Application of Swarm Intelligence Computation Techniques in PID Controller Tuning: A Review

Soumya Ghosal[1], Rajkumar Darbar[2], Biswarup Neogi[3], Achintya Das[4], and Dewaki N. Tibarewala[5]

[1] IT Dept., RCC Institute of Information Technology, Kolkata, India
soumyaghosal.2008@gmail.com
[2] IT Dept., Durgapur Institute of Advanced Tech and Management, Durgapur, India
rajdarbar.r@gmail.com
[3] ECE Dept., Durgapur Institute of Advanced Tech and Management, Durgapur, India
biswarupneogi@gmail.com
[4] ECE Dept., Kalyani Government Engineering College, India
achintya_das123@yahoo.co.in
[5] School of BioScience & Engineering, Jadavpur University, India
biomed_ju@yahoo.com

Abstract. Swarm Intelligence Computation technique is one of the recent and advanced research topic in the field of Artificial Intelligence. This nature – inspired, global optimization technique is used rapidly in various fields , specially it has become one of the most useful method for efficiency improvement of control and distributed optimization aspects. A review study on tuning of PID controller with effective and satisfactory performance analysis via different swarm intelligence computation techniques is presented in this paper. Tuning of PID via traditional methods and genetic algorithm and their limitations in proper tuning, different structure of PID controllers with the objectives for PID tuning and an efficient intelligent PID controller design is presented in the beginning of this paper. Then a brief literature review on PID tuning with different Swarm Intelligence(SI) techniques i.e. Ant Colony Optimization(ACO), Particle Swarm Optimization(PSO), and Bacterial Foraging Optimization Algorithm(BFOA) as well as their advantages and disadvantages in proper tuning is presented in the afterwards . And finally a performance comparison with simulation results of PID tuning via ZN, GA, PSO, BFOA are experimented on four set of system transfer functions and are studied for effective analysis.

Keywords: Swarm Intelligence Computation, Particle Swarm Optimization(PSO), Ant Colony Optimization(ACO), Bacterial Foraging Optimization Algorithm(BFOA), PID controller tuning.

1 Introduction

PID control is a generic feedback control technology and it is vastly used in automatic controllers in industrial control systems. The PID control was first introduced in 1939 in the market and has been successfully used as controller in process control until today. The basic function of the controller is to execute an algorithm based on the control engineers input and hence to maintain the output at a level so that there is no difference between the

S.C. Satapathy et al. (Eds.): Proceedings of the InConINDIA 2012, AISC 132, pp. 195–208.
springerlink.com

process variable and the setpoint[1]. The term 'PID' is an acronym for "proportional, integral, and derivative." A PID controller is a controller that includes elements with those three functions. The popularity of such kind of controller is due to their functional simplicity and reliability. They provide robust and reliable performance for most systems and the PID parameters are tuned to ensure a satisfactory closed loop performance [2]. A PID controller improves the transient response of a system by reducing the overshoot, and by shortening the settling time of a system [3]. For this control loop to function properly, the PID loop must be properly tuned. Ziegler-Nichols[4], Cohen-Coons[5], Astrom and Hagglund[6] tuning methods were some of the primitive tuning methods for PID control. But, due to having greater phase lag, greater overshoot and insufficient tuning results a linear empirical formula had to be introduced. Genetic Algorithm(GA) was an effective solution for these problems. GA was one of the evolutionary computation tuning approaches that could produce better results in PID tuning than the primitive tuning methods[7] having stochastic global searching characteristics that could mimic the process of natural evolution. But later, a set of new intelligent approaches unitedly called swarm intelligence tuning, such as Ant Colony Optimization, Particle swarm optimization, Bacterial Foraging optimization algorithm were introduced which could produce an effective characteristics of positive feedback, search mechanism, distributed computation and constructive greedy heuristic for simpler, efficient and faster tuning than primitive and other evolutionary tuning approaches for having improved characteristics like dynamic adjustment inertia weight factors, etc. Positive feedback search produces advantageous results, distributive computation can be used to avoid pre-matured convergence and greedy heuristic is helpful to find the solution of early stages of search process. The brief survey on PID tuning via different SI techniques is presented later in this paper.

2 The Different Structure of PID Controller and Intelligent PID Controller Design

The practical difficulty with PID control technology is a lack of industrial standards, which has resulted in 46 different architectures of PID controller. For example, the classical architecture of PID controllers is given below.

$$G_c(s) = K_c(1 + \frac{1}{T_i s}) \frac{1 + sT_d}{1 + \frac{sT_d}{N}}$$

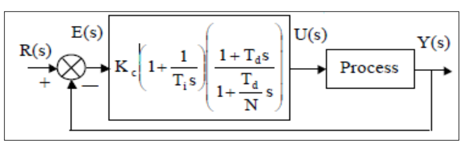


Fig. 1. Classical PID controller in a unity feedback block diagram representation

PID controllers are used in more than 95% of closed-loop industrial processes. Mostly the interest lies in four major characteristics of the closed-loop step response. They are rise time, overshoot, settling time, steady-state error. How the increment of the PID parameters (Kp,Ki,Kd) value can affect the system dynamics is presented in a tabular format in the following way:

Table 1. Performance requirements and objective of PID tuning

Response	Rise Time	Overshoot	Settling Time	S-S Error
K_p	Decrease	Increase	NT	Decrease
K_I	Decrease	Increase	Increase	Eliminate
K_D	NT	Decrease	Decrease	NT

NT: No definite trend. Minor change.

In the next figure, the block diagram of an intelligent PID Controller is given.

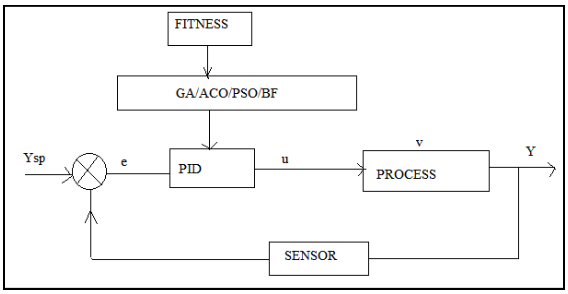


Fig. 2. Block Diagram of an intelligent PID controller[8]

In PID controller, the proportional value determines the reaction of the current error, the integral value determines the reaction based on the sum of recent errors, and derivative value determines the reaction based on the rate at which the error has been changing the weighted sum of these three actions is used to adjust the process via the final control element.

3 Survey on Swarm Intelligence Computation

Swarm intelligence is an algorithm or a device, which is designed for solving distributed problems. It was illumined by the social behavior of gregarious insects and other animals [9]. Compared with the grads algorithm and traditional evolutionary computations, swarm intelligence has following advantageous characteristics: (a) The cooperating particle of the swarm is distributed; (b) There is no control and data of the center and the system is more robust; (c) It can realize cooperation with indirect communication instead of direct communication and the system is more easily to extend; (d) The ability of particle in the population is simple, the operating time of every particle is also very short and it is easy to be realized. Different swarm intelligence algorithms and their uses in PID tuning are discussed below.

4 Ant Colony Optimization(ACO) in PID Tuning

ACO's are very much suited for finding solutions to different optimization problems. A colony of artificial ants cooperates to find good solutions, which are an emergent property of the ant's cooperative interaction .Based on their similarities with ant colonies in nature, ant algorithms are adaptive and robust and can be applied to different versions of the same problem as well as to different optimization problems. The main idea behind ant colony optimization is that when the ants search for food, they initially explore the area surrounding their nest randomly. When one finds a food source, it evaluates it, take some food and goes back to the nest. As they move back, they deposit on the ground a chemical substance called pheromone, which is detectable by other ants. The amount of pheromone that is deposited varies depending on the quantity and quality of the food, and leads other ants to that food source. By the use of this property, the ants can find the shortest path between their nest and the source [11]. A basic algorithm for ACO is given in figure 3.

ACO can be used in PID tuning in the following steps-i) Initialize no. of ants, ii) Then run the process model, iii) Evaluate fitness function, iv) Then update the probability, v) After that calculate the optimum of Kp,Ki,Kd values and finally after reaching the maximum iteration the process stops. ACO was successfully used for PID controller tuning from time as it was able to calculate the optimum value of PID parameters in a very effective manner. A literature review on PID tuning via ACO is given below.

```
input: An instance P of a CO problem model 𝒫 = (S, f, Ω).
InitializePheromoneValues(𝒯)
s_bs ← NULL
while termination conditions not met do
   𝔖_iter ← ∅
   for j = 1, . . . , n_a do
      s ← ConstructSolution(𝒯)
      if s is a valid solution then
         s ← LocalSearch(s)          {optional}
         if (f(s) < f(s_bs)) or (s_bs = NULL) then s_bs ← s
         𝔖_iter ← 𝔖_iter ∪ {s}
      end if
   end for
   ApplyPheromoneUpdate(𝒯, 𝔖_iter, s_bs)
end while
output: The best-so-far solution s_bs
```

Fig. 3. Algorithm of ACO[10]

A research work based on Ant Colony Search-PID tuning is briefly discussed in this paper[51]. In this paper the generation of nodes and path step, the PID controller parameters Kp, Ki, and Kd as the optimized variables, and assumed that each of them has five valid digits, one digit before decimal point and four digits after decimal point and the values of Kp, Ki, and Kd were put on plane O-XY. Here, each decimal digit represents node and connection between them represents the moving path of ants. In transition step, an ant selects the following transition rule :

$$j=\arg\max_{u\in J^{k}_{i}}\left\{\left|\tau(x_{i,}y_{iu})\right|.\left|\eta(x_{i,}y_{iu})\right|^{\beta}\right\},\ \text{if } q\le q_{0}$$

$$J=J,\ \text{if } q>q_{0} \tag{1}$$

In formula (1), τ was considered as the pheromone concentration and $\eta(x_i, y_{ij})$ was the visibility of node (x_i,y_{ij}) and it was proposed as:

$$\eta(x_{i,}y_{ij})=\frac{10-\left|y_{ij}-y_{ij}^{*}\right|}{10} \tag{2}$$

When all of the ants in the colony complete their tours once in the ACS-PID algorithm, the pheromone concentration of each node belonging to the best tour, since the beginning of the trial is updated by the following formulas:

$$\tau(x_{i,}y_{ij}) \leftarrow (1\text{-}\rho).\tau(x_{i,}y_{ij}) + \rho.\Delta\tau(x_{i,}y_{ij})$$
$$\Delta\tau(x_{i,}y_{ij}) = \frac{Q}{ITAE^{*}} \tag{3}$$

Here, ρ is the parameter which governs the pheromone decay. The local update is performed as follows:

$$\tau_{ij}(x_{i,}y_{ij}) \leftarrow (1\text{-}\rho).\tau_{ij}(x_{i,}y_{ij}) + \rho.\tau_0 \tag{4}$$

PID tuning using this adaptive ACS method produces better results than GA-PID, SA-PID, DE-PID. Another approach taken on PID controller tuning via ACO was based on creating incrementally the construction of solutions based on a probabilistic choice of the solution components [12]. In this paper ACO was used for continuous domains, and it was accomplished by the use of a PDF. A PDF could be represented as any function P(x) > 0, such that x that could meet the requirement,

$$\int_{-\infty}^{\infty} P(x)dx = 1 \tag{5}$$

Based on the decision variables Xi, i = 1, 2, . . . , n, each ant constructed a solution performing n steps. At an iteration i, an ant was set to choose a value for the variable Xi. After this, a Gaussian kernel was be created for this iteration. This Gaussian kernel was denoted by this equation:

$$G^{i}(x) = \sum_{l=1}^{k} \omega_l g_l^{i}(x) = \sum_{l=1}^{k} \frac{1}{\sigma_l^{i}\sqrt{2\Pi}} e^{-\frac{(x-\mu_l^{1})^{L}}{2\sigma_l^{2i}}} \quad , i=1,2,\ldots\ldots n. \tag{6}$$

where μ was the mean, ω was the weight, and σ was the standard deviation.

The second step of the algorithm was the pheromone update. All the pheromone information was stored in a solution archive T. For each solution to a problem of n dimensions, the algorithm stored in T the values of its n variables, besides the value of the objective function f(s). All the solutions on this archive was evaluated and then ranked. By this rank they could be sorted. The third and last step was just to update the best solution found, in order that it could be shown when the stop conditions met. A modified ant colony optimization incorporating differential evolution was used in this work. In this design the mutation operation was generated by this function:

$$X_i(t+1) = X_{best}(t) + MF.[X_{i_2}(t) - X_{i_3}(t)] \tag{7}$$

In the above equations, t is the time (generation), MF > 0 is a real parameter, called mutation factor, which controlled the amplification of the difference between two individuals so as to avoid search stagnation. The mutation operation selected the best vector Xbest(t). Then, two individuals were randomly selected and the difference vector was calculated. From the result, it was obtained that with the same preset maximum number of generations, Modified ACO obtained better mean F and minimum F than GA, ES, and ACO methods. Thus, ACO and Modified ACO proved superior in PID tuning than ES and GA. In another work[13] lyapunov function was proposed for accomplishing robust global convergence for tracking error. Parameter tuning of PID was used by Grid-based searching adopted ACO and self adaptive control strategy was also adopted for pheromone decay. Another PID tuning procedure via ACO was furnished by position tracing and elitist strategy was adopted via improved ACO proposed in work[14]. Important ACO optimization, convergence and application strategies are surveyed and published in this journal paper[15].

A research on satisfying the real time control and obtaining better performance, application of ACA (Ant Colony Algorithm) to optimize the parameters of NN-PID controller to improve the on-line self-tuning capability of this controller is presented in this paper[16]. Tuning of Fuzzy PID controller via ACO is also another research work in PID tuning via SI[17]. This paper proposes a proper framework of PID tuning via proper implementation and simulation approach via ACO. One of the nature based algorithm named artificial bee colony similar to ACO was used in PID tuning[18], which was presented by Karaboga in 2005 to optimize numeric benchmark functions [19]. Trajectory tracking comparison for GA and ACO for PID tuning is presented in this paper[20].

ACO is very much efficient in discrete control optimization. It can produce results where the source and destination is known and predetermined. For, crisp result ACO is productive, though it produces some disadvantages also. Its theoretical analysis is difficult for ACO as well as the research on it is experimental rather than theoretical. Also, the time of convergence is uncertain for ACO. PSO is a simpler version of optimization technique for users and also a developed version of this simulation based approach produces better performances in dynamic optimization and constraint handling. In case of problems that are fuzzy in nature usage of PSO is beneficial rather than ACO.

5 Particle Swarm Optimization and Bacterial Foraging Optimization Algorithm in PID Tuning

Particle Swarm Optimization (PSO) is one of the optimization and kind of evolutionary computation technique .The technique is derived from research on swarm such as bird flocking and fish schooling. In the PSO algorithm, instead of using evolutionary operators such as mutation and crossover to manipulate algorithms, a flock of particles are put into the d-dimensional search space with randomly chosen velocities and positions knowing their best values. The velocity and position of each particle, adjusted accordingly to its own flying experience and the other particles flying experience [9]. It was first introduced by Eberhart and Kennedy in 1995[21]. The description of PSO algorithm is given below [22] where the equations are given as (8) and (9) for iteration via PSO.

$$v_{ij}^{(k+1)} = w \times v_{ij}^{(k)} + c_1 \times rand() \times (pbest_{ij} - x_{ij}^{(k)}) + c_2 \times rand() \times (gbest_j - x_{ij}^{(k)}) \tag{8}$$

$$x_{ij}^{(k+1)} = x_{ij}^{(k)} + v_{ij}^{(k+1)} \tag{9}$$

```
Begin
    t→0                //iteration number//
    Initialize X(t)    //X(t): Swarm for iteration t//
    Evaluate f(X(t))   //f(.): fitness function//
While (not termination condition) do
        Begin
            t→t+1
            //Process of PSO//
            Update velocity v(t) and position of each particle x(t) based on (1) and (2)
                if v(t)>vmax
                  v(t)=vmax
                end
                if v(t)<-vmax
                v(t)=-vmax
                end
             //end of the process of PSO//
              Reproduce a new X(t)
              Evaluate f(X(t))
        End
End
```

Fig. 4. Algorithm of PSO

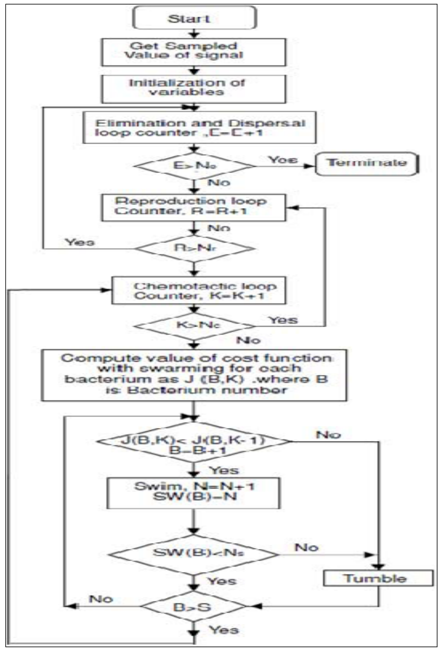


Fig. 5. Flowchart of BFOA

From last decade, Particle Swarm Optimization has been successfully applied in the field of control, design, telecommunication and combinatorial optimization procedures. Around three hundred PSO algorithms are exploited till date. A noble review was done on PSO by De Falco et al[23]. Although PSO has been used mainly to solve unconstrained, single-objective optimization problems, PSO algorithms have been developed to solve constrained problems, multi-objective optimization problems, problems with dynamically changing landscapes, and to find multiple solutions. The application of PSO in nonlinear control domain is also huge. A literature review on its applications in PID controller tuning is discussed below. A research work on designing a robust H2/H∞ PID controller design via PSO in shown in[21]. The controller was designed such that the nominal closed loop system was asymptotically stable and robust stability satisfied the required inequality equation. And the disturbance attenuation performance satisfied the following inequality.

$$J_b = \left\| W_2(s).S(s) \right\|_\infty < 1 \qquad (10)$$

A balanced performance criterion to minimize both *Ja* and *Jb* simultaneously is to minimize *J*∞, The minimization of tracking error J2 by taking integration of error e(t) and by taking inverse laplace transformation ,the robust hybrid controller design was established. Three types of performance estimation was done on PID i.e. the integrated absolute error (IAE), or the integral of squared error (ISE), or integral of time absolute error (ITAE) but after comparing the simulation results of PSO optimization it was seen that the ISE was the best to use with the disturbance condition to get robustness. An Optimal Fractional Order Controller for an AVR System using Particle Swarm

Optimization Algorithm was proposed in this paper[24] where the FOPID PSO designing was done in these steps- i) Randomly initializing the individuals of the population including searching points and velocities in the feasible range, ii)ïFor each initial individual k of the population, the values of the performance criterion were calculated, iii) Then comparing each individual's evaluation value with its personal best p_{id} . The best evaluation value among the p_{id} was denoted as p_g .*iv*) Modifying the member velocity of each individual, v) If the number of iterations reaches the maximum, go to Step vii, otherwise go to Step ii. *vii*) The latest p_g was the optimal controller parameter. A PID controller tuning via PSO for power system stabilizing is shown in this paper[25]. Though ZN method was also applied for PID tuning, PSO based tuning proved faster results for reaching steady state condition and proficient outcomes for damping of the multimachine power system transient and dynamic disturbance. A fuzzy PID controller tuning via an intelligent PSO and Genetic Algorithm approach for AVR system is depicted in [26]. The Craziness based PSO used in this work proved superior in tuning than the binary coded GA used with respect to optimal transient performance and lesser computational time. The work of Fuzzy Logic was to extrapolate intelligently and linearly, the nominal optimal gains in order to determine off-nominal optimal gains for on line off nominal system parameters.

Robust PID controller tuning via PSO [27], PID tuning via Advanced PSO by changing few mathematical structure of original PSO[28], a novel approach based on Stochastic Particle Swarm optimization (SPSO), with dominant eigenvalue shift for designing robust decentralized load frequency control system for interconnected power system[29], PID controller tuning for Hybrid PV-FC-Diesel-Battery Micro Grid Scheme for Village/Resort Electricity Utilization via PSO[30] are some of the PSO application in PID tuning domain. PSO applications in PID tuning for AVR system[31], PID tuning in slider-crank mechanism system [31], evolutionary robotic-vision system and tracking[33] are also some of the advanced research works in the field of PID tuning via PSO. As well as speed control of the Brushless DC Motor (BLDCM) servo system[34], the values of the parameters of a proposed fuzzy PID controllers optimization with minimization of sum square error (SSE)[35], are some of the recent PSO application work in PID controller tuning.

PSO is capable of generating these qualities for its use- its intelligent application in various scientific and research fields, no overlapping and mutation calculation, simple in nature, its a real number code and it is decided directly by the solution. Although, it has few drawbacks also- The method easily suffers from the partial optimism, which causes the less exact at the regulation of its speed and the direction, it is unable to work out the problems of scattering and optimization. Also PSO cannot work out the problems of non-coordinate system, such as the solution to the energy field and the moving rules of the particles in the energy fields.

Another important swarm intelligence technique known as Bacterial Foraging Optimization Algorithm(BFOA) was first introduced and developed by Passino[36]. Inspired by social foraging of bacteria E.Coli ,which can be explained by four processes namely Chemotaxis, Swarming, Reproduction, Elimination and Dispersal [36], this nature based algorithm was generated and it has made a good impact as a global optimization algorithm in distributed optimization and control. Its application in science and engineering domain and analysis is very exquisitely discussed in [37]. Fig.5 shows a flow chart of BFOA method. A brief literature review on PID controller tuning via BFOA is given below.

Bacterial Foraging Optimization method is newer than other swarm intelligence methods (PSO,ACO) and still hard research work is going on development of this heuristic method. For its real world application this method is gaining more popularity to researchers. An application of hybrid genetic algorithm and BFOA in global optimization is shown in[38]. It was shown using on PID of AVR systems and simulation results showed the superiority of BFOA in tuning than the GA. Interface suppression of linear antenna array by amplitude control via BFOA is depicted in[39]. It was found that the nulling method based on BFA was capable of steering the array nulls precisely to the undesired interference directions. Designing a bow-tie antenna for 2.45 GHz Radio Frequency Identification (RFID) readers via bacteria swarm optimization method(BSO) and Nelder-Mead algorithm is proposed in[40]. The BSO proved more efficient result results in parameter tuning than convenient PSO in the simulation result. Different applications and methodologies for tuning of PID via this hybrid SI technique can be found out in [41]. A comparative study of BFOA and PSO and state of art version of PSO for optimizing multi-modal and high dimensional functions clearly shows its efficiency in such optimization problem[42]. Its generates fruitful results in its application in neural networks in load forecasting, fuzz y logic based problems, signal processing, pattern recognition, robust bus architecture design for power systems, social scheduling problems, as well as in controller tuning approaches. Its application in PID tuning are presented in these works[43,44,45]. A fuzzy PID controller tuning approach is shown in[46].Implementation of BFOA in PID tuning for SIMO process on FPGA[47], design approach of PID controller with rejection function against external disturbance in motor control system via BFOA[48], PID tuning in power system stability and in AVR system for obtaining minimum errors and to damn optimally by BFOA[49] are some of the recent research applications of BFOA in PID tuning.

Although having such capabilities for using in multiple engineering and science research domains, BFOA method still needs more development and adaptability for making it , not just a successful, but one of the best swarm optimization methods in real world application. Our review work cannot be established without the favorable help of this book[50].

6 Comparative Analysis of PID Tuning via Traditional, GA and Different SI Techniques

In this work, four systems were selected for simulation using various tuning method such as ZN, PSO, GA, BFO. MATLAB 7.0 software is employed. The four set of systems, considered for the tuning comparisons are as follows:

Set 1:

$$G(s)=\frac{1}{0.21s^2+0.4038s+0.0411} \tag{11}$$

Set 2:

$$G(s)=\frac{1}{0.21s^2+0.1193s+0.0245} \tag{12}$$

Set 3:

$$G(s) = \frac{1}{0.21s^2 + 0.0434s + 0.0299} \tag{13}$$

Set 4:

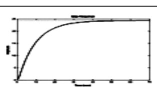
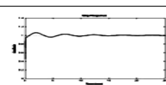
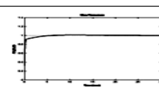
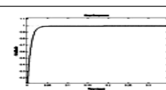
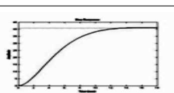
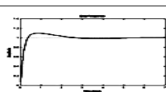
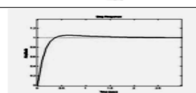
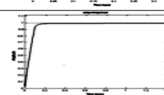
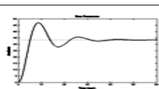
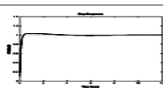
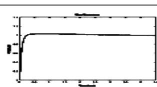
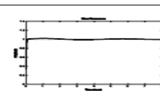
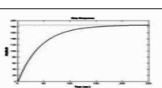
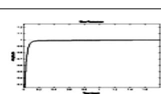
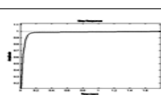
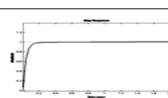
$$G(s) = \frac{1}{0.21s^2 + 0.2369s + 0.0054} \tag{14}$$

The PID parameter values obtained using ZN,GA,PSO and BFO and the closed loop step responses for tuning is shown in next two tables.

Table 2. Different values of PID Parameters after tuning by various tuning methods

Set No	Values of PID parameters using Z-N	Values of PID parameters using PSO	Values of PID parameters using Genetic Algorithm	Values of PID parameters using Bacteria Foraging
Set 1	K_p= 0.0471 T_d = 1.046 T_i = 4.161	K_p= 1.8255 T_d= 2.816 T_i= 0.2219	K_p= 1.2365 T_d = 2.529 T_i = 5.859	K_p = 9.8274 T_d = 2.826 T_i = 0.2196
Set 2	K_p= 0.0083 T_d= 1.8383 T_i= 7.3535	K_p= 1.9045 T_d= 0.8013 T_i =1.0019	K_p= 2.1023 T_d= 0.7929 T_i= 6.9427	K_p = 1.983 T_d = 1.7938 T_i = 6.9938
Set 3	K_p= 0.0028 T_d = 2.165 T_i = 8.66	K_p= 1.255 T_d= 1.8728 T_i= 1.215	K_p= 1.8937 T_d = 1.7856 T_i = 1.0105	K_p = 4.199 T_d = 2.8665 T_i = 0.189
Set 4	K_p= 0.0036 T_d = 3.37 T_i = 13.48	K_p= 4.478 T_d= 1.821 T_i= 3.545	K_p = 3.146 T_d = 2.278 T_i = 4.335	K_p = 5.0257 T_d = 1.147 T_i = 2.726

Table 3. Comparative diagrams of closed loop step responses for various optimization techniques

Set No	Closed loop step response without optimization	Step response after optimizing using PSO	Step response after optimizing using Genetic Algorithm	Step response after optimizing using Bacterial Foraging Optimization
Set 1				
Set 2				
Set 3				
Set 4				

For setting up perfections and better efficacies, each set of transfer functions were tuned by different tuning method in this work. Though in comparison, the swarm intelligence methods proved superior than the traditional ZN method which can be inferred from table no 3. From that table, BFOA method is giving the best optimization among all the tuning methods. PSO and GA showed good results than ZN but surely BFOA produces the better tuning than these EA methods.

7 Conclusion

In this paper we have presented a brief literature review of various applications of swarm computation techniques in PID controller tuning research domain. Alongwith a comparative analysis for different EA and Swarm Intelligence techniques for PID tuning is presented as the critical discussions for using those techniques were also mentioned. For, future works, the dynamic determination of best destination for ACO, adapting fitness sharing for PSO, updating velocity for each individual by taking the best element found in all iterations rather than that of current iteration can be taken for proper implementation to make these techniques more effective in PID tuning as well as global optimization which may become a very effective aspect of artificial and computational intelligence in future.

References

[1] Araki, M.: Control systems. In: Robotics and Automation-Vol II – PID Control. Kyoto University, Japan

[2] Hwa, K.D., Park, J.: Intelligent PID Controller Tuning of AVR System Using GA and PSO. In: Huang, D.-S., Zhang, X.-P., Huang, G.-B. (eds.) ICIC 2005, Part II. LNCS, vol. 3645, pp. 366–375. Springer, Heidelberg (2005)

[3] Astrom, K.J., Hagglund, T.: The future of PID control. Control Eng. Pract. 9(11), 1163–1175 (2001)

[4] Ziegler, G., Nichols, N.B.: Optimum settings for automatic controllers. Trans. ASME 64, 759–768 (1942)

[5] Cohen, G.H., Coon, G.A.: Theoretical Consideration of Retarded Control. Trans. ASME 75, 827/834 (1953)

[6] Astrom, K.J., Hagglund, T.: Automatic tuning of simple regulators with specifications on phase and amplitude margins. Automatica 20, 645–651 (1984)

[7] Krohling, R.A., Rey, J.P.: Design of optimal disturbance rejection PID controllers using genetic algorithm. IEEE Trans. Evol. Comput. 5(1), 78–82 (2001)

[8] Nasri, M., Nezamabadi-pour, H., Maghfoori, M.: A PSO-Based Optimum Design of PIO Controller for a Linear Brushless DC Motor. World Academy of Science, Engineering and Technology 26 (2007)

[9] Kennedy, J., Eberhart, R.C.: Swarm intelligence. Morgan Kaufmann Publisher, San Francisco (2001)

[10] Dorigo, M., Blum, C.: Ant colony optimization theory: A survey. Theoretical Computer Science 344, 243–278 (2005)

[11] Socha, K., Dorigo, M.: Ant colony optimization for continuous domains. European Journal of Operational Research 185(3), 1155–1173 (2009)

[12] Coelho, L., Bernert, D.: A modified ant colony optimization algorithm based on differential evolution for chaotic synchronization. Expert Systems with Applications 37, 4198–4203 (2010)

[13] Duan, H., Liu, S., Wang, D., Yu, X.: Design and realization of hybrid ACO-based PID and LuGre friction compensation controller for three degree-of-freedom high precision flight simulator. Simulation Modelling Practice and Theory 17(6), 1160–1169 (2009)

[14] Duan, H., Wang, D., Yu, X.: Novel Approach to Nonlinear PID Parameter Optimization Using Ant Colony Optimization Algorithm. Journal of Bionic Engineering 3(2), 73–78 (2006)

[15] Dorigo, M., Stützle, T.: The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. In: Handbook of Metaheuristics, vol. 57, pp. 250–285. Springer, Heidelberg (2003)

[16] Cao, C., Guo, X., Liu, Y.: Research on Ant Colony Neural Network PID Controller and Application. In: Proc. SPND, pp. 253–258. IEEE, Qingdao (2007)

[17] Boubertakh, H., Tadjine, M., Glorennec, P.-Y., Labiod, S.: Tuning fuzzy PID controllers using ant colony optimization. In: Proc. Med., pp. 13–18. IEEE, Thessaloniki (2009)

[18] Abachizadeh, M., Yazdi, M.R.H., Yousefi-Koma, A.: Optimal tuning of PID controllers using Artificial Bee Colony algorithm. In: AIM 2010, pp. 379–384. IEEE/ASME, Montreal (2010)

[19] Karaboga, D.: An idea based on honey bee swarm for numerical optimization: Technical report. Erciyes University, Turkey (2005)

[20] Ünal, M., Erdal, H., Topuz, V.: Trajectory tracking performance comparison between genetic algorithm and ant colony optimization for PID controller tuning on pressure process. In: Computer Applications in Engineering Education. Wiley (2010)

[21] Zhang, L.: Simplex method based optimal design of PID controller. Information and Control 33(3), 376–379 (2004)

[22] Ali Al-Waily, R.S.: Design of Robust Mixed H2/H∞ PID Controller Using Particle Swarm Optimization. International Journal of Advancements in Computing Technology 2(5), 53–60 (2010)

[23] De Falco, Cioppa, A.D., Tarntino: Evalution of Particle Swarm Optimization Effectiveness in Classification

[24] Zamani, M., Ghartemani, M., Sadati, N., Parniani, M.: Design of a fractional order PID controller for an AVR using particle swarm optimization. Control Engineering Practice 17(12), 1380–1387 (2009)

[25] Oonsivilai, A., Marungsri, B.: Stability Enhancement for Multi-machine Power System by Optimal PID Tuning of Power System Stabilizer using Particle Swarm Optimization. WSEAS Transactions on Power Systems 6(3), 465–474 (2008)

[26] Mukherjee, V., Ghoshal, S.P.: Intelligent particle swarm optimized fuzzy PID controller for AVR system. Electric Power Systems Research 77(12), 1689–1698 (2007)

[27] Kim, T.H., Maruta, I., Sugie, T.: Particle Swarm Optimization based Robust PID Controller Tuning Scheme. In: Proc. IEEE Conference on Decision and Control, New Orleans, LA, USA, pp. 200–205 (2007)

[28] Jalivand, A., Kimiyaghalam, A., Ashouri, A., Mahdavi, M.: Advanced Particle Swarm Optimization-Based PID Controller Parameters Tuning. In: Proc. 12th IEEE International Multitopic Conference, pp. 429–435 (2008)

[29] Ebrahim, M.A., Mostafa, H.E., Gawish, S.A., Bendary, F.M.: Design of Decentralized Load Frequency Based-PID Controller Using Stochastic Particle Swarm Optimization Technique. In: Proc. IEEE. EPECS 2009, Sharjah, pp. 1–6 (2009)

[30] Sharaf, A., El-Gammal, A.: A novel efficient PSO-self regulating PID controller for hybrid PV-FC-diesel-battery micro grid scheme for village/resort electricity utilization. In: Proc. IEEE EPEC 2010, Halifax, NS, pp. 1–6 (2010)

[31] Gaing, Z.L.: A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Trans. Energy Conversion 19(2), 384–391 (2004)

[32] Kao, C.C., Chuang, C.W., Fung, R.: The self-tuning PID control in a slider–crank mechanism system by applying particle swarm optimization approach. Mechatronics 16(8), 513–522 (2006)

[33] Sulistijono, I.A., Kubota, N.: Evolutionary robot vision and particle swarm optimization for Multiple human heads tracking of a partner robot. In: Proc. CEC 2007, pp. 1537–1541. IEEE, Singapore (2007)

[34] Ren, Y., Xu, X.: Optimization Research of PSO-PID Algorithm for the Design of Brushless Permanent Magnet Machines. In: Fifth IEEE International Symposium on Embedded Computing, Washington, DC, USA (2008)

[35] Dorrah, H.T., El-Garhy, A.M., El-Shimy, M.E.: Design of PSO-Based Optimal Fuzzy PID Controllers for the Two-Coupled Distillation Column Process. In: Proc. 14th International Middle East Power Systems Conference, pp. 181–188. Cairo University, Egypt (2010)

[36] Passino, K.M.: Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control Systems 22(3), 52–67 (2002)

[37] Das, S., Biswas, A., Dasgupta, S., Abraham, A.: Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Applications. In: Abraham, A., Hassanien, A.-E., Siarry, P., Engelbrecht, A. (eds.) Foundations of Computational Intelligence Volume 3. SCI, vol. 203, pp. 23–55. Springer, Heidelberg (2009)

[38] Kim, D., Abraham, A., Cho, J.: A hybrid genetic algorithm and bacterial foraging approach for global optimization. Information Sciences 177(18), 3918–3937 (2007)

[39] Guney, K., Busbug, S.: Interference Suppression Of Linearantenna Arrays By Amplitude-Only Control Using A Bacterial Foraging Algorithm. Progress In Electromagnetics Research, PIER 79, 475–497 (2008)

[40] Mahmoud, K.R.: Design Optimization Of A Bow-Tie Antenna For 2.45Ghz Rfid Readers Using A Hybrid BSO-NM Algorithm. Progress In Electromagnetics Research, PIER 100, 105–117 (2010)

[41] Grosan, C., Abraham, A.: Hybrid Evolutionary Algorithms: Methodologies, Architectures, and Reviews. In: Hybrid Evolutionary Algorithms, vol. 75, Springer, Heidelberg (2007), doi:10.1007/978-3-540-73297-6_1

[42] Biswas, A., Dasgupta, S., Das, S., Abraham, A.: Synergy of PSO and Bacterial Foraging Optimization – A Comparative Study on Numerical Benchmarks. In: Innovations in Hybrid Intelligent Systems, pp. 255–263. Springer, Heidelberg (2007)

[43] Kim, D.H., Cho, J.H.: Adaptive Tuning of PID Controller for Multivariable System Using Bacterial Foraging Based Optimization. In: Szczepaniak, P.S., Kacprzyk, J., Niewiadomski, A. (eds.) AWIC 2005. LNCS (LNAI), vol. 3528, pp. 920–928. Springer, Heidelberg (2005)

[44] Korani, W.M., Dorrah, H.T., Emara, H.M.: Bacterial foraging oriented by Particle Swarm Optimization strategy for PID tuning. In: Proc. CIRA 2009, pp. 445–450. IEEE, Daejeon (2009)

[45] Oyekan, J., Hu, H.: A Novel Bacterial Foraging Algorithm forAutomated tuning of PID controllers of UAVs. In: Proceedings of the 2010 IEEE International Conference on Information and Automation, China, pp. 693–698 (2010)

[46] Su, T.-J., Chen, G.-Y., Cheng, J.-C., Yu, C.-J.: Fuzzy PID controller design using synchronous bacterial foraging optimization. In: Proc. 3rd Interbnational Conference on Information Sciences and Interaction Sciences (ICIS), pp. 639–642. IEEE, China (2010)

[47] Jain, T., Patel, V., Nigam, M.: Implementation of PID Controlled SIMO Process on FPGA Using Bacterial Foraging for Optimal Performance. IJCEE 1(2), 107–110 (2009)

[48] Kim, D.H., Cho, J.H.: Robust Tuning of PID Controller With Disturbance Rejection Using Bacterial Foraging Based Optimization. WSEAS Transactions on Systems (2004)

[49] Manuaba, I., Abdillah, M., Soeprijanto, A., Hery, M.P.: Coordination of PID based power system stabilizer and AVR using combination bacterial foraging techique — Particle swarm optimization. In: Proc. ICMSAO 2011, pp. 1–7. IEEE, Kuala Lampur (2011)
[50] Abraham, A., Das, S., Roy, S.: Swarm Intelligence Algorithms for Data Cluctering. Soft Computing For Knowledge Discovery And Data Mining Book, Part IV
[51] Tan, G., Zeng, Q., Li, W.: Design of PID controller with incomplete derivation based on ant system algorithm. Journal Of Control Theory And Applications 2(3), 246–252 (2004)