Reason - A Technical Journal
ISSN 2277–1654
Volume - XV ● 2016

COOPERATIVE MULTI SWARM OPTIMIZATION WITH AN INTELLIGENT BROADCASTER FOR PID CONTROLLER DESIGN

P. Agarwalla[1] and S.Mukhopadhyay[2]

[1]Heritage Institute of Technology, Kolkata, India. Email: prativa.agarwalla@heritageit.edu
[2]Institute of Radio Physics & Electronics, Kolkata, India. Email: sumitra.mu@gmail.com

Paper received on: April 25, 2017, accepted after revision on: July 23, 2017
DOI: 10.21843/reas/2016/70-79/158778

Abstract: Particle swarm optimization (PSO) is a very efficient optimization tool for solving many scientific and engineering problems. In this paper, an intelligent broadcaster controlled co-operative multi-swarm PSO (IBC-MPSO) has been proposed which improves the fitness and robustness of the PSO technique. The multi-swarm approach with a novel broadcasting mechanism provides diversification in the searching and the involvement of neighborhood operator improves the exploitation of searching of the swarm. The co-operative methodology along with an intelligent broadcaster as a whole achieves good accuracy of the optimization result for the numerical problems. The efficiency of IBC-MPSO optimization technique is comprehensively evaluated for standard popular benchmark optimization problems and compared with several state-of- the-arts PSO. Further, IBC-MPSO is applied for tuning the parameters of a PID controlled both for AVR system and DC motor based system. Result of the experiments illustrates the effectiveness of the IBC-MPSO technique.

Keywords: Particle Swarm Optimization; Diversity; PID Controller; AVR System; DC motor.

1. INTRODUCTION

PSO, proposed by Kennedy and Eberhart, is a very effective optimization technique to find out the global solutions for some complex problems. PSO is developed based on the social cooperative and competitive behavior of bird flocking [1]. During the last two decades, PSO has been successfully implemented to various industrial application, power and control system, clustering applications etc. For a good performance, PSO should have the ability for global search maintaining diversity and good convergence speed. To improve the searching process of PSO, several variants of PSO [2-7] have been developed by the researchers. Here, the performance of the PSO is enhanced using a multi swarm concept along with an intelligent broadcaster (IBC-MPSO) to control over it. The new method is then applied in the domain of optimization problem.

In this work, the total swarm is divided into groups and a broadcaster is present who broadcasts the modified version of the best result to the groups. The broadcaster not only chooses the best result among the groups but also modifies the result by its own intelligence. The new solution provided by the broadcaster helps them not to be trapped at local optima and the interval of broadcasting enhances the possibility for a search of new area. This dynamic nature of the swarm leads the total swarm towards the best solution. The proposed technique is significantly improving the optimum value of both the unimodal and multimodal benchmark functions. The robustness of the proposed algorithm is further evaluated by applying it for tuning of the parameters of a PID controller used in both the Automatic Voltage Regulator (AVR) system and DC motor based system.

The paper is structured as follows: Section.2 presents a general overview of the PSO based optimization techniques. Section.3 describes the new IBC-MPSO algorithm. Section.4 validates the performance of IBC-MPSO on standard benchmark problems and also its application on PID based control system is demonstrated. Finally Section.5 provides the conclusion.

2. REVIEW WORK

In PSO having inertia weight, *w,* the velocity and position updating of a particle *i* at t^th^ iteration are given as follows [1]:

$$v_i\left(t\right)=w*v_i\left(t-1\right)+c_1*rand*\left(p_{best\,i}-x_i\right)+c_2*rand*\left(g_{best}-x_i\right) \quad (1)$$

$$x_i\left(t\right)=x_i(t-1)+v_i\left(t\right) \quad (2)$$

where c_1 and c_2 are cognitive and social learning factor that represents the attraction of a particle towards own best p_{best} and towards the swarm's best g_{best} respectively. Shi and Eberhart [2] proposed a idea to decrease *w* linearly from 0.9 to 0.4 over the course of searching to have a balance between local search and global search. Subsequently, over the years, numerous variants of PSO came into play. Different topological structures have been developed [3], for example, the ring topology (RPSO), the von Neumann topology (VPSO) and so on [4]. Comprehensive learning PSO (CLPSO) proposed by Liang *et al.* [5] was also studied, where particles learn from different dimensions. In another variant of PSO, entitled as DMS-PSO [6] more emphasis is given to the neighborhood structure. To provide the diversity, multi-swarm concept is also implemented [7]. It can be observed that prevention of premature convergence while maintaining the fast-converging nature is still a challenging task in PSO research. Introducing a efficient diversity mechanism in the searching process may be very effective. So, in this proposed work, multi swarm concept is incorporated for maintaining diversity and the concept of broadcasting the best result has been introduced to avoid local trapping. The work is described in the following section in details.

3. CO-OPERATIVE PSO WITH AN INTELLIGENT BROADCASTER

First, the total particle is divided into groups to search the area. They are simultaneously competing with each other which help to achieve the better solutions. A broadcaster is present in the scheme and it continuously monitors the performance of the groups and declares the result after some intervals. Based on the result, broadcasted to the groups, particles are updating their performance. So, it prevents pre-mature convergence as well as helps to improve for the best. To encourage the winning group, the intelligent broadcaster takes a strategy to modify the best result of the groups and then conveys the result to the winning group. The broadcasting mechanism not only introduces diversity but also prevents pre-mature convergence. The schematic diagram of the proposed work is shown in Fig.1 and the key steps of the mechanism are described below.

Step1: First the whole swarm is divided into g number of groups. Same numbers of particles are present in each group and the assignment of a particle to a group is totally random.

Step2: After initialization, a group works independently following a group-strategy where the position and velocity update rule are same as Eq (1) and (2). Thus all the groups have its own $g^g_{best.}$.

Step3: After k^th^ interval, the best solution among the group which is termed as g_{best_group} is broadcasted to each particle by the broadcaster. Also at the very k^th^ iteration, particle of a group updates itself following its own best performance so far (p_{best}) and g^g_{fit}. The term g^g_{fit} is the defined as the solution which is more fitted between own group's g^g_{best} and the broadcasted $g^g_{best\_group}$. The rule of velocity and position updating of the particle corresponding to a group *g* at t^th^ iteration are given in Equation (3) and (4).

$$v_i^{\,g}(t) = w * v^g{}_i(t-1) + c_1 * rand * \left(p^g_{best\,i} - x_i^{\,g}\right) + c_2 * rand * \left(g^g{}_{fit} - x^g{}_i\right) \quad (3)$$

$$x_i^{\,g}(t) = x_i^{\,g}(t-1) + v_i^{\,g}(t) \quad (4)$$

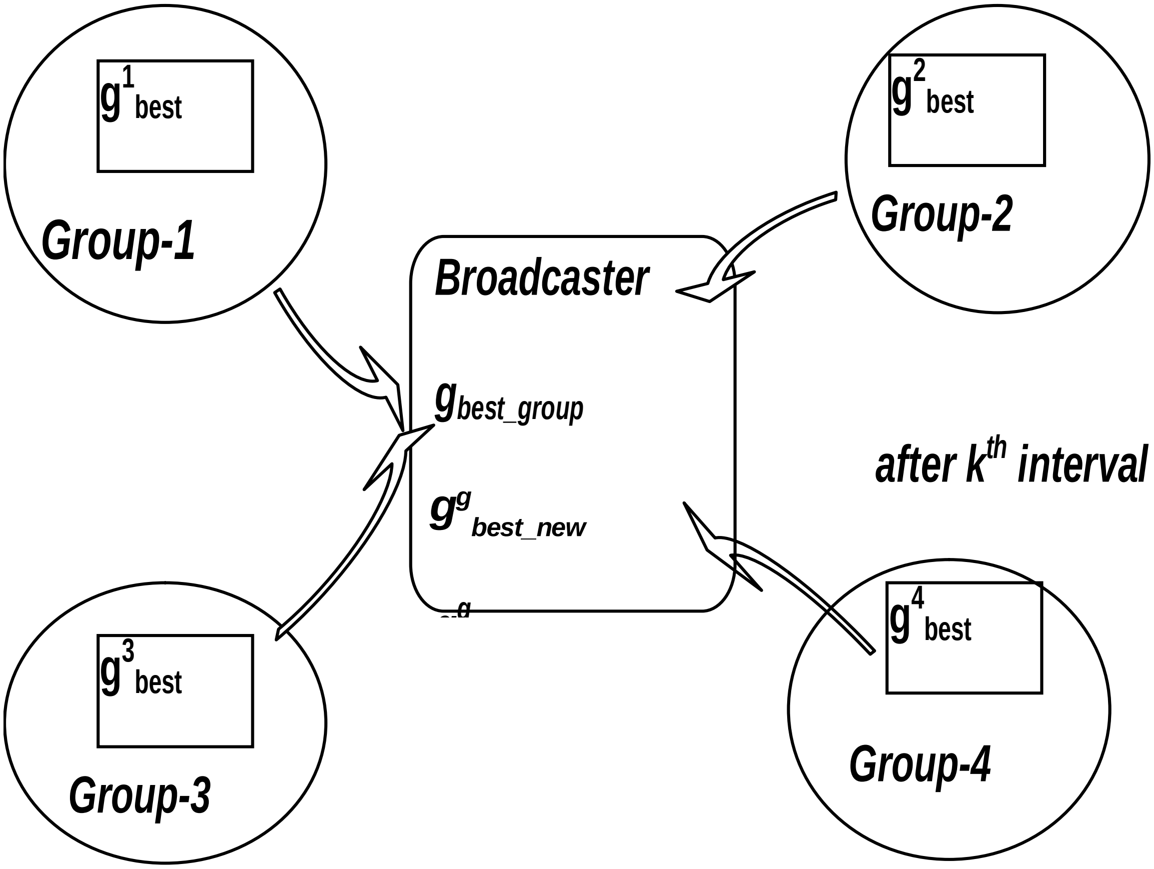


Fig. 1. Basic structure of the proposed IBC-MPSO algorithm.

Step4: The group, whose g^g_{best} is the minimum among all the groups, is termed as the winning group. For the winning group, the broadcaster encourages them and pays more attention. It modifies the result using a neighboring operator [3]. The generation of the modified solution is described in Eq. (5). Here ñ is considered as 0.5 and g^g_{best} is taken from any of the best solution of the groups other than the winning group.

$$g^g_{best\_new} = g_{best\_group} + \rho * (g^g_{best} - g_{best\_group}) \quad (5)$$

Thus, the individual group does not lose their creative feature which increases the diversity of the algorithm as well as the co-operative manner of the work enhances the quality of the solution. The algorithm of IBC-MPSO is given below.

Algorithm1: IBC-MPSO

```
1. Initialization.
(a) To divide the total swarm into g number of groups.
(b) To initialize the position of particles of all subgroups $x^g_k$
(c) To initialize the velocity of particles of all subgroups $v^g_k$
(d) For group=1, 2, 3...g do (for k= 1,2,…,N/g), do $p_k^g=x_k^g$
(e) $g^g$best=arg{min $f(x_k^g)$}
2. Termination check
    (a) If the termination criterion holds stop
    (b Else go to step 3
3. Set t=1(iteration counter)
    For group= 1, 2…g Do
        For k= 1, 2…N/4 Do
            (a) To update velocity according to eq. (1)
            (b) To update velocity according to eq. (2)
            (c) To evaluate fitness of the $k^{th}$ particle $f(x_k^g)$
            (d) If $f(x_k^g)$ is better than $f(p_k^g)$ then $p_k^g=x_k^g$
        End For
            (e) To update $gbest^g$ = arg{min $f(p_k^g)$
    End For
4. If the broadcasting criterion matches
    (a) gbest_group= arg{min f($gbest^g$)} and broadcast
    (b) To generate gbest_new as Eq. (5) for the winning group
    (c) To evaluate fitness of gbest_new f($gbest_{\_new}$)
    (d) If f($gbest_{\_new}$) is better than f($gbest_{\_group}$) then broadcast
5. Set t=t+1.
6. Go to step 3
```

4. EXPERIMENTAL RESULTS

4.1. Experimental Settings

To examine the performance of the proposed algorithm for optimization, first experiments are performed on six benchmark datasets which include both unimodal and multimodal functions having 30 dimensions. The parameter inertia weight, w is changed linearly from 0.9 to 0.4, and the constants c_1, c_2 are set at 1.49445. The maximum number of function evaluation (FES) is considered as 2×10^5 where the population size is 20 and the number of groups is considered as 4 for all the experiments. Experiments are carried out on a machine with a Core 2 Duo CPU running at 2.00 GHz with 4GB of RAM. To lessen statistical errors, each test is repeated independently for 30 times and the mean results are reported. Next, IBC-MPSO is applied for the tuning of PID controller based system. Here two widely used systems are used such as automatic voltage regulator **(**AVR) [8] and DC motor [9]. The results are illustrated in Table 1.

Table 1. Results of different algorithms on 30D benchmark functions (FE=2 lakh)

Function Name (optimum value)		**Search range**		**IBC-MPSO**	**GPSO**	**VPSO**	**RPSO**	**CLPSO**	**DMS-PSO**
f_1	**Sphere (0.00)**	-100, 100	Mean	**7.77E-62**	7.083E□53	1.90-E-38	5.58E-29	1.39E-27	1.95E□54
			Dev	**1.52E-61**	1.710E□52	3.98-E-38	1.42E-28	2.05E-27	8.43E□54
f_2	**Schwefel's (-12569.5)**	-10, 10	Mean	**-84.33E+02**	-13.26 E0+2	-26.92 E+02	-28.93E+02	-11.84E-00	-29.41E+02
			Dev	303.25E-00	332.67 E-00	520.28 E-00	343.96 E-00	**36.13E-00**	36.13 E-00
f_3	**Rosenbrock (0.00)**	-30, 30	Mean	**13.36E-00**	25.51 E-00	29.54 E-00	20.57 E-00	16.95E-00	16.95 E-00
			Dev	**4.59E-00**	25.86 E-00	24.65 E-00	12.45 E-00	12.79E-00	12.79E-00
f_4	**Rastrigin (0.00)**	-5.12 ,5.12	Mean	24.24E-00	25.24 E-00	29.19 E-00	38.76 E-00	**2.44E-14**	27.84 E-00
			Dev	51.01E-00	5.20 E-00	9.65 E-00	8.63 E-00	**5.98E-14**	7.56E-00
f_5	**Ackley (0.00)**	-32, 32	Mean	**9.47E-15**	1.10E-14	1.51E-14	2.664E-14	2.48E-14	2.48E-14
			Dev	1.87E-15	**2.27E-15**	4.10E-15	5.445E-14	4.18E-15	1.791E-15
f_6	**Griewank (0.00)**	-600, 600	Mean	**1.82E-15**	1.64E-02	2.40E-02	8.169E-03	2.01E-14	2.00E-14
			Dev	**2.72E-15**	1.69E-02	2.25E-02	1.780E-02	8.67E-14	1.59E-02

4.2. Results of Optimization on Standard Benchmark Datasets

To validate the efficiency of the proposed IBC-MPSO for optimization, the results obtained by IBC-MPSO for the standard benchmark functions [5] are compared with five different PSO variants, including GPSO [2], the RPSO [4], the VPSO [4], DMS-PSO [6] and CLPSO [5]. Those PSO variants used for the comparison are representative and well-performed PSO algorithms. The optimum value, search range and the results for 2 Lakh FEs are reported in the Table1. The log transformed value of the mean results for those benchmark functions along with the results obtained by all other PSO variants are plotted in the bar charts of Fig.2. Result of IBC-MPSO for unimodal functions (f_1, f_2) show that IBC-MPSO is better than all other variants of PSO. For the multimodal functions, it is very difficult to locate the minimum as the particles are generally trapped to a region. The result of Table 1, for these multimodal functions (f_3-f_6), validates the efficiency of IBC-MPSO in terms of solution accuracy. The result of comparisons indicates that IBC-MPSO has achieved best mean result for the three test functions out of four multimodal functions which justifies the superiority of the proposed algorithm.

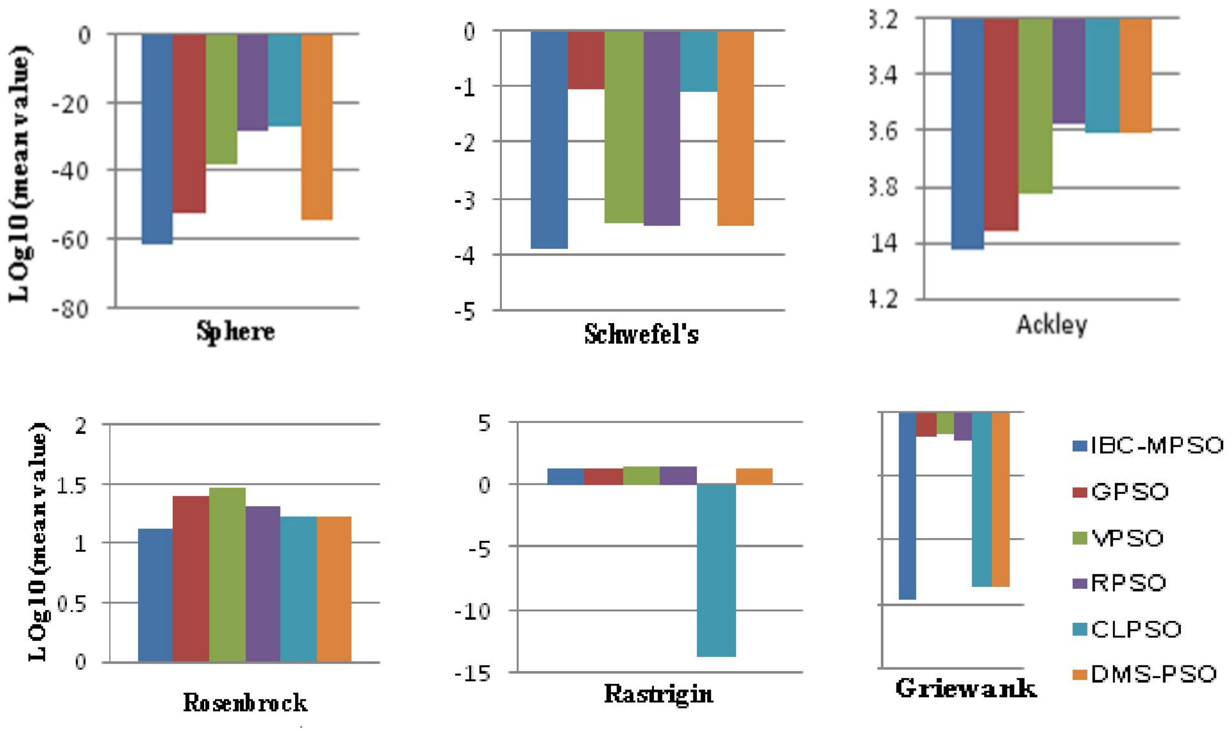


Fig. 2. Bar chart for the comparison of results for different benchmark optimization functions.

4.3. Tuning of PID Controller Based Systems

In this section, the performance IBC-MPSO is evaluated for industry oriented application. For which the problem of tuning the Proportional-integral-derivative (PID) controller is considered. PID controller is broadly used in the industry for controlling the system. PID controller with its three terms, covering modification for transient and steady-state response, delivers simple and efficient solution when applied to real world control problems. Despite the simple configuration, optimal gain tuning of the PID controllers is quite difficult. Tuning of the parameters of PID controller is a challenging task for which the proposed IBC-MPSO has employed. The transfer function of a PID controller having integer order is as follows.

$$C(s) = k_p + \frac{k_i}{s} + k_d s \qquad \textbf{(6)}$$

Where k_p, k_i, k_d are the proportional, derivative and integral constants respectively.

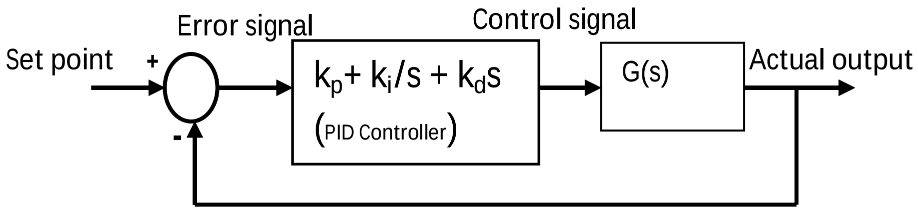


Fig. 3. PID control structure

The fundamental block diagram of a PID control system is shown in Fig. 3. The error signal, *E*(*s*), is used to generate the proportional, integral, and derivative actions. The resulting signal forms the control signal which is applied to the plant. This process will be continued until steady-state error is achieved. The control action can be evaluated using the rise time (t_r), settling time (t_s), overshoot (M_p) and the steady state error (E_{ss}). The performance measurer which is used as optimization objective is *W (K)* [9]. The expression of *W (K)* is as follows. Here K [k_p, k_i, k_d] and â are the constant and weighting factor respectively.

Case Study-1: Tuning of PID Controller used in AVR system: The proposed algorithm is used for tuning of two different systems. The first one is a PID controlled high-order practical automatic voltage regulator system. An AVR system holds the terminal voltage value of a synchronous generator at a particular level [8] and the block diagram of the system with a PID controller is given in Fig. 4. Step response analysis is performed for the AVR system with and without PID controller and the result is shown in Fig. 5 (a). The parameters of the controller are kept within the range of 0 to 1 and using the proposed IBC-MPSO the parameters are calculated. IBC-MPSO is able to find good values of the parameters efficiently as the graph of PID controlled output validates a good stable system. The result of the simulation is summarized in the Table 2 where the time domain specifications of the response for 200 generations are given for two different values of β. The comparison for rise time and settling time for PID controlled system (β=1) is given in the Fig.5 (b). The AVR system having the designed PID controller works better compared to original AVR system.

$$W(K) = (1-e^{-\beta}).(M_p + E_{ss}) + e^{-\beta}.(t_s - t_r) \qquad \textbf{(7)}$$

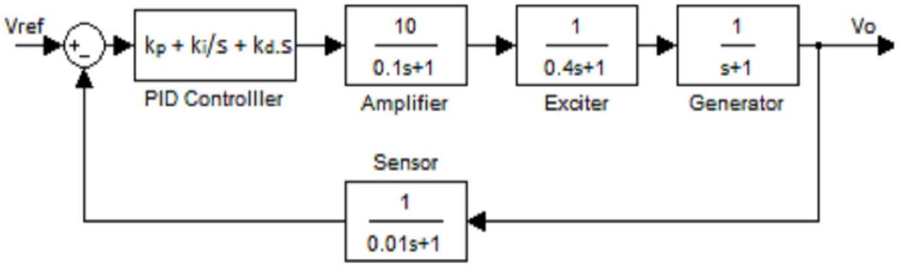
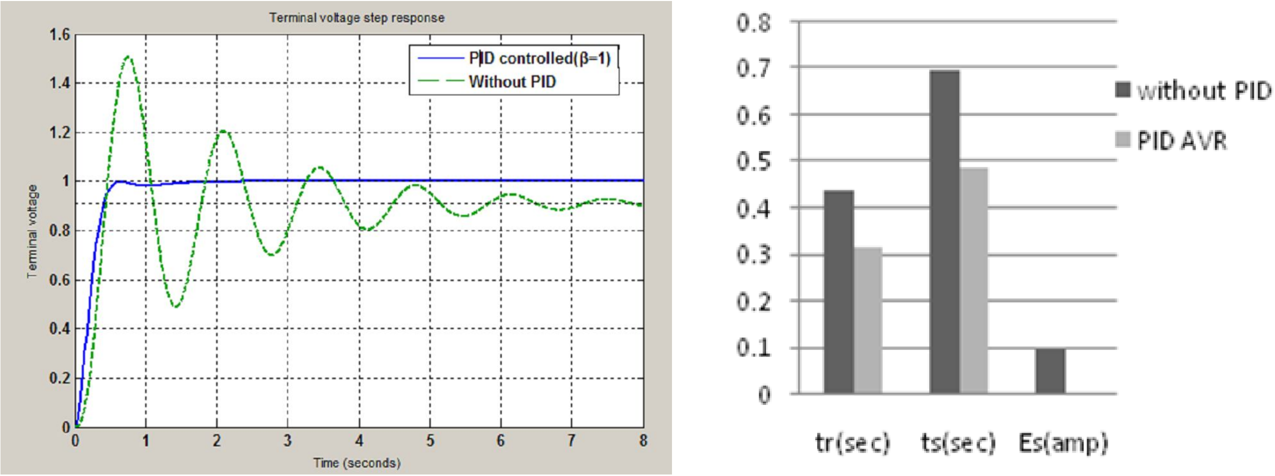


Fig. 4. Block diagram representation of a PID controlled AVR system.

Table 2. Result of response for PID controlled AVR system

β	k_p	k_d	K_i	t_r(sec)	t_s(sec)	Mp(%)	Ess
0.7	0.61	0.201	0.414	0.319	0.49	0.44	0
1	0.59 1	0.196	0.405	0.329	0.514	0	0



(a) (b)

Fig. 5. (a) Step response of the AVR system with and without controller (b) Bar chart to compare the performance of the PID controllerd system for AVR system

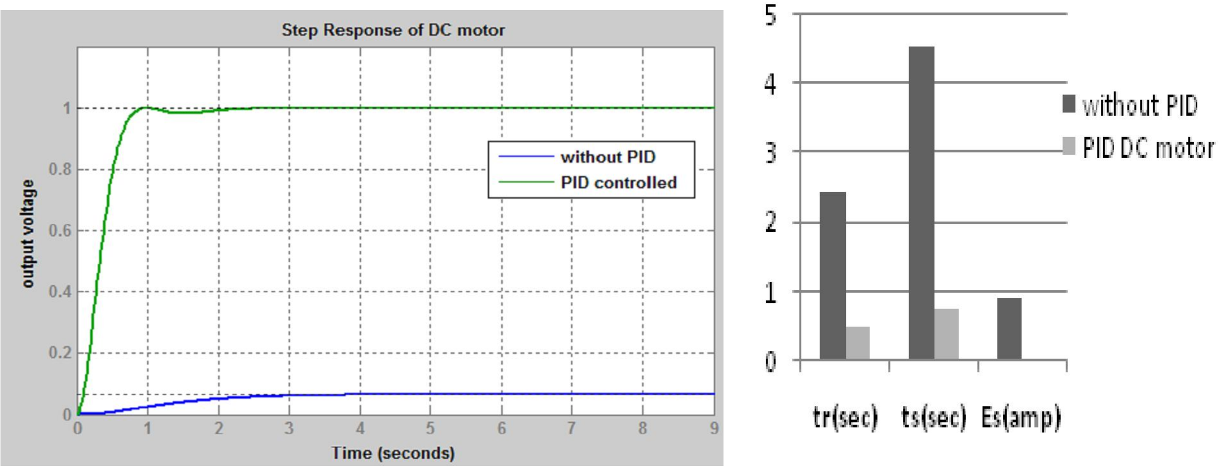
Case Study-2: Tuning of PID Controller for Controlling the speed of DC Motor: Here another plant which is adopted for the evaluation purpose is a DC motor model [9] which is a third order system written as Eq. (8). IBC-MPSO is now applied for controlling the speed of DC motor. The step response of the DC motor system using a PID controller is plotted in Fig. 6(a). The response of the system without a controller is also plotted in the same graph.

The result of the simulation for 200 generations for two different values of â is summarized in the Table 3. The controlled output has a high gain value compared to the uncontrolled system and the response time of the controlled system is also better. The steady state error is also very less. The comparison for rise time and the settling time is given in the Fig. 6(b) as a bar chart form. The graph shows the time domain specification of the PID controlled (β=1) DC motor is better compared to the non controlled DC motor. The controlled output has a high gain value compared to the uncontrolled system and the response time of controlled system is also better. The steady state error is also very less. The comparison for rise time and the settling time is given in the Fig 6(b) as a bar chart form. The graph shows that the time domain specification of the PID controlled (β=1) DC motor is better compared to the non controlled DC motor.

$$G(s) = \frac{1}{s^3 + 9s^2 + 23s + 15} \tag{8}$$

Table 3. Result of response for PID controlled DC motor

β	K_p	K_d	k_i	t_r(sec)	t_s(sec)	Mp (%)	Ess
1	60.23	19.34	40.9 6	0.50	0.78	0	3. 32E-03
0. 7	59.77	19.01	40.5 7	0.51	0.79	0	2.53E-03



(a) (b)

Fig. 6. (a) Step response of the DC motor (b) Bar chart for the comparison of performance of the PID controllerd system for DC motor

5. CONCLUSION

In this paper, an intelligent broadcaster controlled co-operative multi swarm PSO (IBC-MPSO) has been proposed to find more accurate and fast converging solution for complex problems. A broadcaster conveys the better solution of the problem to the groups in a cooperative manner. The advantage of IBC-MPSO is experimentally verified for different benchmark problems. It works better compared to all other investigated PSO variants in many cases. IBC-MPSO is also applied for the tuning problem of PID controller efficiently which proves the robustness of the proposed algorithm. In future, work can be done on the sensitivity of optimization factors for finding the best optimal value of the objective. The time domain specification parameters can be further modified by updating the algorithm. Also, the above mentioned problem may be formulated as a multi-objective problem and a variant of PSO in multi-objective domain can be applied.

REFERENCES

[1] Eberhart, R. and Kennedy, J., A New Optimizer using Particle Swarm Theory, In: 6th International Symposium on Micro Machine Human Science, IEEE Press, pp.39-43, 1995

[2] Shi, Y. and Eberhart, R.C., Empirical Study of Particle Swarm Optimization, In: IEEE Congress on Evolutionary Computation, IEEE Press, pp. 1945–1950, 1999.

[3] Suganthan, P.N., Particle Swarm Optimizer with Neighborhood Operator, In: IEEE Congress on Evolutionary Computation, IEEE Press, pp. 1958–1962, 1999.

[4] Kennedy, J. and Mendes, R., Population Structure and Particle Swarm Performance, In: IEEE Congress on Evolutionary Computation, IEEE Press, pp. 1671–1676, 2002.

[5] Liang, J.J., Qin, A.K., Suganthan, P.N. and Baskar, S., Comprehensive Learning Particles Swarm Optimization for Global Optimization of Multimodal Functions, IEEE Trans. Evol. Comput, Vol.10, No.3, pp.281–295, 2006.

[6] Liang, J.J. and Suganthan, P.N., Dynamic Multiswarm Particle Swarm Optimizer, In: Swarm Intelligence Symposium, pp.124-129, 2005.

[7] Zhang, J. and Ding, X., A Multi-Swarm Self-Adaptive and Cooperative Particle Swarm Optimization, Engg Appl. of Artificial Intel, Vol.24, pp.958–967, 2008.

[8] Gaing, Z.L., A Particle Swarm Optimization for Optimum Design of PID Controller in AVR System, IEEE Trans. on Energy Conversion, Vol.19, No.2, pp.384-391, 2004.

[9] Solihin, M.I., Tack, L.F. and Kean, M.L., Tuning of PID Controller Using Particle Swarm optimization (PSO), In: International Conference on Advanced Science Engineering and Information Technology, Malaysia, pp.458-461, 2011.