INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 18, Issue: 4, Month: August, Year: 2023
Article Number: 5294, https://doi.org/10.15837/ijccc.2023.4.5294

CCC Publications

New Meta-heuristic - Based Approach for Identification and Control of Stable and Unstable Systems

M. Azegmout, M. Mjahed, A. El Kari, H. Ayad

Mohamed Azegmout
Department of Applied Physics
Cadi Ayyad University, Marrakech, Morocco
*Corresponding author: mohamed.azegmout@ced.uca.ma

Mostafa Mjahed
Department of Mathematics and Systems
Royal School of Aeronautics, Marrakech, Morocco
mjahed.mostafa@gmail.com

Abdeljalil El Kari
Department of Applied Physics
Cadi Ayyad University, Marrakech, Morocco
a.elkari@uca.ma

Hassan Ayad
Department of Applied Physics
Cadi Ayyad University, Marrakech, Morocco
h.ayad@uca.ma

Abstract

Nowadays, the use of meta-heuristic algorithms (MAs) for tackling complicated engineering issues has shown significant promise, therefore applying MAs to optimum model parameters and PID parameters can be quite beneficial. As a result, this paper looks at the capabilities of four recently released resilient MAs in optimizing model parameters and PID parameters for various system behaviors. Hence, these four meta-heuristic algorithms are used such as Ant Colony Optimization (ACO), Cultural Algorithm (CA), Invasive Weed Optimization (IWO), and Black Hole Algorithm (BHA). The key contribution of this study is the employment of many meta-heuristics at the same time with the same objective function while taking into consideration each algorithm parameters for identification and control, then compared to traditional techniques such as Least square (LS) and Reference Model (RM). Thus, the most efficient algorithm is the one that yields the lowest cost function, has the lowest standard deviation (SD), and uses the least amount of CPU time. Regarding identification, simulation findings showed that CA algorithm has the best cost, lowest standard deviation (SD) and fewest CPU time 2.7838e-13, 7.1108e-13 and 3.1395(s), respectively. As for control system, it is shown that created intelligent-based controllers are more dependable than reference model controllers in stabilizing the behaviors of the various examined processes, with the IWO algorithm finds the best gains of PID and converges the fastest with best cost 3.2905e-10 and CPU time 48.8732(s). Moreover, ACO and BHA both failed to achieve satisfactory results in terms of accuracy and CPU time compared to others algorithms. Additionally, studies also showed that optimization methods has good performance, resilient and effective.

Keywords: Identification, Automatic Control, Ant Colony Optimization (ACO), Invasive Weed Optimization (IWO), Cultural Algorithm (CA), Black Hole Optimization (BHA), PID, Least Squares, Reference Model.

1 Introduction

A number of scientific and technical disciplines commonly experience identification and control issues, they also do so in a variety of applications that depend more and more on sophisticated control techniques.

System identification is the first and most crucial step in the design of a controller. A controller can be created using a variety of control methods in accordance with a recognized system model to meet the necessary specifications. The choice of a suitable identification model and an assessment of the model's parameters build up system identification. Fortunately, most engineering systems and industrial processes have well-understood structures, making it easy to develop a certain type of models that can accurately capture the real system. As a result, the challenge of system identification is typically simplified to that of parameter estimation.

A fundamental method that is frequently used for parameter estimate is the least-squares approach [1], both the static and dynamic systems' parameters have been effectively identified using it. But only model structures of systems with the attribute of linearity in the parameters are appropriate. This strategy may be invalid if the model structure is not linear for the parameters [2]. Heuristic optimization approaches appear to be a more promising strategy and offer an effective way of solving this challenge [3, 4, 5, 6, 7]. They appear to be a viable replacement for conventional methods.

Proportional integral derivative (PID) controllers have long been employed in process industries because to their simplicity, usability and reliability. The precise and effective tuning of parameters is the main challenge for PID controllers. PID parameter tuning is complicated in reality due to the nonlinearity and time delay that are common in controlled systems. Numerous PID tuning techniques were put forth. Despite requiring a step input application with paused process, the Ziegler-Nichols (ZN) approach is experimental and frequently used [8]. The need for prior knowledge of plant models is one of the disadvantages of this approach. Once the controller has been tuned using the ZN approach, a good but not ideal system response will be obtained.

Numerous artificial intelligence (AI) approaches, including neural networks, fuzzy systems, and neuro-fuzzy logic, have been extensively used in the last 20 years to optimize PID controller settings [9, 10, 11]. Along with these methods, a variety of heuristic approaches have garnered a lot of attention recently for their high efficacy and capacity to find the best solution in a problem space. These meta-heuristic optimization algorithms, which include ACO, PSO, AFS, BFO, ABC and other algorithms, demonstrate their capacity to solve nonlinear design issues in practical applications across almost all branches of science, engineering, and industry, according to research that have been published, PSO and ACO are the two most commonly employed algorithms [12]. For examples of these applications include those that deal with transportation issues such as those involving unmanned aerial vehicles (UAVs) [13, 14, 15, 16], system identification [3, 4, 5, 6, 7], control system [17, 18, 19, 20], fault diagnosis, power systems [21, 22] and others application [12].

The PID controller design employing ACO, CA, IWO, and BHA in accordance with the identified system will also be covered in this work, these methods entail the use of data modeling and cost function with optimization techniques to adjust PID settings. These controller parameter techniques are based on a cost function that they wish to minimize, for tuning PID controller settings, there are six (6) commonly used cost functions (Integral Absolute Error (IAE), Integral Square Error (ISE), Integral Time Absolute Error (ITAE), Integral Time Square Error (ITSE), Mean Square Error (MSE), Integral Error (IE)) [17]. For instance, in [23] a novel hybrid Ant colony optimization for DTC control framework has been implemented, the simulation's results demonstrated the efficiency of the intelligent ACO-DTC control, which outperforms the traditional DTC in terms of speed, stability, accuracy, and torque ripples. For the best PID controller design, a improved Invasive weed optimization method (IWO) based on chaos theory has been developed, five benchmark functions and PID controller parameter for DC motor are used to examine the chaotic invasive weed development process, results on optimization issues demonstrate the faster convergence rate and higher accuracy of the improved chaotic weed algorithm [18]. In [19] the design of a quantitative PID controller using Black hole algorithm (BHA) is suggested for Ball and Beam system, According to the findings simulation, the suggested quantitative PID controller is capable of compensating the ball and beam system with stable behavior and preferable time response. For unmanned tilt-rotor flight control and trajectory tracking, proportional-derivative parameters based on particle swarm optimization (PSO) were developed in [14], for the best tuning of the controllers' settings, both Particle Swarm Optimization (PSO) and traditional Reference Model (RM) approaches are used and then compared. In [15], a PID quadcopter controller's parameters were determined via genetic algorithm optimization (GA) and compared to reference model. In [16] a comparative study of optimized based-control system of a single-rotor, medium-scale rotorcraft has been made by using cuckoo search (CS), ant colony optimization (ACO), particle swarm optimization (PSO) and genetic algorithm (GA) compared to manual tuning the four algorithm perform better, The ACO algorithm converges the quickest in hover trim settings and determines the optimum gains for the selected goal function, as for forward trim the optimum algorithm was determined to be GA. In both hover and forward flight, The two tuned flight controllers were successful in maintaining trim condition and regaining control after an external disruption.

Although the aforementioned MAs were frequently used to solve various engineering challenges, there is no one algorithm that can be used to find the best solution for all optimization issues. Certain algorithms could offer a superior solution to a specific problem, but not to others. So, it's important to assess each algorithm's suitability for a certain optimization issue. As a result, this study looks at how well four MAs perform when it comes to optimizing model and PID parameters for various system characteristics. Ant Colony Optimization (ACO), Cultural Algorithm (CA), Invasive Weed Optimization (IWO), and Black Hole Optimization (BHA) are some of these algorithms. to compare them we take into account their ease of use, lower run-time, more precise findings, fast convergence, high convergence accuracy, and good robustness.

This study is organized as follows: Section 2 describes the steps used to identify and control the analyzed stable and unstable processes using conventional approaches (least squares and reference model) and intelligent techniques (ACO, CA, IWO and BHA). Four chosen Meta-heuristic algorithms (MAs) are introduced and described in Section 2.3. In section 3, the recommended PID-based on ACO, CA, IWO, and BHA controllers for system stabilization are synthesized after the parameters of the models characterizing the four behaviors are calculated using four intelligent approaches. The results of the simulation are compiled in Section 4. Finally, a summary of this study is offered in the last section 5.

2 Conventional and Intelligent Methods Used for Identification and Control

The topic covered in this section is creating a PID controller for each process based on the reference model and four selected intelligent algorithms, while also estimating the model parameters using Least Square (LS), we examine four systems that represent four distinct types of behaviors in order to undertake this research.

We take into consideration the specified transfer function as a model equation for the four systems.

$$G_{mi}\left(p\right) = \frac{k{\omega_n}^2}{p^2 + 2\xi\omega_n p + {\omega_n}^2} \tag{1}$$

Therefore, in order to identify each system, the parameters k, ω_n and ξ must be estimated.

2.1 Least Square Method (LS)

The basic idea behind this approach is to choose a set of parameters for a model that will be built in a way that minimizes the sum of squares of the difference between the anticipated and actual values of the model [24]. The system's input-output connection is determined through model identification using algebraic differential equations. Since Friedrich Gauss's publication of his least squares estimations method [1], this methodology has been used. Linear, nonlinear, deterministic, and stochastic plants can all benefit from it. Controls, optimizations, and fault detection benefited considerably from this field's progress [25]. In order to provide an appropriate identification, transfer function models, state-space models, and polynomial models were calculated.

A vector of measurements is given for the parameter identification:

$$y = [y_1 y_2 \cdots y_N]^T \tag{2}$$

And utilizing the model and its parameters, we want to create an estimated vector:

$$\hat{y} = [\hat{y}_1 \hat{y}_2 \cdots \hat{y}_N]^T \tag{3}$$

Therefore, it is essential to develop a model that enables the calculation of the variable, the discrete version of this model is desired:

$$\hat{y}(k+1) = -\sum_{n=0} a_n y(k-n-1) + \sum_{m=0} b_m u(k-m+1) \tag{4}$$

n and m, respectively, stand for the denominator and numerator's degrees ($m < n$). θ is the vector of parameters that must be determined (5) in order to minimize the J criterion in (6).

$$\theta = [a_0, a_1, b_0, b_1] \tag{5}$$

$$J(\theta) = \sum_{i=n+1}^{N} (y(i) - \hat{y}(i))^2 \tag{6}$$

The following function in (7) expresses the second order model in discrete mode as a function of the parameters that need to be estimated, a_0, a_1, b_0 and b_1.

$$G(z) = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0} \tag{7}$$

The transfer function in (1) of $G_{mi}(p)$, when the sample period is set to T_e, has the following Z-transform:

$$G_m(z) = kk_1 \frac{(z - z_0)}{(z - \rho^{j\varphi})(z - \rho^{-j\varphi})} \tag{8}$$

with

$$\begin{cases} \rho = e^{-\xi\omega_n T_e}, \\ \varphi = T_e\omega_n\sqrt{1-\xi^2} = T_e\omega_p \end{cases}$$

And

$$\begin{cases} k_1 = 1 - \rho\left(\frac{\xi}{\sqrt{1-\xi^2}} + sin(\varphi) + cos(\varphi)\right) \\ z_0 = \frac{-1}{k_1}\rho^2 + \rho\left(\frac{\xi}{\sqrt{1-\xi^2}} + sin(\varphi) - cos(\varphi)\right) \end{cases}$$

By identifying equations (7) and (8), we can find the parameters of the vector provided by equation (9)

$$\begin{cases} a_0 = e^{-2\xi\omega_n T_e} \\ a_1 = -2\sqrt{a_0}cos(\omega_p T_e) \\ b_0 = ka_0 + k\sqrt{a_0}\left(\xi\frac{\omega_n}{\omega_p}sin(\omega_p T_e) - cos(\omega_p T_e)\right) \\ b_1 = k - k\sqrt{a_o}\left(\xi\frac{\omega_n}{\omega_p}\sin(\omega_p T_e) + \cos(\omega_p T_e)\right) \end{cases} \tag{9}$$

According to equation (9), we have:

$$\begin{cases} \xi\omega_n T_e = \frac{-log(a_0)}{2} \\ \omega_n T_e\sqrt{1-\xi^2} = cos^{-1}\left(\frac{-a_1}{2\sqrt{a_0}}\right) \end{cases} \tag{10}$$

mentioning that: $A = \frac{-log(a_0)}{2}$ $B = cos^{-1}\left(\frac{-a_1}{2\sqrt{a_0}}\right)$

The calculated model parameters of (1) are stated in (11) as follows:

$$\begin{cases} \xi = \pm\frac{1}{\sqrt{1+\frac{B^2}{A^2}}} \\ \omega_n = \frac{A}{\xi T_e} \\ k = \frac{b_0+b_1}{1+a_0+a_1} \end{cases} \tag{11}$$

2.2 The Reference Model Approach (RM)

The reference model control strategy entails modifying the behavior of a "n" order system by approximately converting its performance values to those of a first or second order. We want a behavior without overshoot O_v with a settling time T_{s0} for the control of the four processes examined in this section. The PID controller we employ, together with his transfer function $C(p)$ in (12), and its parameters K_p, K_i, and K_d.

$$C(p) = K_p + \frac{K_i}{p} + K_d p \tag{12}$$

The four systems closed loop transfer functions $F(p)$, denominators $D_F(p)$, and correctors $C(p)$ are identified, along with the desired characteristic (T_{s0}, τ), which is used to evaluate the denominator $D_{ref}(p)$ of the desired system:

$$D_{ref}(p) = \left(p + \frac{1}{\tau}\right)\left(p + \frac{\alpha}{\tau}\right)^{n-1};\ T_{s0} = 3\tau \tag{13}$$

The dominant pole is positioned at $(\frac{-1}{\tau})$ in the complex plane, while the other $(n-1)$ poles are positioned to its left. To ensure the poles' dominance at $(\frac{-1}{\tau})$, the parameter "α" is set much bigger than 1. The PID controller parameters are then obtained by equating the coefficients of $D_F(p)$ and $D_{ref}(p)$ [26].

2.3 Meta-heuristic Algorithms used

2.3.1 Ant Colony Optimization (ACO)

The Ant Colony Optimization Algorithm (ACO), is a graph-based meta-heuristic technique that has been used to resolve many challenging combinatorial optimization issues. In 1992, Marco Dorigo first proposed the core idea of ACO in his PhD thesis [27], the core concept of ACO is to represent the problem as the pursuit of the least expensive path across a graph. Artificial ants search this network in search of useful pathways. Because of their incredibly simple actions, ants typically only find roads independently that are of rather low quality. The global collaboration of the colony ants leads to the discovery of better pathways.

ACO has been utilized by several studies to examine PID tuning techniques and to enhance the performance of ACO [23]. After finishing a tour, each ant in the suggested method changes the pheromones left on the paths it walked, in accordance with local pheromone updating rules (14).

$$\tau(k)_{ij} = \tau(k-1)_{ij} + \frac{0.01\theta}{F} \tag{14}$$

Where F is the cost of fitness function for the ant's tour, θ is the global update coefficient for pheromones, and at iteration k, $\tau(k)_{ij}$ is the difference in pheromone value between $nest(i)$ and (j).

According to the global pheromone updating rule, the best and worst ant colony tours' pheromone pathways are changed according to equations (15) and (16):

$$\tau(k)_{ij}^{best} = \tau(k)_{ij}^{best} + \frac{\theta}{F_{best}} \tag{15}$$

$$\tau(k)_{ij}^{worst} = \tau(k)_{ij}^{worst} - \frac{0.3\theta}{F_{worst}} \tag{16}$$

With τ^{best} and τ^{wost} denote, respectively, the pheromones of the pathways the ant took on the route with the best F_{best} and worst F_{worst} costs.

The poorest tour of the iteration's trails has less pheromones, while the greatest tour of the colony's trails has much more pheromones. Since pheromone evaporation (16) enables the ant algorithm to forget its prior experience, ACO can then focus its search and avoid being trapped in local minima. Equation (17) expresses concentrations in terms of the evaporation constant λ [28].

$$\tau(k)_{ij} = \tau(k)_{ij}^{\lambda} + \left[\tau(k)_{ij}^{best} + \tau(k)_{ij}^{worst}\right] \tag{17}$$

2.3.2 Invasive Weed Optimization (IWO)

Weed algorithms take their inspiration from the way weeds grow naturally. This strategy was suggested in 2006 by Mehrrabian and Lucas [29]. Naturally occurring weed development poses a serious threat to beneficial plants because it spreads quickly. The IWO algorithm's optimization is based on the fact that weeds are recognized for having a high degree of flexibility and stability in nature.

The invasive weed meta-heuristic algorithm is a population-based optimization method that establishes the overall best value of a mathematical function by imitating the compatibility and irrationality of weed colony behavior. Colonies of invasive plants, which occur in agriculture, served as inspiration for this tactic. Unintentionally growing plants are known as weeds. Despite the fact that certain weeds may be beneficial and useful, weeds are defined as those that grow in a way that obstructs human needs and activities. A straightforward numerical optimization approach based on colonized weed was first published by Mehrabian and Lucas in 2006. They called it "The Invasive Weed Optimization Approach". Utilizing fundamental characteristics like spawning, growth, and competition, this approach is straightforward but effective in coming to optimal solutions in a weed colony [18].

Step 1 Initialization of population space : In equation (18) the IWO begins by creating a random $M \times N$ matrix, with M and N stand for the population number and the quantity of decision variables, respectively. The number of weeds and the decision variables are therefore represented by the rows and columns, in the randomly generated solution matrix:

$$Population = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_j \\ \vdots \\ X_M \end{bmatrix} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,i} & \cdots & x_{1,N} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,i} & \cdots & x_{2,N} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ x_{j,1} & x_{j,2} & \cdots & x_{j,i} & \cdots & x_{j,N} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ x_{M,1} & x_{M,2} & \cdots & x_{M,i} & \cdots & x_{M,N} \end{bmatrix} \tag{18}$$

With the $X_j = j$th solution and ith decision variable of the jth solution is $x_{j,i}$.

Step 2 Reproduction: Based on their fitness value, each seed matures into a blooming plant that subsequently produces seeds. From S_{max} to S_{min}, the amount of grass grains decreases linearly, as detaild in (19):

$$S_i = (S_{min} + \frac{S_{max} - S_{min}}{f_{min} - f_{max}} \times (f(X_j) - f_{max}), j = 1, 2, \ldots, M \tag{19}$$

Step 3 The Spread of Seeds: The following equation (20) provides the seeds produced by the typical distribution group, together with the average planting position and standard deviation (SD):

$$\sigma^t = \frac{(T-t)^{\beta}}{(T)^{\beta}}(\sigma_{initial} - \sigma_{final}) + \sigma_{final} \tag{20}$$

Where σ^t the current iteration's standard deviation, t current iteration, T max iterations, the initial and final standard deviation are $\sigma_{initial}$, σ_{final}, respectively, β is a nonlinear modulus, often known as a nonlinear modulation index, and it's selected by user as an algorithm parameters.

After assessing the standard division, the following new solutions are generated (21):

$$X_r^{(new)} = Mrand(0, \sigma_t) + X_j \ , j = 1, 2, \ldots \ M, \ r = 1, 2, 3, \cdots \ S_i \tag{21}$$

Where $Mrand(0, \sigma_t)$ is a random value matrix with mean 0 and standard division σ_t with dimension $1 \times N$, and X_j is the population's solution jth. Based on this, $X_r^{(new)}$ is the new solution rth.

Step 4 Competitive deprivation: The worst-fitting grass is removed from the colony, leaving a set number of plants, if the total number of grasses in the colony exceeds the maximum permissible number M_{max}.

Step 5 The technique is then repeated until the maximum number of iterations is achieved, saving the grass's minimal colony cost function each time.

2.3.3 Cultural Algorithm (CA)

Robert Reynolds (1994) established the concept of cultural algorithms [30], which are distinguished by being a subset of evolutionary computation that make use of the belief space knowledge mechanism. You may think of cultural algorithms as an expansion of genetic algorithms.

Cultural algorithms are dual inheritance systems that have two fundamental parts: "Population Space" and "Belief Space" [31]. The algorithm begins with a population space that is produced randomly and is denoted by $P_S(t)$. Additionally, suitable values are used to initialize the overall structure of the belief space given by $B_S(t)$. The $Obj()$ function is then used to assess the fitness functions of the people in $P_S(t)$. To keep the belief space $B_S(t)$ up to date using $Update()$ function, the $Accept()$ function selects from the $P_S(t)$ a predetermined number of people with superior fitness functions. In order to create a new generation of people, $P_S(t+1)$, the method $Influence()$ is employed. The following subsections provide an explanation of CA used for both applications.

The population space is experiencing the process of evolution, and it contains a variety of cultural data that is evolving with time. The belief space of CA serves as a cultural information repository and contains the knowledge and common behavior that individuals have learned. In contrast to the original version of CA, which only included one knowledge source (situational information) [30], the updated version now includes normative, historical, topographical, and domain knowledge sources. These cultural facts and knowledge will be employed as support materials to modify people's behavior in the future and push them in the direction of the search space's global optima, the most used knowledge component in CA are situational and normative knowledge sources [32] and we are going to focus on that two knowledge.

The belief space can be formally defined as in (22):

$$B^t = [S^t, N^t] \tag{22}$$

Where B^t stands for belief space, S^t for situational knowledge, N^t for normative knowledge and t denotes the number of iterations. Detailed explanations are given for the first two knowledge sources (Situational and Normative) which are used for our problem.

- Situational Knowledge

  The situational knowledge source stores the best information from each era of the population space, it represented by $S^t = [s_1, s_2, \cdots, s_n]$. The $Update()$ function updates the situational knowledge component (23):

$$S^{t+1} = \begin{cases} X_l^t & if\ f(X_l^t) < f(S^t) \\ S^t & otherwise \end{cases} , l = 1, 2, \ldots, n_{accepte} \tag{23}$$

  where $n_{accepte}$ is the amount of approved elite individuals required to update the belief space, and X_l^t is the lth accepted person at iteration t.

- Normative Knowledge

  Each problem dimension is represented by a set of data in normative knowledge, which may be expressed in equation (24):

$$N^t = \begin{bmatrix} I_1^t & I_2^t & \cdots & I_n^t \\ L_1^t & L_2^t & \cdots & L_n^t \\ U_1^t & U_2^t & \cdots & U_n^t \end{bmatrix} \tag{24}$$

  At iteration t, N^t indicate the normative knowledge source, the belief boundary of the problem's jth dimension is shown by $I_j^t = \left[x_{min,j}^t, x_{max,j}^t\right]$, the lower and upper normative boundaries of the problem's jth dimension are represented, respectively, by $x_{min,j}^t$ and $x_{max,j}^t$, The values of the fitness function for the lower and upper normative bounds are L_j^t and U_j^t, respectively.

  In order to avoid overly narrow belief intervals, CA takes a cautious approach while updating the normative knowledge source. $Update()$ function updates the following elements of the normative knowledge source:

$$x_{min,j}^{t+1} = \begin{cases} x_{i,j}^t & if\ x_{l,j}^t \leq\ x_{min,j}^t\ or\ f(X_l^t) < L_j^t \\ x_{min,j}^t & otherwise \end{cases} \tag{25}$$

$$x_{max,j}^{t+1} = \begin{cases} x_{i,j}^t & if\ x_{l,j}^t \geq\ x_{max,j}^t\ or\ f(X_l^t) < U_j^t \\ x_{max,j}^t & otherwise \end{cases} \tag{26}$$

$$L_j^{t+1} = \begin{cases} f(X_l^t) & if\ x_{l,j}^t \leq\ x_{min,j}^t\ or\ f(X_l^t) < L_j^t \\ L_j^t & otherwise \end{cases} \tag{27}$$

$$U_j^{t+1} = \begin{cases} f(X_l^t) & if\ x_{l,j}^t \geq\ x_{max,j}^t\ or\ f(X_l^t) < U_j^t \\ U_j^t & otherwise \end{cases} \tag{28}$$

  In the equations provided (25)(26)(27)(28), $l = 1, 2, \ldots, n_{accepte}$, X_l^t denotes the lth accepted person at iteration t, and $x_{l,j}^t$ signifies its jth variable.

Step 1 : Population space initialization

Population space $P_S(t)$ is initialized inside the search space in the following way:

$$X_{i,j}^0 = \phi(l_j, u_j), \quad i = 1, 2, \ldots, N\ ;\ j = 1, 2, \ldots, n \tag{29}$$

With $X_{i,j}^0$ denotes the jth component of the ith person in the population space N, l_j and u_j denote the variable's lower and upper bounds, $\phi()$ stands for the uniform random function, n is the number of variables.

Step 2 : Belief space initialization

The second step involves initializing the belief space $B_S(t)$ with the proper values and setting the iteration number $t = 0$, $B^0 = [S^0, N^0]$. Assuming, for instance, that $x_{min,j}^0 = -\infty$, $x_{max,j}^0 = \infty$, $L_j^0 = \infty$, and $U_j^0 = \infty$.

Step 3 : Fitness assessment

The objective functions of the people in the population space are assessed in this stage.

Step 4 : The updating of the belief space

This phase involves setting the iteration to $t = t + 1$ and updating the knowledge sources with data from the population space's people. Examples include updating the situational knowledge source using equation (23) and updating the normative knowledge source with equations (25)-(28).

Step 5 : The population space's influence

Then, to generate a new population of people $P_S(t+1)$, the knowledge contained in the belief space's knowledge sources is utilised. For instance, the effect of two types of information (situational and normative) may be utilized to decide the characteristics of a new generation of people. Situational knowledge is used to establish the direction of the search, while the normative knowledge component determines the step size. The following is a statement of this influence function:

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t + |size(I_j^t)N_{i,j}(0,1)| & if\ x_{i,j}^t < s_j^t \\ x_{i,j}^t - |size(I_j^t)N_{i,j}(0,1)| & if\ x_{i,j}^t > s_j^t \\ x_{i,j}^t + size(I_j^t)N_{i,j}(0,1) & otherwise \end{cases} \tag{30}$$

Step 6 : Criteria for stopping

Until the stopping requirement is met and the best solution is produced, repeat **Step 3**-**Step 5**.

2.3.4 Black Hole Algorithm (BHA)

Black hole optimization [33] is one of the most current strategies that has been created and effectively used to address optimization issues. It is a population-based strategy that employs a mechanism inspired by the phenomenon of black holes to guide the produced population toward the best solution.

Step 1 : Initialization

The suggested BHA employs a set of candidate solutions, referred to as the stars, in the issue of n dimensional search space, where each dimension is bounded by upper and lower limits. The best star, or the one with the best objective function value, is identified as the black hole x_{BH} after each star fitness value has been evaluated.

Step 2 : Movement of the stars

The black hole has the capacity to swallow the nearby stars. The black hole begins consuming the nearby stars after the stars are placed into motion, and it has been located. Therefore, all the stars are drawn to the black hole, and its equation for absorbing stars is as described in (31):

$$x_i(t+1) = x_i(t) + rand \times (x_{BH} - x_i(t)), \quad i = 1, 2, \ldots, N \tag{31}$$

With $x_i(t+1)$ and $x_i(t)$ are the position of the ith star at iterations $(t+1)$ and t, respectively, $rand$ is an arbitrary number between 0 and 1. The black hole is located in our search area at x_BH. N is the total number of potential solutions, given that it has the best fitness value and draws all other stars to it, it's fascinating to observe that the black hole is immobile.

Step 3 : The updating of black hole

While traveling near a black hole, a star can dive deeper (due to its lowest cost of fitness function). By picking this star, the black hole is thus updated.

Step 4 : Placement of the stars

As previously stated, if a star passes through the black hole's event horizon, it is aspirated. In other words, a star dies if its distance from the black hole is smaller than the Schwarzschild radius. In order to keep the number of solutions or candidate stars constant, a new star must appear and be randomly dispersed around the search region. Using the following equation (32), the BHA calculates the event's horizon's radius:

$$R = \frac{f_{BH}}{\sum_{N}^{i=1} fi} \tag{32}$$

With f_{BH} indicates the cost function value of the best star, fi denotes the cost function value of the ith star, and N indicates the total number of stars in the population.

When the distance (measured as a Euclidean distance) between a feasible solution and the black hole (the best candidate) is less than R, the candidate is removed, and a new candidate is produced and disseminated randomly around the problem space.

3 Identification and Control of different systems by the proposed approach

In this section, we will try to identify and control the four processes that were studied $G_{mi}(p)$ using traditional technique (LS, RM) and four intelligent approaches (ACO, CA, IWO, BHA), and we will compare the outcomes with the conventional method least squares for identification and reference model for correction or control. In order to make a fair comparison, the number of iterations was assumed to be the same for all algorithms as shown in Table 1 and Table 2.

3.1 Identification of different systems behaviors

In order to choose the best model, it is necessary to use prior knowledge about the target system to pick a class of models from which to conduct the best model search. In this instance, given the several systems under examination, the model $G_{mi}(p)$ in equation (1) is chosen. The estimate of the parameters k, ω_n, and ξ is also necessary for the identification of four systems, just like it is for the Least Square technique (LS).

The task of identifying will be approached as an optimization issue after our identification model has been constructed. Comparing a system's temporal response to the model's using just input and output data is the principle underlying parameter estimation. The parameters are computed to minimize the difference between the system output that is expected and what actually occurs. As a result, as described in (33), the adaptation or fitness function used for this application takes into consideration the quadratic error between the real process and the recognized model, the parameters of each algorithms for identifcation application are shown in table 1.

$$f = \sum_{i=1}^{N} (y(i) - \hat{y}(i))^2 \tag{33}$$

Table 1: Adjustement of ACO, CA, IWO, and BHA parameters for identification application

Algorithms	Parameters	Symbol	Values
ACO	Dimensions of the problem	$X = (x_\xi, x_{\omega_n}, x_k)$	3
	Number of ants	m	800
	Maximum number of iteration	k	100
	Weight of pheromone	θ	0.06
	Positive pheromone	-	0.2
	Negative pheromone	-	0.3
	Evaporation	λ	0.95
	Nodes	-	101
CA	Dimensions of the problem	$n = (n_\xi, n_{\omega_n}, n_k)$	3
	Number of population	N	400
	Maximum number of iteration	t	100
	Alpha	α	0.2
	Acceptance ratio	p_{accept}	0.35
IWO	Dimensions of the problem	$X = (x_\xi, x_{\omega_n}, x_k)$	3
	Maximum number of iteration	T	100
	Initial population	M	200
	Max population	M_{max}	400
	Minimum quantity of seeds	S_{min}	0
	Maximum quantity of seeds	S_{max}	5
	Exponent of variation reduction	β	4
	Initial standard deviation value	$\sigma_{initial}$	0.75
	Final standard deviation value	σ_{final}	1e-06
BHA	Dimensions of the problem	$X = (x_\xi, x_{\omega_n}, x_k)$	3
	Number of stars	N	400
	Maximum number of iteration	T	100

3.2 Control of different systems behaviors

The systems that will be controlled here are the ones that were identified. In PID controller parameter tuning, The target function is the step response characteristic that contains settling time (T_s), maximum overshoot (O_v), and steady state error E_{ss}. the fitness function that has been adopted is in equation (34) and table 2 represents the parameters for each algorithm used for control systems.

$$Z \ = \ O_v \ + \ |T_s - T_{so}| \ + \ E_{ss} \tag{34}$$

With ($T_{so} = 0.01s$) settling time desired, overshoot ($O_v = 0\%$) and a steady state error null.

Table 2: Adjustement of ACO, CA, IWO, and BHA parameters for control application

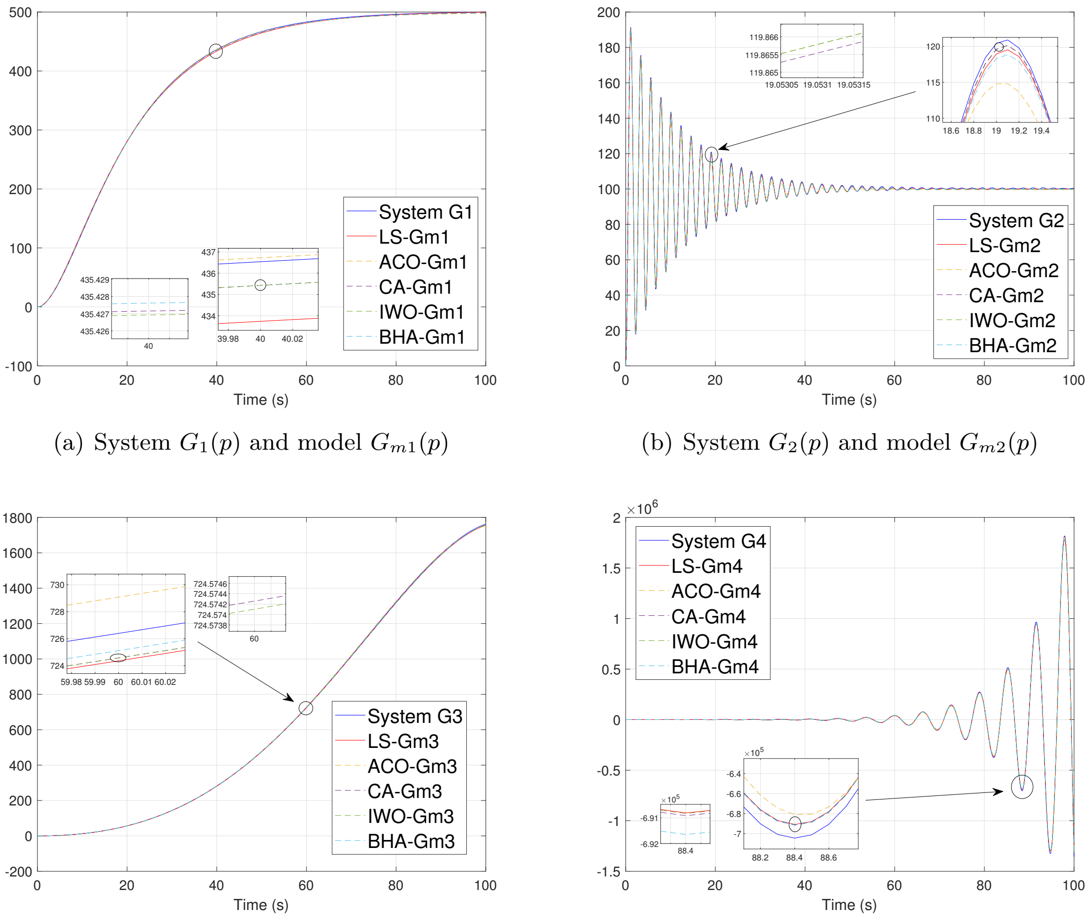
Algorithms	Parameters	Symbol	Values
ACO	Dimensions of the problem	$X = (x_P, x_I, x_D)$	3
	Number of ants	m	100
	Maximum number of iteration	k	30
	Weight of pheromone	θ	0.001
	Positive pheromone	-	0.2
	Negative pheromone	-	0.3
	Evaporation	λ	0.95
	Nodes	-	1001
CA	Dimensions of the problem	$n = (n_P, n_I, n_D)$	3
	Number of population	N	400
	Maximum number of iteration	t	30
	Alpha	α	0.2
	Acceptance ratio	p_{accept}	0.35
IWO	Dimensions of the problem	$X = (x_P, x_I, x_D)$	3
	Maximum number of iteration	T	30
	Initial population	M	20
	Max population	M_{max}	40
	Minimum quantity of seeds	S_{min}	0
	Maximum quantity of seeds	S_{max}	5
	Exponent of variation reduction	β	4
	Initial standard deviation value	$\sigma_{initial}$	7.5
	Final standard deviation value	σ_{final}	1e-06
BHA	Dimensions of the problem	$X = (x_P, x_I, x_D)$	3
	Number of stars	N	400
	Maximum number of iteration	T	30

4 Simulations and Results

In this part, the outcomes of several simulations are shown. Using a computer with an i5 CPU running at 2.5 GHz, 8 GB of RAM, and a 256 GB SSD drive, all the algorithms were programmed in MATLAB 2020b. Plots showing the identification results utilizing the two various proposed methodologies, LS and four intelligent methods (ACO, CA, IWO and BHA) are first provided in Figure 1, along with a contrast of the two approaches. In Figure 3, simulation results displaying the four behaviors regulated or controlled by intelligent approaches for PID correctors and model reference controller are also shown and discussed.

4.1 Identifications results

Following the four-step response in figure 1, it is clear that the four systems and model plots produced by ACO, CA, IWO, BHA and LS are relatively comparable, despite the fact that four intelligent methods yield more effective outcomes than LS, and table 3 shows the estimated model parameters for different systems.



(a) System $G_1(p)$ and model $G_{m1}(p)$

(b) System $G_2(p)$ and model $G_{m2}(p)$

(c) System $G_3(p)$ and model $G_{m3}(p)$

(d) System $G_4(p)$ and model $G_{m4}(p)$

Figure 1: Step responses of the system $G_1(p), \ldots, G_4(p)$ and the model $G_{m1}(p), \ldots, G_{m4}(p)$ identified by LS, ACO, CA, IWO and BHA

Table 3: Estimated model parameters using LS, ACO, CA, IWO, and BHA

Methods	Model $G_{mi}(p)$ based on LS, ACO, CA, IWO and BHA					
	$G_{m1}(p)$			$G_{m2}(p)$		
	k	ω_n	ξ	k	ω_n	ξ
LS	501.4961	0.1025	1.1394	100.0022	2.8003	0.0304
ACO	499	0.1	1.088	100	2.79	0.03
CA	500	0.1	1.1	100	2.8	0.03
IWO	500.0008	0.1	1.1	100	2.8	0.03
BHA	499.9995	0.0999	1.1	100.0441	2.7998	0.030271
	$G_{m3}(p)$			$G_{m4}(p)$		
	k	ω_n	ξ	k	ω_n	ξ
LS	150.2160	0.0378	-0.6042	99.9804	1	-0.1
ACO	165.8	0.0369	-0.587	69	1	-0.104
CA	150	0.0378	-0.6048	99.9995	1	-0.1
IWO	149.9849	0.037801	-0.60482	99.8207	1	-0.10002
BHA	151.2243	0.037718	-0.60351	101.5912	0.99999	-0.099835

The values of the objective function for 10 runs of the employed methods are shown in Table 4. As can be observed, the best values of the cost function is owned by CA with the best cost (=2.7838e-13) and SD (=7.1108e-13). It also has the fastest CPU (=3.1395 s), demonstrating its best efficiency by balancing the capacities of exploration and exploitation, the powerful operators which is using both situational knowledge and normative knowledge sources in the CA algorithm have been able to outperform the other algorithms in the large-scale and complicated challenge of identification, especially for unstable systems as shown in Table 5.

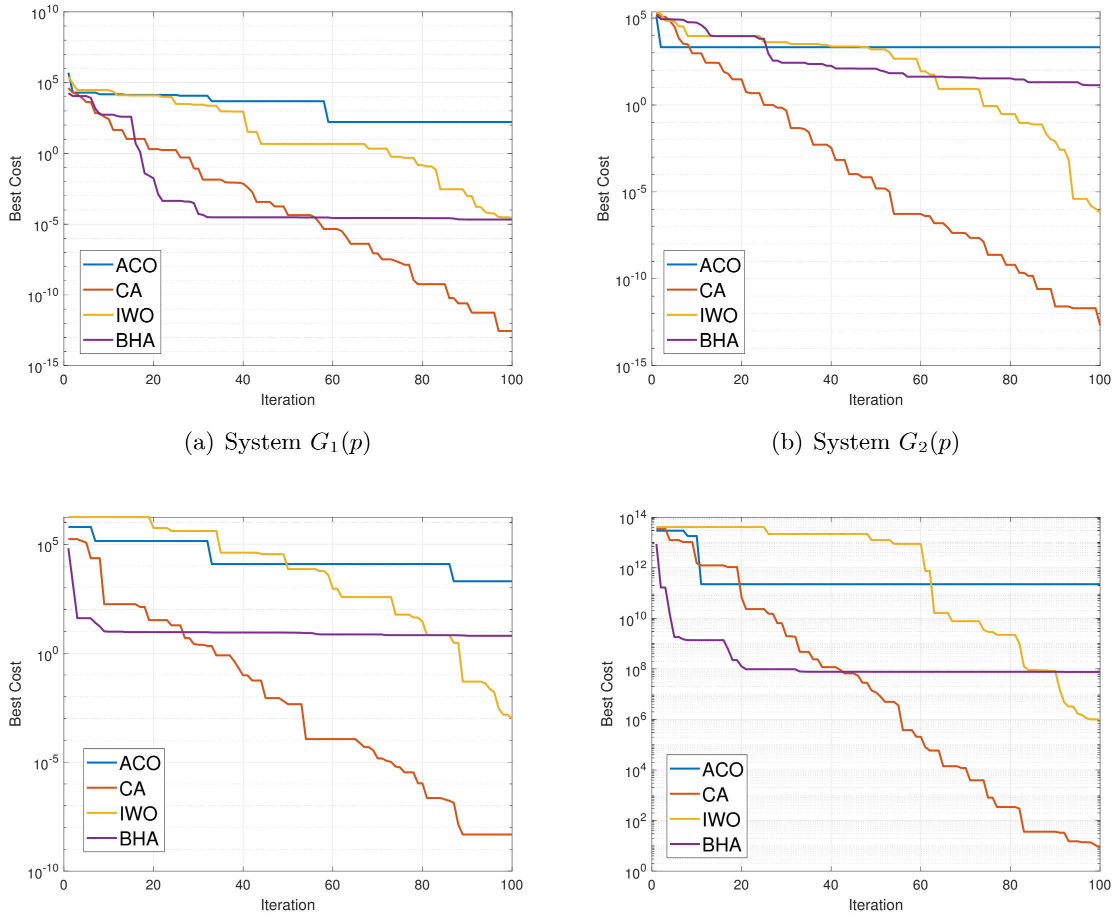
Table 4: The objective function (33) values of the system $G_1(p)$ for 10 runs of the meta-heuristic algorithms used. Values that are significant are bold.

Number of runs	ACO	CA	IWO	BHA
1	1575	3.8385e-13	55.3414	0.0019
2	5784	1.2884e-12	0.7089	1.24
3	8014	2.3550e-12	2.9561e-05	2.1459e-05
4	6135	1.7742e-12	0.8336	9.8432e-04
5	404.18	1.1661e-12	40.0042	0.0022
6	11468	3.5245e-13	1907	1.0431
7	164.18	1.9259e-12	311.0612	0.0227
8	1055	9.2095e-13	66.7968	0.1161
9	11557	9.1687e-13	0.0223	0.1288
10	13852	**2.7838e-13**	9.0129	0.0763
Best	164.18	**2.7838e-13**	2.9561e-05	2.1459e-05
Worst	13852	**2.3550e-12**	1907	1.24
Average	6001	**1.1362e-12**	239.1582	0.2632
Standard Deviation (SD)	5120	**7.1108e-13**	593.8296	0.4678
Best CPU Time (s)	4.572	**3.1395**	5.2506	19.2908

Table 5: Comparison results of the best runs utilizing ACO, CA, IWO, and BHA for different systems $G_i(p)$ to minimize the objective function in (33)

Methods	Systems	Performance				
		Best	Worst	Average	(SD)	CPU Time(s)
ACO	$G_1(p)$	164.1858	13852.1689	6001.1083	5120.4399	4.572
	$G_2(p)$	2114.7106	18199.4445	10262.4695	4461.2297	4.6367
	$G_3(p)$	1.9747e+03	4.9590e+04	2.8210e+04	1.6616e+04	3.3323
	$G_4(p)$	2.2184e+11	1.3794e+13	8.2932e+12	4.3109e+12	6.9851
CA	$G_1(p)$	2.7838e-13	2.355e-12	1.1362e-12	7.1108e-13	3.1395
	$G_2(p)$	2.1546e-13	2.4001e-12	1.1323e-12	7.6299e-13	4.5657
	$G_3(p)$	4.7613e-09	63.0391	14.7164	22.2056	1.9514
	$G_4(p)$	8.4201	6.8346e+09	1.4185e+09	2.4914e+09	4.7415
IWO	$G_1(p)$	2.9561e-05	1907.8004	239.1582	593.8296	5.2506
	$G_2(p)$	6.5514e-07	18.5218	2.4471	5.9315	5.3731
	$G_3(p)$	9.8903e-04	1.5599e+03	289.9662	543.8815	2.266
	$G_4(p)$	9.8920e+05	2.0962e+12	2.2305e+11	6.5840e+11	3.3926
BHA	$G_1(p)$	2.1459e-05	1.24	0.26321	0.46777	19.2908
	$G_2(p)$	13.7295	1776.3125	327.4037	531.1237	16.8769
	$G_3(p)$	6.3414	1.9439e+04	5.3125e+03	6.7216e+03	13.4429
	$G_4(p)$	7.6708e+07	2.1975e+11	7.6796e+10	8.3543e+10	17.4635

Figure 2 displays the convergence graphs for each of the four optimization algorithms, with CA and IWO gives better performances in terms of accuracy and CPU time compared to others algorithms. Moreover, CA surpasses other algorithms in finding the unstable system $G_4(p)$, which is challenging due to the abundance of data, with the best cost of (=8.4201). ACO and BHA gives better performance for stable systems $G_1(p)$, but it tends to not minimize quite well the objective function in (33) for the other systems. However, Ant colony optimization (ACO) is effective at solving discrete issues, but when working with a lot of data, it unavoidably suffers from limitations in convergence speed and solution accuracy [34].



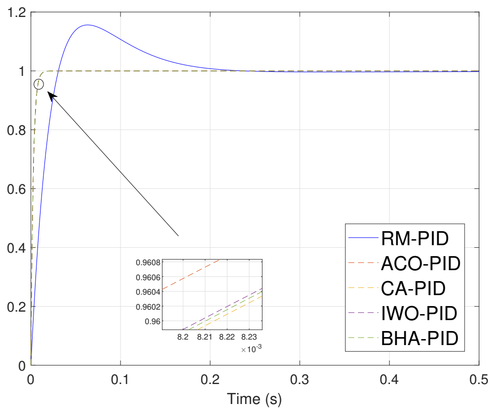
(a) System $G_1(p)$ (b) System $G_2(p)$

(c) System $G_3(p)$ (d) System $G_4(p)$

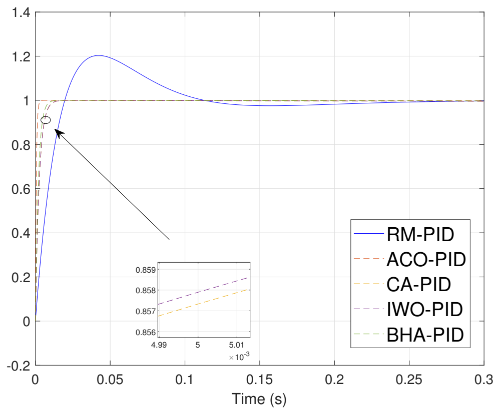
Figure 2: Convergence of the ACO, CA, IWO, and BHA to the optimum value for different systems using fitness function in (33)

4.2 Control results

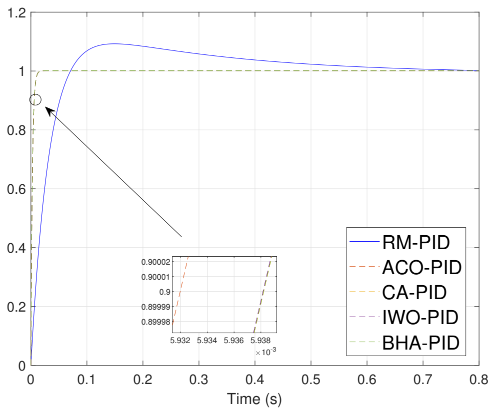
In order to develop PID controller parameters for the four identified systems, we combine the conventional Reference Model (RM) technique with four intelligent controls (ACO, CA, IWO and BHA). we desire a system with an overshoot of ($O_v = 0\%$) and a settling time ($T_{s0} = 0.01s$), Table 6 provides a summary of PID settings, and Figure 3 displays the results of a simulation of the closed-loop step response for the four controlled systems. and Table 7 shows the PID controller's performance as determined by conventional and intelligent approaches in relation to overshoot (O_v) and settling time (T_s) values.



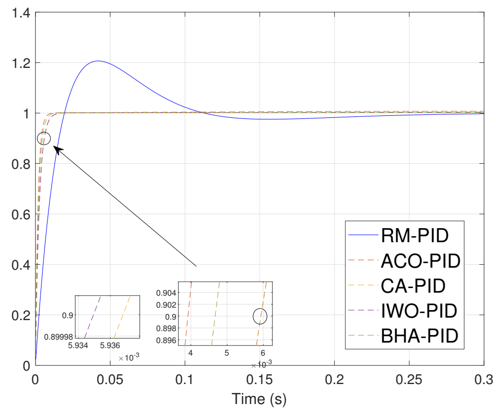
(a) System $G_{m1}(p)$



(b) System $G_{m2}(p)$



(c) System $G_{m3}(p)$



(d) System $G_{m4}(p)$

Figure 3: Step responses of the systems $G_{m1}(p), \ldots, G_{m4}(p)$ controlled by RM, ACO, CA, IWO and BHA

Table 6: PID Parameter tuning based on RM, ACO, CA, IWO and BHA

Methods	Model $G_{mi}(p)$ based on LS, ACO, CA, IWO and BHA					
	$G_{m1}(p)$			$G_{m2}(p)$		
	K_p	K_i	K_d	K_p	K_i	K_d
RM	216	540	12.5	3.43	34.43	0.11
ACO	9	8	79	0.45	14.47	2.15
CA	7.27	9.95	78.53	0.16	0.55	0.49
IWO	2.44	14.57	78.68	0.15	0.61	0.49
BHA	14.52	12.67	78.62	0.16	1.83	0.64
	$G_{m3}(p)$			$G_{m4}(p)$		
	K_p	K_i	K_d	K_p	K_i	K_d
RM	1847.87	5039.66	196.2	18.13	182.16	0.6
ACO	590	937	1810	0.94	84.27	3.9
CA	650.95	19.98	1804.05	0.46	7.94	2.57
IWO	464.83	454.2	1809.39	0.38	14.18	2.57
BHA	647.51	297.23	1804.11	0.93	10.74	3.27

Table 7 displays the values of the objective function for the 10 runs of the algorithms used for the system G_{m1}. As can be seen, IWO-PID has the best cost (=3.2905e-10) while also converge the fastest with CPU time (=48.8732 s). without forgetting that CA-PID has the best SD (=7.5467e-08) but in term of CPU time it takes a long time to converge to optimal solution. Table 8 lists the values of cost function for control of different systems $(G_{m1}(p), \ldots, G_{m4}(p))$.

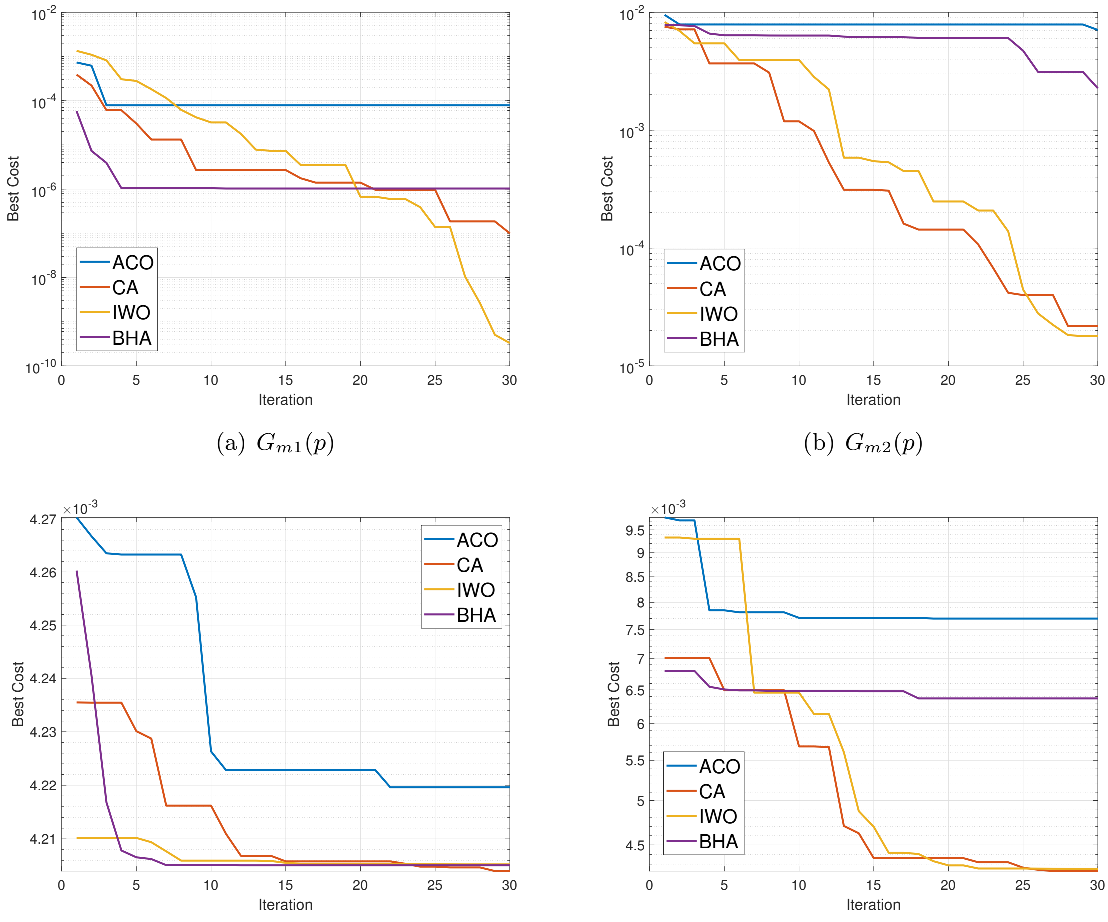
Table 7: The objective function (34) values of the system $G_{m1}(p)$ for 10 runs of the meta-heuristic algorithms used. Values that are significant are bold.

Number of runs	ACO	CA	IWO	BHA
1	2.2396e-04	1.6238e-07	3.6876e-08	1.5938e-04
2	7.8667e-05	3.0624e-07	1.0429e-08	1.2794e-05
3	3.0269e-04	2.3524e-07	2.1642e-08	5.0703e-06
4	5.3046e-04	9.5822e-08	1.1676e-09	4.5972e-05
5	1.1824e-04	1.5771e-07	**3.2905e-10**	7.1006e-06
6	3.6468e-04	2.6905e-07	2.1557e-08	1.0316e-06
7	0.0019	1.1275e-07	2.6086e-04	8.2930e-05
8	2.3518e-04	1.8182e-07	2.6097e-04	1.0894e-05
9	1.3502e-04	9.9823e-08	3.0009e-04	3.1620e-06
10	8.2892e-04	1.0198e-07	1.8825e-08	1.5428e-05
Best	7.8667e-05	9.5822e-08	**3.2905e-10**	1.0316e-06
Worst	0.0019	3.0624e-07	**3.0009e-04**	1.5938e-04
Average	4.7462e-04	1.7228e-07	**8.2203e-05**	3.4376e-05
Standard Deviation (SD)	5.5807e-04	7.5467e-08	**1.3276e-04**	5.0753e-05
Best CPU Time (s)	92.6657	311.6901	**48.8732**	342.8826

Table 8: Comparison results of several runs utilizing ACO, CA, IWO, and BHA to minimize the objective function in (34)

Methods	Systems	Performance				
		Best	Worst	Average	(SD)	CPU Time(s)
ACO	$G_{m1}(p)$	7.8667e-05	0.0019	4.7462e-04	5.5807e-04	92.6657
	$G_{m2}(p)$	0.0071	0.0092	0.0079	7.8315e-04	97.4224
	$G_{m3}(p)$	0.004219	0.004553	0.004342	1.1339e-04	89.4102
	$G_{m4}(p)$	0.0076982	0.0097668	0.0094758	6.2696e-04	104.6407
CA	$G_{m1}(p)$	9.5822e-08	3.0624e-07	1.7228e-07	7.5467e-08	311.6901
	$G_{m2}(p)$	2.1866e-05	1.7496e-04	7.3372e-05	5.9248e-05	369.7557
	$G_{m3}(p)$	0.004204	0.004207	0.004205	1.3536e-06	377.6429
	$G_{m4}(p)$	0.0042298	0.0095878	0.0079801	0.0025821	294.9629
IWO	$G_{m1}(p)$	3.2905e-10	3.0009e-04	8.2203e-05	1.3276e-04	48.8732
	$G_{m2}(p)$	1.7853e-05	0.0025	3.2683e-04	7.6233e-04	38.4879
	$G_{m3}(p)$	0.0042053	0.0062771	0.0047261	5.9079e-04	51.6781
	$G_{m4}(p)$	0.004253	0.0095874	0.0090514	0.001686	54.9531
BHA	$G_{m1}(p)$	1.0316e-06	1.5938e-04	3.4376e-05	5.0753e-05	342.8826
	$G_{m2}(p)$	0.0023	0.0069	0.0043	0.0015	299.2424
	$G_{m3}(p)$	0.0042051	0.0042093	0.0042068	1.2547e-06	310.0056
	$G_{m4}(p)$	0.006971	0.0096194	0.0089169	0.00094863	301.2313

It's critical to understand the convergence rates of MAs while evaluating them in order to determine how quickly they reach to the best solution. The convergence rate of the methods being used is seen in Figure 4. IWO-PID and CA-PID had the quickest convergence rate, as demonstrated, and they could reach values close to the ideal value with the fewest iterations (less than 30 iterations). In terms of reaching the ideal value, BHA-PID also displayed acceptable outcomes. As for ACO-PID even after more than 30 cycles, he was unable to successfully attain the ideal solution.



(a) $G_{m1}(p)$ (b) $G_{m2}(p)$

(c) $G_{m3}(p)$ (d) $G_{m4}(p)$

Figure 4: Convergence of the ACO, CA, IWO, and BHA to the optimum value for different systems of fitness function in (34)

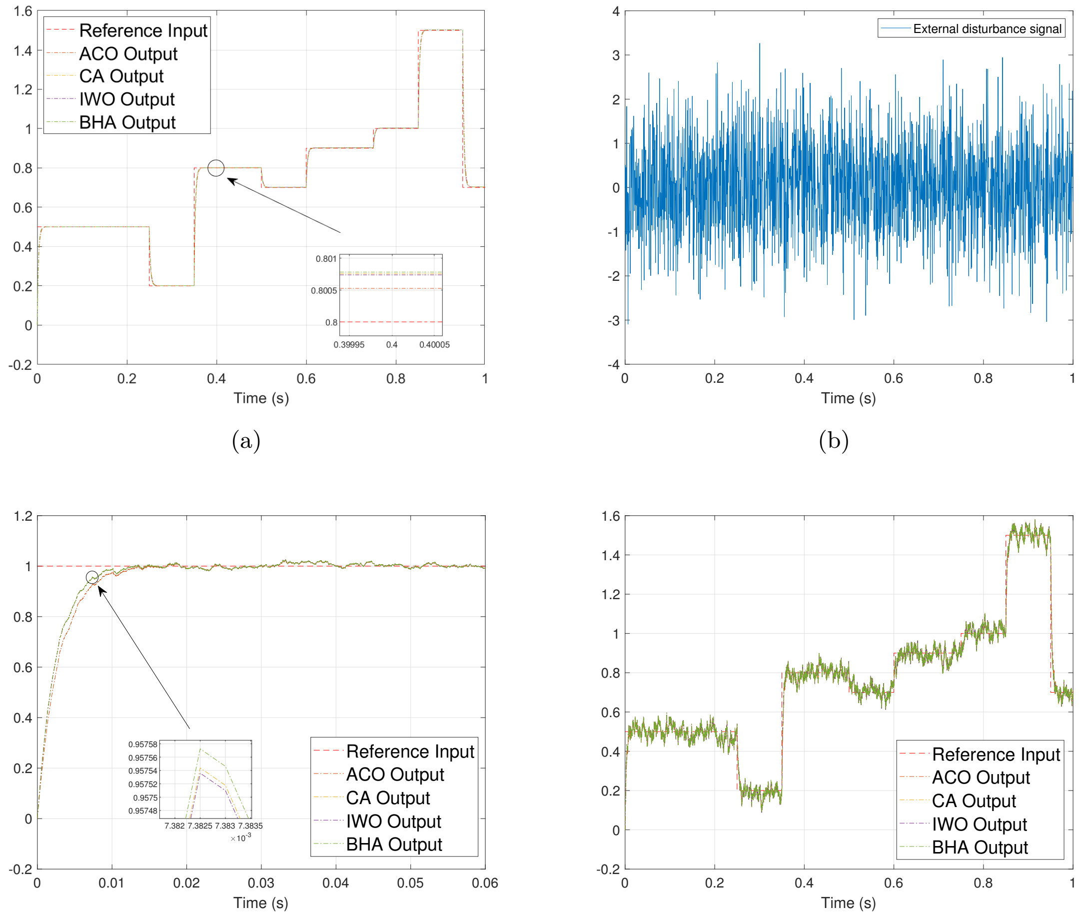
Table 9: PID Controller performance obtained with RM and intelligent methods

Methods	Performances							
	$G_{m1}(p)$		$G_{m2}(p)$		$G_{m3}(p)$		$G_{m4}(p)$	
	$O_v(\%)$	$T_s(s)$	$O_v(\%)$	$T_s(s)$	$O_v(\%)$	$T_s(s)$	$O_v(\%)$	$T_s(s)$
RM	15.6	0.17	20.21	0.1892	14.9118	0.3	20.65	0.1886
ACO	0	0.0099	0	0.0023	0	0.01	0	0.0066
CA	0	0.01	0	0.01	0	0.01	0	0.01
IWO	0	0.01	0	0.01	0	0.01	0	0.01
BHA	0	0.01	0	0.0078	0	0.01	0	0.0074

The selection of the fitness function in (34) is demonstrated, all of these systems are intelligently handled with high accuracy to avoid any solutions with an overshoot, a huge quadratic error and undesired settling times as shown in Table 9. However, ACO-PID and BHA-PID struggled in a minor way to find the desired settling time for the unstable systems $G_{m2}(p)$ and $G_{m4}(p)$, it is evident from the comparisons that the intelligent control performs better than RM-PID. The four systems settling times are short when using the conventional technique for the control task, but there is still an overshoot that has to be eliminated.

For the example of $G_{m1}(p)$ with variable input signal amplitude, the effectiveness of ACO-PID, CA-PID, IWO-PID, and BHA-PID was validated. The system efficiently tracked the multiple simultaneous changes in the input signal, as shown in figure 5.(a), with desired settling time and overshoot which is $0.01s$ and 0%, respectively.

To evaluate the robustness of ACO-PID, CA-PID, IWO-PID, and BHA-PID, the external disturbance has been added as shown in figure 5.(b), an external disturbance signal with varying amplitude that is added in a closed loop is used to represent this perturbation, it is clear from figure 5.(c) that the ACO-PID, CA-PID, IWO-PID, and BHA-PID ensured a signal that was reasonably close to the reference signal and rejected the unwanted effects of the disturbance, the test has also been done for variable reference input as shown in figure 5.(d). As a result, it is confirmed that the intelligent control is remarkably reliable and effective.



(a) (b) (c) (d)

Figure 5: (a) Response of $G_{m1}(p)$ for variable reference input, (b) External disturbance signal, (c) Response of $G_{m1}(p)$ in presence of disturbance signal, (d) Response of $G_{m1}(p)$ in presence of disturbance signal for variable reference input

4.3 Discussion and Comparison of the New Meta-Heuristic Algorithm used with Recently Published Papers

In the technical literature, a variety of meta-heuristic approaches have been used to different optimization issues. In [22] a 14 recently introduced reliable EAs have been used to optimize energy production from the Karun-4 hydropower reservoir, the moth swarm algorithm (MSA), with values of best cost, SD and CPU Time for MSA were 0.147, 0.0029 and 19.70 s respectively, was discovered to have produced the best result compared to others algorithms. The author in [16], ant colony optimization (ACO), particle swarm optimization (PSO), cuckoo search (CS), and genetic algorithm (GA) were used in a comparative study of optimized based-control system of a single-rotor, medium-scale rotorcraft for a certain flight regime, these algorithms outperform manual tuning. In hover trim settings, compared to others algorithms the optimal CPU time and best gains for the chosen objective function are determined by the ACOR which is an ACO for continuous domains, with best cost

1.340e-7, SD 0.119 and CPU Time 77.59 min. The author in [18] proposed invasive weed optimization based on chaos theory for optimal PID control of DC motor, results shows that chaotic invasive weed (CIWO) has better performance compared to simple IWO, in reducing the overshoot from 5.7644 % to 1.23 % and settling time from 0.690 s to 0.344 s with value of cost function (8.0369). An adaptive particle swarm optimization (APSO) was used for identification and control of an unstable nonlinear system in [6] and compared to linearly decreasing weight particle swarm optimization (LDW-PSO) and GA with the same identified system and model, number of population and same searching range, result shows that APSO captured the real system parameters over LDW-PSO and GA with best cost and SD were 2.1954e-24 and 4.2636e-22, respectively, the test of proposed APSO was also done for control process of finding the best gain of PID controller, the objective function which is the sun of squared errors was minimized, results show that APSO is superior to others algorithm in terms of best cost and standard deviation (SD) with 1.0976 and 0.0045, respectively. However, There isn't a single MAs that can be applied to solve every optimization problem. For a certain issue only, some algorithms could provide a better solution than others.

Table 10: Comparison between the new Meta-heuristic algorithm used and various algorithm that have just been published for different optimization issue.

Reference	Optimization problem	Algorithms used	Performances				
			Best	Worst	Average	SD	CPU Time
[22]	Hydropower Energy Generation	MSA	0.1470	0.1559	0.1489	0.0029	19.70 (s)
		SOS	0.1473	0.2618	0.1615	0.0352	42.08 (s)
		WCA	0.1509	0.2350	0.1670	0.0284	43.17 (s)
[16]	Tuning of a PID-Based Controller of Hover trim conditions in Medium-Scale Rotorcraft	ACO	1.340e-7	-	1.434e-7	0.119	77.59 (min)
		PSO	1.499e-7	-	1.521e-7	0.020	94.70 (min)
		GA	2.563e-7	-	2.608e-7	0.065	91.21 (min)
		CS	1.566e-7	-	1.763e-7	0.240	119.68 (min)
[18]	Tuning of a PID-based controller of DC Motor	IWO	23.0944	-	-	-	-
		CIWO	8.0369	-	-	-	-
[6]	System Identification	GA	0.1734	1.6408	0.7456	0.6241	-
		LDW-PSO	8.3425e-14	7.8441e-8	3.7530e-12	4.4612e-9	-
		APSO	2.1954e-24	8.5327e-21	4.9653e-22	4.2636e-22	-
	Control system	GA	1.2904	1.3846	1.328	0.9045	-
		LDW-PSO	1.1251	1.2064	1.134	0.0382	-
		APSO	1.0976	1.1075	1.1031	0.0045	-
Proposed Algorithms	Identification of system $G_1(p)$	ACO	164.1858	13852.1689	6001.1083	5120.4399	4.572 (s)
		CA	2.7838e-13	2.355e-12	1.1362e-12	7.1108e-13	3.1395 (s)
		IWO	2.9561e-05	1907.8004	239.1582	593.8296	5.2506 (s)
		BHA	2.1459e-05	1.24	0.26321	0.46777	19.2908 (s)
	Tuning of PID-based controller of system $G_{m1}(p)$	ACO	7.8667e-05	0.0019	4.7462e-04	5.5807e-04	92.6657 (s)
		CA	9.5822e-08	3.0624e-07	1.7228e-07	7.5467e-08	311.6901 (s)
		IWO	3.2905e-10	3.0009e-04	8.2203e-05	1.3276e-04	48.8732 (s)
		BHA	1.0316e-06	1.5938e-04	3.4376e-05	5.0753e-05	342.8826 (s)

5 Conclusion

The research in this paper are primarily concerned with the capacities of four resilient MAs that were only recently introduced to optimize model and PID settings for varied system characteristics. These algorithms include ACO, CA, IWO and BHA are the most newest algorithms and they have been used for the first time in one paper for both optimization problem. As a result, they seem like a good alternative to traditional methods (LS, RM).

These MAs performance was compared using the following five factors: Best cost, Worst cost, Average cost, Standard deviation (SD), and CPU time (s). In order to compare the results of 10 separate runs of each algorithm and each algorithm's parameters were evaluated. For identification tasks, it was discovered that CA came in first position and has the best model for parameters model for various systems, for the first system, it had the best objective function 2.7838e-13, lowest standard deviation (SD) 7.1108e-13 and the quickest CPU time 3.1395 (s), even for unstable system it has a better performances compared to others algorithm. As for PID parameters for the first system, it shows that IWO-PID was able to converge to optimal solution in fewest CPU Time 48.8732 (s) and has lowest cost function 3.2905e-10. Additionally, the robustness of proposed algorithm is assessed, from the first system's simulation, the findings demonstrate that ACO-PID, CA-PID, IWO-PID, and BHA-PID are resistant to random disturbances.

The performance of the MAs may be improved and a potent strategy for identification and control can be proposed using a variety of novel hybrid algorithms that can be further developed for this relevant sector.

References

[1] H. W. Sorenson. (1970); Least-squares estimation: from Gauss to Kalman, *IEEE Spectrum*, 7(7), 63–68, IEEE.

[2] K.J. Astrom.; B. Wittenmark. (1995); *Adaptive Control*, Addison-Wesley, Massachusetts, 1995.

[3] El Gmili. N.; Mjahed. M.; El Kari. A.; Ayad. H. (2017); An Improved Particle Swarm Optimization (IPSO) Approach for Identification and Control of Stable and Unstable Systems, *International Review of Automatic Control (IREACO)*, 10(3), 229–239, 2017.

[4] Siti. I.; Mjahed. M.; Ayad. H.; El Kari. A. (2018); Identification and Control of Stable and Unstable Systems by Using Genetic Algorithm Method, *2018 International Conference on Control, Automation and Diagnosis (ICCAD)*, IEEE.

[5] Fayti. M.; Mjahed. M.; Ayad. H.; El Kari. A. (2022); Differential Evolution Approach for Identification and Control of Stable and Unstable Systems, *2022 8th International Conference on Control, Decision and Information Technologies (CoDIT)*, 1, 218–223, 2022, IEEE.

[6] Alireza. A.; Hamidreza. M. (2011); System identification and control using adaptive particle swarm optimization, *Applied Mathematical Modelling*, 35(3), 1210–1221, 2011.

[7] Wei-Der. C. (2007); Nonlinear system identification and control using a real-coded genetic algorithm, *Applied Mathematical Modelling*, 31(3), 541—550, 2007.

[8] J. G. Ziegler.; N. B. Nichols. (1993); Optimum Settings for Automatic Controllers, *Journal of Dynamic Systems, Measurement and Control*, 115(2B), 220–222, 1993, ASME.

[9] El Hamidi. K.; Mjahed. M.; El Kari. A.; Ayad. H. (2018); Neural and Fuzzy Based Nonlinear Flight Control for an Unmanned Quadcopter, *International Review of Automatic Control (IREACO)*, 10(3), 98–106, 2018.

[10] Mjahed. M. (2019); Flight control system design using neural networks, *International Robotics & Automation Journal*, 5(3), 96–98, 2019.

[11] D.H. Vu.; S. Huang.; T.D. Tran.; T.Y. Vu.; V.C. Pham (2019); A Robust Adaptive Control using Fuzzy Neural Network for Robot Manipulators with Dead-Zone, *International Journal of Computers Communications and Control*, 14(5), 692–710, 2019.

[12] J. Tang.; G. Liu.; Q. Pan (2021); A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends, *IEEE/CAA Journal of Automatica Sinica*, 8(10), 1627—1643, IEEE, 2021.

[13] Abbas. N. H.; Sami. A. R. (2018); Tuning of PID Controllers for Quadcopter System using Cultural Exchange Imperialist Competitive Algorithm, *Journal of Engineering*, 24(2), 80–99, 2018.

[14] El Gmili. N.; Mjahed. M.; El Kari. A.; Ayad. H. (2020); Particle swarm optimization based proportional-derivative parameters for unmanned tilt-rotor flight control and trajectory tracking, *Automatika*, 61(2), 189–206, 2020.

[15] Siti. I.; Mjahed. M.; Ayad. H.; El Kari. A. (2015); Control of a quadcopter using reference model and genetic algorithm methods, *2015 Third World Conference on Complex Systems (WCCS)*, 1–6, IEEE.

[16] Mpanza. L.J.; Pedro. J.O. (2021); Optimised Tuning of a PID-Based Flight Controller for a Medium-Scale Rotorcraft, *Algorithm*, 14(6), 178, 2021.

[17] Stephen. B. J.; Emmanuel. G. D.; Afeez. A.; David. O. O.; Ban. M. K. (2022); Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems, *Heliyon*, 8(5), 2405—8440, 2022.

[18] Mojgan. M.; Mahdi. Y. (2019); Improved invasive weed optimization algorithm (IWO) based on chaos theory for optimal design of PID controller, *Journal of Computational Design and Engineering*, 6(3), 284—295, 2019.

[19] Ibtihal. H. I.; Hazem. I. A. (2021); Quantitative PID Controller Design using Black Hole Optimization for Ball and Beam System, *Iraqi Journal of Computers, Communications, Control and Systems Engineering (IJCCCE)*, 21(3), 65—75, 2021.

[20] Ibtissem. C.; Noureddine. L.; Pierre. B. (2012); Tuning PID Controller Using Multiobjective Ant Colony Optimization, *Applied Computational Intelligence and Soft Computing*, 2012.

[21] Tulasichandra. S. G.; Ramana. P. (2019); Black hole optimised cascade proportional derivative-proportional integral derivative controller for frequency regulation in hybrid distributed power system, *International Journal of Swarm Intelligence*, 4(2), 155—174, 2019.

[22] Sharifi. M.R.; Akbarifard. S.; Madadi. M.R; (2022); Optimization of hydropower energy generation by 14 robust evolutionary algorithms, *Scientific Reports*, 12, 7739 (2022).

[23] Mahfoud. S.; Derouich. A.; El Ouanjli. N.; Quynh. N.V.; Mossa. M.A. (2022); A New Hybrid Ant Colony Optimization Based PID of the Direct Torque Control for a Doubly Fed Induction Motor, *World Electric Vehicle Journal*, 13(5), 78, 2022.

[24] Mjahed. M. (2016); *Identifications of linear systems, Reading notes*, Royal School of Aeronautics, Marrakech, Morocco, 2016.

[25] O. Pajonk. (2009); *Overview of System Identification with Focus on Inverse Modeling*, Institute of scientific computing, Technical University of Braunschweig, 2009.

[26] Mjahed. M. (2014); *Linear control systems, Lectures notes*, Royal School of Aeronautics, Marrakech, Morocco, 2014.

[27] Dorigo. M. (1992); *Optimization, Learning and Natural Algorithms*, Ph.D Thesis, Politecnico di Milano, Italian, 1992.

[28] H. A. Varol.; Z. Bingul. (2004); A new PID tuning technique using ant algorithm, *Proceedings of the 2004 American Control Conference*, 3, 2154–2159, IEEE.

[29] A.R. Mehrabian.; C. Lucas. (2006); A novel numerical optimization algorithm inspired from weed colonization, *Ecological Informatics*, 1(4), 355–366.

[30] R.G. Reynolds. (1994); An introduction to cultural algorithms, *Proceedings of the Third Annual Conference on Evolutionary Programming*, 131–139.

[31] M. Khalid.; Q. Luo.; H. Duan. (2012); A cultural algorithm based particle swarm optimization approach to linear brushless DC motor PID controller, *Scientific Research and Essays*, 7(3), 318–326, 2012.

[32] A. Maheri.; S. Jalili.; Y. Hosseinzadeh.; R. Khani.; M. Miryahyavi. (2021); A comprehensive survey on cultural algorithms, *Swarm and Evolutionary Computation*, 62(1), 2021.

[33] Hatamlou. A. (2013); Black hole: A new heuristic optimization approach for data clustering, *Information Sciences*, 222, 175–184, 2013.

[34] Jin. Y.; Xiaoming. Y.; Sheng. Liu.; (2021); A heterogeneous guided ant colony algorithm based on space explosion and long–short memory, *Applied Soft Computing*, 113(B), 2021.

Cite this paper as:
Azegmout, M.; Mjahed, M.; . El Kari, A; Ayad, H. (2023). New Meta-heuristic - Based Approach for Identification and Control of Stable and Unstable Systems, *International Journal of Computers Communications & Control*, 18(4), 5294, 2023.
https://doi.org/10.15837/ijccc.2023.4.5294