

Optimal Tuning of PID Controller using Adaptive Hybrid Particle Swarm Optimization Algorithm

S. Morkos, H. Kamal

Sawsan Morkos Gharghory

Electronics Research Institute
Dokki, Cairo, Egypt
E-mail: sawsan@eri.sci.eg

Hanan Ahmed Kamal

Faculty of Engineering, Cairo University
Giza, Egypt
E-mail: Hanan_ak2003@yahoo.com

Abstract:

Particle swarm optimization (PSO) has proved its ability as an efficient search tool in many optimization problems. However, PSO is easy to be trapped into local minima due to its mechanism in information sharing. Under this circumstance, all the particles could quickly converge to a position by the attraction of the best particle; all particles could hardly be improved. To overcome premature convergence of the standard PSO algorithm, this paper presents an adaptive hybrid PSO, namely (AHPSO) by employing an adaptive mutation operator for local best particles instead of applying the mutation operator to the global best particle as has been done in previous work. The developed algorithm is a new approach which allows the swarm to be more diverse by making better exploration of the local search space instead of global search space investigated by previous researchers. The proposed algorithm holds on the properties of simple structure, fast convergence, and at the same time enhances the variety of the population, and extends the search space. It is applied to self-tuning of proportional-integral-derivative-(PID) controller in the ball and hoop system which represents a system of complex industrial processes. The results are compared with those obtained by applying standard PSO, and adaptive hybrid PSO based on global best particles. It has been shown that the developed AHPSO local best algorithm is faster in convergence and the obtained results are proved to have higher fitness than the other two algorithms.

Keywords: PSO, Adaptive mutation, PID Controller, and ball and hoop system.

1 Introduction

The PID controller was the most popular controller of this century because of its remarkable effectiveness, simplicity of implementation and broad applicability. In practice, it is hard to obtain optimal tuning for PID controller. Most of PID tuning is done manually which is difficult and time consuming. In order to use PID controller better, the optimal tuning of its parameters have become an important research field [1]. People have made lots of research, and proposed some advanced PID control methods, such as expert PID control based on knowledge inference[2], self-learning PID control based on regulation, neural network PID control based on connection mechanism[3], and intelligent PID control based on fuzzy logic[4,5]. Genetic algorithm (GA) has

been applied to self-tuning of PID parameters, too [6]. However, GA has the disadvantages of premature and slow convergence rate, and the need to set up many parameters. Recently, the computational intelligence has proposed particle swarm optimization (PSO) [7, 8] as opened paths to a new generation of advanced process control. The PSO algorithm, proposed by Kennedy and Eberhart [7] in 1995, was an evolution computation technology based on population intelligent methods. In comparison with genetic algorithm, PSO is simple, easy to realize and has very deep intelligent background. It is not only suitable for scientific research, but also suitable for engineering applications in particular. Thus, PSO received widely attentions from evolution computation field and other fields. Now the PSO has become a hotspot of research.

Many efforts on the enhancement of traditional PSO have been proposed, by combining the PSO with other techniques, especially evolutionary computation techniques. The research effort in [9] has developed a hybrid method combining two heuristic optimization techniques, GA and PSO, for the global optimization of multimodal functions. The work in [10], obtained better results by applying PSO first followed by applying GA in their profiled corrugated horn antenna optimization problem. The effort in [11] has introduced a new integrated genetic swarm optimization algorithm (IGSA), combining the strengths of PSO with GA. It is applied in the tuning of PID controllers for the ball and hoop system. A genetic programming based adaptable evolutionary hybrid particle swarm optimization algorithm, have presented in [12], for avoiding premature convergence to local minima by the introduction of diversity in the swarm.

In addition to incorporate evolutionary algorithms into PSO, another research trend is to merge evolutionary operators like selection, crossover and mutation to the PSO. By applying selection operation in PSO, the particles with the best performance are copied into the next generation; therefore, PSO can always keep the best performed particles [13]. By applying crossover operation, information can be swapped between two individuals to have the ability to "fly" to the new search area as that in evolutionary programming and GA [14]. Among the three evolutionary operators, the mutation operators are the most commonly applied evolutionary operators in PSO. The purpose of applying mutation to PSO is to increase the diversity of the population and the ability to have the PSO to escape from the local minima. One approach is to mutate PSO parameters such as the position of the best neighborhood, as well as the inertia weight [15]. Another approach is to prevent particles from moving too close to each other so that the diversity could be maintained and therefore escape from being trapped into local minima. In [16], the particles are relocated when they are too close to each other. In [17], collision-avoiding mechanisms are designed to prevent particle from colliding with each other and therefore increase the diversity of the population. In [18], deflection and stretching techniques as well as a repulsion technique are incorporated into the original PSO to avoid particles to move toward the already found global minima, so that the PSO can have more chances to find as many global minima as possible. Chen [19] presented a Gaussian mutation operator with adaptive mutation probability. Wang [20] proposed an adaptive mutation on the basis of average velocity of swarm. Pant [21] used an adaptive Cauchy mutation operator in PSO, which was based on beta distribution. Tang [22] proposed Local Search PSO, namely LSPSO by applying an adaptive mutation operator which dynamically adjusts the step size of local search in terms of the size of current search space in order to improve the global search ability of PSO. He introduced one technique for mutating the global best particle by a value which based on the difference between the maximum and minimum value for each dimension of the search space. However, the results given by LSPSO as well as the standard PSO, PSO with both Gaussian and Cauchy mutation fall into local optima for some types of test functions.

In the present work, a new adaptive hybrid PSO, called AHPSO local best is proposed by applying an adaptive mutation operator, which differs from the above mutation techniques by using three types of mutation operators instead of using one technique. The main idea

of AHPSO local best is to generate an operator that can adaptively select the most suitable mutation method in each generation according to each stage of the problem. Three types of mutation operators are used in this work, Gaussian, Cauchy, and Levy mutation operators. In the proposed algorithm, the local best particles are to be mutated by the selected mutation operator instead of applying the mutation operator to the global best particle as in the previous literatures. This can be accomplished by searching the neighborhood of the global best particles in each generation, resulting in more exploration of the search space and increasing the diversity of the population and the ability to have the PSO to avoid the local optima. The new developed algorithm is carried out for the optimal tuning of PID controller to the ball and hoop system. The performance of the system is compared with the standard PSO and adaptive PSO using the global best particle. Experimental studies on tuning the parameters of PID controller for the ball and hoop problem show that AHPSO local best performs better than the standard PSO and adaptive PSO based on global best particle search technique. The obtained results have higher fitness and faster convergence.

The rest of the paper is organized as follows. Section 2 describes the standard PSO. In Section 3, PSO with adaptive mutation operator is described and the proposed AHPSO local best algorithm is presented in detail. An overview of the control problem which will be solved is provided in Section 4. Experimental results and discussions are presented in Section 5. Finally, Section 6 concludes the whole work.

2 Particle Swarm Optimization

PSO is a stochastic optimization technique [7] which operates on the principle of social behavior like bird flocking or fish schooling. Like other evolutionary algorithms, PSO is also a population-based search algorithm and starts with an initial population of randomly generated solutions called particles which fly through the search space. Each particle represents a candidate solution to the optimization problem, and has a velocity and a position. The position of a particle is influenced by the best position visited by itself i.e. its own experience and the position of the best particle in its neighborhood i.e. the experience of neighboring particles. The best particle in the population is denoted by (global best), while the best position that has been visited by the current particle is denoted by (local best). Consequentially, each particle is influenced by the best performance of any member in the entire population due to the sharing information between them. The performance of each particle is measured using a fitness function that varies depending on the optimization problem. Each particle in the swarm is represented by the following characteristics:

X_i : The current position of the particle i .

V_i : The current velocity of the particle i .

P_i : The best position of particle i so far, and P_g is the best position found in the whole swarm so far. Equations (1) and (2) are used for updating both of the velocity and the position of each particle.

$$V_i = w_i.V_i + c_1.r_1.(P_i - X_i) + c_2.r_2.(P_g - X_i) \quad (1)$$

$$X_i = X_i + V_i \quad (2)$$

Where: c_1 and c_2 are the cognitive coefficients and r_1 , and r_2 are random real numbers drawn from $U(0, 1)$, w is the inertia weight which is used to achieve a balance in the exploration and exploitation of the search space and plays very important role in PSO convergence behavior. The

inertia dynamically reduces during a run from 1.0 to near 0 in each generation which facilitates a balance in the exploration and exploitation of the search space, it is determined as follows:

$$w_i = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \cdot iter \quad (3)$$

Where $iter_max$, is the maximum number of iterations, and $iter$ is the current number of iteration. Several topologies exist in literature for the particles to communicate with one another. The topologies are ring, star, pyramid and master-slave topologies [23]. Among the topologies, the star is the best topology.

3 Adaptive particle swarm optimization

Different types of mutation operators can be used to increase the diversity of the population and to help PSO jump out of local minima. The type of mutation operator may be more effective or worse depending on the stage of optimization process. In the present work, three types of mutation operators are applied at different stages of the problem for more exploration of search space. An adaptive method for selecting the mutation operator that is suitable for each stage of the problem was proposed in this paper. Also, the developed approach search for the best neighborhood of the global best particle to be mutated by the selected mutation operator. The proposed mutation operators are presented in the following section as follow:

-Cauchy Mutation operator

$$V_g = V_g \exp(\delta) \quad (4)$$

$$X_g = X_g + V_g \delta_g \quad (5)$$

Where:

X_g and V_g represent position and velocity of the global best particle.
 δ and δ_g denote Cauchy random numbers with the scale parameter of 1.

-Gaussian Mutation operator

$$V_g = V_g \exp(N) \quad (6)$$

$$X_g = X_g + V_g N_g \quad (7)$$

Where:

X_g and V_g represent position and velocity of the global best particle.
 N and N_g are Gaussian distribution numbers with the mean equals 0 and the variance equals 1.

-Levy Mutation operator

$$V_g = V_g \exp(L(\alpha)) \quad (8)$$

$$X_g = X_g + V_g L_g(\alpha) \quad (9)$$

$L(\alpha)$ and $L_g(\alpha)$ are random numbers generated from Levy distribution with a parameter α which is set to 1.3 [24].

3.1 The Adaptive mutation operator

The method proposed for adaptive mutation uses the three mutation operators described above. Initially, the selection ratio is set equal to 1/3 for the three mutation operators, by this ratio; the number of particles mutated by each operator is calculated. Then, each mutation operator is applied to the swarm particles according to its selection ratio and finally, the resulting offspring fitness is evaluated. The mutation operators that result in higher fitness values of offspring have the most chance to be selected than the other one with lower fitness values of offspring. Gradually the most suitable mutation operator will be chosen automatically and control all the mutation behavior in the whole swarm. The steps for selecting the best mutation operators are described as follows [24]:

1- The progress value for each operator at each generation is evaluated as:

$$prog_i(t) = \sum_{j=1}^{M_i} f(P_j^i(t)) - \min(f(P_j^i(t)), f(c_j^i(t))) \quad (10)$$

Where:

$P_j^i(t)$, and $c_j^i(t)$ denote the fitness of a parent and its child produced by mutation operator i at generation t , and M_i is the number of particles that select mutation operator i to mutate.

2- The reward value for each operator is calculated as:

$$reward_i(t) = \exp\left(\frac{prog_i(t)}{\sum_{j=1}^N prog_j(t)}\alpha + \frac{S_i}{M_i}(1 - \alpha)\right) + c_i P_i(t) - 1 \quad (11)$$

Where:

S_i is the number of particles whose children have a better fitness than themselves after the mutation by operator i , $P_i(t)$ is the selection ratio of mutation operator i at generation t , α is a random weight between (0,1), N is the number of mutation operator, and c_i is a penalty factor for mutation operator i , and is defined as:

$$c_i = \begin{cases} 0.9 & \text{if } S_i = 0 \text{ and } P_i(t) = \max_{j=1}^N (P_j(t)) \\ 1, & \text{otherwise} \end{cases} \quad (12)$$

The mutation operator with maximum reward has the best chance to mutate the best local particles selected during each generation.

3- The selection ratio to the next generation for the mutation operator is updated as follows:

$$P_i(t+1) = \frac{reward_i(t)}{\sum_{j=1}^N reward_j(t)}(1 - N - \gamma) + \gamma \quad (13)$$

Where, γ is the minimum selection ratio for each mutation operator and is set equal to 0.01 in our problem. The selection ratio for the next generation depends on four factors: the progress value, the minimum selection ratio, the previous selection ratio, and the ratio of successful mutation operator. The selection of each mutation operator may be updated each generation or after a fixed number of generations, in this paper we update the selection ratio after each generation.

3.2 The proposed AHPSO local best particles algorithm

The standard *PSO* was inspired by the social and cognitive behavior of swarm. According to the analysis given in [25], particles are largely influenced by its previous best particles and the global best particle. Once the best particle has no change in a local optimum, all the rest particles will quickly converge to the position of the best particle. The present work proposes searching neighbors of the global best particle to be mutated in each generation, rather than selecting the global best particle for mutating. As a result, it would be helpful for the best particles to jump out the local minima, and the whole swarm would move to better position. This can be accomplished by applying the adaptive mutation operator described above to the neighborhood of the global best particle in each generation. The framework of PSO algorithm with one of the three mutations operators according to its selection ratio to mutate the best neighborhood particles of the global best particle is given as follows:

- 1- Generate the initial position and velocity for each particle in the swarm randomly.
- 2- Evaluate the fitness of each particle, and determine the local and the global best fitness for each particle in the swarm.
- 3- Set the initial selection ratio equal 1/3.
- 4- Update each particle according to equation (1) and (2)
- 5- For each particle i , if its fitness is smaller than the fitness of its previous best position (P_i) update P_i .
- 6- Update the fitness of the best position (P_g) of all particle if there is a particle with fitness smaller than the current best fitness P_g .
- 7- Apply each one of mutation operator to number of particles according to its selection ratio.
- 8- Evaluate the progress and the reward values to select the best one from the three above mutation operators, and then update the selection ratio of each operator for the next generation.
- 9- Mutate the best neighborhood particles of the global best particle with the best mutation operator (with maximum reward), and select the best one from the mutated best neighborhood to produce (P_g^*).
- 10- Compare P_g^* and P_g to select the better to reproduce in the next generation.
- 11- Stop if the stop criterion is satisfied otherwise, go to step 4.

4 Plant System

The Ball and Hoop system illustrates the dynamics of a steel ball that is free to roll on the inside of a rotating circular hoop. There is a groove on the inside edge of the hoop so that a steel ball can roll freely inside the hoop. This introduces the complexity of the rolling radius of the ball being different to the actual radius of the ball as illustrated in Figure 1 where angle θ is the hoop angular position. The position of the ball is given by:

- 1- γ is the position of the ball on the hoop periphery with respect to a datum point.
- 2- Ψ is the slosh angle which measures the deviation of the ball from its rest position.

A fourth order system for the Ball and Hoop system with the following transfer function[26] is:

$$G(s) = \frac{1}{S^4 + 6S^3 + 11S^2 + 6S} \quad (14)$$

The ball and hoop apparatus is difficult to control optimally using a PID controller because the system parameters are constantly changing. The parameters of PID controller will be tuned offline separately, using PSO, AHPSO global best, and AHPSO local best algorithms as shown in figure 2.

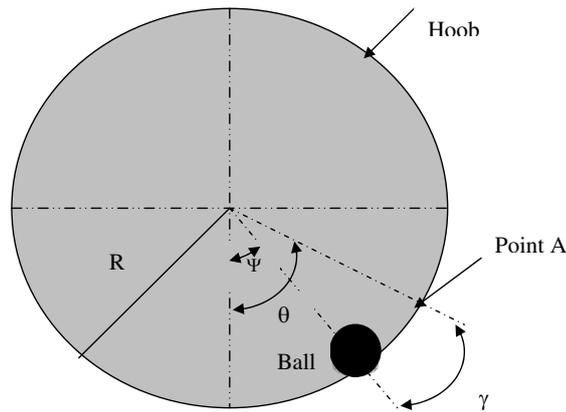


Figure 1: The ball and hoop system

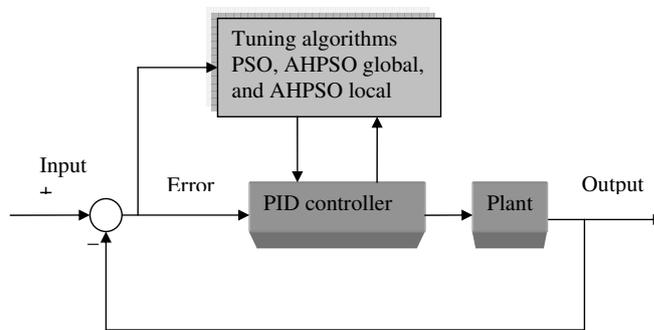


Figure 2: The structure of the proposed algorithms in tuning PID controller for the plant system

Various objective functions based on error performance criterion are used to evaluate the performance of the above algorithms. Each objective function is fundamentally the same except for the section of code that defines the specific error performance criterion being implemented to optimize the performance of a PID controlled system. The Performance index is calculated over a time interval T. Performance indices used to estimate the best parameters of PID controller are given by:

- **Integral of the Square of the Error (ISE)**

$$I_{ISE} = \int_0^T e^2(t)dt \quad (15)$$

- **Mean of the Square of the Error (MSE)**

$$I_{MSE} = \frac{1}{n} \sum_{i=1}^n (e(t))^2 \quad (16)$$

- **Integral of Absolute Magnitude of the Error (IAE)**

$$I_{IAE} = \int_0^T |e(t)|dt \quad (17)$$

Where: e is the error calculated over a time interval T . The effectiveness of the proposed AHPSO local best algorithm in comparison with the other two algorithms is tested using the above three performance indices.

5 Simulation Result

To evaluate the performance of AHPSO based on local best particles, experiments have been carried out for optimal tuning of PID controller to the ball and hoop system. The performance results of PID controller tuned by AHPSO local best search in comparison with PSO, and AHPSO global best particle in the swarm is analyzed using IAE, ISE, and MSE performance indices. Cost functions achieved by each algorithm are averaged over 10 runs for 30 generations. The resulted time response and cost function for the three algorithms using three performance indices are shown in Figures 3-8 respectively. Tables 1-3 give comparison of cost function values and the transient response characteristics for PSO, AHPSO global best, and AHPSO local best algorithms using IAE, ISE and MSE performance indices.

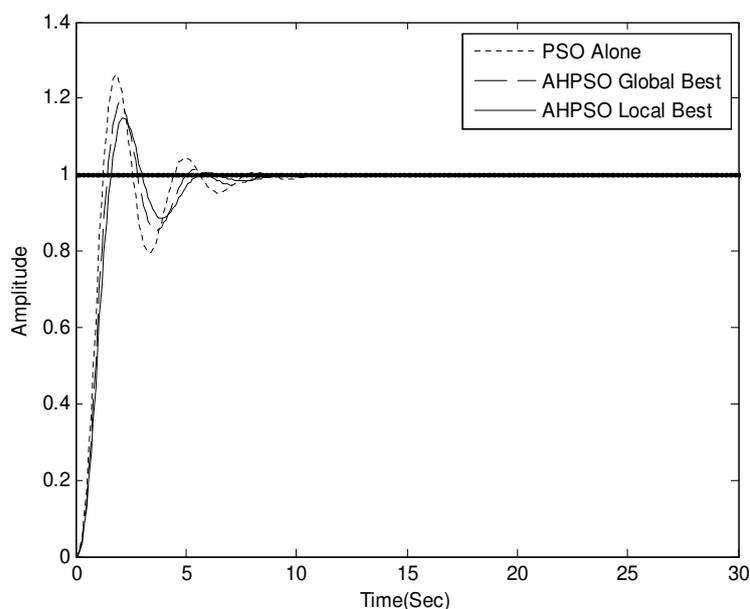


Figure 3: System response using IAE for PSO, AHPSO global particle, and AHPSO local particle

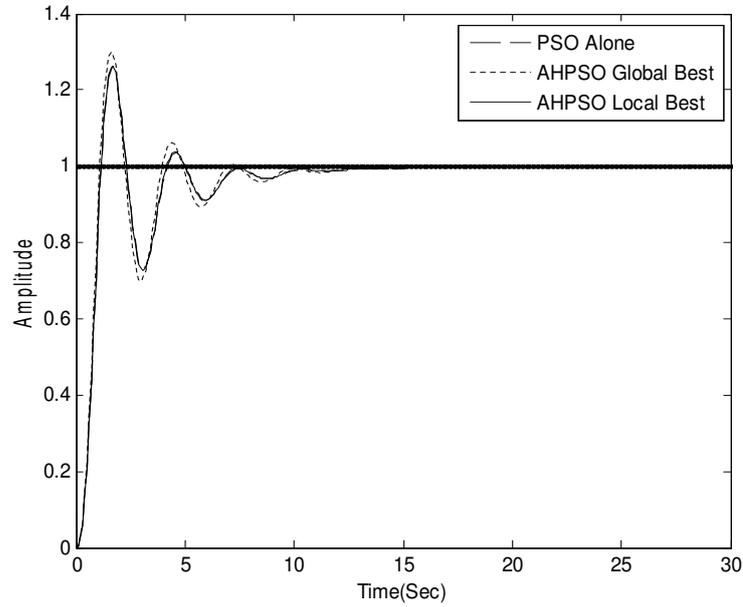


Figure 4: System response using ISE for PSO, AHPSO global particle, and AHPSO local particle

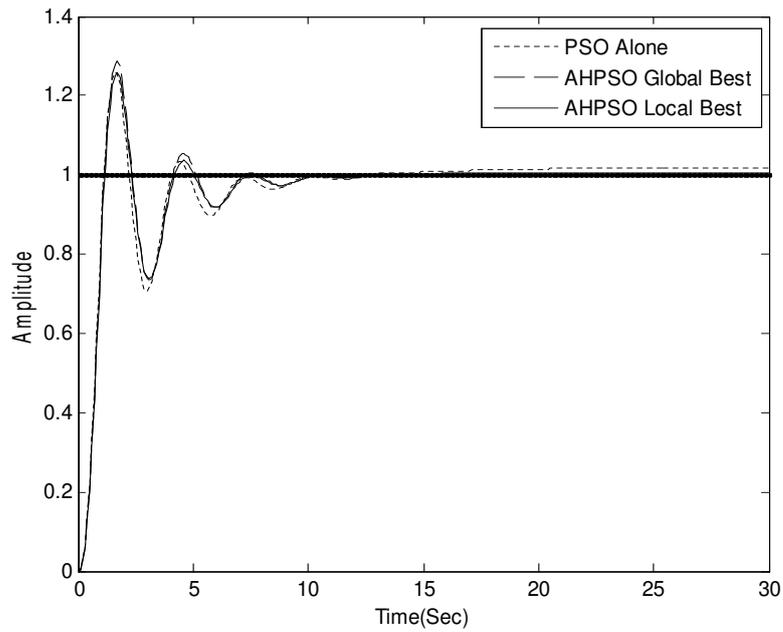


Figure 5: System response using MSE for PSO, AHPSO global particle and AHPSO local particle

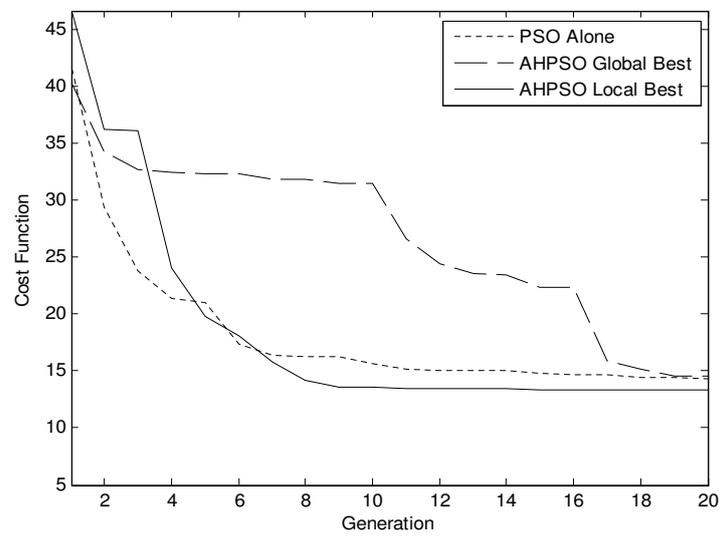


Figure 6: Cost function using IAE for PSO, AHPSO global particle, and AHPSO local particles

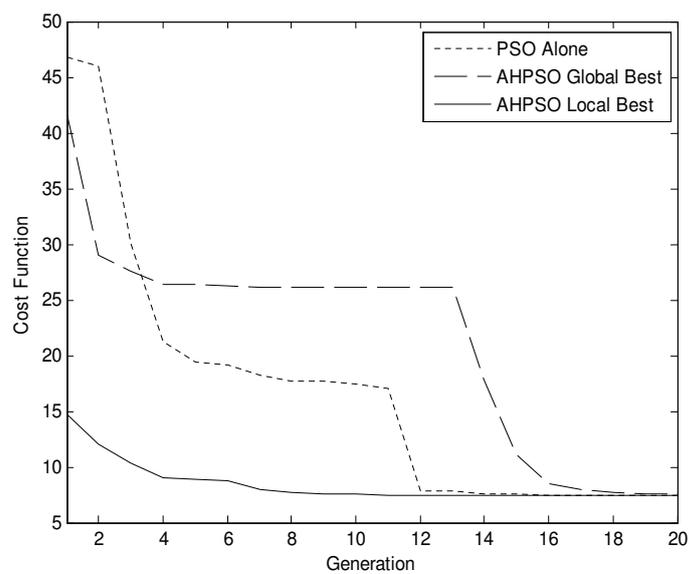


Figure 7: Cost function using ISE for PSO, AHPSO global particle, and AHPSO local particles

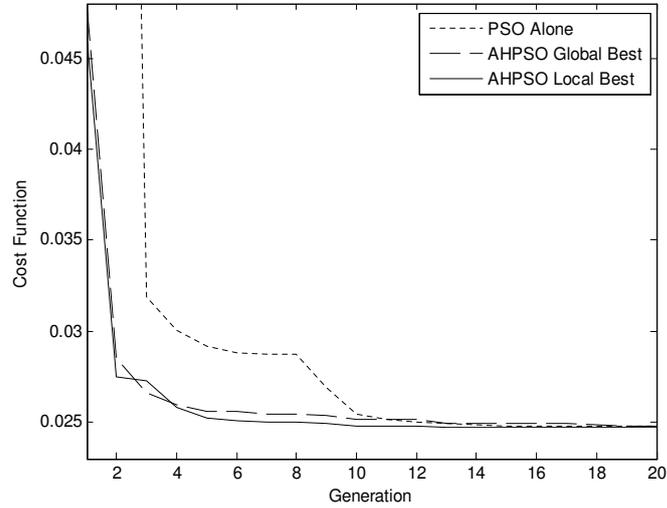


Figure 8: Cost function using MSE for PSO, AHPSO global particle, and AHPSO local particles

Table 1: Transient response characteristics using IAE criteria

Criteria IAE	Standard PSO	AHPSO Global	AHPSO Local
Rise Time T_r	1.12	1.29	1.17
Peak Value M_p	25%	15.5%	14%
Settling Time T_s	7.2	7.58	5.1
Peak Time T_p	1.27	2.30	2.25
Cost Function	13.39	13.38	13.36

Table 2: Transient response characteristics using ISE criteria

Criteria ISE	Standard PSO	AHPSO Global	AHPSO Local
Rise Time T_r	1.02	0.97	0.83
Peak Value M_p	25.95%	28%	25.89%
Settling Time T_s	9.54	9.29	9.14
Peak Time T_p	1.72	1.72	1.67
Cost Function	7.46	7.49	7.43

Table 3: Transient response characteristics using MSE criteria

Criteria MSE	Standard PSO	AHPSO Global	AHPSO Local
Rise Time T_r	1.012	1.01	0.84
Peak Value M_p	25.9%	29%	25.6%
Settling Time T_s	9.59	9.4	9.32
Peak Time T_p	1.72	1.72	1.69
Cost Function	0.0248	0.0248	0.0247

Simulation results demonstrate the superiority of AHPSO based on local search comparing to the other algorithms. In terms of overshoot (peak value), AHPSO local search has a lower overshoot by 1.3% than AHPSO global search and 8.8% than PSO for IAE performance index. In ISE performance index, the improvement is about 1.6% over AHPSO global search and has 0.047 over PSO. For MSE, the improvement is about 0.23% over PSO, and about 2.6% over AHPSO global search using MSE performance index.

For cost function the results are as follows: the improvement is about 0.22% over PSO, and 0.149% over AHPSO global search using IAE performance index. For ISE performance index it is about 0.4% over PSO, and 0.8% over AHPSO global search. The improvement using MSE performance index is about 0.68% over PSO, and 0.4% over AHPSO global search.

As for settling time, AHPSO local search has minimum settling time to reach the final minimum cost function. The improvement is about 29% over PSO, and 32.7% over AHPSO global search using IAE. It equals 4.2% over PSO, and is 1.6% over AHPSO global search for ISE. Also, it is about 2.8% over PSO, and is 0.85% over AHPSO global search using MSE metric.

6 Conclusion

This paper is concerned with developing adaptive operator for the selection of best mutation technique of three investigated mutation techniques: Cauchy, Gaussian, and Levy techniques. Instead of applying single mutation operator, several mutation operators are applied at different stages for best performance. A new particle swarm algorithm based on adaptive mutation operator to local best particles namely AHPSO local best is proposed in this paper. Instead of applying the best mutation operators to the global particle, it is applied to the neighbors' of global best particle. Besides, the paper investigates the use of AHPSO based on local best particles for tuning PID controller parameters for the ball and hoop system and compares the system performance with the standard PSO and AHPSO based on global best particles. The performance of the three algorithms is analyzed based on three performance indices; IAE, ISE, and MSE. Experimental results show the superiority of the AHPSO local search over the other techniques for the optimal tuning of PID controller to the ball and hoop system. Also, it has been shown that the developed algorithm is faster in convergence and gives higher fitness value than the other algorithms, at the same time enhances the variety of the population, and extends the search space. Also, the time response characteristics of the proposed algorithm are better than other techniques. In future research, we intend to apply the technique to different set of practical constrained problems to show the robustness of the technique. Also, comparison of the effectiveness of different mutation operators on particles velocity for different types of problems will be studied.

Bibliography

- [1] Astroöm K., and Hagglund T., "The future of PID control". *Control Engineering Practice*, 9, pp. 1163-1175, 2001.
- [2] Conradie A., Miikkulainen R., and Aldrich C., "Adaptive Control Utilizing Neural Swarming", In *Proceedings of the Genetic and Evolutionary Computation Conferences, USA*, 2002.
- [3] Hossein Shayeghi, Heidar Ali Shayanfar and Aref Jalili, "Multi Stage Fuzzy PID Load Frequency Controller in a Restructured Power System", *Journal of Electrical Engineering*, Vol. 58, No. 2, pp. 61-70, 2007.
- [4] Saban Cetin, and Ozgür Demir, "Fuzzy PID Controller with Coupled Rules for a Nonlinear Quarter Car Model", *World Academy of Science, Engineering and Technology* Vol. 41, pp.238-241, 2008.
- [5] Aye Aye Mon, "Fuzzy Logic PID Control of Automatic Voltage Regulator System", *Proceedings of PWASET*, Vol. 38, Feb., 2009.
- [6] Cipperfield A. Flemming P., and Fonscea C., "Genetic Algorithms for Control System Engineering", in *Proceedings Adaptive Computing in Engineering Design Control*, pp- 128-133, 1994.
- [7] Kennedy J. and Eberhart C., "Particle Swarm Optimization", *Proceedings of the IEEE International Conference on Neural Networks, Australia*, pp. 1942-1948, 1995.
- [8] Oliveira, P. M., Cunha, J. B., and Coelho, J. o. P., "Design of PID controllers using the Particle Swarm Algorithm.", *Twenty-First IASTED International Conference: Modeling, Identification, and Control (MIC 2002)*, Innsbruck, Austria. 2002.
- [9] Yi-Tung Kao and Erwie Zahara, "A hybrid genetic algorithm and particle swarm optimization for multimodal functions", *Applied Soft Computing* Vol. 8, pp 849-857, 2008.
- [10] Robinson, J., Sinton, S., and Rahmat-Samii, Y., "Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna", *IEEE International Symposium on Antennas & Propagation. San Antonio, Texas. June, 2002*.
- [11] H. A. Kamal, "A new integrated GA/PSO Algorithm for Optimal tuning of PID Controller", *the Mediterranean Journal of Measurement and Control*, Vol. 6, No. 1, pp.18-24, January 2010.
- [12] M. Rashid and A. Rauf Baig, "A genetic programming based adaptable evolutionary hybrid particle swarm optimization algorithm", *International Journal of Innovative Computing, Information and Control (ICIC)*, Vol. 6, Nu. 1, January 2010.
- [13] Angeline, P. J., "Using selection to improve particle swarm optimization", *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 1998)*, Anchorage, Alaska, USA. 1998.
- [14] Lvbjerg, M., Rasmussen, T., and Krink, T., "Hybrid particle swarm optimizer with breeding and subpopulations", *Proceedings of the third Genetic and Evolutionary Computation Conference (GECCO)*, Vol. 1, pp. 469-476, 2001.

- [15] Miranda, V., and Fonseca, N., " New evolutionary particle swarm algorithm (EPSO) applied to voltage/VAR control", The 14th Power Systems Computation Conference (PSCC'02), Seville, Spain, June, 2002.
- [16] Lvbjerg, M., and Krink, T., "Extending particle swarms with self-organized criticality", Proceedings of the Fourth Congress on Evolutionary Computation (CEC-2002).
- [17] Blackwell, T., and Bentley, P. J., (2002). "Don't push me ! Collision-avoiding swarms". IEEE Congress on Evolutionary Computation, Honolulu, Hawaii USA, 2002.
- [18] Parsopoulos, K. E., and Vrahatis, M., "On the computation of all global minimizers through particle swarm optimization", IEEE Transactions on Evolutionary Computation, (accepted for special issue on PSO, 2004.
- [19] J. Chen, Z. Ren and X. Fan, "Particle swarm optimization with adaptive mutation and its application research in tuning of PID parameters," in Proc. 1st International Symposium on Systems and Control in Aerospace and Astronautics, pp. 990-994, 2006.
- [20] H. Wang, Y. Liu C. H. Li, and S. Y. Zeng, "A Hybrid Particle swarm algorithm with Cauchy Mutation," IEEE Swarm Intelligence Symposium, Honolulu, Hawaii, pp. 356-360, 2007.
- [21] Pant, M. Thangaraj, R. Abraham, A. , "Particle swarm optimization using adaptive mutation," in Proc. 19th International Conference on Database and Expert Systems Application, pp. 519-523, 2008.
- [22] Jun Tang, and, X. Zhao, "A Hybrid Particle Swarm Optimization with Adaptive Local Search", journal of networks, Vol. 5, No.4, April 2010.
- [23] Fatih Ta?getiren M and Yun-Chia Liang, "A Binary Particle Swarm Optimization Algorithm for Lot Sizing Problem", Journal of Economic and Social Research, Vol.5 No.2, pp. 1-20, 2004.
- [24] C. Li, S. Yang and I. A. Korejo. "An Adaptive Mutation operator for Particle Swarm". Proceedings of the 2008 UK Workshop on Computational Intelligence, pp. 165-170, 2008.
- [25] H. Wang, Y. Liu C. H. Li, and S. Y. Zeng, "A hybrid particle swarm algorithm with Cauchy mutation," IEEE Swarm Intelligence Symposium, Honolulu, Hawaii, pp. 356-360, 2007.
- [26] I. Griffin, "On-line PID Controller Tuning using Genetic Algorithms", MSc. Thesis School of Electronic Engineering Dublin City University, 2003.