processes

MDPI

Article

Metaheuristic Optimization Algorithm Based Cascaded Control Schemes for Nonlinear Ball and Balancer System

Farhan Zafar [1], Suheel Abdullah Malik [1], Tayyab Ali [1], Amil Daraz [2,*], Atif M. Alamri [3], Salman A. AlQahtani [4] and Farkhunda Bhatti [5]

1 Department of Electrical and Computer Engineering, Faculty of Engineering and Technology, International Islamic University Islamabad, Islamabad 44000, Pakistan; farhanzafar113@yahoo.com (F.Z.); suheel.abdullah@iiu.edu.pk (S.A.M.); tayyab.ali@iiu.edu.pk (T.A.)
2 School of Information Science and Engineering, NingboTech University, Ningbo 315100, China
3 Software Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11495, Saudi Arabia; atif@ksu.edu.sa
4 Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11495, Saudi Arabia
5 Department of Electronics Engineering, Mehran University of Engineering & Technology Jamshoro, Jamshoro 76062, Pakistan; farkhundafarhan@gmail.com
* Correspondence: amil.daraz@nbt.edu.cn

Abstract: The ball and balancer system is a popular research platform for studying underactuated mechanical systems and developing control algorithms. It is a well-known two-dimensional balancing problem that has been addressed by a variety of controllers. This research work proposes two controllers that are proportional integral derivative-second derivative-proportional integrator (PIDD^2-PI) controller and tilt integral derivative with filter (TID-F) controller in a multivariate, electromechanical, and nonlinear under-actuated ball and balancer system. Integral Time Absolute Error (ITAE) is an objective function used for designing controllers because of its ability to be more sensitive to overshooting as well as reduced settling time and steady-state error. As part of the analysis, four metaheuristic optimization algorithms are compared in the optimization of proposed control strategies for cascaded control of the ball and balancer system. The algorithms are the Grey Wolf optimization algorithm (GWO), Cuckoo Search algorithm (CSA), Gradient Base Optimization (GBO), and Whale Optimization Algorithm (WOA). The effectiveness of proposed controllers PIDD^2-PI and TID-F is investigated to be better in terms of transient time response than proportional integral derivative (PID), proportional integral-derivative (PI-D), proportional integral-proportional derivative (PI-PD) and proportional integral derivative-second derivative-proportional derivative (PIDD^2-PD). Moreover, these two proposed controllers have also been compared with recently published work. During the analysis, it is shown that the proposed control strategies exhibit significantly greater robustness and dynamic responsiveness compared to other structural controllers. The proposed controller WOA-PIDD^2-PI reduced the 73.38% settling time and 88.16% rise time compared to classical PID. The other proposed controller GWO-TID-F reduced 58.06% the settling time and 26.96% rise time compared to classical PID. These results show that proposed controllers are particularly distinguished in terms of rise time, settling time, maximum overshoot, and set-point tracking.

Keywords: underactuated system; ball and balancer; optimization; PIDD^2-PI; TID-F; grey wolf optimization algorithm; cuckoo search algorithm; gradient based optimization and whale optimization

Citation: Zafar, F.; Malik, S.A.; Ali, T.; Daraz, A.; M. Alamri, A.; AlQahtani, S.A.; Bhatti, F. Metaheuristic Optimization Algorithm Based Cascaded Control Schemes for Nonlinear Ball and Balancer System. *Processes* **2024**, *12*, 291. https://doi.org/10.3390/pr12020291

Academic Editors: Claudio Urrea and Dan Zhang

Received: 21 December 2023
Revised: 16 January 2024
Accepted: 22 January 2024
Published: 29 January 2024

1. Introduction

The under actuated mechanical systems (UMS) are challenging to control as they are inherently unstable, nonlinear and have complex dynamics [1]. Because of the under actuation property UMS require reliable, efficient and fast controllers. This broad research area of UMS control can be divided into two main categories: set-point regulation and trajectory tracking, that aim to stabilize UMSs in the face of various uncertainties and external disturbances [2]. The UMS have fewer actuators than degrees of freedom to be controlled, so many traditional non-linear control methods cannot be used directly in this scenario. It is well known that UMS exists in a number of forms, such as the Acrobat, Pendubot, cart-pole systems, crane systems, rotating pendulums, inertia wheel pendulums, beam-and-ball systems, magnetic suspension systems, translational oscillators with rotational actuators (TORAs), vertical takeoff and landing (VTOL) aircraft, and surface vessels. The ball and balancer system with two degrees of freedom is one of the best-known examples of an under actuated mechanical systems. The position control of ball in this system is a major challenge that is usually addressed by applying various control methodologies. To achieve the required balancing objective, various controllers have been tested upon the ball and balancer system including feedback linearization, energy, back stepping, sliding mode, and fuzzy logic. Overall, the results of these studies have demonstrated the effectiveness and potential of advanced control methods in improving the performance of systems. A new hybridized Chaotic state of matter search with Elite opposition-based learning (CSMSEOBL) [3] algorithm is proposed to tune the gains of PID controller to improve the transient response of ball and balancer system. Their results showed that the CSMSEOBL based PID controller outperforms classical PID controllers and other optimization techniques such as PSO-PID, SFS-PID, and SMS-PID. An artificial intelligence-based deep reinforcement learning (AI-RL) PID controller and a genetic algorithm-based PID controller are compared for linearized ball and balancers [4]. Their analysis reflected that DDPG-FC-350-E-PID outperformed all other approaches and achieves the best performance. For improving time domain response fuzzy PID and Reinforcement Learning (RL) controllers are investigated in [5]. There were various fuzzy controller proposed for motion control and trajectory tracking of ball and beam system in [6–9]. Moreover a fuzzy based adaptive integral control action had been proposed that significantly reduces the steady state error because of the integral control [10]. The effect of parameters variation has been verified by introducing PD tuned fuzzy logic controller in time domain and the results are compared with classical PD controller [11]. The goal to provide a quick and accurate response with little tracking error has achieved in [12] by proposing two types of ball stabilization controllers, a classical PID controller and a Lead/Lag compensation controller and Lead/Lag controller performed the best. Traditional PID controllers face a draw back when it comes to adjusting their settings. There are different methods to help with tuning these PID parameters. There are several examples of PID tuning [13,14] with ITAE [8,15–17]. ITAE have strong emphasis on minimizing steady-state error and settling time. In addition it is more sensitive to overshooting compared to IAE.

Using metaheuristics to solve multi-objective problems has have been proven to be an efficient and cost-effective way. There are several popular and relatively recent metaheuristics that can be used to tune controllers. Particle Swarm Optimization (PSO), Simulated Annealing (SA) and Genetic algorithm (GA) are used to optimize the gains of PID controller to control a ball and balancer system [18]. Their results were compared with classical PID in terms of delay time, rise time, and settling time and all shown a very good percentage of improvement. A fuzzy logic controller optimized with chicken swarm optimization algorithm (CS0) has been proposed and showed improvement in transient response of the ball and balancer system [19]. Whale optimization algorithms are used [20] to regulate PID parameters for tracking control of robot manipulator. As evidenced by settling times, errors, and convergence times, as well as the robustness of the WOA-PID for tuning parameters for PID controllers for robot tracking, it had proved effective for tracking robots with or without disturbances. The control of such systems was also addressed by using intelligent controllers or autonomous decision-making methods [21–24]. The simulation results indicate that sliding mode control (SMC) control scheme excels in dynamic performance as well as disturbance rejection compared to PID, fuzzy control , and LQR control [25]. An internal model control based scheme had proposed to give zero steady state error in tracking of ball and balancer system [26,27]. A neural integrated fuzzy and its hybridization with PID had been implemented to control the position of ball and angle of plate [28]. In their work steady-state error analysis and time response analysis are used to evaluate each controller's performance. Some fractional order controllers also contribute to stabilize ball and balancer system [29–31]. The FOMC performed better in terms of less chattering, improved trajectory control [32] and speed then SMC.

Fuzzy controllers are quite good in handling imprecise and nonlinear systems, but they can be complex to design and computationally intensive. Sliding mode controllers are robust to uncertainties but may exhibit chattering, and they can be sensitive to modelling errors. Neural controllers offer adaptability and effectiveness in complex systems but lack interpretability and may require significant computational resources for training.

PID controllers remain a reliable choice due to their simplicity, ease of implementation, and well-established track record in a wide range of applications. Their ability to provide stable performance make them advantageous, especially in scenarios where a precise mathematical model is available.

There are various configurations of PID are available. $PIDD^2$ controllers are often used to control position, velocity and acceleration feedback [33], as our aim is to control position of ball so we choose $PIDD^2$. TID-F controller is easier to tune with enhanced disturbance rejection capacity, and gives outstanding durability to parameter variation [34]. Four metahueristic approaches are selected and applied for tuning the controllers with optimal gains.

This paper investigates the position control and tracking performance of ball and balancer system with a tilt integral derivative controller with filter based on GWO/WOA and a proportional integral derivative second derivative controller based on GWO/WOA. In this context, this research paper has following main contributions:

- Design and implementation of two novel controllers TID-F and $PIDD^2$-PI for stabilizing underactuated ball and balancer system and optimization with different metaheuristic algorithms like WOA, CSA, GBO and GWO.
- The performance of the system has been analyzed using set point tracking analysis, and step response analysis of the proposed control strategies by employing ITAE as an error reducing function.
- An evaluation of the control strategies WOA/GWO-TID-F and WOA/GWO-$PIDD^2$-PI with numerous different control schemes like PID, PI-D, PI-PD, $PIDD^2$-PD and optimization techniques. Robustness has been verified by analyzing some test cases with different parameters of the ball and balancer system, as well as with reference tracking to different ball positions.

Following is a brief outline of rest of the paper: A two-degree-of-freedom ball balancer system is described in Section 2, different controllers are described in Section 3, metaheuristic algorithms are described in Section 4 a summary of the findings and an analysis is presented in Section 5, while conclusion with future prospects are discussed in Section 6.

2. Dynamic Modeling of Ball and Balancer System

A ball and balancer system typically refers to a mechanical or dynamic arrangement involving a ball and a balancing mechanism. For a variety of applications, this system is often designed to ensure stability or equilibrium. Quanser Lab's equipment 2D ball and balancer system is used in this research represents the system depicted in Figure 1. The plate accommodates a ball that is freely moving. A gimbal with two degrees of freedom can be used to rotate the plate in any direction. Overhead USB cameras and vision units are used to determine the ball's position. The Quanser Rotary Servo Base Unit (SRV02) mathematical modeling instructions are as follows

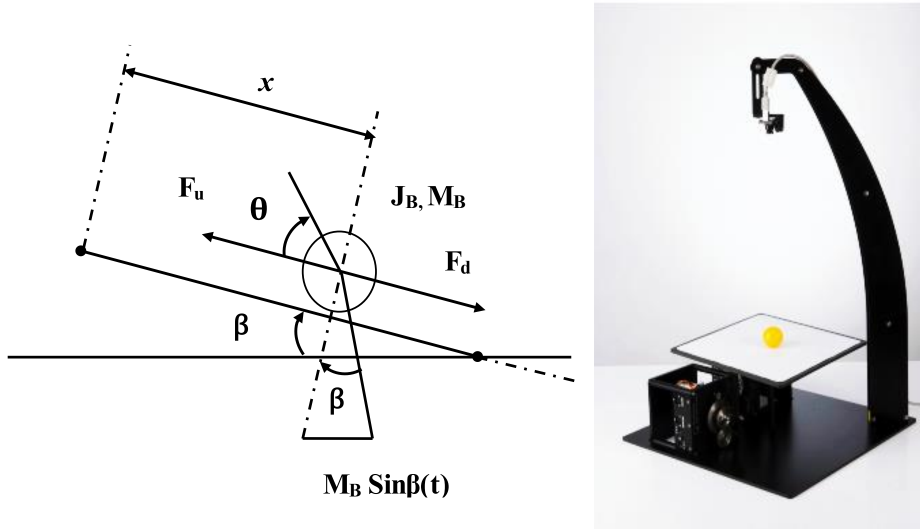


Figure 1. Diagram of Ball and balancer [35].

In order to move a ball while preserving the servo load angle following equation can be used [35]:

$$M_B\ddot{x}(t) = \sum F = F_u - F_d \tag{1}$$

As a result of inclination, the positive force is written as:

$$F_d = M_B g sin\beta(t) \tag{2}$$

Rotational forces are generated by the rotation of the ball:

$$F_d = \frac{\tau_B}{R_B} = \frac{J_B\ddot{\theta}(t)}{R_B} \tag{3}$$

where J_B is the moment of inertia of the ball is, θ is the ball angle and R_B is the radius of the ball. When the moment of the ball is applied in the x-direction, we have put $x = r\theta \Rightarrow \theta = \frac{x}{R_B}$ in (3):

$$F_d = \frac{J_B\ddot{x}(t)}{R_B^2} \tag{4}$$

Putting the value of F_u and F_d in (1):

$$M_B\ddot{x}(t) = M_B g sin\beta(t) - \frac{J_B\ddot{x}(t)}{R_B^2} \tag{5}$$

Now adding servo motor (SRV02) dynamics. The equation of motion representing the position of the ball rotation the ball of the servo motor SRV02 load gear $\alpha_L(t)$ and the beam angle $\beta(t)$ as $sin\beta(t) = \frac{2h}{L_T}$. Where L_T is the length of the table or plate and h is the height of the table or plate. Taking the sine of the load angle of the servo motor (SRV02):

$sin\alpha(t) = \frac{h}{r_a} \Rightarrow h = sin\alpha_L(t)r_a$. Where r_a is the distance between the couple joint and output gear-shaft.

$sin\beta(t) = \frac{2sin\alpha_L(t)r_a}{L_T}$ Put in (5):

$$M_B\ddot{x}(t) + \frac{J_B\ddot{x}(t)}{R_B^2} = \frac{2M_B g sin\alpha_L(t)r_a}{L_T} \tag{6}$$

Trigonometric value $sin\alpha_L(t)$ is a non-linear function. We use an approximation $sin\alpha_L(t) = \alpha_L(t)$ for linearization:

$$M_B\ddot{x}(t) + \frac{J_B\ddot{x}(t)}{R_B^2} = \frac{2M_Bg\alpha_L(t)r_a}{L_T} \tag{7}$$

Over all Transfer Function of ball and balancer system is denoted as $G(s)$, where the transfer function of servo motor is $G_{SR}(s)$ and ball & balancer system is $G_{BB}(s)$ as shown in Figure 2. So the transfer function of the ball and balancer system is:

$$G(s) = G_{SR}(s).G_{BB}(s) \tag{8}$$

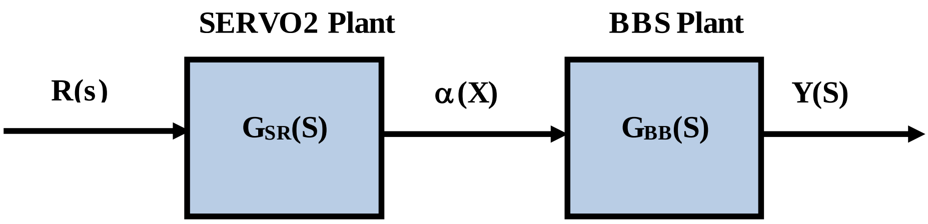


Figure 2. Open loop block diagram of Ball and balancer.

From (7):

$$\begin{aligned} M_B + \frac{J_B}{R_B^2}\ddot{x}(t) &= \frac{2M_Bg\alpha_L(t)r_a}{L_T} \\ \ddot{x}(t) &= \frac{2M_Bgr_aR_B^2}{L_T(M_BR_B^2 + J_B)}\alpha_L(t) \end{aligned} \tag{9}$$

Let model gain is the co-efficient of $\alpha_L(t)$ as:

$$K_1 = \frac{2M_Bgr_aR_B^2}{L_T(M_BR_B^2 + J_B)}\alpha_L(t) \tag{10}$$

$$\ddot{x}(t) = K_1\alpha_L(t) \tag{11}$$

The moment of inertia of the ball is calculated by using the values given in Table 1: $J_B = \frac{2}{5}M_BR_B^2 = \frac{2}{5}(0.003)(1.96)^2 = 0.0046$ kgm^2. Put in (10):

$$K_1 = \frac{2(0.003)(9.8)(2.54)(1.96)^2}{27.5((0.003)(1.96)^2 + 0.0046)} = 1.3 \tag{12}$$

Putting this in (11) we get

$$\ddot{x}(t) = 1.3\alpha_L(t) \tag{13}$$

Table 1. Parameter values for the proposed system [35].

Parameter	Symbol	Value
Length of table or plate	L_t	27.5 cm
Distance between couple joint and output gear shaft	r_a	2.54 cm
Radius of ball	R_B	1.96 cm
Mass of the ball	M_B	0.003 Kg

The Laplace transformation of the linear equation of motion is $s^2X(s) = 1.3\alpha_L(s) \Rightarrow \frac{X(s)}{\alpha_L(s)} = \frac{1.3}{s^2}$

$$G_{BB}(s) = \frac{X(s)}{\alpha_L(s)} = \frac{1.3}{s^2} \tag{14}$$

The angular rate of the SRV02 load shaft with $w_1(t)$, an input voltage $v_1(t)$, $k_2 = 1.76$ a steady state gain, and $\tau = 0.0285s$ is a time constant of the motor then transfer function of the motor is:

$$G_{SR}(s) = \frac{k_2}{(\tau s + 1)s} = \frac{1.76}{(0.0285s + 1)s} \tag{15}$$

Putting the values of $G_{BB}(s)$ and $G_{SR}(s)$ in (8):

$$G(s) = \frac{1.76}{(0.0285s + 1)s} \cdot \frac{1.3}{s^2} = \frac{2.28}{0.0285s^4 + s^3} \tag{16}$$

3. Controller Design

Two robust control strategies including the PIDD^2-PI, and TID-F tuned with meta-heuristic optimization algorithms, WOA and GWO are propose for the position control analysis of ball and balancer systems.

3.1. PID Controller

The ball and balancer system is a classic example of a control problem, where the objective is to keep a ball balanced on a platform by controlling the motion of the platform. One of the most commonly used control techniques for such systems is PID control. PID control is a feedback control technique. In the first step, classical PID controller is implemented tuned with the proposed schemes GWO & WOA as shown in Figure 3.

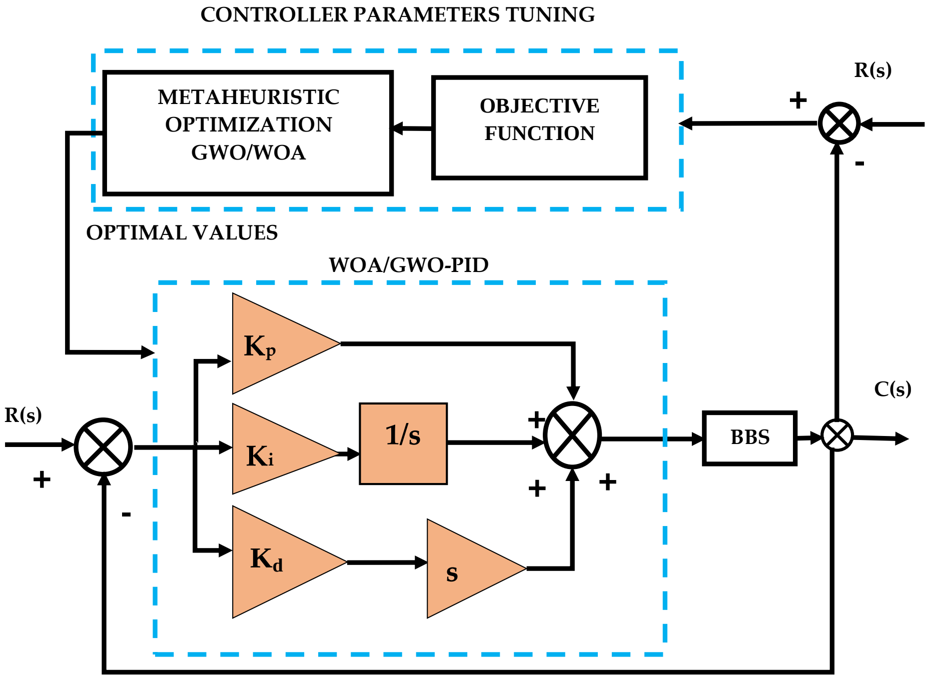


Figure 3. Control scheme of PID.

A PID control method uses feedback control. Depending on the amount of difference between the actual and desired outputs of the system, the control input is adjusted employing proportional, integral, and derivative terms. Control input is adjusted according to the error with the proportional term. As time passes, the integral term adjusts for accumulated errors. By using a derivative term, errors over time can be adjusted.

Ball and balancer systems are controlled by PID by measuring the ball position and comparing it with a set point that is desired. An appropriate control input is determined based on the error in the PID controller, which then adjusts the platform position as a result. This process is repeated continuously to maintain the ball at the required position.

The cascaded scheme for control is applied which is consisting of two loops. The inner loop and the outer loop as shown in Figure 4.

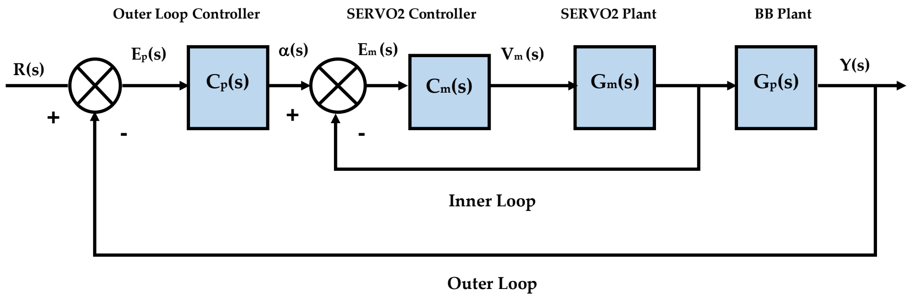


Figure 4. Block diagram of Ball and Balancer.

In order for the outer loop to be stabilized, the inner loop must first be stabilized. An inner loop keeps track of the angle of the motor Hence; the controller of inner loop should be programmed so the motor angle tracks the reference signal. Ball angles are controlled by outer loops using inner feedback loops. It is therefore necessary to begin with the inner loop. Firstly the inner loop is stabilized, with the inner loop gains Kp = 5.9462, Ki = 0.0136, and Kd = 0.0305. The closed inner loop transfer function using PID is:

$$G_{IL}(s) = \frac{0.05368s^2 + 10.47s + 0.02394}{0.0285s^3 + 1.054s^2 + 10.47s + 0.0239} \tag{17}$$

After the reduction block, the ball and balancer system is reduced as shown in Figure 5 below:

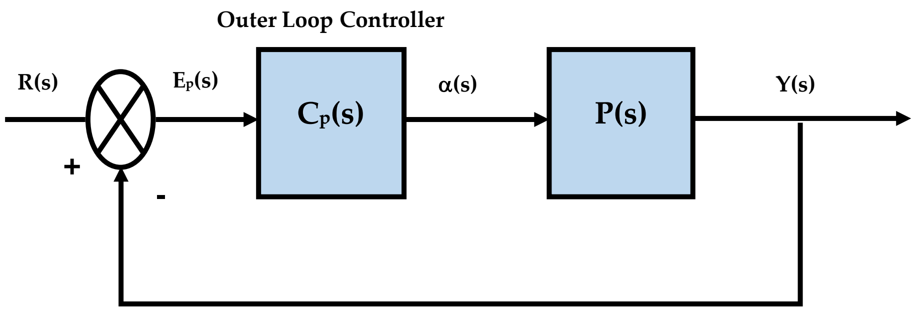


Figure 5. Reduced Block diagram of Ball and Balancer.

$$P(s) = \frac{0.06978s^2 + 13.6s + 0.03112}{0.0285s^5 + 1.054s^4 + 10.47s^3 + 0.02394s^2} \tag{18}$$

The error function E(s) of a complete system (inner and outer loop) for the PID controller:

$$E(s) = \frac{0.0285s^5 + 1.054s^4 + 10.47s^3 + 0.2394s^2}{0.0285s^6 + 1.054s^5 + (A)s^4 + (B)s^3 + (C)s^2 + (D)s + 0.03112K_i} \tag{19}$$

$A = 10.47 + 0.06978K_d$
$B = 0.2394 + 13.6K_d + 0.06978K_p$
$C = 0.03112K_d + 13.6K_p + 0.06978K_i$
$D = 0.03112K_p + 13.6K_i$

3.2. PIDD^2-PI Controller

The Proposed controller for the ball and balancer system is PIDD^2 coupled with the PI controller as shown in Figure 6.

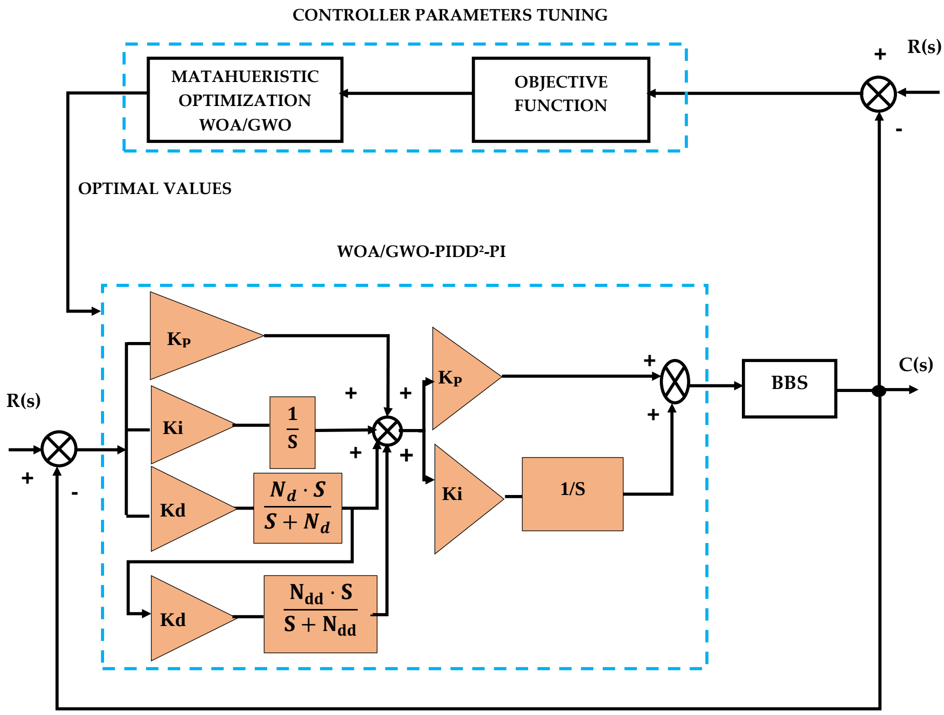


Figure 6. Control scheme of PIDD^2-PI.

Proportional Integral Derivative Double Derivative (PIDD^2) regulates a process variable to a desired set point. The PIDD^2 algorithm is an extension of the classical PID controller and adds a second derivative term to improve the system's performance. The PI (Proportional Integral) controller is a basic feedback control system that uses two control actions to regulate a process variable. An integral term represents the cumulative error over time in PIDD^2 and PI controllers. Proportional terms represent the current error between the set point and the actual process variable. The combination of PIDD^2 and PI control improves the performance. The PIDD^2 algorithm adds a second derivative term to the control signal, which helps to damp any overshoot or oscillations in the system response. On the other hand, PI controller adjust the control signal continuously based on the accumulated error of the system to eliminate steady-state errors. Error signals are calculated as differences between set points and process variables. A control signal is calculated via the PIDD^2 algorithm by combining integral, proportional, derivative, and double derivative terms. The process variable is measured and compared to the desired set point. The PI controller adjusts the control signal based on the accumulated error to eliminate steady-state error. As a result of the control signal, the actuator adjusts the process variable. The control loop is repeated to continuously regulate the process variable to the desired set point based on the next measurement of the process variable.

Overall, the PIDD^2 coupled with the PI controller provides a more advanced and robust control system that can handle a wider range of process dynamics and disturbances compared to the classical PID controller.

The transfer function of PIDD^2 and PI are written as:

$$G_{PIDD^2}(s) = K_p + \frac{K_I}{s} + K_D\left(\frac{N_d s}{s + N_d}\right) + K_D\left(\frac{N_d s}{s + N_d}\right).K_{DD}\left(\frac{N_{dd} s}{s + N_{dd}}\right) \tag{20}$$

$$G_{PI}(s) = K_p + \frac{K_I}{s} \tag{21}$$

The error function for ball and balancer systems using equations of PIDD^2-PI with ball and balancer system equation is written as:

$$E(s) = \frac{(0.0285s^5 + 1.054s^4 + 10.47s^3 + 0.02394s^2)(s + N_d)(s + N_{dd})}{[s(s + N_d)(s + N_{dd})](A) + [K_p s(s + N_d)(s + N_{dd}) + Bs^2(s + N_{dd}) + CN_{dd}s^3][D]} \tag{22}$$

$A = 0.0285s^5 + 1.054s^4 + 10.47s^3 + 0.02394s^2$
$B = K_D N_d$
$C = K_D N_d K_{DD}$
$D = 0.0978s^2 + 13.6s + 0.03112$

3.3. TID-F Controller

Filtered tilt integral derivative controller is a combination of TID with filter. The control scheme is shown in Figure 7. It is a feedback controller having four parameters (Kp, N, Ki, and Kd).

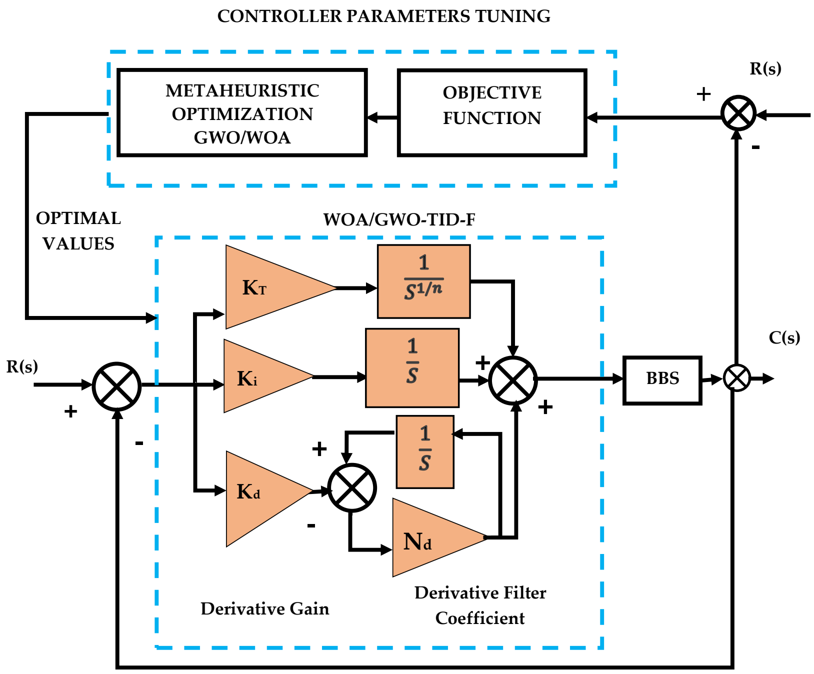


Figure 7. Control scheme of TID-F.

TID design [36] have some similarities with PID design, but they also have some differences; PID is modified by replacing "(1/s) n" with a real number (n) in place of the proportional constant. TID-F controller is mathematically represented as follows:

$$G_{TID-F} = \frac{K_T}{s^{\frac{1}{n}}} + \frac{K_I}{s} + K_D\left(\frac{N_s}{s + N}\right) \tag{23}$$

K_T, K_D, and K_I represent proportional/tilt, integral and derivative constants on controllers, respectively. The TID controller can be characterized as a combination of fractional order (FO) and integer controllers. TID has an advantage over FO and integer controllers. This method quickly eliminates disturbances between integers and FOs Where derivative filter coefficient is defined by the parameter N.

The error function for ball and balancer system using equation of TID-F with ball and balancer equation is written as

$$E(s) = \frac{s^{\frac{1}{n}}(s + N)(0.0285s^5 + 1.054s^4 + 10.47s^3 + 0.02394s^2)}{s^{1+\frac{1}{n}}(s + N)(A) + [K_T s(s + N) + K_i s^{\frac{1}{n}}(s + N) + K_D N s^{2+\frac{1}{n}}][B]} \tag{24}$$

$$A = 0.0285s^5 + 1.054s^4 + 10.47s^3 + 0.02394s^2$$
$$B = 0.06978s^2 + 13.6s + 0.03112 \quad (24)$$

4. Tuning of Controllers Using Metaheuristic Optimization Techniques

Heuristic computational techniques are important for solving optimization problems because they provide efficient and effective solutions when exact methods are impractical or too time-consuming. These techniques can be used in situations where the problem is constantly changing or dynamic, requiring quick adjustments to the solution approach. Overall, heuristic computational techniques are an essential tool for solving optimization problems in a timely and effective manner.

4.1. Whale Optimization Algorithm (WOA)

Humpback whales' natural hunting behavior is used to design the WOA [20,37–39]. Figure 8 is showing the pseudo code of WOA. In order to solve complex optimization problems, population-based optimization is used. It involves two main behaviors: searching and encircling prey. The algorithm uses these two behaviors to iteratively improve the fitness of the solution set. In the search phase, the whales move randomly toward the prey, while in the encircling phase, they surround the prey to trap it. By combining these two behaviors, the WOA algorithm forms a multi-objective optimization problem, with the goal of minimizing fitness. Brief description of its working is following.

Pseudo-code of Whale Optimization Algorithm (WOA)

```
Establish the initial population of whales Xi (i =1, 2, 3,...n)
X*= Agents who provide the best search results
while (t < Iterations maximum)
    for each search agent
    Update a, A, C, 1, and p
        if1 ( p < 0.5 )
            if2 ( |A| < 1)
        The current search agent's position should be updated by Equation
            else if2 ( |A| ≥ 1)
                Pick an agent at random (Xrand)
        The current search agent's position should be updated by Equation
            end if2
        else if1 (p ≥ 0.5)
                Update the position of the current search by Equation
        end if1
    end for
Make sure that no search agent goes beyond the search space and amend it if it does
    A fitness calculation should be performed for each search agent
    Update X* if there is a better solution
    t = t + 1
  end while
  return X*
```

Figure 8. Pseudo code of WOA.

4.1.1. General Structure of WOA

The algorithm only requires a small number of control parameters to be tuned, using only one parameter (time interval) needing adjustment. It is based on the assumption that populations of humpback whales searches for food in a multidimensional space, in which individuals' positions are represented by decision variables, and the distance between individuals and food is reflected in objective costs. There are three operational phases involved in whale action during its time-dependent location: shrinking encircling the prey, bubble-net attacks, and searching for prey. These operational processes are described and mathematically expressed in the following subsections.

4.1.2. Encircling Prey

Humpback whales have the ability to detect the location of prey and surround them. WOA assumes that the current most appropriate candidate solution represents the target prey or is near the optimal design since the exact position of the optimal design in the search space is unknown beforehand. While the algorithm seeks to identify the most efficient search agent, the remaining search agents adjust their positions around the most efficient search agent. The following equations are used to describe this behavior mathematically:

$$\overrightarrow{D} = |\overrightarrow{C}.\overrightarrow{X^*}(t) - \overrightarrow{X}(t)| \tag{25}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X^*}(t) - \overrightarrow{A}.\overrightarrow{D} \tag{26}$$

$$\overrightarrow{A} = 2\overrightarrow{a}.\overrightarrow{r} - \overrightarrow{a} \tag{27}$$

$$\overrightarrow{C} = 2.\overrightarrow{r} \tag{28}$$

In the given equation, X denotes the overall optimal position, X^* represents the position of a whale, t is the current iteration, a represents a linear reduction between 2 and 0.

Where D is the distance vector, that specifies the difference between the current position $X(t+1)$ denotes the updated position vector for the next and A is the coefficient vector used to update the position which depends on two variables r and a where a is a parameter that controls the spiral updating mechanism. r is a random vector ranging between 0 and 1, used to introduce randomness in the algorithm.

4.2. Grey Wolf Optimization (GWO)

GWOs are motivated by grey wolves' intelligent hunting tactics [40–44] and social structure. It is generally believed that grey wolves are ranked at the top of their habitat's hierarchy. It is common for grey wolves to live in groups of 5 to 12 individuals. Its primary goal is to develop the candidate solution during each iteration which makes GWO different from other metaheuristic optimization algorithms. In other words, GWO imitates grey wolf hunting behavior, which involves locating and attacking prey.

Grey wolves undergo the following stages of hunting, which has ben shown in Figure 9 in terms of pseudo code. The process of following, pursuing, and moving forward with the prey continually following, encircling, and harassing the prey. Targeting prey with an attack. According to the several wolf roles that aid in the advancement of the hunting process, the GWO pack is organized into four groups. It has been determined that alpha is the most successful hunting strategy out of the four, with beta, delta, and omega representing the others. In nature, grey wolves are divided into four groups based on their dominance structures. The creators of this algorithm carried out a thorough trial and discovered that taking. A grey wolf population is established as a random population in the GWO search procedure, similar to previous swarm intelligence algorithms. The four wolf groups and their positions are then established, and the distances to the intended prey are calculated. An update is made on each wolf as it symbolizes a potential solution during the search process. To prevent the local optima from stagnating, keep up the exploration and exploitation. Its mathematical model differs from that of other population-based algorithms in that it determines the global optimum by calculating the value of the global average.

To mimic grey wolves hunting and encircling their prey in the wild, it moves a solution around another in an n-dimensional space.

```
Pseudo-code of Grey Wolf Optimization Algorithm (GWO)
Input: Size of Problem, Size of Population
Output: best solution
Start
        Grey wolf populations are initialized Xi (i = 1, 2, ... , n)
        a, A, and C are initialized
        Assessing and grading search agents according to their fitness values.
(Xα = the most suitable solution in the search agent, Xβ = the second best solution
In the search agent, and Xδ = the third most optimal solution in the search agent.)
          t= 0
          While (t < Maximum number of iterations)
                For each search agent
                      Using Equation to update the current position of the search agent
           End for
        Bringing the three a's, A's, and C's up to date
        All search agents are graded on their fitness values
        Update the positions of Xα, Xβ, and Xδ
         t= t+1
        End while
End
```

Figure 9. Pseudo code of GWO.

5. Simulation Results and Discussion

In this section, position control of the ball on ball and balancer model with two degrees of freedom is designed and developed in Simulink/MATLAB. The problem is defined using an objective function or fitness function for optimization process, such as convergence of a metaheuristic algorithm toward the global optima of controller adjusted parameters.

The classical PID controller is initially applied to the ball and balancer system, but the results are unsatisfactory as shown in Figure 10.

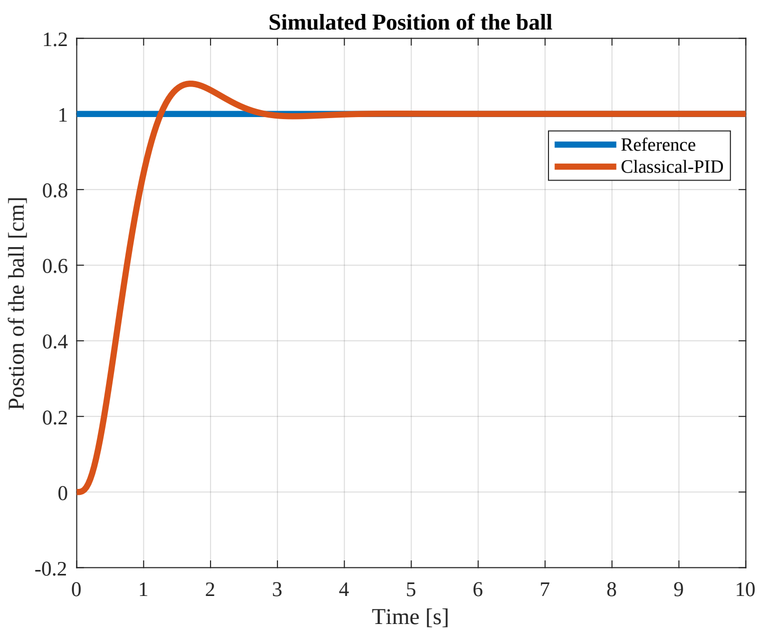


Figure 10. Control of the simulated position of the ball by using classical PID controller.

Table 2 shows that the classical PID controller has large settling time and overshoot. To overcome this situation cascaded control strategy is used for the implementation of the controller. The response of the system is deliberately divided into two sections. Firstly the response of proposed controllers with various others is analyzed by applying different metahueristic approaches like GWO, CSA, GBO and WOA respectively.

Table 2. Characteristics of step response for classical PID.

Name of Controller	Controller Parameters			Step Response Characteristics		
	Kp	Ki	Kd	Rise Time (s)	Settling Time (s)	Max. Overshoot
Classical-PID	3.45	0.0012	2.11	0.7721	2.4505	7.9829

The step response comparisons of controllers tuned by GWO are shown in Figure 11.

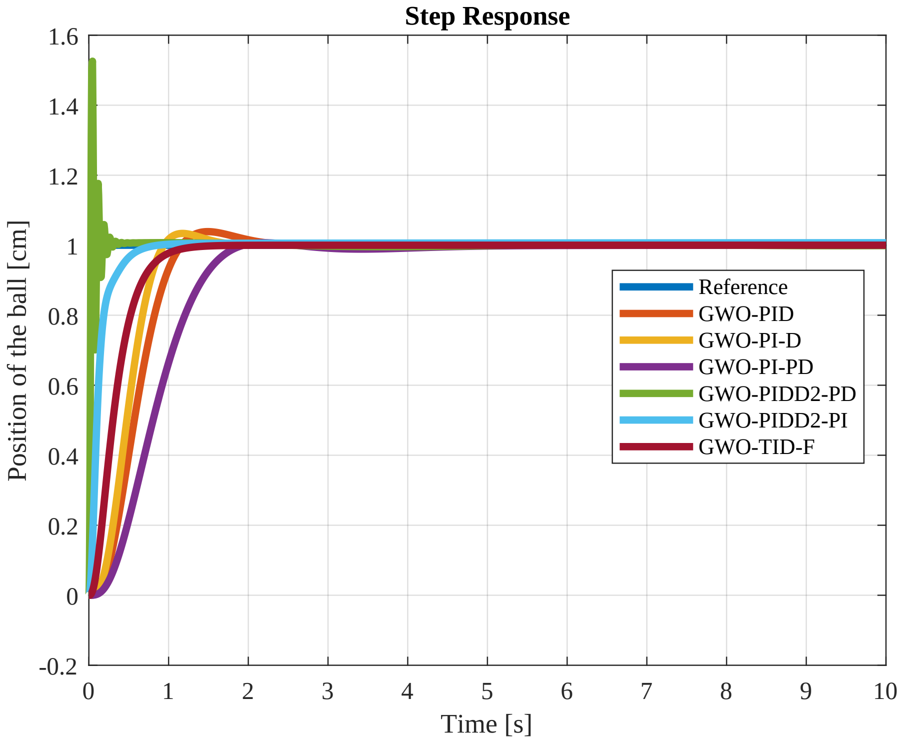


Figure 11. Step responses of the controllers tuned by GWO.

The proposed control schemes along with the various controllers, like PID, PI-D, PI-PD and PIDD^2-PI, are then optimized with the recent metaheuristic algorithm including GWO, CSA, GBO, and WOA and results are compared. The optimal gains for controllers tuned by GWO, CSA, GBO and WOA are shown in Table 3. These optimal gains are used further for the improved controller performances.

The GWO is used to tune the PID, PI-D, PI-PD, PIDD^2-PD,PIDD^2-PI and TID-F controllers with a focus on minimizing overshoot. However, despite achieving fast rise times, the PIDD^2-PD controller exhibits a large amount of overshoot. In contrast, the proposed PIDD^2-PI controller outperforms all other controllers in terms of performance, with a very small settling time 0.6421 s & TID-F settling time 1.0277 s along zero overshoots as shown in Table 4. It is clear that GWO-PIDD^2-PI & GWO-TID-F perform excellent, and GWO-PIDD^2-PI is quicker in rise and settling time than GWO-TID-F.

Table 3. Control parameters tuned by GWO, CSA, GBO and WOA.

Controller	Control Parameters									
	Kp1	**Ki1**	**Kd**	**Nd**	**Kdd**	**Ndd**	**Kp2**	**Ki2**	**kd2**	**n/nf**
GWO-PID	5.0548	0.017	2.8546	-	-	-	-	-	-	-
GWO-PI-D	7.4572	0.054	3.5443	-	-	-	-	-	-	-
GWO-PI-PD	2.3	2.5703	2.54	-	-	-	2.2227	-	-	-
GWO-PIDD^2-PD	1.648	4.921	3.8956	220	0.0267	115.5	25.792	1.4241	-	128.6
GWO-PIDD^2-PI	0.0003546	0.0415	13.9834	355.46	0.1007	124.55	0.376	0.0137	-	-
GWO-TID-F	12.238	0.000043	2.1444	554.234	-	-	-	-	-	126.532
CSA-PID	33.7239	1.0565	6.8412	-	-	-	-	-	-	-
CSA-PI-D	49.9232	2.5232	12.5026	-	-	-	-	-	-	-
CSA-PI-PD	21.3624	2.9843	7.6519	-	-	-	1.324	-	–	
CSA_PIDD^2-PD	12.648	3.921	2.8956	325	0.13	425	3.792		2.4241	335
CSA_PIDD^2-PI	0.0003546	0.0415	14.9834	355.46	0.1	124.55	0.378	0.0137		
CSA- TID-F	757.7291	0.4193	9.2916	494.1739	-	-	-	-	-	680.9719
GBO-PID	0.0121	0.0103	6.5043	-	-	-	-	-	-	-
GBO-PI-D	0.01	0.0107	6.4416	-	-	-	-	-	-	-
GBO-PI-PD	31.2998	0.0316	8.8136	-	-	-	0.01	-	-	-
GBO-PIDD^2-PD	1.876	14.676	3.832	425	0.023	500	13.957	-	2.437	229
GBO-PIDD^2-PI	1.15×10^{-6}	0.0115	5.6677	350	0.159	35	0.378	0.0137	-	-
GBO-TID-F	632.9403	0.0001501	4.9453	966.157	-	-	-	-	-	966.157
WOA-PID	3.575	0.0388	2.7664	-	-	-	-	-	-	-
WOA-PI-D	7.3475	0.3425	3.4567	-	-	-	-	-	-	-
WOA-PI-PD	4.3675	4.3675	5.54	-	-	-	3.815	-	-	-
WOA-PDD^2-PD	25	40	1.7171	324	0.1	475	14.253	-	2.6	500
WOA-PDD^2-PI	1.20×10^{-6}	0.00115	15.177	550	0.159	75	0.475	0.0023	-	-
WOA- TID-F	32.9978	0.001	1.9445	962.4152	-	-	-	-	-	316.5016

Table 4. Step response characteristics tuned by GWO.

Controller	Performance Parameters		
	Rise Time (s)	**Settling Time (s)**	**% Overshoot**
GWO-PID	0.6852	1.8686	3.6163
GWO-PI-D	0.5437	1.3747	2.9435
GWO-PI-PD	1.067	1.7463	0.6184
GWO-PIDD^2-PD	0.0147	0.2669	54.6068
GWO-PIDD^2-PI	0.287	0.6421	0
GWO-TID-F	0.5638	1.0277	0

Figure 12 shows that when controllers are tuned using the CSA, the PID and PIDD^2-PD controllers exhibited a very high degree of overshoot that is 31.0255% and 37.9147%. On the other hand, the PI-D controller has smaller overshoot that is 11.122%, while both the PI-PD and PIDD^2-PI controllers achieve zero overshoot. Furthermore, the PIDD^2-PI controller settled faster than the PI-PD controller. A very short rise time is achieved by the

PIDD^2-PD controller. TID-F has good rise and settling time with large overshoots. PI-PD and PIDD^2-PI controllers are better tuned by CSA. Step response characteristics are shown in Table 5.

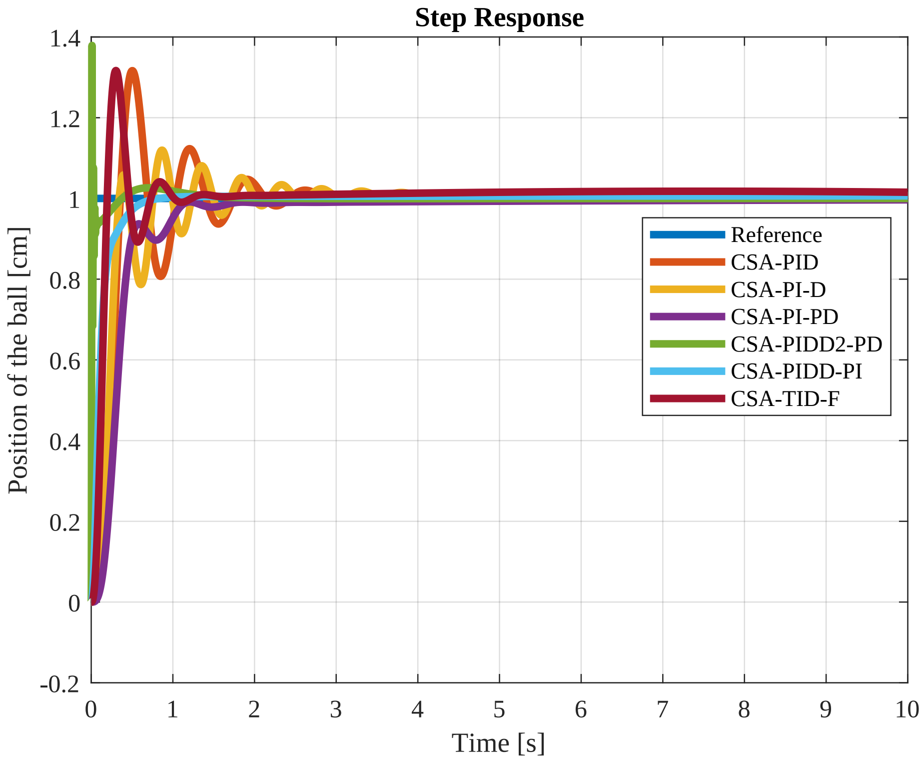


Figure 12. Step responses of the controllers tuned by CSA.

Table 5. Step response characteristics tuned by CSA.

Controller	Performance Parameters		
	Rise Time (s)	**Settling Time (s)**	**% Overshoot**
CSA-PID	0.1972	2.3242	31.0255
CSA-PI-D	0.1892	2.3922	11.122
CSA-PI-PD	0.3322	1.0797	0
CSA-PIDD^2-PD	0.0039	0.9545	37.9147
CSA-PIDD^2-PI	0.3516	0.7998	0
CSA- TID-F	0.1231	1.1808	29.6977

Figure 13 shows that, the proposed PIDD^2-PI controller is optimized very efficiently using GBO settled in 1.6919 s with zero overshoot, Unlike other controllers such as PID, PI-D, and PI-PD, which are having high overshoot. The PIDD^2-PD controller achieves a quick response with a small overshoot 5.5756%. Further step response characteristics are shown in Table 6.

Table 6. Step response characteristics tuned by GBO.

Controller	Performance Parameters		
	Rise Time (s)	**Settling Time (s)**	**% Overshoot**
GBO-PID	0.1554	0.8513	22.3784
GBO-PI-D	0.2255	1.9791	7.2446
GBO-PI-PD	0.237	1.4301	3.8612
GBO-PIDD^2-PD	0.0108	2.5234	5.5756
GBO-PIDD^2-PI	0.0878	1.6919	0
GBO-TID-F	0.2041	0.6558	9.9344

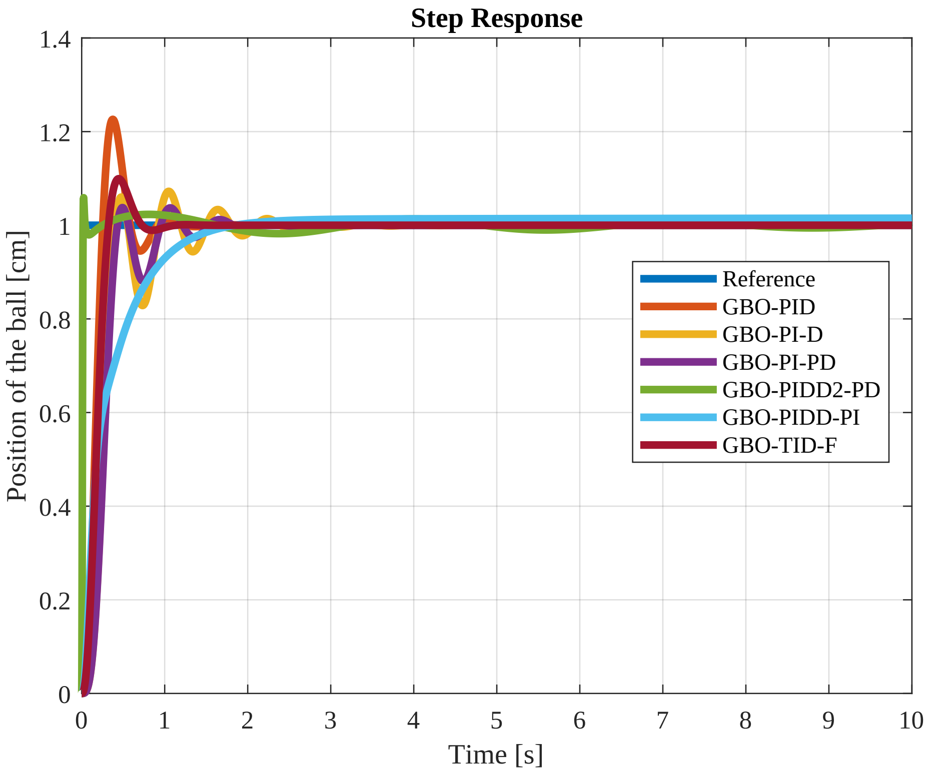


Figure 13. Step responses of the controllers tuned by GBO.

Figure 14 shows that the WOA algorithm is capable of optimizing a range of controllers, including PID, PID-D, PI-PD, PIDD^2-PD, and the proposed PIDD^2-PI. Based on WOA optimization, the PIDD^2-PD controller for the Ball and Balancer system achieves quick settlement in 0.3965 s and a really fast rise time that is 0.0059 s but having overshoot. On the other hand WOA-PDD^2-PI and WOA-TID-F are giving zero overshoots with 0.7 s and 1.21 s settling time, respectively, as shown in Table 7. Hence it is clear that GWO and WOA tuning schemes exhibit the most impressive performance among all.

Table 7. Step response characteristics tuned by WOA.

Controller	Performance Parameters		
	Rise Time (s)	**Settling Time (s)**	**% Overshoot**
WOA-PID	1.0526	1.8388	0.1134
WOA-PI-D	0.5395	1.5682	4.6004
WOA-PI-PD	1.5755	4.3581	2.2757
WOA-PDD^2-PD	0.0059	0.3965	6.1159
WOA-PDD^2-PI	0.0914	0.6521	0
WOA-TID-F	0.6504	1.2139	0

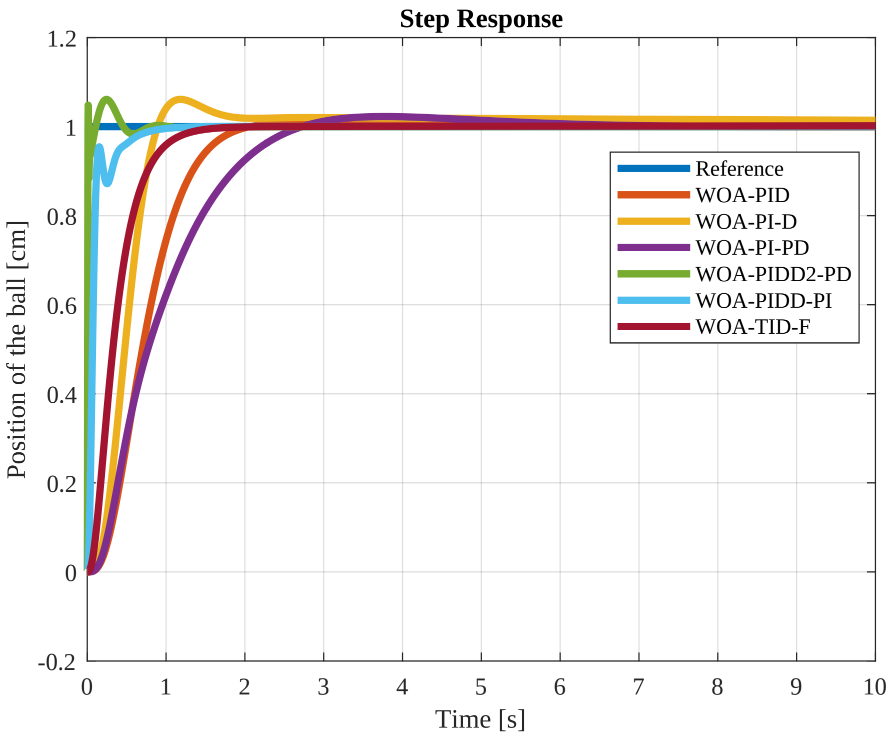


Figure 14. Step responses of the controllers tuned by WOA.

This section explains the response of proposed and comparison controllers tuned with different metahueristic approaches.

Figure 15 shows that PID is effectively optimized with WOA, but it tend to result in overshoots when used with CSA and GBO i-e 31.02% and 22.3% respectively. Accordingly, not all optimization algorithms are suitable for all control structures.

Figure 16 shows that the WOA efficiently optimized PI-D control method with 1.5682 s settling time but 4.6% overshoots as shown in Table 7. However, when applied to CSA and GBO, it tends to result in overshoots and take longer to settle compared to other control methods shown in Tables 5 and 6.

Figure 17 indicates that the PI-PD control method is effectively optimized by GWO with a significantly smaller settling time than the WOA that is 1.7463 s and 4.3581 s respectively . However, it can lead to overshoots when used with CSA and GBO shown in Tables 5 and 6.

Figure 18 shows that the PIDD^2-PD control method is efficiently optimized with GWO, CSA, GBO and WOA with rapid rise time shown in Tables 4–7. While it results in significant overshoots with GWO, and only minor overshoots with GBO, CSA, and WOA.

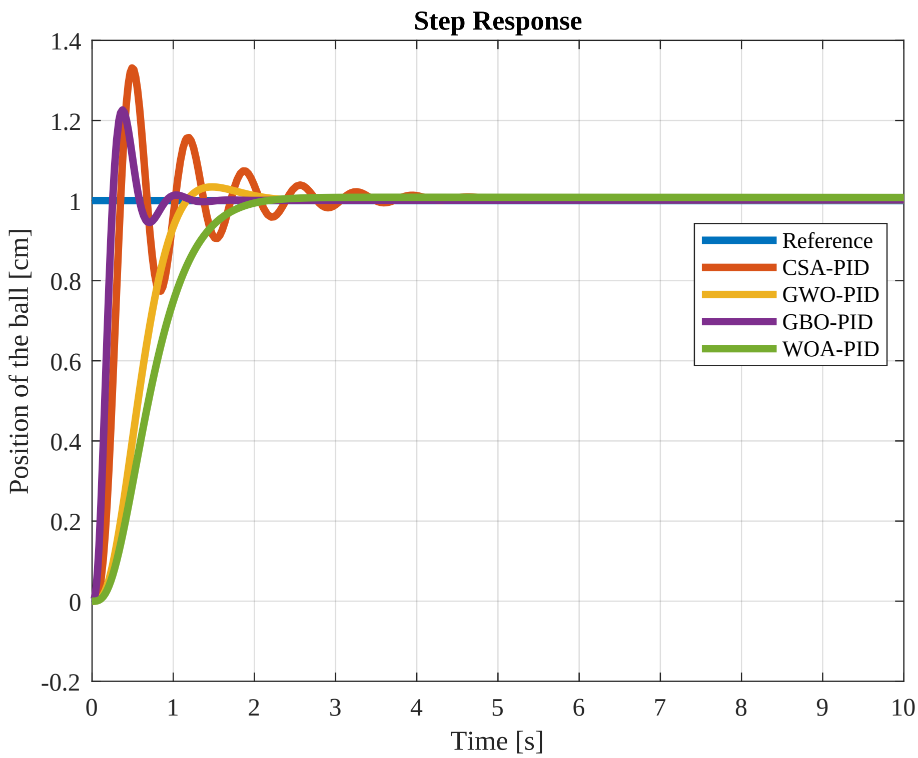


Figure 15. PID controller response with CSA/GWO/GBO/WOA.

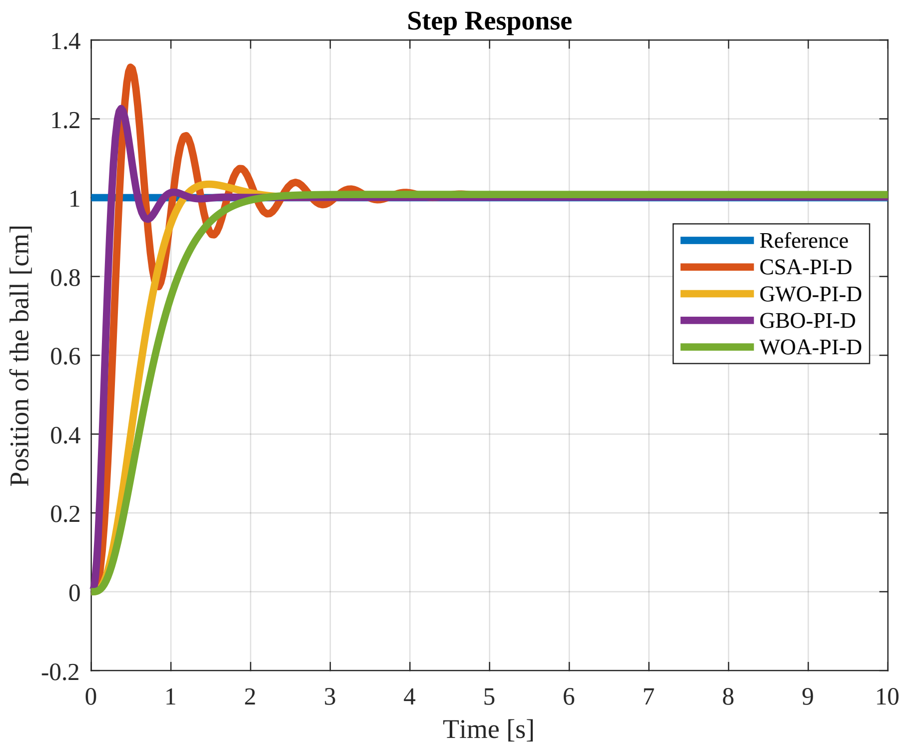


Figure 16. PI-D controller responses with CSA/GWO/GBO/WOA.

Figure 19 shows that the proposed controller PIDD^2-PI is more effectively optimized by the GWO, CSA, GBO, and WOA shown in Tables 4–7. This control method is associated with quick rise time and very short settling time, with no overshoots observed. These findings indicate that the PIDD^2-PI control method can deliver satisfactory results when applied to all optimization techniques.

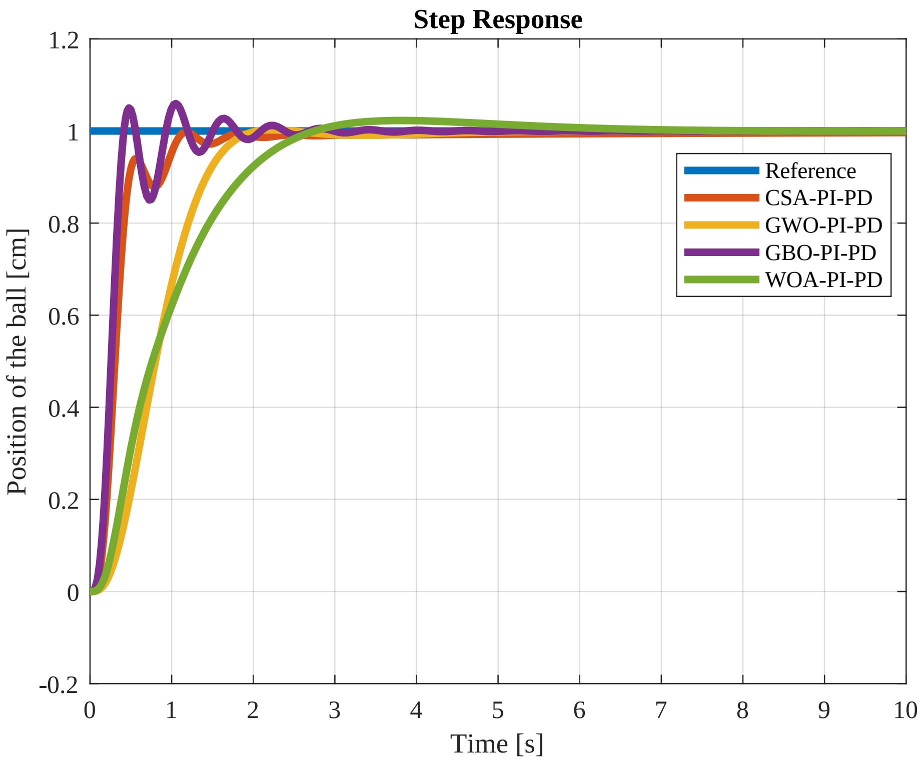


Figure 17. PI-PD controller response with CSA/GWO/GBO/WOA.

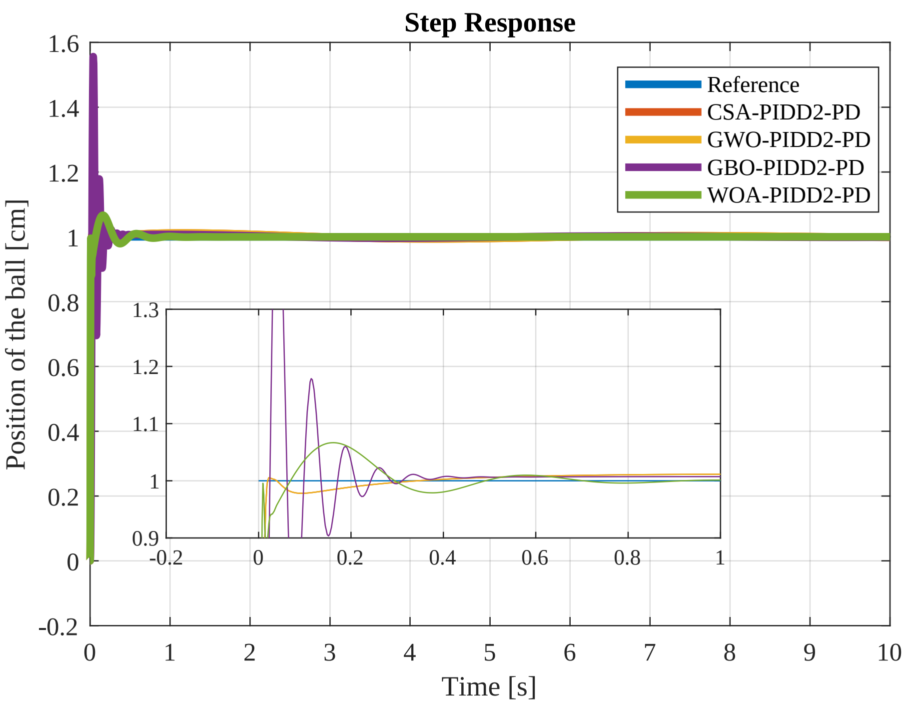


Figure 18. PIDD^2-PD controller response with CSA/GWO/GBO/WOA.

Figure 20 demonstrates that TID-F control method is effectively optimized by the GWO and WOA but gives some overshoots when tuned by CSA, GBO. TID-F tuned GWO, and WOA gives satisfactory results in rise and settling time with no overshoots. According to the analysis of the results, PIDD^2-PI and TID-F performed best with WOA and GWO, respectively.

In Figure 21a, the rise time is plotted against each controller. In Figure 21b, the Settling time is plotted against each controller. In Figure 22, the maximum overshoot is plotted against each controller.

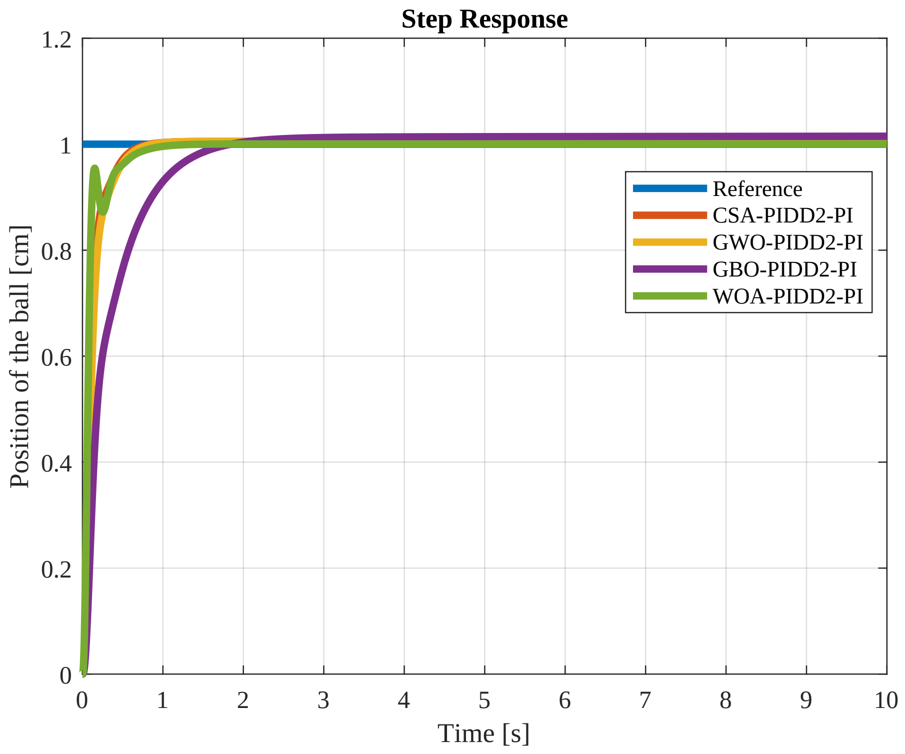


Figure 19. PIDD^2-PI response with CSA/GWO/GBO/WOA.

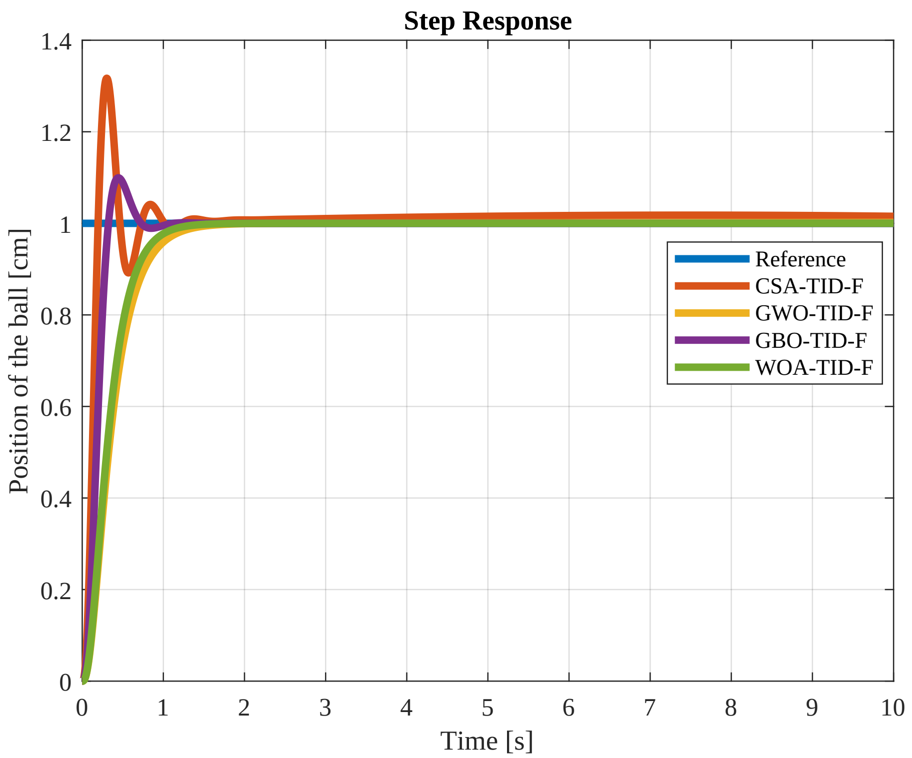
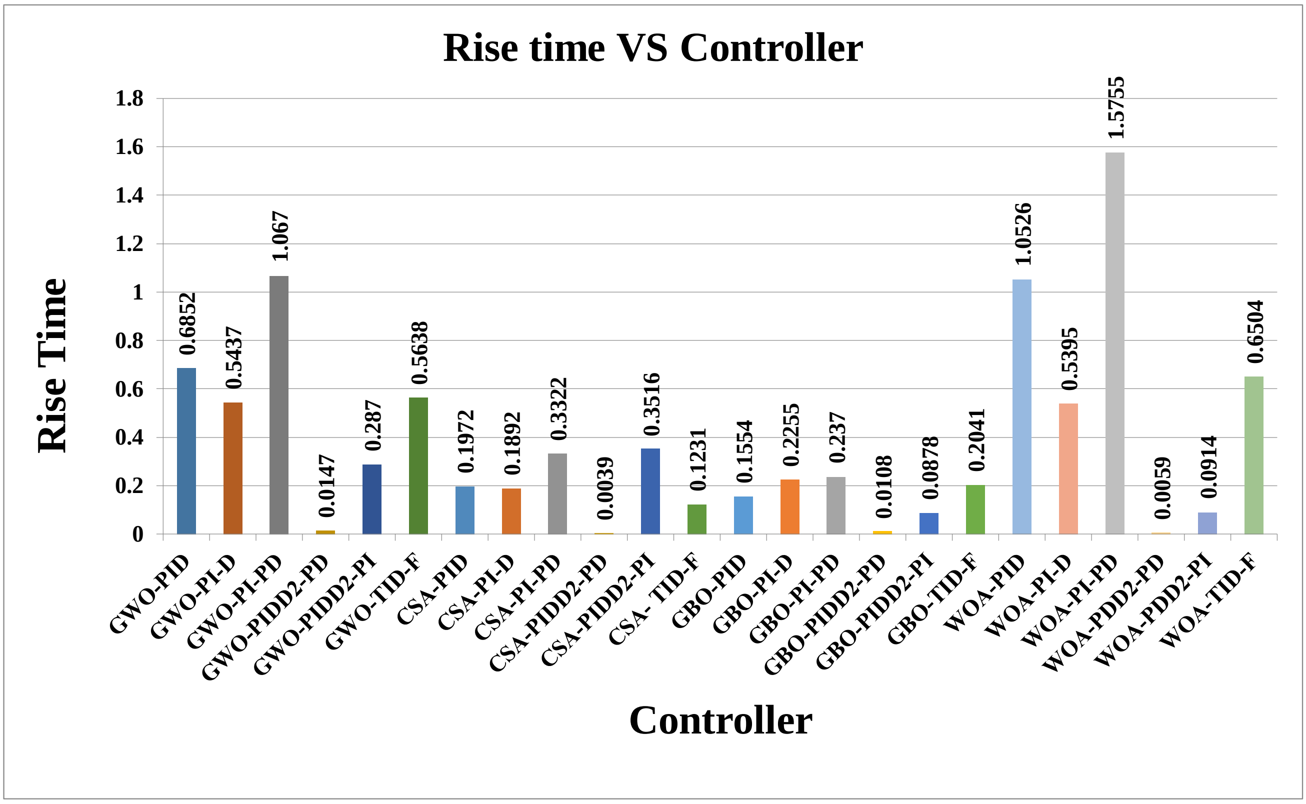
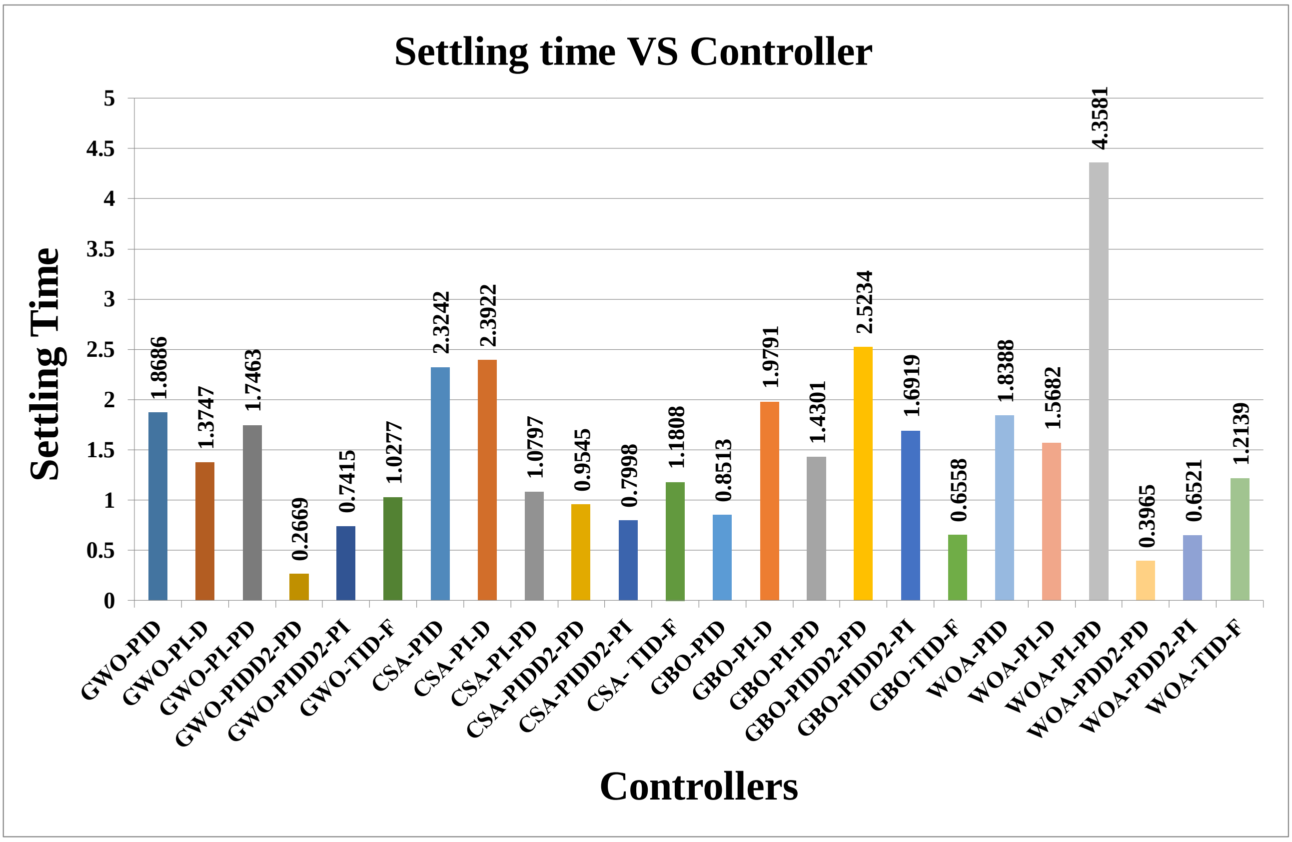


Figure 20. TID-F controller responses with CSA/GWO/GBO/WOA.



(**a**)



(**b**)

Figure 21. Controllers comparison. (**a**) Graphs of controllers versus rise time. (**b**) Graph of controller response versus setting time.

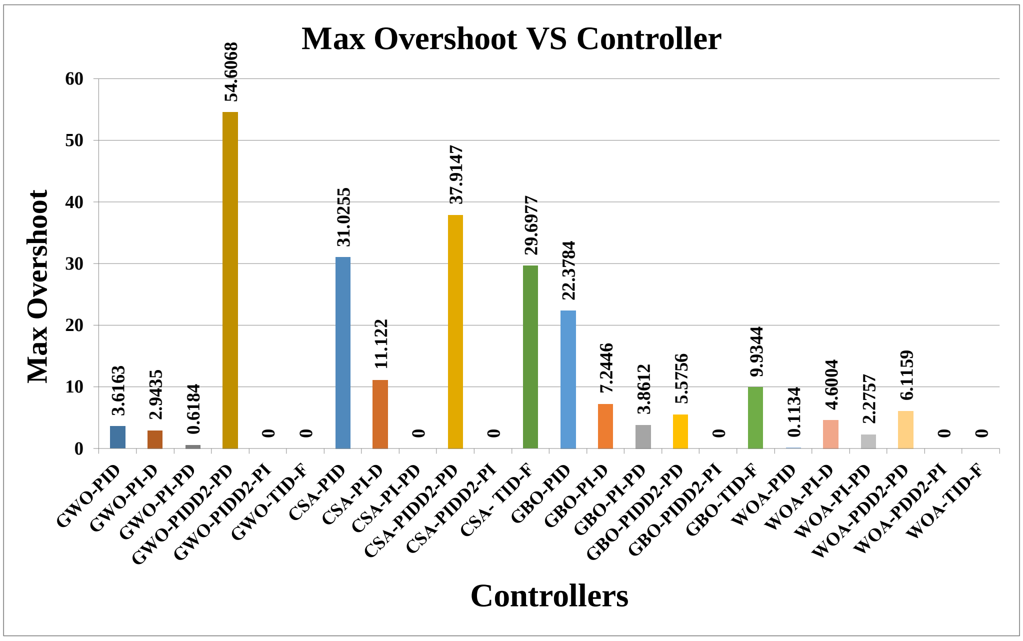


Figure 22. Graphs of controllers versus maximum overshoot.

5.1. Case Study

This section provides analysis and comparison of proposed control strategies with relevant published research work. It demonstrates the improved performance and robustness. The robustness of proposed control schemes is verified by applying change in parameters of ball and balancer system. In addition step response performance and set point tracking capabilities of proposed strategies are validated.

5.1.1. Case 1

A comparison is made between the proposed controllers GWO-TID-F & GWO-PIDD^2-PI and SMS-PID & CSMSEOBL-PID [3] by using the model parameters used in Table 1 for a ball and balancer system.

5.1.2. Step Response Comparison with SMS-PID & CSMSEOBL-PID

The step response comparison of proposed control schemes GWO-TID-F & GWO-PIDD^2-PI with SMS-PID & CSMSEOBL-PID [3] are shown in Table 8.

Table 8. Performance Comparison of purposed controllers with SMS-PID & CSMSEOBL-PID [3].

Controller	Performance Parameters		
	Rise Time (s)	**Settling Time (s)**	**% Overshoot**
SMS-PID [3]	0.57	2.1	0.496
CSMSEOBL-PID [3]	0.447	2.2	0
GWO-TID-F	0.5637	1.0277	0
WOA-PIDD^2-PI	0.0914	0.6521	0

Figure 23 shows the comparison of step responses for GWO-TID-F & GWO-PIDD^2-PI versus SMS-PID & CSMSEOBL-PID. It shows that the proposed controller WOA-PIDD^2-PI gives an excellent response in terms of rise and settling time with zero overshoot. WOA-PIDD^2-PI reduced 79.55% rise time and decreased 70.35% settling time compared to CSMSEOBL-PID. The 2nd proposed controller GWO-TID-F reduced 53.28% settling time compared to CSMSEOBL-PID with zero overshoot.

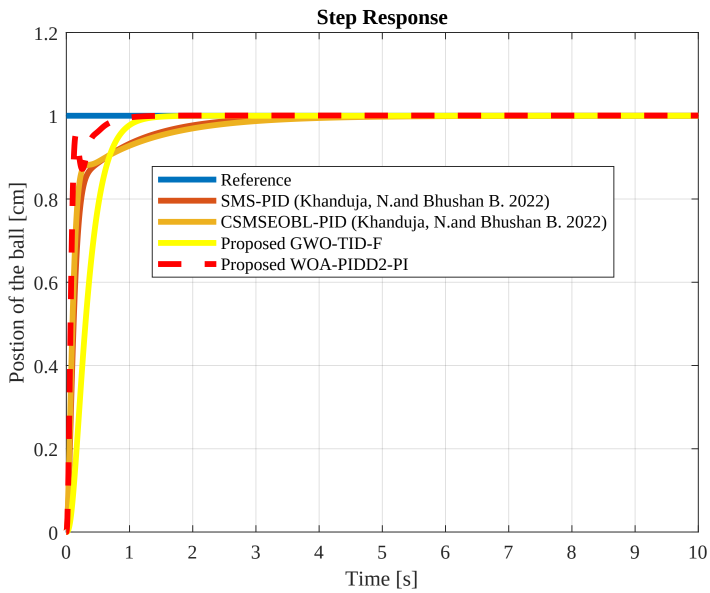


Figure 23. Step response comparisons of GWO-TID-F & WOA-PIDD^2-PI versus SMS-PID & CSMSEOBL-PID [3].

5.1.3. Set Point Tracking with SMS-PID & CSMSEOBL-PID

The set point tracking performance is compared by using the reference track signal used in [3] . Comparison of the tracking performance is made with GWO-TID-F & WOA-PIDD^2-PI versus SMS-PID & CSMSEOBL-PID [3].

Figure 24 shows the comparison of tracking responses for GWO-TID-F & WOA-PIDD^2-PI versus SMS-PID & CSMSEOBL-PID. It is evident that the proposed controller WOA-PIDD^2-PI tracks the input signal ideally. The other proposed controller GWO-TID-F tracks the input signal excellently compared to SMS-PID & CSMSEOBL-PID with zero overshoot. Figure 25 shows the comparison of step responses for GWO-TID-F & WOA-PIDD^2-PI versus DDPG-FC-350-R-PID [4]. Figure 26 is showing the comparison of proposed controllers versus SMS-PID & CSMSEOBL-PID.

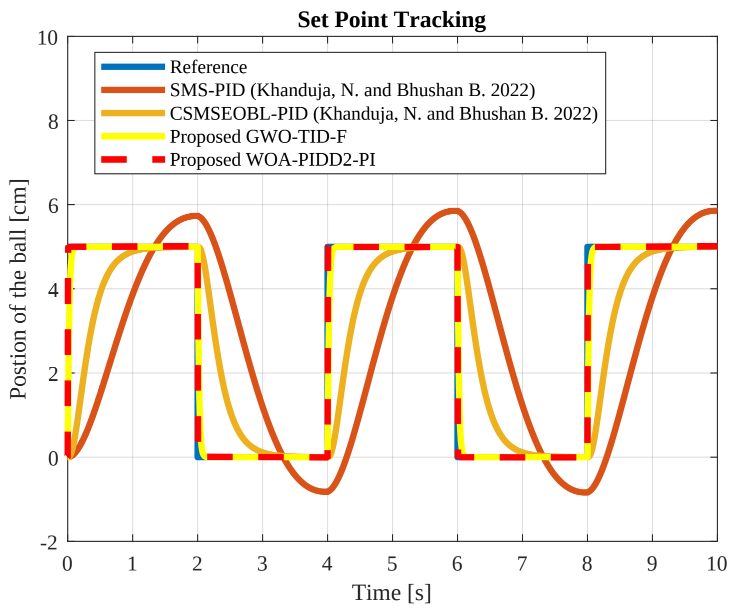


Figure 24. Tracking response comparisons of GWO-TID-F & GWO-PIDD^2-PI versus SMS-PID & CSMSEOBL-PID [3].

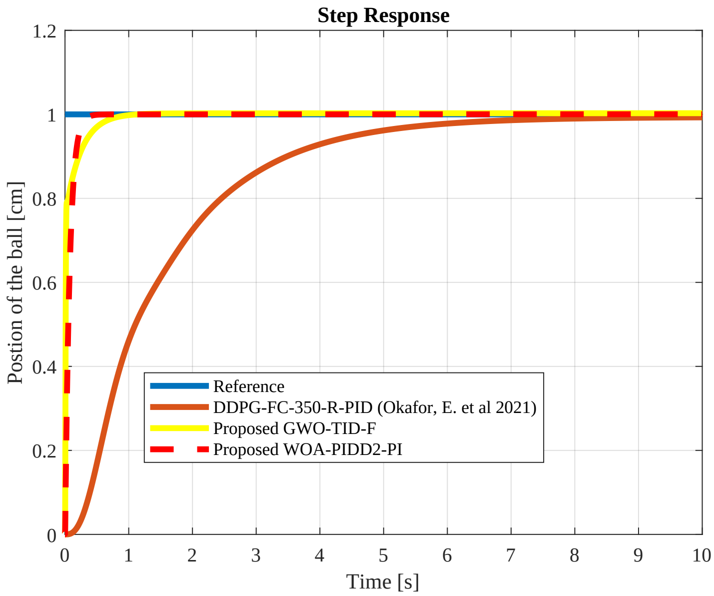


Figure 25. Step response comparisons of GWO-TID-F & GWO-PIDD^2-PI versus DDPG-FC-350-R-PID [4].

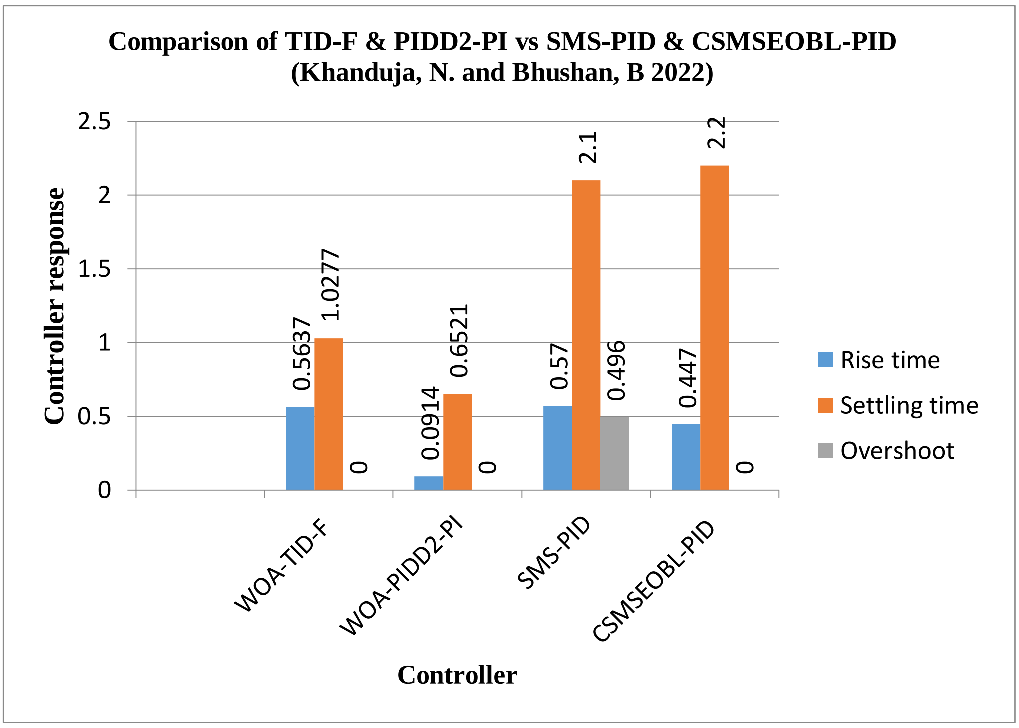


Figure 26. Comparison b/w proposed controller vs. SMS-PID & CSMSEOBL-PID [4].

5.1.4. Case 2

In CASE 02, the proposed controllers are tested with another related controller deep deterministic policy gradient based PID DDPG-FC-350-R-PID [4] by taking different model parameters. The motor is treated as a first-order system and denote it as $G_m(s)$. The gain compensator $k_m = -0.6854$ and time constant $T_m = 0.187$ [4]. Servo Motor transfer function is:

$$G_m(s) = \frac{k_m}{T_m s + 1} = \frac{-0.6854}{0.187s + 1} \tag{29}$$

Transfer function of Ball Balancer is:

$$G_{BB} = \frac{5g}{7s^2} = \frac{7.007}{s^2} \tag{30}$$

Combine transfer function of the Plant is [4]:

$$G_p(s) = G_m(s)G_{BB}(s) = \frac{4.803}{0.187s^3 + s} \tag{31}$$

5.1.5. Step Response Comparison with DDPG-FC-350-R-PID

The step response comparison of proposed control schemes GWO-TID-F & WOA-PIDD^2-PI using the transfer function of ball and plate system as given in (25) with DDPG-FC-350-R-PID [4] is shown in Table 9.

Table 9. Performance Comparison of proposed controllers versus DDPG-FC-350-R-PID [4].

Controller	Performance Parameters		
	Rise Time (s)	**Settling Time (s)**	**% Overshoot**
DDPG-FC-350-R-PID [4]	3.0440	5.6865	0.01170
GWO-TID-F	0.2201	0.6237	0
WOA-PIDD^2-PI	0.1579	0.2841	0

It is clear that proposed controller WOA-PIDD^2-PI gives an excellent response in terms of rise and settling time with zero overshoot. WOA-PIDD^2-PI reduced 94.81% rise time and decreased 95% settling time compared to DDPG-FC-350-R-PID, as plotted in Figure 27. The other proposed controller GWO-TID-F reduced 92.76% rise time & decreased 89.50% settling time compared to DDPG-FC-350-R-PID with zero overshoot.

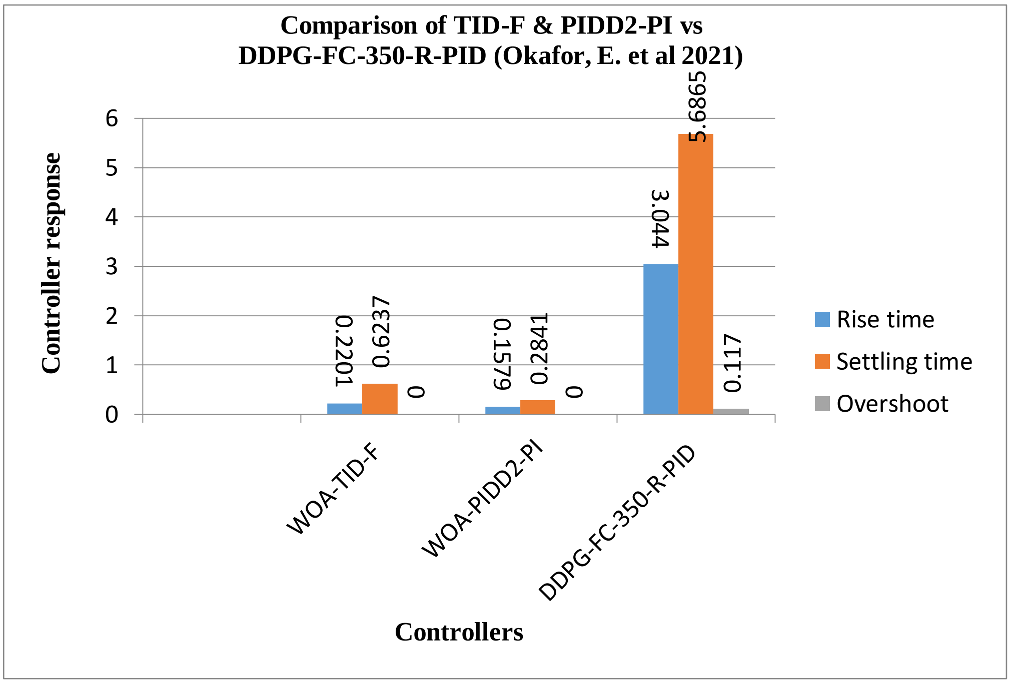


Figure 27. Comparison b/w proposed controller vs. DDPG-FC-350-R-PID [4].

6. Conclusions & Future Work

In this work, the PIDD^2 with PI and TID with filter (TID-F) are proposed control strategies for a ball and balancer system with ITAE as the objective function. The proposed controllers are optimized with CSA, GWO, GBO and WOA. The findings of this research are:

- PIDD^2-PI is best tuned with WOA as compared other optimization algorithms like CSA, GWO, and GBO.
- Optimal tuning of TID-F is achieved with GWO over other optimization algorithms like CSA, WOA, and GBO.
- The WOA-PIDD^2-PI and GWO-TID-F controller gives the best response in terms of settling time, rise time and overshoots then others PID, PI-D, PI-PD & PIDD^2-PD.
- The proposed controller WOA-PIDD^2-PI reduced 73.38% settling time, 88.16% rise time compared to classical-PID with zero overshoot
- The proposed controller GWO-TID-F reduced 58.06% settling time, 26.96% rise time compared to classical-PID with zero overshoot
- WOA-PIDD^2-PI reduced 83.96% rise time and 68.94% settling time compared to SMS-PID [3] with zero overshoot
- GWO-TID-F reduced 51.06% settling time compared to SMS-PID [3] with zero overshoot.
- WOA-PIDD^2-PI reduced 79.55% rise time and 70.35% settling time compared to CSMSEOBL-PID [3] with zero overshoot.
- GWO-TID-F reduced 53.28% settling time compared to CSMSEOBL-PID [3] with zero overshoot.
- WOA-PIDD^2-PI reduced 94.81% rise time and 95% settling time compared to DDPG-FC-350-R-PID [4] with zero overshoot.
- GWO-TID-F reduced 92.76% rise time & 89.03% settling time compared to DDPG-FC-350-R-PID [4] with zero overshoot.

As a result of this research study, it can be concluded that TID-F is best tuned with GWO, and that PIDD^2-PI is perfectly tuned with WOA. In comparison with related recent published work, GWO-TID-F and WOA-PIDD^2-PI control schemes are robust and provide superior performance in terms of rising and settling times with zero overshoot with a ball and balancer system. The ball and balancer system likely aims to maintain stability while efficiently controlling the position or movement of the ball. Numerical improvements indicate better stability and enhanced performance of the proposed controllers compared to other methods. This is particularly important in applications where precise control is required, such as in robotics, manufacturing, or even in consumer electronics.

Controllers that exhibit numerical improvements may result in more energy-efficient systems. This is especially relevant in battery-powered devices or systems where minimizing energy usage is essential for prolonged operation. In real-world applications, especially those involving dynamic environments or interacting with external factors, a faster response time can be crucial. For example, in autonomous vehicles or robotic systems, quick and precise control responses are essential for avoiding obstacles or adapting to changing conditions. The ball and balancer system may encounter external disturbances or uncertainties. If the proposed controllers demonstrate numerical improvements in terms of robustness, it implies a better ability to handle disturbances and uncertainties. In practical applications, this robustness can be critical for ensuring reliable performance in varied and unpredictable environments.

Both control strategies provide excellent set point tracking. There are following few future recommendations: There may be room for further optimization of the controller parameters to improve its performance in specific scenarios and implemented to other under actuated systems. The PIDD^2-PI and TID-F controller can be hybridized with other metaheuristic algorithms such as the Teaching–Learning-Based optimization (TLBO), League Championship Algorithms (LCA), exchange market algorithm (EMA), seeker optimization algorithm (SOA) and social-based algorithm (SBA) etc., to solve complex control problems. Real-time implementation of the PIDD^2-PI and TID-F controller on embedded hardware platforms can be a future research direction to explore its efficacy in industrial automation systems.

Author Contributions: Conceptualization, F.Z., S.A.A. and A.M.A.; Data curation, T.A., A.D. and F.B.; Formal analysis, T.A.; Funding ac-quisition, S.A.A. and A.M.A.; Investigation, T.A. and A.D.; Methodology, F.Z. and A.D.; Project administration, S.A.M. and S.A.A.; Resources, S.A.A.; Software, F.Z., T.A. and F.B.; Supervision, S.M.; Validation, F.B.; Writing—original draft, F.Z.; Writing—review & editing, S.A.M., T.A., A.D., F.B., A.M.A. and S.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Research Supporting Project Number (RSP2024R421), King Saud University, Riyadh, Saudi Arabia.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Mehmood, N. Non-Linear Control Techniques for Stabilization of Underactuated Mechanical Systems. Ph.D. Thesis, Capital University, Bexley, OH, USA, 2019.
2. Boscariol, P.; Richiedei, D. Robust point-to-point trajectory planning for nonlinear underactuated systems: Theory and experimental assessment. *Robot. Comput.-Integr. Manuf.* **2018**, *50*, 256–265. https://doi.org/10.1016/j.rcim.2017.10.001.
3. Khanduja, N.; Bhushan, B. Hybrid State of Matter Search Algorithm and its Application to PID Controller Design for Position Control of Ball Balancer System. *Iran. J. Sci. Technol. Trans. Electr. Eng.* **2022**, *46*, 785–804. https://doi.org/10.1007/s40998-022-00506-4.
4. Okafor, E.; Udekwe, D.; Ibrahim, Y.; Bashir Mu'azu, M.; Okafor, E.G. Heuristic and deep reinforcement learning-based PID control of trajectory tracking in a ball-and-plate system. *J. Inf. Telecommun.* **2021**, *5*, 179–196. https://doi.org/10.1080/24751839.2020.1833137.
5. Shuprajhaa, T.; Sujit, S.K.; Srinivasan, K. Reinforcement learning based adaptive PID controller design for control of linear/nonlinear unstable processes. *Appl. Soft Comput.* **2022**, *128*, 109450. https://doi.org/10.1016/J.ASOC.2022.109450.
6. Fan, X.; Zhang, N.; Teng, S. Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme. *Fuzzy Sets Syst.* **2004**, *144*, 297–312. https://doi.org/10.1016/S0165-0114(03)00135-0.
7. Ivanov, V. A review of fuzzy methods in automotive engineering applications. *Eur. Transp. Res. Rev.* **2015**, *7*, 707. https://doi.org/10.1007/s12544-015-0179-z.
8. Rastin, M.A.; Talebzadeh, E.; Moosavian, S.A.A.; Alaeddin, M. Trajectory tracking and obstacle avoidance of a ball and plate system using fuzzy theory. In Proceedings of the 2013 13th Iranian Conference on Fuzzy Systems (IFSC), Qazvin, Iran, 27–29 August 2013; pp. 1–5. https://doi.org/10.1109/IFSC.2013.6675631.
9. Robayo Betancourt, F.I.; Brand Alarcon, S.M.; Aristizabal Velasquez, L.F. Fuzzy and PID controllers applied to ball and plate system. In Proceedings of the 2019 IEEE 4th Colombian Conference on Automatic Control (CCAC), Medellin, Colombia, 15–18 October 2019; pp. 1–6. https://doi.org/10.1109/CCAC.2019.8921113.
10. Pattanapong, Y.; Deelertpaiboon, C. Ball and plate position control based on fuzzy logic with adaptive integral control action. In Proceedings of the 2013 IEEE International Conference on Mechatronics and Automation, Takamatsu, Japan, 4–7 August 2013; pp. 1513–1517. https://doi.org/10.1109/ICMA.2013.6618138.
11. Nikita, N.; Bhushan, B. Effect of Parameter Variation of Ball Balancer System Using PD and Fuzzy Control. In Proceedings of the 2022 IEEE Delhi Section Conference (DELCON), New Delhi, India, 11–13 February 2022; pp. 1–6. https://doi.org/10.1109/DELCON54057.2022.9753641.
12. Oussama, H.; Mohamed, B.; Abdennour, Z.; Hİchem, S. Tracking control of a ball on plate system using PID controller and Lead/Lag compensator with a double loop feedback scheme. *Eur. J. Sci. Technol.* **2021**, 28, 375–380. https://doi.org/10.31590/ejosat.1000437.
13. Lawrence, B. Tuning of A PID Controller for Optimal Performance of Ball and Beam System. *Int. J. Eng. Res.* **2020**, *V9*, 1–5. https://doi.org/10.17577/IJERTV9IS040009.
14. Ahamad, N.; Uniyal, S.; Sikander, A.; Singh, G. A comparative study of PID controller tuning techniques for time delay processes. *UPB Sci. Bull. Ser. C Electr. Eng.* **2019**, *81*, 129–142.
15. Rao, C.S.; Santosh, S.; V, D.R. Tuning optimal PID controllers for open loop unstable first order plus time delay systems by minimizing ITAE criterion. *IFAC-PapersOnLine* **2020**, *53*, 123–128. https://doi.org/10.1016/j.ifacol.2020.06.021.
16. Buchi, R. Optimal ITAE Criterion PID Parameters for PTn Plants Found with a Machine Learning Approach. In Proceedings of the 2021 9th International Conference on Control, Mechatronics and Automation (ICCMA), Belval, Luxembourg, 11–14 November 2021; pp. 50–54. https://doi.org/10.1109/ICCMA54375.2021.9646211.
17. Martins, F.G. Tuning PID controllers using the ITAE criterion. *Int. J. Eng. Educ.* **2005**, *21*, 867.

18. Surana, A.; Bhushan, B. Design and Comparison of PSO, SA and GA tuned PID Controller for Ball Balancer Arrangement. In Proceedings of the 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), Erode, India, 15–17 September 2021; pp. 1–5. https://doi.org/10.1109/ICECCT52121.2021.9616686.
19. Oglah, A.A.; Msallam, M.M. Real-time implementation of Fuzzy Logic Controller based on chicken swarm optimization for the ball and plate system. *Int. Rev. Appl. Sci. Eng.* **2022**, *13*, 263–271. https://doi.org/10.1556/1848.2021.00360.
20. Loucif, F.; Kechida, S.; Sebbagh, A. Whale optimizer algorithm to tune PID controller for the trajectory tracking control of robot manipulator. *J. Braz. Soc. Mech. Sci. Eng.* **2020**, *42*, 1280. https://doi.org/10.1007/s40430-019-2074-3.
21. Liu, H.; Liang, Y. Trajectory tracking sliding mode control of ball and plate system. In Proceedings of the 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010), Wuhan, China, 6–7 March 2010; pp. 142–145. https://doi.org/10.1109/CAR.2010.5456649.
22. Bang, H.; Lee, Y.S. Embedded Model Predictive Control for Enhancing Tracking Performance of a Ball-and-Plate System. *IEEE Access* **2019**, *7*, 39652–39659. https://doi.org/10.1109/ACCESS.2019.2907111.
23. Tudić, V.; Kralj, D.; Hoster, J.; Tropčić, T. Design and Implementation of a Ball-Plate Control System and Python Script for Educational Purposes in STEM Technologies. *Sensors* **2022**, *22*, 1875. https://doi.org/10.3390/s22051875.
24. Knuplez, A.; Chowdhury, A.; Svecko, R. Modeling and control design for the ball and plate system. In Proceedings of the IEEE International Conference on Industrial Technology, Maribor, Slovenia, 10–12 December 2003; pp. 1064–1067. https://doi.org/10.1109/icit.2003.1290810.
25. KASSEM, A.; HADDAD, H.; ALBITAR, C. Commparison Between Different Methods of Control of Ball and Plate System with 6DOF Stewart Platform. *IFAC-PapersOnLine* **2015**, *48*, 47–52. https://doi.org/10.1016/j.ifacol.2015.09.158.
26. Han, G.X.; Meng, S.J.; Huang, D.D.; Hu, Y.F. Zero Steady-state Error Tracking Control for Ball and Plate System Based on Principle of Internal Model. *Int. J. Control Autom. Syst.* **2023**, *21*, 890–899. https://doi.org/10.1007/s12555-021-0138-5.
27. Pinagapani, A.K.; Mani, G.; K R, C.; Pandian, K. Composite Disturbance Rejection Control for Ball Balancer System. *Procedia Comput. Sci.* **2018**, *133*, 124–133. https://doi.org/10.1016/j.procs.2018.07.016.
28. Singh, R.; Bhushan, B. Real-time control of ball balancer using neural integrated fuzzy controller. *Artif. Intell. Rev.* **2020**, *53*, 351–368. https://doi.org/10.1007/s10462-018-9658-7.
29. Borah, M.; Majhi, L.; Roy, P.; Roy, B. Design of a Fractional Order PD Controller Tuned by Firefly Algorithm for Stability Control of the Nonlinear Ball and Plate System. In Proceedings of the 2014 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), Tamilnadu, India, 8–10 May 2014.
30. Borah, M.; Roy, P.; Roy, B.K. Enhanced Performance in Trajectory Tracking of a Ball and Plate System using Fractional Order Controller. *IETE J. Res.* **2018**, *64*, 76–86. https://doi.org/10.1080/03772063.2017.1343157.
31. Roy, P.; Das, A.; Roy, B.K. Cascaded fractional order sliding mode control for trajectory control of a ball and plate system. *Trans. Inst. Meas. Control* **2018**, *40*, 701–711. https://doi.org/10.1177/0142331216663826.
32. Das, A.; Roy, P. Improved Performance of Cascaded Fractional-Order SMC over Cascaded SMC for Position Control of a Ball and Plate System. *IETE J. Res.* **2017**, *63*, 238–247. https://doi.org/10.1080/03772063.2016.1258336.
33. Siciliano, B.; Sciavicco, L.; Villani, L.; Oriolo, G. Mobile robots. In *Robotics: Modelling, Planning and Control*; Spring: Berlin/Heidelberg, Germany, 2009; pp. 469–521.
34. Lurie, B.J. Three-Parameter Tunable Tilt-Integral-Derivative (TID) Controller. U.S. Patent 5,371,670, 6 December 1994.
35. Innovate educate, Q. Rotary Motion Servo Plant: SRV02 Rotary Experiment #17:2D Ball Balancer 2D Ball Balancer Control Using QUARC. Available online: https://nps.edu/documents/105873337/0/56+-+2D+Ball+Balancer+Control+-+Instructor+Manual.pdf/709c97d2-0fae-426c-9e2a-4b36e8411edf?t=1436282347000 (accessed on 20 January 2024).
36. Esgandanian, A.; Daneshvar, S. A Comparative Study on a Tilt-Integral-Derivative Controller with Proportional-Integral-Derivative Controller for a Pacemaker. *Int. J. Adv. Biotechnol. Res.* **2016**, *7*, 645–650.
37. Sun, W.z.; Wang, J.s.; Wei, X. An Improved Whale Optimization Algorithm Based on Different Searching Paths and Perceptual Disturbance. *Symmetry* **2018**, *10*, 210. https://doi.org/10.3390/sym10060210.
38. Chakraborty, S.; Kumar Saha, A.; Sharma, S.; Mirjalili, S.; Chakraborty, R. A novel enhanced whale optimization algorithm for global optimization. *Comput. Ind. Eng.* **2021**, *153*, 107086. https://doi.org/10.1016/j.cie.2020.107086.
39. Nadimi-Shahraki, M.H.; Zamani, H.; Asghari Varzaneh, Z.; Mirjalili, S. A Systematic Review of the Whale Optimization Algorithm: Theoretical Foundation, Improvements, and Hybridizations. *Arch. Comput. Methods Eng. State Art Rev.* **2023**, *30*, 4113–4159. https://doi.org/10.1007/s11831-023-09928-7.
40. Hou, Y.; Gao, H.; Wang, Z.; Du, C. Improved Grey Wolf Optimization Algorithm and Application. *Sensors* **2022**, *22*, 3810. https://doi.org/10.3390/s22103810.
41. Precup, R.E.; Voisan, E.I.; Petriu, E.M.; Tomescu, M.L.; David, R.C.; Szedlak-Stinean, A.I.; Roman, R.C. Grey Wolf Optimizer-Based Approaches to Path Planning and Fuzzy Logic-based Tracking Control for Mobile Robots. *Int. J. Comput. Commun. Control* **2020**, *15*. https://doi.org/10.15837/ijccc.2020.3.3844.
42. Djerioui, A.; Houari, A.; Machmoum, M.; Ghanes, M. Grey Wolf Optimizer-Based Predictive Torque Control for Electric Buses Applications. *Energies* **2020**, *13*, 5013. https://doi.org/10.3390/en13195013.

43. Sharma, I.; Kumar, V.; Sharma, S. A Comprehensive Survey on Grey Wolf Optimization. *Recent Adv. Comput. Sci. Commun.* **2022**, *15*, 1. https://doi.org/10.2174/2666255813999201007165454.
44. Şen, M.A.; Kalyoncu, M. Grey wolf optimizer based tuning of a hybrid LQR-PID controller for foot trajectory control of a quadruped robot. *Gazi Univ. J. Sci.* **2019**, *32*, 674–684.