Chapter 4

Design of FOPID controller using grey wolf optimizer

4.1 Introduction

Most of numerical optimization technique available in the literature are inspired by hunting and searching strategy of a particular species present in nature. The leadership quality and hierarchical behaviour of grey wolves is still untouched for optimization of FOPID controller parameters. The grey wolves are popular because of their efficient hunting strategy in packs and thus fit in the top of the food chain. This chapter presents an efficient optimization algorithm based on the social behaviour of grey wolves known as grey wolf optimization to optimize the parameters of the FOPID controller for various systems.

4.2 Grey Wolf Optimizer

Grey wolf also known as the timber wolf (Scientific name Canis lupus) is suited to canidae family and supposed to be an apex predator. Most of the times they prefer to live in groups or pack [169]. The average pack size lies between 5 to 12. The grey wolves are too illiberal to their social leading hierarchy as shown in Figure 4.1. The leader may be a male or a female called Alpha (α) and he/she will be the most responsible member to make decision for arranging food i.e. hunting, choosing best place to rest, time to walk etc. Decisions of alpha are imposed to the other group members. However, few democratic conducts are noticed when an alpha respect the other wolves in the group. In any meeting or assemblies, the whole group acknowledges the leader alpha by keeping their tails downward. Alpha is also known as the dominant wolf among the group because his/ her orders are obeyed by each member in the group [170], [171]. The alpha is not essentially the most powerful member but he/she has the best capability to manage or lead the group. Only alpha wolf has permission to mate in the group. This shows the importance of organization and discipline of the pack than its strength.

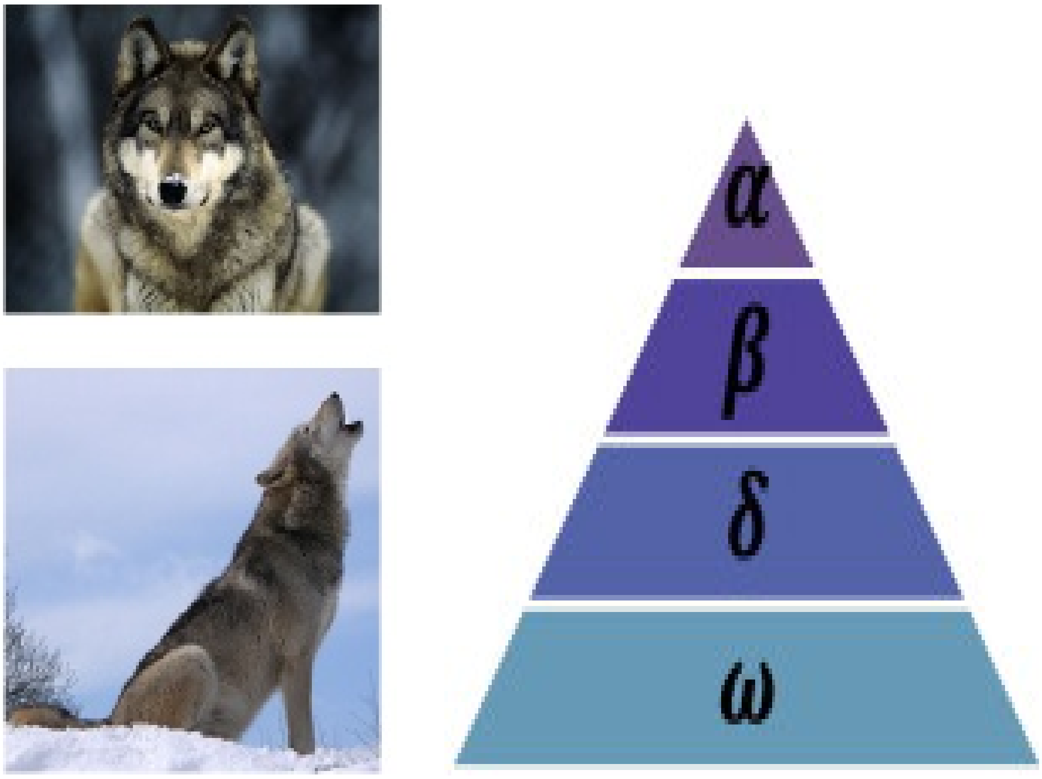

Fig. 4.1. Social leading hierarchy of grey wolves

The next level (i.e. second level) of the hierarchy is called Beta (β). The betas are the foremost helping hands that assist the alpha for making any decision or any other activity of the group. The beta can be either a male or a female wolf and he/she may be the best candidate to be next leader of the group when alpha passes away. The wolf beta respects the leader alpha and also commands to the other members at lower-level. It is the most faithful advisor for the alpha wolf and best commander for rest of the group. It reinforces the commands of the leader alpha to other group members and followed by their response back to the leader.

The third level of the hierarchy is being represented by Delta (δ) which searches for the prey in different direction other than alpha (α) and beta (β). He/she acts as subordinate of beta wolf. The delta (δ) wolf plays a prominent role during the search process as it encourages other wolves to follow the leader involving him. Delta has to respect the orders of alpha and beta and also dominate the wolves at fourth or lowest level of the hierarchy.

The fourth and lowest level of the hierarchy of the wolves is represented as Omega (ω). These wolves are like scapegoat for the pack which always obey the dominant wolves and permit to eat at last. Although the omega wolves does not have much importance for the group but internal fighting and crisis is observed while losing any omega from the pack. In a few cases omega act as babysitters in the group. This hierarchy consists of scouts, hunters, sentinels, caretakers and elders who have their individual responsibilities for the pack. Scouts supervise the boundaries of their territory and inform the pack in dangerous conditions. Hunters are responsible for providing food for the pack and also help the leader in hunting the prey. Sentinels provide safety and guaranteed protection for the pack. Caretakers are like doctors or health care members of the pack, they care for the sick, wounded, frail and injured wolves in the pack. Lastly, the elders share precious experiences of their lives useful for the leaders of the pack.

Besides the social hierarchy of the grey wolves, the hunting strategy of the pack also shows one more attractive social behavior of the grey wolves. As mentioned in [172] the hunting procedure of the grey wolves involves the following three major steps:

- Tracking and approaching the hunt or prey.
- Encircling and harassing the prey until it get tiered or, stops racing for its life.
- Attack or strike to the prey.

These hunting steps are depicted Figure 4.2.

Fig. 4.2. (A) Chasing, approaching, and tracking prey (B–D) Pursuing, harassing, and encircling (E) Stationary situation and attack.

The hierarchy nature, social behavior and incredible hunting strategy of the grey wolves motivate to present a mathematical model to design as optimization algorithm known as Grey Wolf Optimizer.

4.3 Mathematical Modeling of the Algorithm

In this subsection social behavior of the hunting process of grey wolves like social hierarchy, encircling, hunting, attacking prey and searching for a prey is mathematically modeled[169], [171], [173]. Hence, the outline of GWO algorithm is presented as follows:

a) Social hierarchy

While designing the GWO algorithm, to get the efficient mathematical model of the social hierarchy of grey wolves, assume the best solution as the leader Alpha (α). Accordingly, the second and third finest solutions are considered as Beta (β) and Delta (δ) respectively. Remaining members of the solution are considered to be Omega (ω). Similar to the social behavior of the grey wolves, optimization (hunting) in the GWO algorithm is guided by the leader Alpha (α) and his/her subordinates Beta (β) and Delta (δ).

b) Encircling prey

The grey wolves encircle the prey during the hunting action. Encircling behavior of the wolves is presented by the following mathematical model equations:

$$\vec{D} = \left|\vec{C}.\vec{X}_P(t) - \vec{X}(t)\right| \tag{4.1}$$

$$\vec{X}(t+1) = \vec{X}_P(t) - \vec{A}.\vec{D} \tag{4.2}$$

where $\vec{A}$ and $\vec{C}$ are coefficient vectors, $\vec{X}_P$ denotes the position vector of the prey, $\vec{X}$ specify the position vector of the grey wolf, t denote the present iteration.

Coefficient vectors $\vec{A}$ and $\vec{C}$ are calculated using the equations:

$$\vec{A} = 2\vec{a}.r_1 - \vec{a} \tag{4.3}$$

$$\vec{C} = 2.r_2 \tag{4.4}$$

where r_1 and r_2 are randomly chosen variables in the range of [0, 1] and numerical values of $\vec{a}$ are linearly decreased from 2 to 0 during the entire optimization period.

To observe the effectiveness of mathematical Equations 4.1 and 4.2, an example of a two-dimensional position vector and few of the feasible neighbors are demonstrated in Figure 4.3. One can easily observed from the figure that a grey wolf having the position of (X, Y) can modify location according to the location of the prey (X^*, Y^*). Various places around the best wolf can be reached from the present location by simply correcting the values of vectors $\vec{A}$ and $\vec{C}$. For example, the location $(X^* - X, Y^*)$ can be attained by putting $\vec{A} = (1, 0)$ and $\vec{C} = (1, 1)$. The probable updated locations of the wolf in 3D space is shown in Figure 4.4. The random variables r_1 and r_2 permit to attain any position among all the locations shown in Figure 4.3 and Figure 4.4.

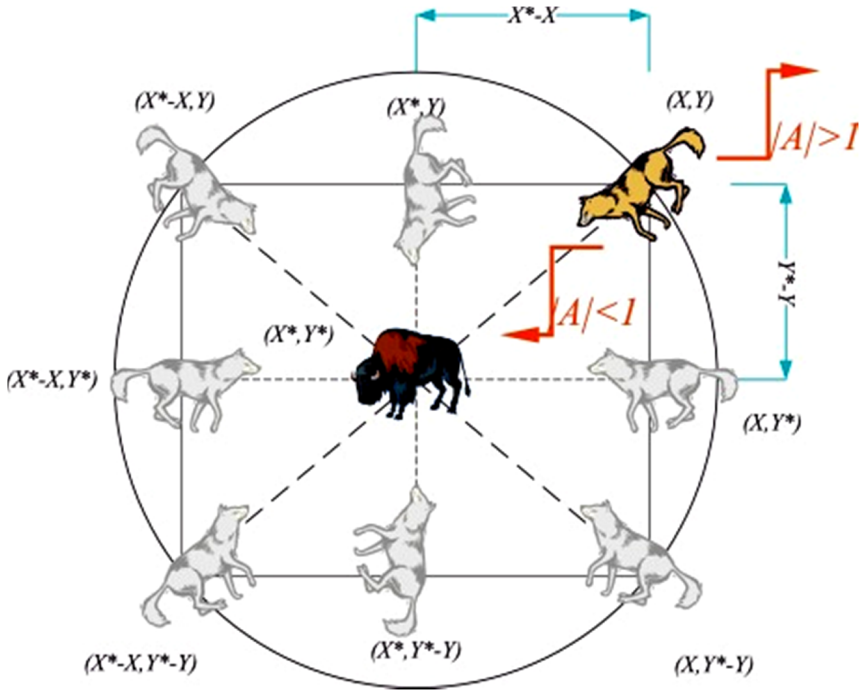

Fig. 4.3. Two-dimensional position vector and few of the feasible neighbors of the wolves

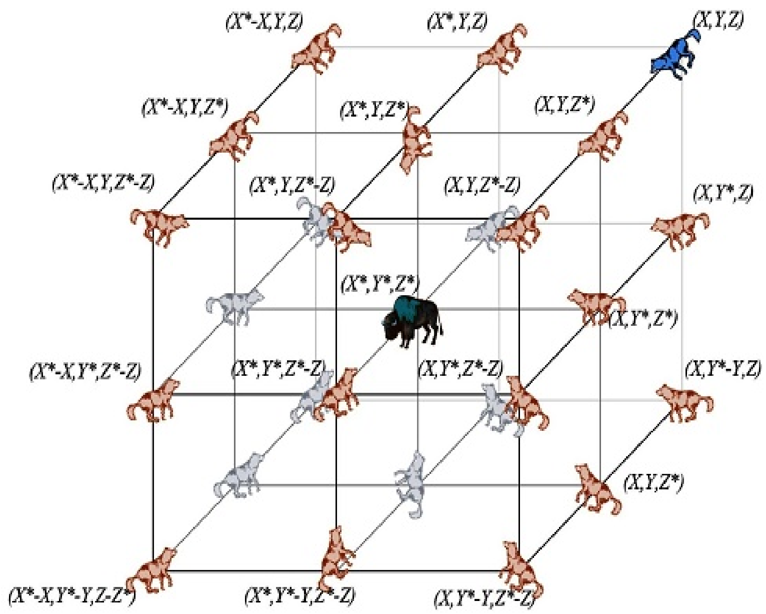

Fig. 4.4. The probable updated locations of the wolf in 3D space

Hence, a grey wolf may upgrade its location inside the search space around the prey in any arbitrary position by using Equation 4.1 and 4.2. The identical perception can be followed in an n-dimensional search space and the wolves will shift in a hyper-cubes (or hyper-spheres) about the fittest solution.

c) Hunting

Grey wolves are capable to identify the location of the food (prey) and encircle them. The hunting of prey is guided by the leader alpha moreover beta and delta might also take part in hunting. However, in a theoretical search space there is no clue about the location of the prey (i.e. the optimum solution). To obtain a suitable mathematical model of the hunting process of the wolves. The best three members of the hierarchy (i.e. alpha, beta and delta) having best information about the location of the prey are assumed. Hence, best three solutions are preserved and rest of the members of lower hierarchy update their locations according to the top search agents.

As α, β and δ are considered as the best results of the algorithm, the positions of the remaining wolves are renewed as:

$$\left.\begin{aligned} \vec{D}_{\alpha_i} &= \left|C_1.\vec{X}_{\alpha_i}(t) - \vec{X}_i(t)\right| \\ \vec{D}_{\beta_i} &= \left|C_2.\vec{X}_{\beta_i}(t) - \vec{X}_i(t)\right| \\ \vec{D}_{\delta_i} &= \left|C_3.\vec{X}_{\delta_i}(t) - \vec{X}_i(t)\right| \end{aligned}\right\} \tag{4.5}$$

$$\left.\begin{aligned} \vec{X}_{i1} &= \vec{X}_{\alpha_i}(t) - A_1.\vec{D}_{\alpha_i} \\ \vec{X}_{i2} &= \vec{X}_{\beta_i}(t) - A_2.\vec{D}_{\beta_i} \\ \vec{X}_{i3} &= \vec{X}_{\delta_i}(t) - A_3.\vec{D}_{\delta_i} \end{aligned}\right\} \tag{4.6}$$

$$\vec{X}_i(t+1) = \frac{\vec{X}_{i1} + \vec{X}_{i2} + \vec{X}_{i3}}{3} \tag{4.7}$$

where i and $\vec{X}_i(t+1)$ denote the number of iterations and the top search agent of i^{th} iteration respectively.

Process of updating the position according to the best search agents (i.e. α, β and δ) in a 2-dimenssional search space is shown in Figure 4.5.

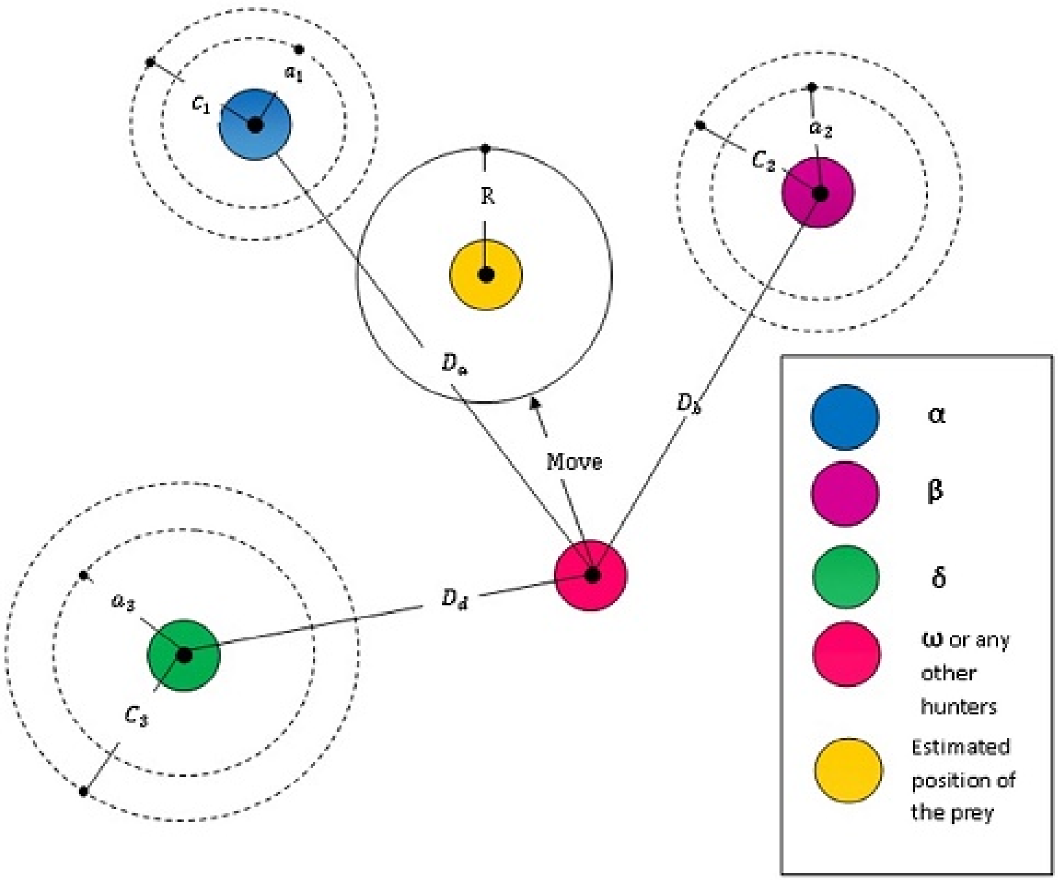

Fig. 4.5. Process of updating the position according to the best search agents

It is noticeable that last position of prey will be an arbitrary place within a circle defined by the location of three best solutions alpha, beta, and delta in the defined search space. So it can be concluded that alpha, beta, and delta not only approximate the location of the prey but also helps to update the location of the other wolves about the prey.

d) Attacking prey (exploitation)

As discussed above the grey wolves terminate the hunting process by attacking the prey when it get tiered to take rest. In order to model the approaching prey mathematically the value of '$\vec{a}$' is reduced. Here, it is noticeable that variation range of $\vec{A}$ also reduced by

'$\vec{a}$'. Hence, $\vec{A}$ has a random value in the period $[-2a, 2a]$ where value of 'a' decreases from 2 to 0 over the entire approximation iteration. If the value of random variable $\vec{A}$ lie between $[-1, 1]$, the position of the wolf in next iteration will be any location between the present location and the location of the prey. Figure 4.6 (a) illustrate that the wolves are compel to attack the prey if $|A| < 1$. In the proposed GWO algorithm the search agents i.e. wolves are allowed to update their position corresponding to the location of the alpha, beta, and delta; and also attack to the prey.

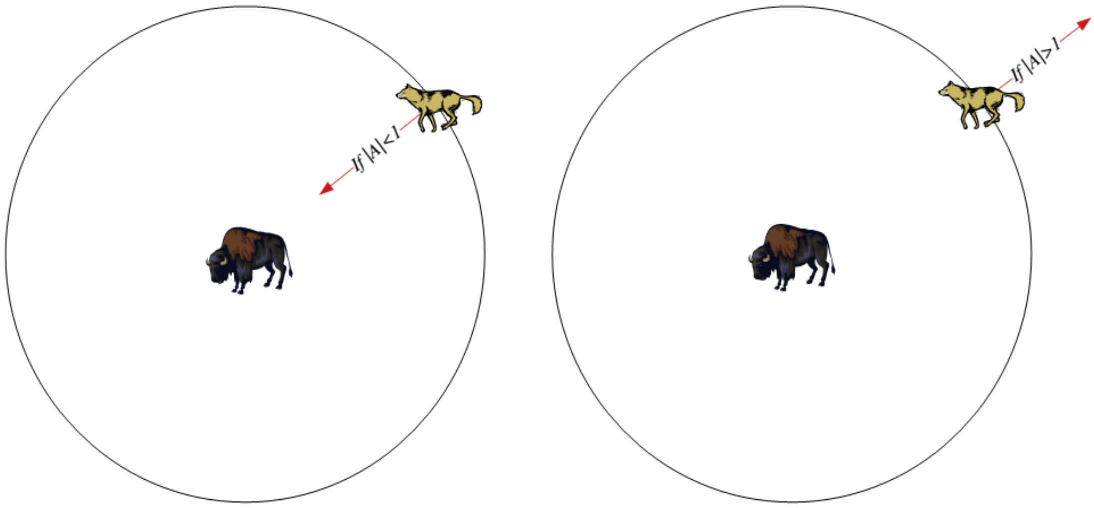

Fig. 4.6. (a) the wolves who are compelled to attach the prey, (b) The wolves who are compelled to separate from the prey to find a better solution

In some cases the GWO algorithm get trapped in it's local solutions. It is due to the fact that the encircling mechanism proposed in the algorithm show a very less emphasis on exploration than it required.

e) Search for prey (exploration)

Every movement of the grey wolves is directly governed by the present position of the best agents alpha, beta, and delta. Accordingly they separate (diverge) from each other for searching prey or come together (converge) to attack the prey.

To present a mathematical model the divergent phenomenon, $\vec{A}$ is used with the random values either less than -1 or greater than 1 to force the wolves to diverge from the food or prey. This stresses the exploration and allows the GWO algorithm for global search. It is shown in Figure 4.6(b) that $|A| > 1$ compel the wolves to separate from the prey to find a better prey or global optimum solution.

Further, the other coefficient vectors $\vec{C}$ also support the exploration. As it can be observed from Equation 4.4, the values of $\vec{C}$ retain a random number between $[0, 2]$.

The vector $\vec{C}$ offers an arbitrary weight to the prey to stochastically increase ($C > 1$) or decrease ($C < 1$) the effect of the position of the prey while calculating the distance in Equation 4.1. This supports the optimization process to demonstrate a random action during the optimization which enhance the exploration and also avoid local optima. It is necessary to mention here that different from A, the values of C does not decrease linearly. Value of C is strictly chosen randomly during all the iteration from starting till end to enhance the exploration. So, the coefficient vector C is very useful to avoid local optima especially during last iteration. In a different concept vector C is considered as an obstacle in path of hunting which prevent the wolves from approaching the prey quickly and easily. Vector C does the same by applying a random weight to the prey position which makes it difficult for the wolf to reach the prey.

During the entire iterations, three best search agents (i.e. alpha, beta, and delta wolves) evaluate the possible position of the prey and each candidate updates their distances from the prey. The value of the variable 'a' is decreased from $2\ to\ 0$ to put stress on exploration and exploitation respectively. Moreover, the candidate solutions are likely to move away (diverge) from the prey if $\vec{A} > 1$ and unite (converse) towards the prey if $\vec{A} < 1$. This process continue till the GWO algorithm is terminated by satisfying the termination criterion. The flowchart of the GWO algorithm is shown in Figure 4.7. The pseudo code for the algorithm is written in Figure 4.8.

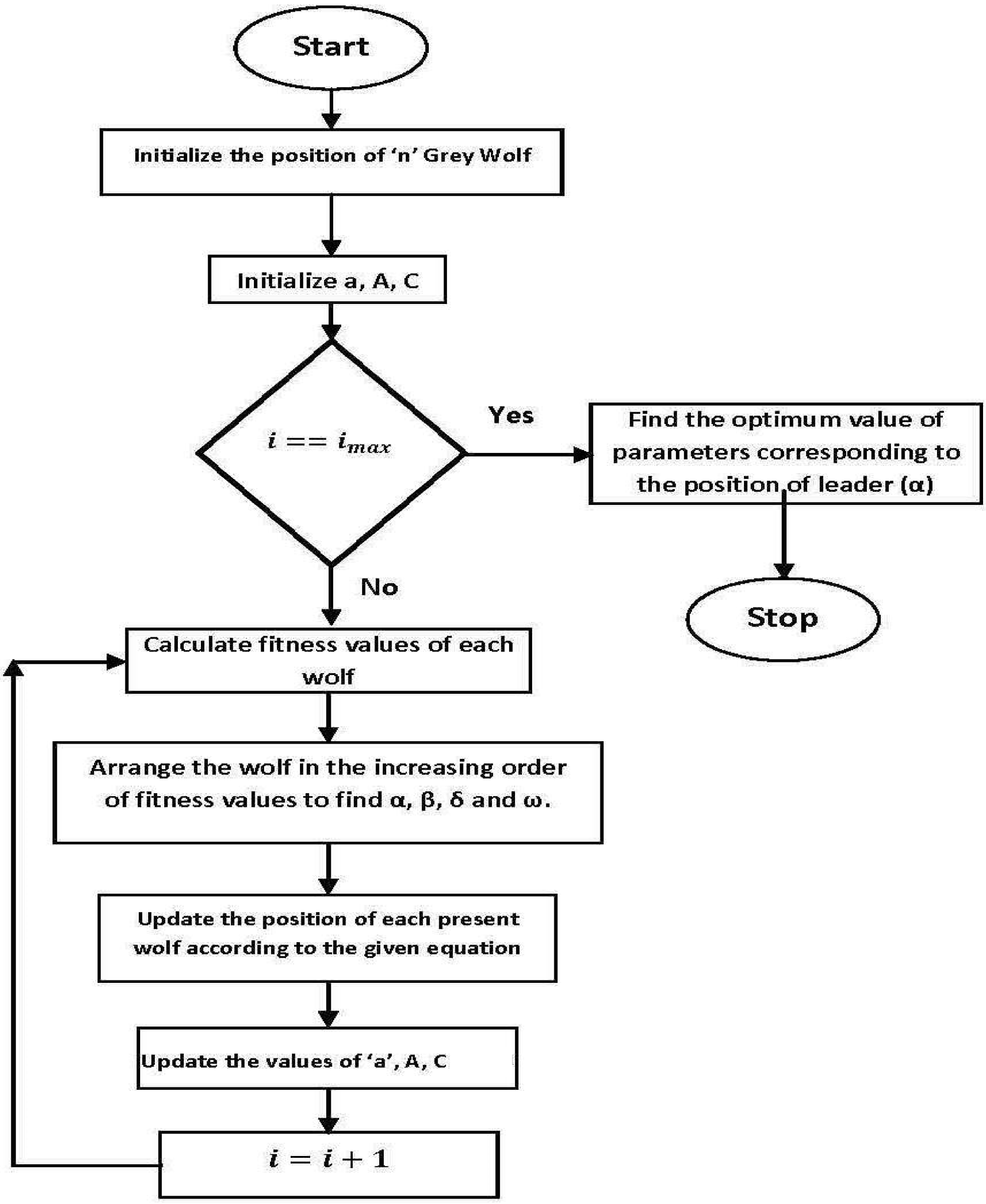

Fig. 4.7. Flowchart of the GWO algorithm

```
Initialize the grey wolf population X_i (i = 1,2 ..... n)
Initialize a, A, C
Calculate the fitness of each search agent
X_α = the best search agent
X_β = the second best search agent
X_δ = the third best search agent
While (t < Max. number of iterations)
    for each search agent
        Update the position of the current search agent by equation (4.2)
    end for
    Update a, A, C
    Calculate the fitness of all the search agents
    Update X_α, X_β, X_δ
    t = t + 1
end while
return X_α
```

Fig. 4.8. Pseudo code of the GWO algorithm

To observe the theoretical operation and ability of GWO algorithm, some points to be considered:

- The social hierarchy presented here, helps the algorithm to preserve the fittest solutions in each iteration.
- The encircling process defines a circular area around the fittest solutions which can be increased to higher dimensions as per the requirement.
- The random coefficient vectors A and C support the agent solutions to have n-spheres with various arbitrary radii.
- The hunting process allows the wolf to establish the possible position of the prey.
- The numerical values of the variable 'a' and 'A' assure the exploration and exploitation process.
- The adaptive values of variable 'a' and 'A' permit the algorithm to easily discriminate among exploration and exploitation.

- Total optimization process in divided into two part with decreasing values of A. the first half of the total iterations are dedicated for exploration ($|A| \geq 1$) while the other half are committed to exploitation ($|A| < 1$).
- There are only parameters to regulate (‘a’ and ‘C’) in the algorithm.

In the present work the GWO-algorithm is implemented for optimizing the FOPID controller parameters for various systems discussed in previous chapter. In the proposed algorithm ISE is used as the fitness function and best result is considered after 100 iterations in every case of optimization.

4.4 Illustrative Examples

4.4.1 Design of an FOPID controller for third order linear plant

Consider the plant in section 3.3.1 it is observed that both controllers (i.e. ZN-PID and NM-FOPID) fails to fulfil all the design requirement of the system. Hence, parameters of FOPID controller are optimized using GWO which gives the optimized value as: $K_P = 99.9464, K_I = 96.6096, K_D = 5.77882, \lambda = 0.12538$ and $\mu = 0.12456$. The step response of the closed-loop system with GWO-FOPID controller is compared with ZN-PID and the system without controller in Figure 4.9. Frequency response of the system with GWO-FOPID controller is shown in Figure 4.10.

Figure 4.9 depict a support to GWO-FOPID controller that fulfils all the design criteria of the system. Performance characteristics of the plant with GWO-FOPID is compared with the conventional ZN-PID controller in Table 4.1 that validates the effectiveness of the proposed algorithm for the plant.

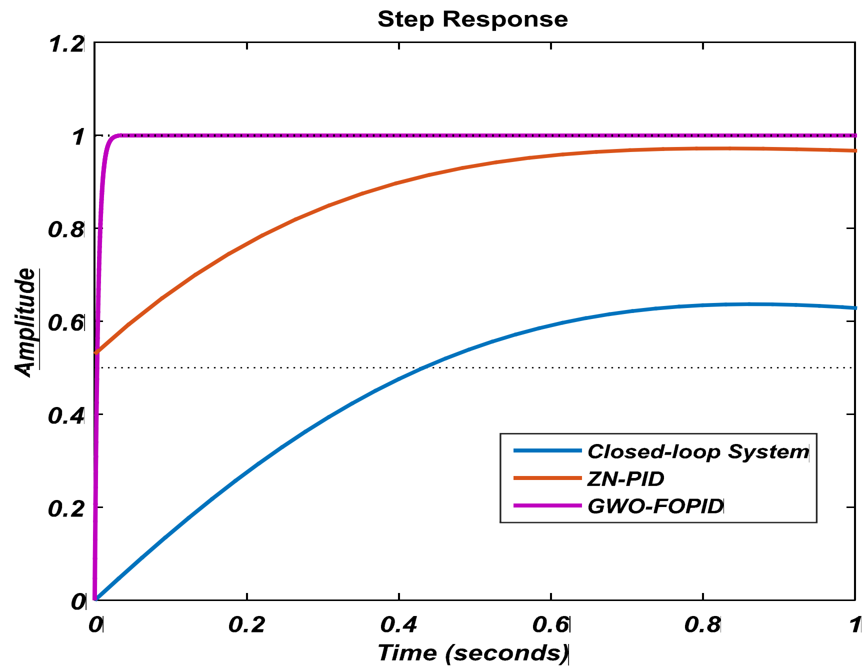

Fig. 4.9. Comparison of step response of the closed-loop system with GWO-FOPID controller with ZN-PID and the system without controller

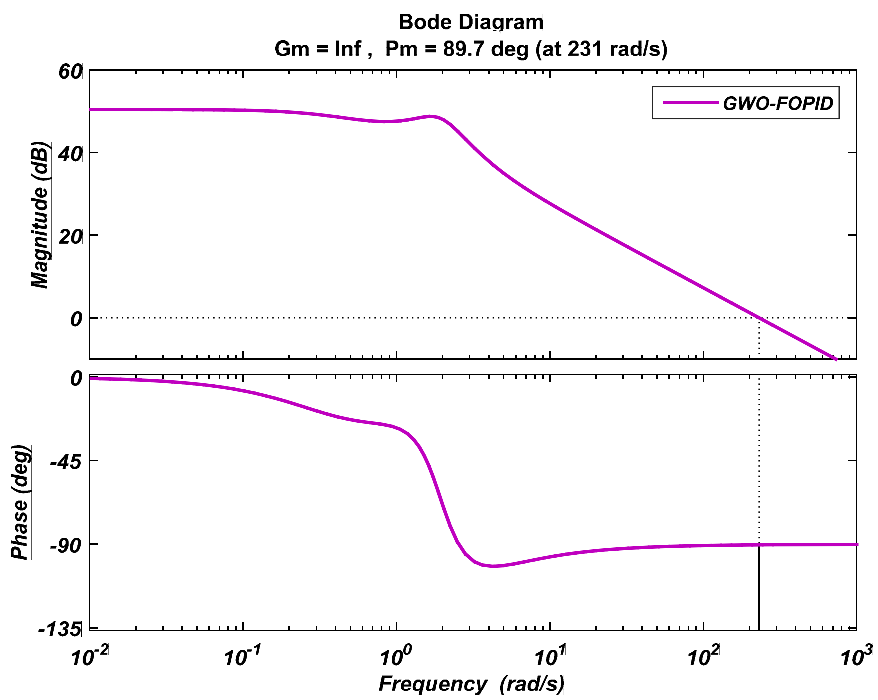

Fig. 4.10. Frequency response of the system with GWO-FOPID controller

Table 4.1 Comparison of performance characteristics of ZN-PID and GWO-FOPID

Controller	Rise-time	Settling-time	Peak Overshoot	Gain Margin	Phase Margin
ZN-PID	0.5491	3.2580	0.9997	∞	∞
GWO-FOPID	0.0100	0.0176	0.6753	∞	89.7

4.4.2 Design of a FOPID controller for Systems with Time Delay

To validate the effectiveness of the GWO-algorithm for time-delayed system FOPID controller is designed for two different types of system considered from section 3.3.2.

4.4.2.1 Second order system with time delay

Let the system in section 3.3.2.1 of the previous chapter is considered and FOPID controller is designed. The parameter of the FOPID controller is optimized using GWO-algorithm which gives $K_P = 5.8208, K_I = 3.3408, K_D = 5.5551, \lambda = 0.75712$ and $\mu = 0.65412$. The GWO-FOPID controller shows faster controller action and better performance than the classical ZN-PID controller as shown in figure 4.11.

Performance characteristics of the GWO-FOPID controller is compared with the classical ZN-PID controller in Table 4.2. It is observe that the GWO-FOPID controller provides lesser settling time and very less peak overshoot than the classical ZN-PID controller.

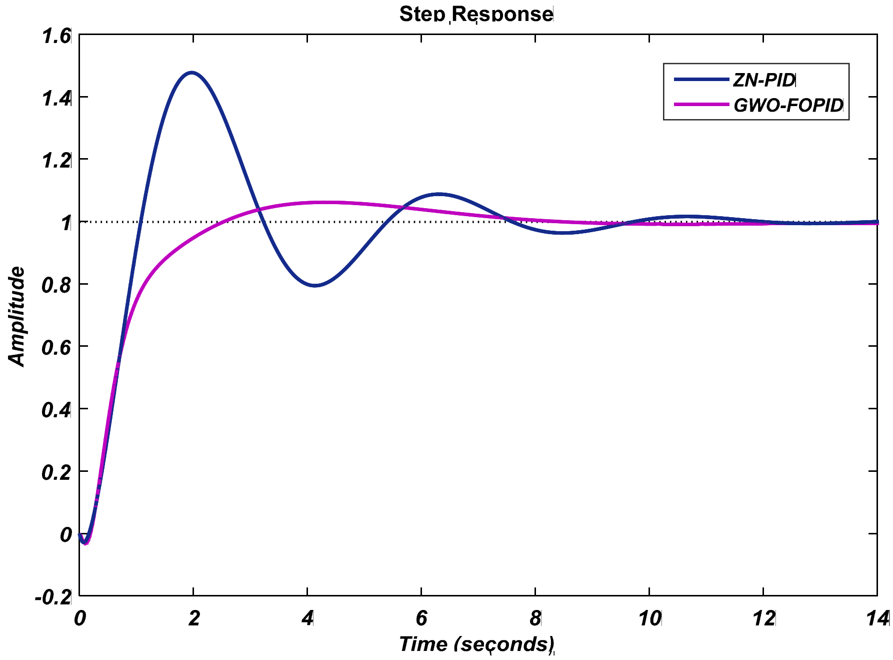

Fig. 4.11. Comparison of step response of the closed-loop time-delayed system with GWO-FOPID and ZN-PID controller

Table 4.2 Comparison of performance characteristics of time-delayed system with ZN-PID and GWO-FOPID controller

Controller	Rise-time	Settling-time	Peak Overshoot
ZN-PID	0.6871	9.2096	47.6680
GWO-FOPID	1.3195	7.0761	6.4650

4.4.2.2 Non-minimum phase system with time delay

The NMP-system under consideration is from section 3.3.2.2. The optimized value of the GWO-FOPID controller are obtained as: $K_P = 0.1005, K_I = 0.8056, K_D = 3.6640, \lambda = 0.0034$ and $\mu = 0.4626$. As shown in Figure 4.12, the GWO-FOPID controller provides faster control action than the classical ZN-PID controller for NMP-system.

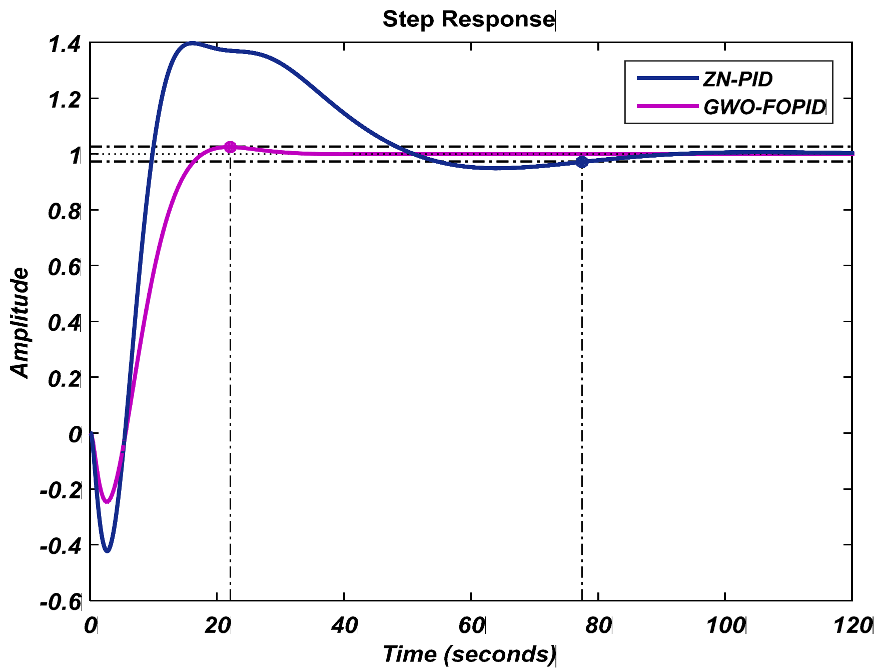

Fig. 4.12. Comparison of step response of the closed-loop NMP-system with ZN-PID and GWO-FOPID controller

Table 4.3 Comparison of performance characteristics of ZN-PID and GWO-FOPID

Controller	Rise-time	Settling-time	Peak Overshoot
ZN-PID	3.3664	77.4635	39.7292
GWO-FOPID	7.9301	22.9301	2.4970

The performance characteristics of NMP-system is compared with the classical ZN-PID controller in Table 4.3 which offer the significant improvement in settling-time and peak overshoot.

4.4.3 Design of an FOPID controller for magnetic levitation system

MLS from section 3.3.3 is considered here. In this section, an FOPID controller is designed by optimizing its parameters using GWO-algorithm. The optimized value of the controller parameters after 100 iterations are obtained as: $K_P = 99.057, K_I = 99.773, K_D = 220.881, \lambda = 0.4345$ and $\mu = 1.4258$. The step response of the MLS with GWO-FOPID is compared with the classical ZN-PID controller and TE-PID controller in Figure 4.13.

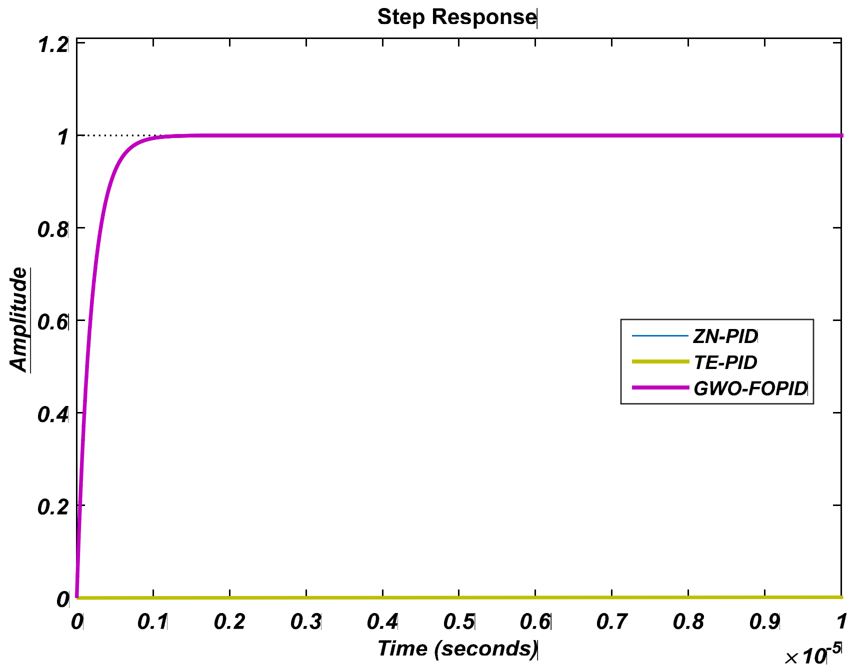

Fig. 4.13. Comparison of step response of the closed-loop MLS with ZN-PID, TE-PID and GWO-FOPID controller

Table 4.4 Comparison of performance characteristics of MLS with ZN-PID, TE-PID and GWO-FOPID controller

Controller	Rise-time	Settling-time	Peak Overshoot
ZN-PID	0.0096	0.503	37.6922
TE-PID	0.0068	0.2637	22.1133
GWO-FOPID	$4.27 * 10^{-7}$	$7.61 * 10^{-7}$	0.0157

The performance characteristics of all the three controllers are compared in Table 4.4 which validates the faster control action and negligible overshoot of the GWO-FOPID controller over the classical PID controllers.

4.4.4 Design of an FOPI controller for a non-monotonic phase system

Non-monotonic phase system from section 3.3.4 is observed. An FOPI controller is designed and its parameters are optimized using GWO-algorithm. The optimized set of parameters of FOPI controller are as: $K_P = 2125, K_I = 25, \lambda = 0.24$. Comparison of closed-loop system with GWO-FOPI controller with classical ZN-PI is shown in Figure 4.14 which shows the faster control action of the proposed controller. The frequency response is compared in Figure 4.15 which present an improved gain of the system.

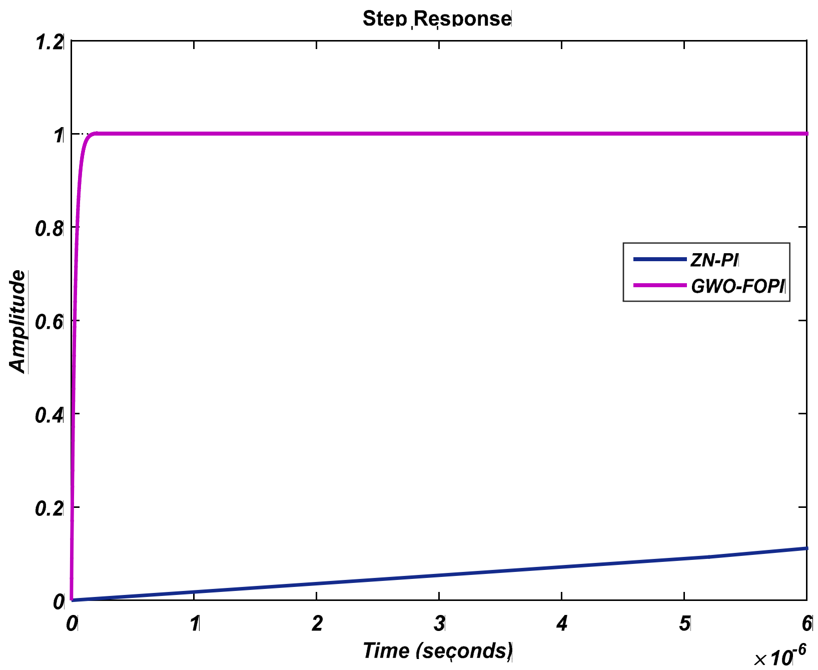

Fig. 4.14. Comparison of step response of the closed-loop DC-buck regulator with ZN-PI and GWO-FOPI controller

The performance characteristics compared in Table 4.5 validates the faster control action and improved phase margin of the DC-buck regulator system with GWO-FOPID controller over the classical PID controllers. Moreover, maximum overshoot of the system is also reduced to a negligible value. Hence, the proposed algorithm present an efficient controller for non-monotonic phase system like DC-buck regulator.

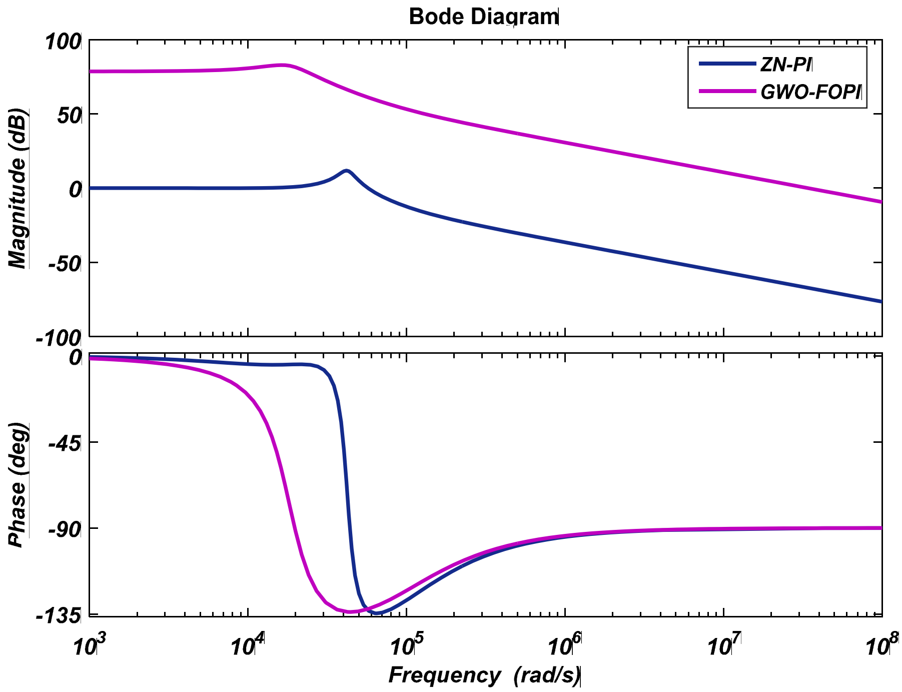

Fig. 4.15. Comparison of frequency response of the DC-buck regulator with ZN-PI and GWO-FOPI controller

Table 4.5 Comparison of performance characteristics of DC-buck regulator using ZN-PI and GWO-FOPI controller

Controller	Rise-time	Settling-time	Peak Overshoot	Gain Margin	Phase Margin
ZN-PID	$2.75 * 10^{-5}$	$7.575 * 10^{-4}$	51.5141	∞	46.9
GWO-FOPID	$6.417 * 10^{-8}$	$1.122 * 10^{-7}$	0.1160	∞	89.9

4.4.5 Design of an FOPID controller for a spherical tank system

Assume the STS from section 3.3.5. The optimized set of FOPID controller parameters using GWO-algorithms are obtained as: $K_P = 999.97, K_I = 99.985, K_D = 49.57, \lambda = 0.10232$ and $\mu = 0.29372$.

The step response of the closed-loop STS with GWO-FOPID is compared with that of the classical ZN-PID controller designed in section 3.7 in Figure 4.16.

The performance characteristics of the STS with GWO-FOPID is compared with classical ZN-PID controller in Table 4.6 which offer a significant improvement in rise-time, settling-time and peak overshoot.

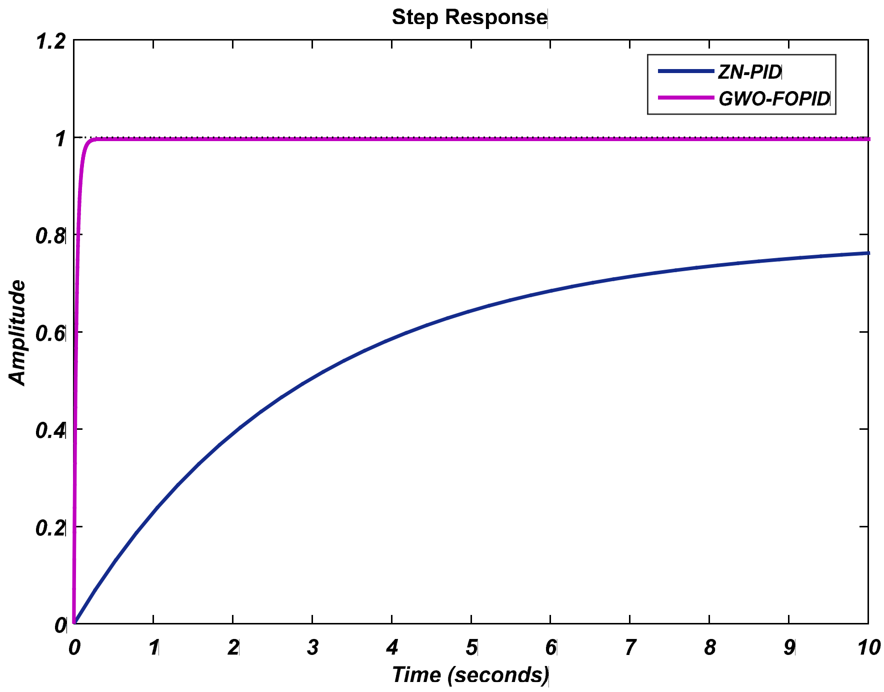

Fig. 4.16. Comparison of step response of the closed-loop STS with ZN-PID and GWO-FOPID controller

Table 4.6 Comparison of performance characteristics of ZN-PID and NM-FOPID

Controller	Rise-time	Settling-time	Peak Overshoot
ZN-PID	84.644	193.0292	1.8564
GWO-FOPID	0.0773	0.1481	0

4.4.6 Design of an FOPID controller for AVR system

In this section, concept of meta-heuristic algorithm GWO algorithm is used to find the fittest value of the FOPID controller parameters for AVR system considered from section 3.3.6. The starting position of the wolves (i.e. initial values of controller parameters) are calculated using ZN method. A nonlinear fitness function suggested in [145] is used for obtaining optimum value of the controller parameters for AVR system. The new fitness function includes time domain as well as frequency domain characteristics of the system and has total eight terms [145]. The importance of each term is defined by a weight factor (β). The fitness function is defined as:

$$W(\tilde{Z}) = \left(1 - e^{\beta}\right)(M_P + E_{ss}) + e^{-\beta}(t_s - t_r) \tag{4.8}$$

where $\tilde{Z} = (K_{P,}K_{I,}K_D)$ is a parameter of PID controller, β is a weighting factor, M_P, E_{ss}, t_r and t_s represent the peak overshoot, steady-state error, rise time and settling time. The position vector of each wolf is defined by five parameters i.e $K_{P,}K_{I,}K_D, \lambda$ and μ. Hence, $\tilde{Z} = (K_{P,}K_{I,}K_D, \lambda\,, \mu)$ is taken for designing an FOPID controller. Since a group of twelve wolves is considered for this work, the whole group is represented by a matrix with dimension 12×5. For optimization of parameters of FOPID controller two different values of $\beta = 1$ and 1.5 is considered as in [145]. The optimized values of the parameters of the proposed FOPID controller for different values of β is shown in Table 4.7.

Table 4.7 The optimized values of the parameters of GWO-FOPID controller for different values of β.

Beta(β)	K_P	K_I	K_D	λ	μ
1	11.7509	3.6876	0.7380	0.9960	1.6428
1.5	5.8679	0.9682	0.3673	1.0683	1.6150

The step response of the closed-loop AVR system with GWO-FOPID controller for both values of β is compared with that of ZN-PID controller in Figure 4.17. Here GWO-FOPID controller present faster control action with negligible maximum overshoot.

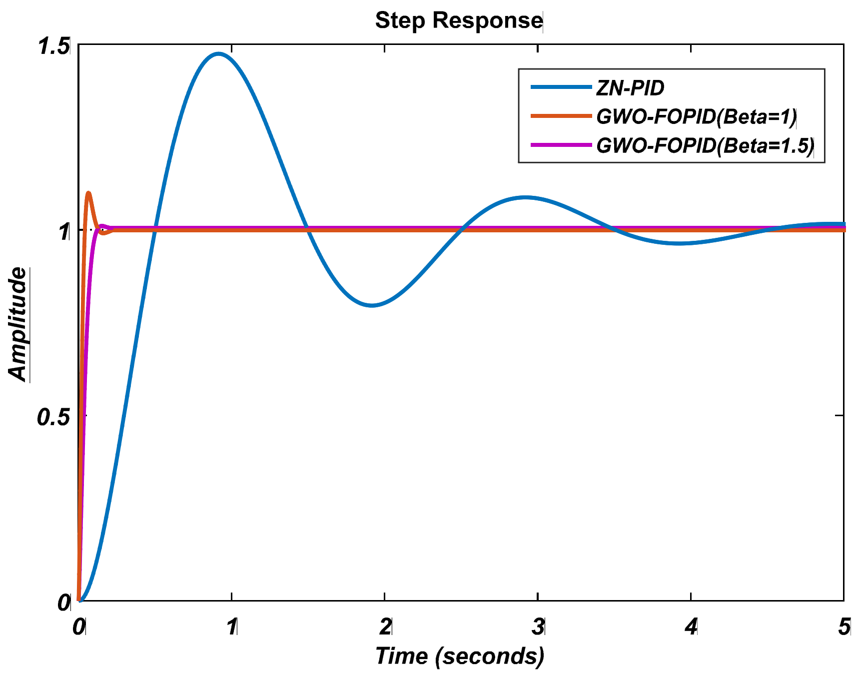

Fig. 4.17. Comparison of step response of the closed-loop AVR system with GWO-FOPID controller for both values of β and ZN-PID controller

The performance characteristics of the AVR system with GWO-FOPID controller is compared with the classical ZN-PID controller in Table 4.8. Here it is clear that the GWO-FOPID controller not only enhance the time domain characteristics of the AVR system but it also improves the gain margin and phase margin.

Table 4.8 Comparison of performance characteristics of AVR system with ZN-PID and GWO-FOPID controller for both values of β

Controller	Rise-time	Settling-time	Peak Overshoot	Gain Margin	Phase Margin
without controller	0.2607	6.9865	65.7226	4.62	16.1
ZN-PID	0.3435	4.2656	47.3931	30.3	27.4
GWO-FOPID (β=1)	0.0292	0.1107	9.8347	25.2	60.2
GWO-FOPID (β=1.5)	0.0714	0.1062	0.9825	31.1	70.6

4.4.7 Robustness analysis of GWO-FOPID controller for AVR system:

In the present work robustness of the proposed GWO-FOPID controller is validated by considering three different types of parameter uncertainties in the AVR system.

4.4.7.1 Uncertainty in amplifier:

Consider the parameters of the amplifier change from original value $K_A = 10, \tau_A = 0.1$ to $K_A = 14, \tau_A = 0.007$.The step response of the terminal voltage of the AVR system with GWO-FOPID for both values of β is shown in Figure 4.18. It is clear that the proposed algorithm provide a robust controller for uncertainty in amplifier parameters.

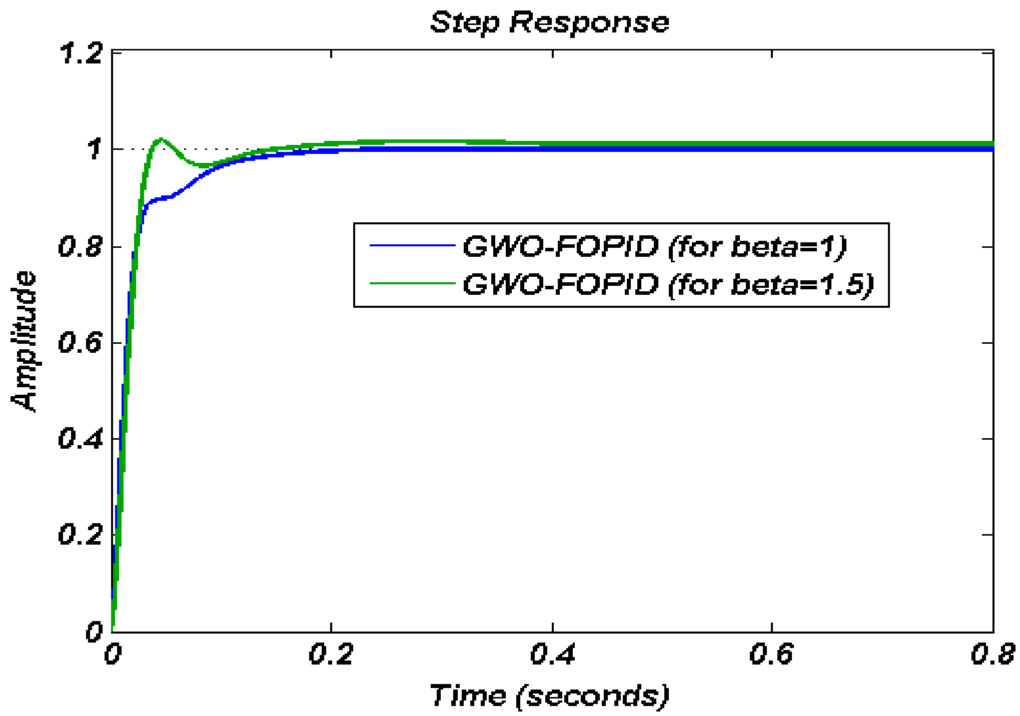

Fig. 4.18. Comparison of step response of AVR system with uncertainty in amplifier

4.4.7.2 Uncertainty in exciter:

Parameter change of the exciter from original value $K_E = 1, \tau_E = 0.4$ to $K_E = 1.2, \tau_E = 0.5$ is considered here. The step response of output voltage of the AVR system with GWO-FOPID shown in Figure 4.19 proves the robust nature of the proposed controller with possible exciter uncertainty.

4.4.7.3 Uncertainty in generator :

Changes in generator parameter is considered from $K_G = 1, \tau_G = 1$ to $K_G = 0.7, \tau_G = 1.6$. The step response of output voltage of the AVR system with GWO- FOPID for both the values of β is shown in Figure 4.20. The response present the robust behavior of the proposed controller with possible uncertainty in generator parameters.

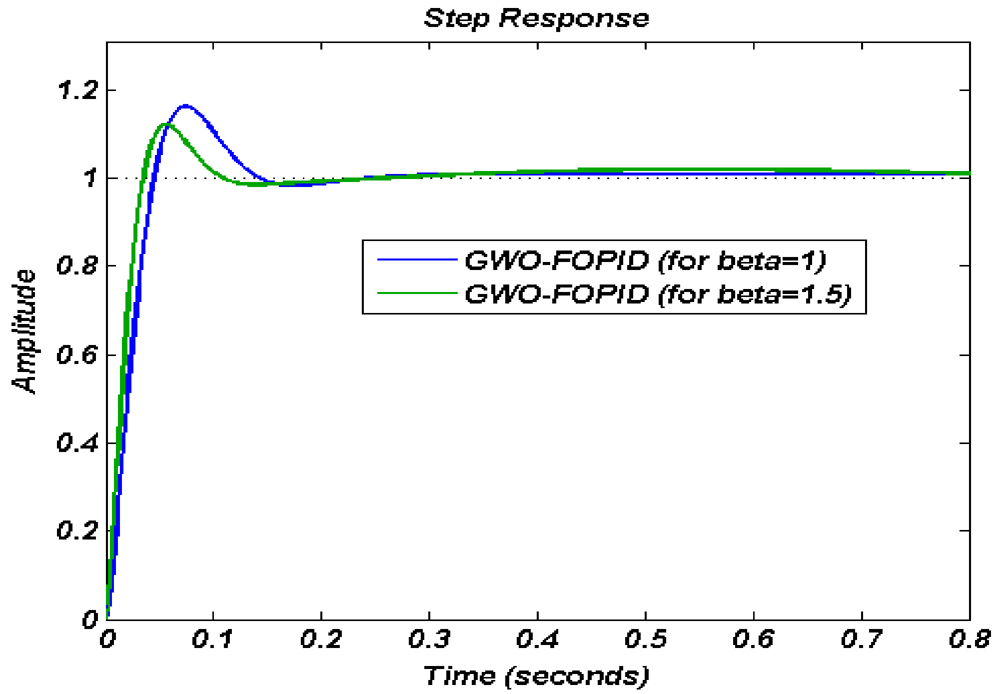

Fig. 4.19. Comparison of step response of AVR system with uncertainty in exciter

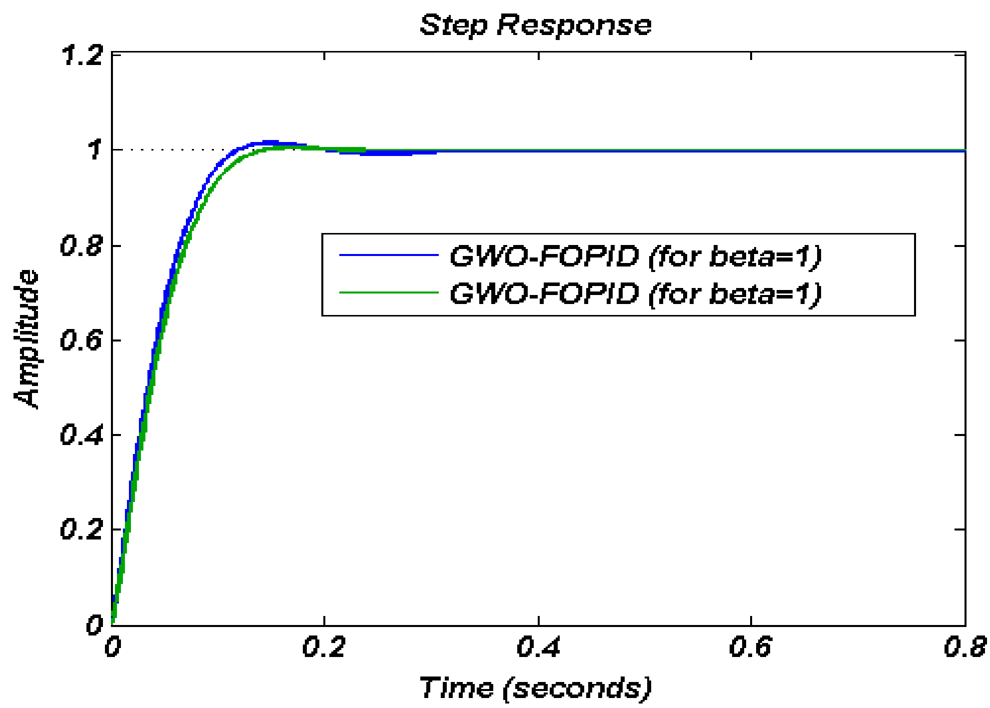

Fig. 4.20. Comparison of Step response of AVR system with uncertainty in generator

The above analysis proves the novelty of the proposed FOPID controller for AVR system. Additionally, the GWO-FOPID controller also presents minimum rise-time, settling-time and peak overshoot of the system as shown in Table 4.7.

4.5 Summary

This chapter elaborated a novel technique for optimization of FOPID controller parameters. Performance of the algorithm is verified by designing the FOPID controller for different types of system and effectiveness of the algorithm is validated by comparing the simulation results with classical ZN-PID controller.

Moreover, the performance of the proposed algorithm is also tested for AVR system having uncertainty in amplifier, exciter and generator. A negligible change in step response of the system is considered with different uncertainties which shows the robust performance of the GWO-FOPID controller. The performance of the algorithm fails to design an efficient FOPID controller for some systems such as time-delayed system where larger rise-time is achieved. Hence, some possible amendment may be performed to obtain more refine and best global solution. A modified version of the algorithm is presented in the next chapter.