstable system. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, *10*(4), 849-857.

[15] Idir, A., Bensafia, Y., Khettab, K., & Canale, L. (2023). Performance improvement of aircraft pitch angle control using a new reduced order fractionalized PID controller. *Asian Journal of Control*, *25*(4), 2588-2603.

[16] Idir, A., Bensafia, Y., & Canale, L. (2024). Influence of approximation methods on the design of the novel low-order fractionalized PID controller for aircraft system. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, *46*(2), 1-16.

[17] Idir, A., Akroum, H., Tadjer, S. A., & Canale, L. (2023, June). A comparative study of integer order PID, fractionalized order PID and fractional order PID controllers on a class of stable system. In *2023 IEEE International Conference on Environment and Electrical Engineering and 2023 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)* (pp. 1-6). IEEE.

[18] M.K.Rout,D.Sain, S. K. Swain and S. K. Mishra, "PID controller design for cruise control system using genetic algorithm," in 2016 Int. Conf. on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, pp. 4170–4174, 2016.

[19] D. Izci, S. Ekinci, M. Kayri and E. Eker, "A novel improved arithmetic optimization algorithm for optimal design of PID controlled and Bode's ideal transfer function based automobile cruise control system," Evolving Systems, vol. 13, no. 3, pp. 453–468, 2022.

[20] Hlangnamthip, S., Thammarat, C., Sinsukudomchai, C., & Puangdownreong, D. (2024, January). Optimal Tuning of **PIDA** Controller for Vehicle Cruise Control System by Modified Bat Algorithm. In *2024 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)* (pp. 108-111). IEEE.

[21] R. Pradhan and B. B. Pati, "Optimal FOPID controller for an automobile cruise control system,"in 2018 Int. Conf. on Recent Innovations in Electrical, Electronics & Communication Engineering (ICRIEECE), Bhubaneswar, India, pp. 1436–1440, 2018.

[22] Gunasekaran, P., Sivasubramanian, R., Periyasamy, K., Muthusamy, S., Mishra, O. P., Ramamoorthi, P., & Geetha, M. (2024). Adaptive cruise control system with fractional order ANFIS PD+ I controller: optimization and validation. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, *46*(4), 184.

[23] Abdulnabi, A. R. (2017). PID controller design for cruise control system using particle swarm optimization. Iraqi Journal for computers and Informatics (IJCI), 43(2), 30–35.

[24] Izci, D., & Ekinci, S. (2021, June). An efficient FOPID controller design for vehicle cruise control system using HHO algorithm. In *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)* (pp. 1-5). IEEE.

[25] Morovatdel, M., & Taraghi Osguei, A. (2024). Designing a PID Controller for a Cruise Control System using Genetic Algorithm. *Computational Sciences and Engineering*.

[26] Saravanan, G., Pazhanimuthu, C., Lalitha, B., Senthilkumar, M., & Kannan, E. (2024, June). Red Panda Optimization Algorithm-Based PID Controller Design for Automobile Cruise Control System. In *2024 International Conference on Smart Systems for Electrical, Electronics, Communication and Computer Engineering (ICSSEECC)* (pp. 33-37). IEEE.

[27] Pradhan, R., Majhi, S. K., Pradhan, J. K., & Pati, B. B. (2017). Performance evaluation of PID controller for an automobile cruise control system using ant lion optimizer. *Engineering Journal*, *21*(5), 347-361.

[28] Oleiwi, B. K., & Abood, L. H. (2024). Enhanced PD Controller for Speed Control of Electric Vehicle Based on Gorilla Troops Algorithm. *Journal Européen des Systèmes Automatisés*, *57*(4).

[29] Bensafia, Y., Khettab, K., & Idir, A. (2022). A novel fractionalized PID controller using the sub-optimal approximation of FOTF. *Algerian Journal of Signals and Systems*, *7*(1), 21-26.

[30] Benaouicha, K., Idir, A., Akroum, H., & Bensafia, Y. (2024). Fractionalized order PID controller design for three tanks liquid level control. *Studies in Engineering and Exact Sciences*, *5*(2), e8915-e8915.