Vol:1 Issue:1 ISSN 2278 - 215X

An Improved Hybrid Approach to Modify Global Best of PSO

Dr. Damanjeet Kaur, Assistant Professor, Department of Electrical Engineering,

UIET, Panjab University, Chandigarh

***Abstract*—In this paper, an improved hybrid approach is developed to find global minima of the problem. The proposed approach consists of population based approaches PSO and DE. In the proposed approach, the global best obtained from PSO is modified with the help of DE. When global best of PSO remains same for large number of iterations. Then a mutant vector corresponding to random particles is generated and diversity is increased so that better results can be obtained. In this paper, difference of more than two vectors is taken to increase more randomness. The proposed approach is tested on standard functions also and results are compared with PSO.**

***Index Terms*—DE, PSO, Global Best**

I. Introduction

PARTICLE Swarm Optimizations is population based approach. It was initially developed as an alternate to Genetic Algorithm (GA). GA is successful in giving optimal solution to problem but the convergence of the algorithm takes lot of time. The convergence is actually dependent on the fact that how finely the parameters are tuned. There is diversity in the population but parameters selection is really a rigorous work. There are some operators in GA which are applied in sequence to improve the solution of the problem. Because of longer convergence time, PSO came into existence. PSO is considered a solution to the problem. In few years PSO became very popular and has wide applications in various fields like biomedical, power systems, data clustering etc. It has advantage of simplicity and less number of parameters tuning. Otherwise also tuning of parameters is not that much required as in GA. It searches automatically for the optimum solution in the search space, and the involved search process is random.

In PSO like other evolutionary algorithms a population of individuals (particles) is generatedrandomly within the given search space of the variables. The initial random population (swarm) is random solution of the problem. To check the quality of solution, for each generated solution fitness is calculated and compared with other individuals. The fittest population is kept and different operators are generally applied in evolutionary algorithms to modify the obtained solution. While PSO does not have operators like crossover and mutation in GA. But individuals (particles) update themselves with the internal velocity. They also have memory, which is important to the algorithm.PSO has ability of maintaining balance between local minima and global minima which helps to find solution of the problem in particular direction. With passing time it has been observed now that PSO sometimes goes into local minima and converges very quickly in few iterations and stop searching better solutions. This problem of local minima occurs because of the lesser diversity in the population and during updation of velocity and position vector, some solutions fly from the search space. This problem of local tapping can be overcome if there is diversity in population. The biggest advantage of PSO is its ability to give direction of search in which solution lies. Due to this, in this paper, the diversity in population is introduced if swarm best obtained is not changing for a large number of iterations. To increase diversity, differential evolution is combined with the PSO.

In this paper, after given number of iterations if global best is not changing then the trial vectors are generatedand considered as part of the population which increases diversity in the particles and gives optimal solution.

The present paper is organized into four sections. Section I introduces particle swarm optimization; SectionII is about the algorithm of PSO and DE while Section III and IV deals with results and discussions and conclusions respectively.

II. Basic Algorithm of PSO and DE

PSO and DE are population based approaches. Like all population based approaches in these techniques, a population is generated and modified in each iteration to get better solution. The detailed algorithms of these are as given below:

A. Particle Swarm Optimization

In PSO, initially particles (individuals) of the swarm (population) are randomly generated refining their knowledge of the given search space. Each particle *p* in a PSO thus have a position (Pos_p^k) and velocity (Vel_p^{k+1})at k^{th} iteration which directs the flying of the particles. All of

Damanjeet Kaur is presently working as Assistant Professor in Electrical & Electronics Engineering, UIET, Panjab University,Chandigarh, India. (djkb14@rediffmail.com)

Vol.1 Issue 1 ISSN 2278-215X

particles have fitness values which are evaluated by the fitness function to be optimized, and one can find the optimal solution of the problem through the generation. In each generation, each particle in PSO traces a trajectory in the search space; constantly updating a velocity vector by way of two kinds of search memories. One is the particle's best memory, called *pbest*, and the other is the swarm's best memory, called *gbest*. After generations, the PSO can find the best solution according to the best solution memories based on the best solutions found so far by that particle as well as others in the population (swarm). The equations involved in PSO are as given below [1-2]:

$$Vel_p^{k+1} = \mathrm{int}(wVel_p^k + c_1 rand_1(pbest - Pos_p^k) + c_2\, rand_2(gbest - Pos_p^k) \qquad (1)$$

$$pos_p^{k+1} = pos_p^k + Vel_p^{k+1} \qquad (2)$$

where c_1 and c_2 are random number between 0 and 1; *w* provides balance between global and local explorations. *w* often decreasesfrom 0.9 to 0.4 during the iterations. It is generally set using the following equation:

$$w = w_{\max} - ((w_{\max} - w_{\min}) / k_{\max}) * k \qquad (3)$$

The above procedure is repeated till the search satisfies the termination condition.

B. Differential Evolution (DE)

Differential evolution algorithm is population based approach in which at iteration *G=1*, a population is generatedwithin the search space of variables. The fitness function is calculated for the generated population.In the next operation *G=G+1*, a mutant vector is produced by selecting random particles from the population at *G=1*as given below:

$$Mu\tan t\,vector = Particle_{4,G} + F(Particle_{1,G} - Particle_{2,G} - Particle_{3,G}) \qquad (4)$$

where*F*is constant having value between 0 and 1; $Particle_{1,G}$, $Particle_{2,G}$,$Particle_{3,G}$ and $Particle_{4,G}$ are four randomly selected particles from population at G^{th} particle.

In DE, trial vectors are generated by applying crossover between earlier generated population at G^{th} iteration and mutant vector. The trial vector is obtained by comparing the random number generated between 0 and 1 and crossover probability selecting a crossover probability (P_{cross}). If random number is greater than theP_{cross} then design variable of trail vector will be from mutant vector otherwise from G^{th} iteration generation.

In next step, fitness function is calculated for trial vectors, if fitness of trial vector is less than earlier fitness of *Pbest*. Then in population trial vectors will take the place of earlier population. The above procedure repeated till the search satisfies the termination condition. The termination condition may be maximum number of iterations or the convergence criteria set.

III. Algorithm of the Proposed Approach

The algorithm of the proposed approach is as given below:

1. A swarm of particle at initial iteration within the given search space of the variables. Each particle p has position Pos_p^k and velocity vel_p^k at k^{th} iteration which directs the flying of the particles.
2. Evaluate fitness function for the generated swarm.
3. Store *Pbest* and *Gbest* for each particle and swarm at each iteration.
4. Update position and velocity vector as given in equation (1) and (2) respectively.
5. If*Gbest* at *k+1* iteration is better than *Gbest* at *k*. Then replace it with new ones.
6. If *Gbest* is not changing for given number of iterations.
7. Generate a mutant vector as given in equation (4).
8. Generate trial vectors by combining *Pbest* and mutant vector by selecting P_{cross}.
9. Calculate fitness function for the generated trial vector.
10. If fitness of trial vector is better than *Pbest* of PSO. Replace the earlier particle with trial vector.
11. Calculate *Pbest* and *Gbest* for the updated swarm as given in PSO.

The above steps are repeated till maximum number of iterations is achieved.

IV. Results and Discussions

In this paper, the proposed approach is tested on standard functions, Ackley, Rosenberg, and Rastrigin. The results are also compared with PSO method also. The final values obtained using proposed approach and PSO are tabulated in Table I. The range of variables for various functions along with the dimensions of variables are tabulated in 2nd and 3rd column of the Table I. The results obtained for PSO and proposed approach are tabulated in 5th column of the Table I.

The results for Rastrigin, Ackely and Rosenbrock functions are shown in Fig 1, 2 and 3 respectively. In all the figures the *Gbest* v/s iterations are shown. At the top of fig name of function along with the dimensions of variables are displayed and finally best *Gbest* is also visible in the figures of various functions.

On comparing the results of PSO and proposed approach, it is clear from 5th column of Table I that the results obtained using proposed approach are fairly better than PSO.

Hence it can be concluded that the proposed approach introduces sufficient of randomness and gives optimal solution in less number of iterations.

TABLE I
COMPARISON OF PSO AND PROPOSED APPROACH FOR DIFFERENT FUNCTIONS

Method	Function	Range of variables	*D*	Value of Objective Function
PSO	Ackley	[-100 100]	30	19.867
Proposed		[-100 100]		$-8.817 * 10^{-16}$
PSO	Rosenberg	[-100 100]	10	3.415
Proposed		[-100 100]		0.000269
PSO	Rastrigin	[-100 100]		39.79
Proposed				0

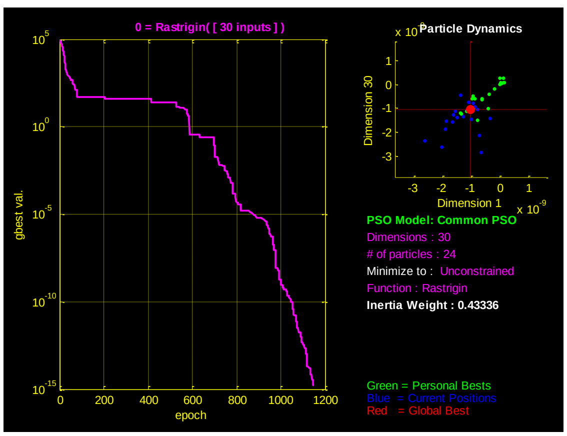

Fig 1 (a) Gbest v/s. Iterations Using PSO for Rastrigin Function

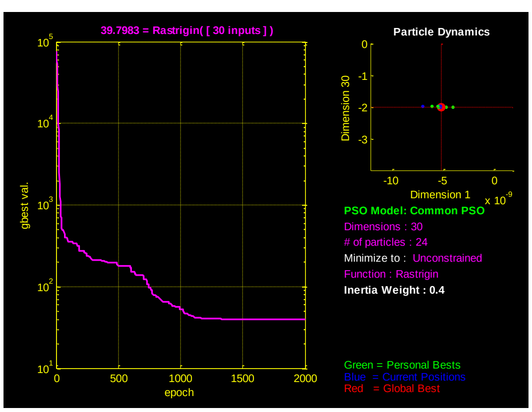

Fig 1 (b) Gbest v/s. Iterations Using Proposed Approach for Rastrigin Function

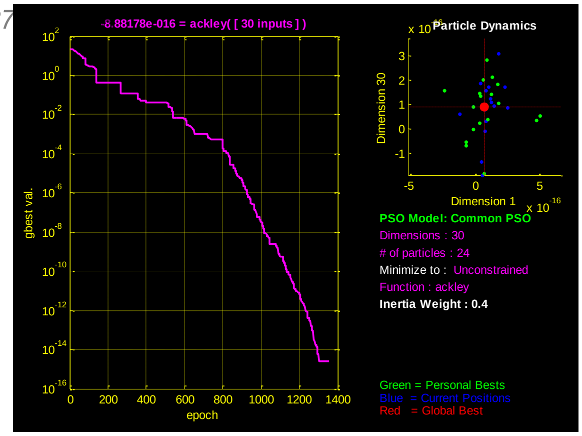

Fig2 (a) Gbest v/s. Iterations Using PSO for Ackley Function

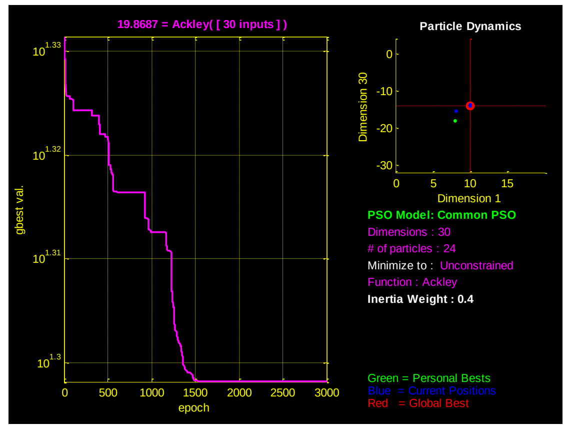

Fig 2 (b) Gbest v/s. Iterations Using PSO for Ackley Function

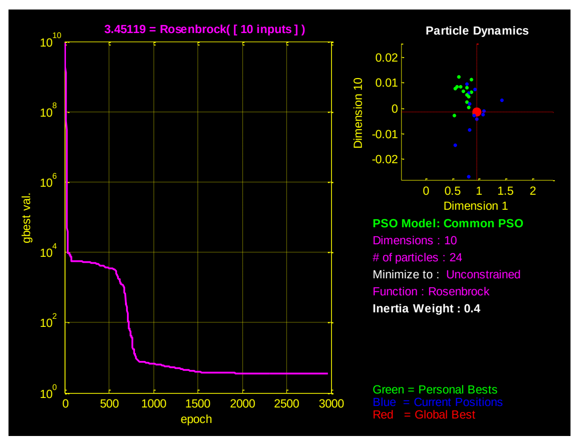

Fig 3 (a) *Gbest* v/s. Iterations Using PSO for Rosenbrock Function

e:1 ISSN 2278 - 215X

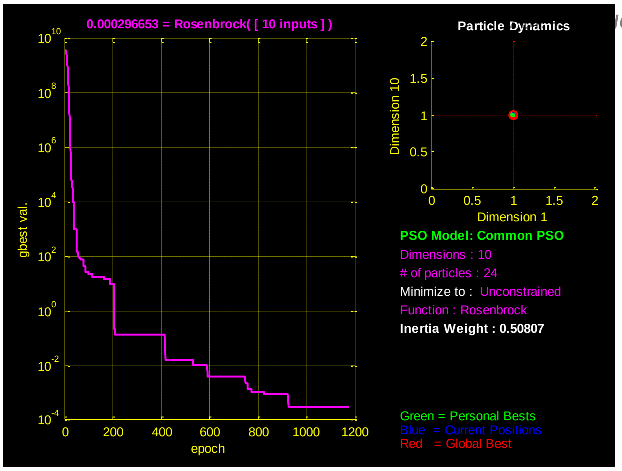

Fig 3 (b) *Gbest* v/s. Iterations Using Proposed Approach for Rosenbrock Function

V. Conclusions

In this paper, a hybrid approach to find global minima in PSO is developed. In this approach, the global best obtained in PSO is modified by creating randomness in search space with the help of DE. A mutant vector considering random particles is generated which provides an improved direction for the solution. The proposed approach is easy to implement and gives optimal solution of the problem.

References

[1] J. Kennedyand R. Eberhart, "Particle Swarm Optimization," *IEEE Proceedings on 1st International Conference on Neural Networks (Perth, Australia)*, 1942-1948, 1995.

[2] J. Kennedy and R. Eberhart, " A Discrete Binary Version of the Particle Swarm Algorithm," *IEEE Conference on Systems, Man, and Cybernetics, Orlando, FA*, 4104-4109, 1997.

[3] K. Price, R. Stornand J. Lampinen, Differential Evolution-A Practical Approach to Global Optimization, Springer, Berlin Heidelberg New York, 2005.