International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Enriched Particle Swarm Optimization Created By Varying Parameters for Optimization

Pooja Verma, Raghav Prasad Parouha

Abstract*: *Particle swarm optimization (PSO) is one of the most capable algorithms that reside to the swarm intelligence (SI) systems. Recently, it becomes very popular and renowned because of the easy implementation in complex/real life optimization problems. However, PSO has some observable drawbacks such as diversity maintenance, pre convergence and/or slow convergence speed. The ultimate success of PSO depends on the velocity update of the particles. Velocity has a significant dependence on its multiplied coefficient like inertia weight and acceleration factors. To increase the ability of PSO, this paper introduced an enriched PSO (namely ePSO), to solve hard optimization problems more precisely, efficiently and reliably. In ePSO novel gradually decreased inertia weight (as an alternative of a fixed constant value) and new gradually decreased and/or increased acceleration factors (meant for cognitive and social modules) is introduced. Proposed ePSO is used to solve four well known typical unconstrained benchmark functions and four complex unconstrained real life problems. The overall observation shows that proposed new algorithm ePSO is fitter than the compared algorithms significantly and statistically. Moreover, the convergence accuracy and speed of ePSO are also improved effectively.

Keywords*: *Unconstrained optimization; Particle swarm optimization; Inertia weight; Acceleration coefficients.

I. INTRODUCTION

Recently, nature inspired (NI) optimization algorithms introduced for extremely nonlinear and complex optimization problems. This is because of their flexibility, capability of global search, avoidance of local optima and derivation free mechanisms. These methods entirely depends on the concepts of animals and insects behaviors that are sustained in the environment, so that they are also named as algorithm of swarm intelligence (SI) which includes PSO (particle swarm optimization) [1], ABC (artificial bee colony) [2], ACO (ant colony optimization) [3], FA (firefly algorithm) [4], KH (krill herd) [5], BA (bat algorithm) [6], CS (cuckoo search) [7], AAA (artificial algae algorithm) [8], TSA (tree seed algorithm) [9], GWO (grey wolf optimizer) [10], SSA (social spider algorithm) [11], MFA (moth flame algorithm) [12], SSO (salp swarm optimizer) [13], WOA (whale optimization algorithm) [14], DEA (dolphin echolocation algorithm) [15], CSO (cat swarm optimizer) [16] and LOA (lion optimization algorithm) [17].

Currently, among the several SI algorithms, PSO and its kinds continuously are being used to deal with problems that are very complicated in the area of optimization. Also, it resolves numerous practical problems of the various fields [18–24], because of its advantages like easy implementation, few parameters to be controlled and rapid convergence. Although PSO endure from premature/slow convergence and trapped in local optima problem. On the way to overcome such problems, a number of alternatives of PSO were anticipated in the literature.

Shi and Eberhart [25] upgrade the implementation of the standard PSO via inertia weight which is linearly time varying that may effective to solve optimization problems. A new scheme to upgrade velocity of particles of standard PSO given by Liang et al. [26], which increase early convergence and swarm's diversity. Orthogonal culture strategy was offered to update the position/velocity of the PSO for better diversity and direction of particle enhancements by Zhan et al. [27]. Xinchao [28] updated a system which includes novel velocity that is determined by perturbed global best and outcome of which is the avoidance of swarm's diversity loss. In order to investigate to a greater extent proposed algorithm by Wang et al. [29] not only improve effectiveness of traditional PSO, but also functional altered velocity informs process to avoid early convergence in standard PSO. Tsoulos [30] transformed pattern of velocity update by adding some similarity check, local search and stopping rule.

A novel PSO (HPSO-TVAC) introduced by Ratnaweera et al. [31] where social and cognitive component of the PSO were controlled by newly generated time varying factors. Its results give some favorable results to optimization problems. Yang et al. [32] changed inertia weight dynamically to update the velocity formula of the PSO. In [33] the recommended method applies detailed information of the whole neighborhoods for conducting the particle to fly in proper directions. To control particle's neighborhood in PSO topological arrangement is used in [34]. Lianget al. [35] used small neighborhood's information and dynamic multiple swarms to diversify the swarm. The FGLT-PSO (fusion global-local-topology) method based a new PSO was developed in [36].

Manuscript published on 30 September 2019
* Correspondence Author
Pooja Verma*, Department of Mathematics, Indira Gandhi National Tribal University, Amarkantak Madhya Pradesh – 484887, India. Email: pooja.v0903@gmail.com
Raghav Pasad Parouha, Department of Mathematics, Indira Gandhi National Tribal University, Amarkantak Madhya Pradesh – 484887, India. rparouha@gmail.com, +91-94071577

MTVPSO (a modified time-varying PSO) for solving nonconvex ELD (economic load dispatch) problems initiated by Parouha [37] which relies on novel time-varying acceleration and inertia weight coefficients and provide better results compared among other effective algorithms.

Observably, parameters of PSO are few and its parameter tuning is quite simple, which enhance the algorithm quality and has become a widely used method. The crucial parameter of affecting implementation of the PSO algorithm is inertia weight.

It makes particles encouraging to global exploration whenever inertia weight is large and particles tend to local exploitation when it is small. Also, the important parameter of PSO, the cognitive and social improves particle's ability of learning optimal solution individually and population optimal solution and hence global and local search of algorithm is balanced.

Inspired by the above fact (advantages, disadvantages and parameter influences of PSO) and effective literature survey, we put forward an enriched PSO (termed as ePSO) algorithm, that have new gradually decreased inertia weight factor and novel gradually decreased and/or increased acceleration coefficients (cognitive and social parameter). Present algorithm proposed the characteristics of novel gradually varying parameters i.e. inertia weight, cognitive and social parameter makes ePSO algorithm more random that may enhance global and local search balance of algorithm as well as improvise the algorithm's ability to avoid fall into local optimal solution.

The rest part of this paper described as follows: In the section 2 an overview of the classical PSO is presented, section 3 covers the proposed technology. Results and discussion are given in section 4, and at last impact of this paper along with few future research advices discussed in section 5

II. PSO OVERVIEW

Initially, Particle swarm optimization (PSO) was very first invented Russell Eberhart and James Kennedy [1] in 1995. The PSO algorithm contains basically two main procedures in which first and second procedure is to update velocity and position update respectively. The population of particles randomized in *j*-dimensional search space in starting stage. A particle '*i*' has a velocity vector at iteration't' is given as $v_i^t = (v_{i1}^t, v_{i2}^t, \dots, v_{ij}^t)$ and $x_i^t = (x_{i1}^t, x_{i2}^t, \dots, x_{ij}^t)$ is a position vector. The prominent part of the solution is $p_{best_i}^t = (p_{best_{i1}}^t, p_{best_{i2}}^t, \dots, p_{best_{ij}}^t)$ which is best solution acquire by ith particle at current iteration (t). The g_{best}^t (global best) is the best $p_{best_i}^t$ among the entire particles. The updated velocity and position of particle 'i' will be formulated as

$$v_{ij}^{t+1} = wv_{ij}^t + c_1 r_1 \left(p_{best_{ij}}^t - x_{ij}^t\right) + c_2 r_2 \left(g_{best_j}^t - x_{ij}^t\right) \quad (1)$$

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \quad (2)$$

where, v_{ij}^t: t^{th} existing velocity and x_{ij}^t: t^{th} present position of the *i*th particle; t: t^{th} present iteration of the algorithm, $1 \le t \le T$; T: the maximum iteration number; i: *i*th particle of the swarm, $1 \le i \le NP$, NP: the swarm/population size. c_1 and c_2 are cognitive and social acceleration coefficient respectively. Usually c_1 and $c_2 \in [0, 2]$ [38]. r_1 and r_2 are two random uniformly distributed that belongs to range [0, 1] and the inertia weight which is denoted as w fixed to 1 in the standard PSO [1].

Obviously, among all optimization algorithms PSO is one of the modest and well-organized global optimization algorithms. From literature review, it is observed that the PSO are much affected by three parameters: w, c_1 and c_2. Suitable tuning of these parameters gives particles different behavior and improves the search quality.

III. PROPOSED TECHNOLOGY

PSO is a swarm intelligence (SI) searching approach which is entirely depends upon the population and hence renowned as a population based optimization algorithm. Motivated by the superiority (like simplicity, ease of implementation, efficiency of computation, robustness and having ability to find a global optimum) and inferiority (such as early convergence) as well as the requirement of the population based processes (like (a) in early stages: ability to improve particles to wander without gathering around the local optima upon the entire search space and (b) in next stages: to find optimum (near the global optima) solution efficiently).

In PSO, the inertia weight (w) plays the prominent role in controlling effectiveness of previous velocity increases of the particles and acceleration coefficients (c_1 and c_2) values are employ to covey that particles gets more affected either locally or globally

Encouraged by above observations (i.e. by the advantages, disadvantages, influences of the parameter, need of the population based optimization and effective survey of literature), an enriched particle swarm optimization (namely ePSO) presented and it is based on novel gradually varying (decreasing and/or increasing) parameters w, c_1 and c_2. The mathematical demonstration for proposed w, c_1 and c_2 are given as follows.

$$w = w_{max} + (w_{min} - w_{max}) \left(\frac{t}{T}\right)^2$$

$$c_1 = c_{1max} \left(\frac{c_{1min}}{c_{1max}}\right)^{\left(\frac{t}{T}\right)^2} ; \; c_2 = c_{2min} \left(\frac{c_{2max}}{c_{2min}}\right)^{\left(\frac{t}{T}\right)^2}$$

In ePSO, velocity (by using equation (3)) and position (by using equation (4)) of the particle 'i' are formulated as,

$$v_{ij}^{t+1} = \left(w_{max} + (w_{min} - w_{max}) \left(\frac{t}{T}\right)^2\right) v_{ij}^t + \left(c_{1max} \left(\frac{c_{1min}}{c_{1max}}\right)^{\left(\frac{t}{T}\right)^2}\right) r_1 \left(p_{best_{ij}}^t - x_{ij}^t\right) + \left(c_{2min} \left(\frac{c_{2max}}{c_{2min}}\right)^{\left(\frac{t}{T}\right)^2}\right) r_2 \left(g_{best_j}^t - x_{ij}^t\right) \quad (3)$$

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \quad (4)$$

The new enhanced implementation of presented algorithm (ePSO) contains (i) the proposed inertia weight w gradually reduced because of which local exploitation and global exploration abilities is maintained due to larger inertia weight facilitates a global exploration and a smaller inertia weight tends to a local exploitation and (ii) c_1 and c_2 gradually decreased and increased respectively improve the global search capability of PSO in early stage because of initially big and lesser value of c_1 and c_2, pushes particles to move in entire solution and then value of c_1 will decrease and of c_2

will increase (as iteration increases,) which pull the particles to the global solution. After doing an extensive experiments on unconstrained benchmark function as well as engineering design optimization using introduced ePSO the control parameter is decided as, $w_{max} = 0.9$ and $w_{min} = 0.4$, $c_{1max} = 2.5$ and $c_{1min} = 0.5$ and $C_{2max} = 2.5$ and $c_{2min} = 0.5$ which are used in the entire calculation. The influences and/or behavior of proposed w, c_1 and c_2 during iteration (search process) for ePSO are illustrated in Fig. 1.

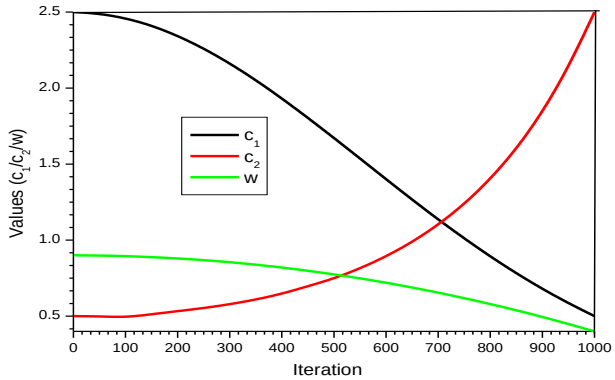


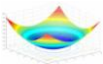
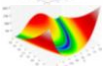
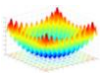
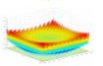
Fig. 1. Variation of parameters (w, c_1 and c_2) during iterations

Pseudo code of the projected ePSO.

```
For each particle
   Initialize particle and velocity randomly
END
Do
   For each particle
      Calculate function value
      Update P_best
   End
   Choose the g_best
   For each particle
      Calculate velocity using equation (3)
      Update position using equation   (4)
   End
While stopping criteria (like maximum iterations) is not
attained.
```

IV. RESULT AND DISCUSSION

This section discussed the performance of the ePSO which is tested on four typical unconstrained benchmark function namely Sphere, Rosenbrock function, Rastrigin and Griewank. The detail of these function is described below.

Name	Formula	Mesh in 2D	Optimal value
Sphere	$\sum_{i=1}^{n} x_i^2$		[-100, 100] $x^* = (0,0 \ldots, 0)$, $f(x^*) = 0$
Rosenbrock	$\sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$		[-32, 32] $x^* = (1,1 \ldots, 1)$, $f(x^*) = 0$
Rastrigin	$10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i)\right)$		[-5.12, 5.12] $x^* = (1,1 \ldots, 1)$, $f(x^*) = 0$
Griewank	$\sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) - 1$		[-600, 600] $x^* = (1,1 \ldots, 1)$, $f(x^*) = 0$

Further, four complex unconstrained real life problems of ePSO (URLPs) that is (i) URLP1: estimation of parameter for frequency modulated (FM) sound waves, (ii) URLP2: design of a gear train, (iii) URLP3: spread spectrum radar polly phase code design, and (iv) URLP4: Lennard-Jones potential problems (the details of listed RWA can be found in [28,57]) are selected to testify ePSO practicability on real life problems. Program is coded on C-Free 4.0 standard and conducted on Core-i7 Intel®, 2.20 GHz, 8 GB RAM computer. Every table in this paper the best result compared to all the algorithms highlighted as Italic bold standards. Results and discussion detail of numerous examinations are as follows.

A. On four unconstrained benchmark functions

Here, in this section ePSO is applied to analyze above listed 4 typical unconstrained benchmark functions and compared with traditional PSO, classical DE, PSODE (PSO-DE based hybrid algorithm) [39]. Population size (NP = 40), the functions dimension (j/d = 10) and stopping criterion (1500 iterations and 20 independent runs for each function) and which is same as in [39], for a fair comparison. The best objective function value of 4 typical unconstrained benchmark functions is analyzed numerically such as mean (Mn) and standard deviations (Std.) which is illustrated in Table 1.

Table 1. Results on four unconstrained benchmark problems

function	Criteria	ePSO	PSODE	PSO	DE
Sphere	Mn	***0.000e+000***	7.9601e-001	1.8051e-045	8.3857e-038
	Std.	***0.000e+000***	5.8400e-001	1.6596e-089	3.5160e-074
Rosenbrock	Mn	***2.017e-159***	2.2864e-051	1.6916e+000	2.8662 e+000
	Std.	***3.251e-198***	2.3016e-101	1.7038e+003	1.6892 e+000
Rastrigin	Mn	***1.071e-018***	3.1011e-002	3.3311e+000	2.0894e+000
	Std.	***1.251e-021***	2.8418e-004	2.9464e+000	1.4189e+000
Griewank	Mn	***0.000e+000***	0.8804e+000	3.8822 e-002	3.8496e-002
	Std.	***0.000e+000***	1.2502e+000	6.0670 e-004	7.8891e-004

From Table 1, it is perceived that the presentation of ePSO improved than other algorithms and have less standard deviations (Std.) on each benchmark functions which shows its stability. Concretely, the suggested ePSO is either superior or comparable.

Furthermore, comparison of convergence speed between ePSO with the other algorithms reported in [39] over a set of 4 considered typical unconstrained benchmark. The representation of graph (convergence graph) is depicted separately in Fig. 2(a-d). Moreover, performance (reported in [40]) of proposed ePSO with PSO, DE and PSODE are reported in Fig. 3. From figs. 2(a-d) and Fig. 3 it can be decide that the newly proposed ePSO has quick convergence capability which indicates that performance of ePSO is the superior (as it clearly visible in performance pi-charts that covers maximal area) than other methods respectively. Overall the proposed ePSO algorithm is a absolutely good performer in solving unconstrained function optimization.

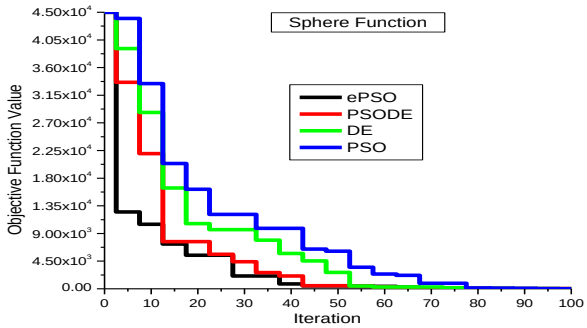


Fig. 2(a). Convergence of ePSO with others for Sphere

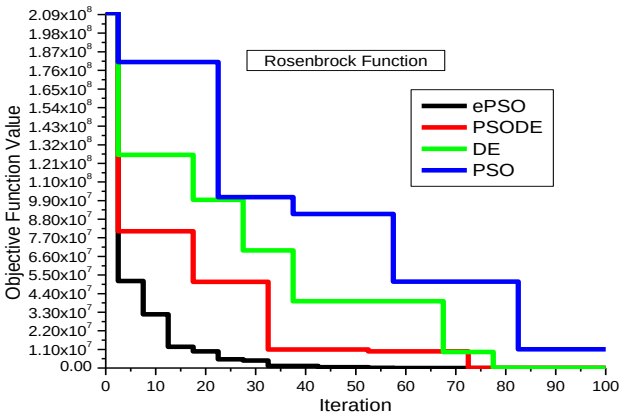


Fig. 2(b). Convergence of ePSO with others for Rosenbrock

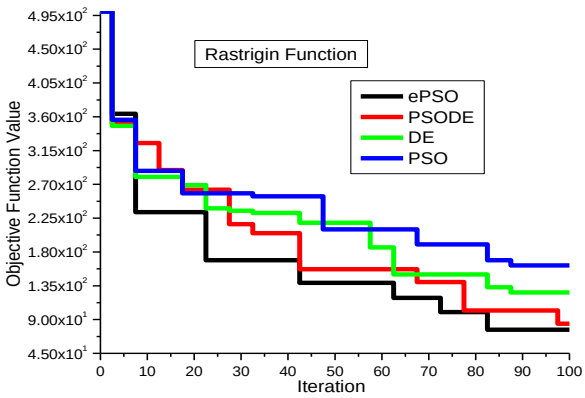


Fig. 2(c). Convergence of ePSO with others for Rastrigin

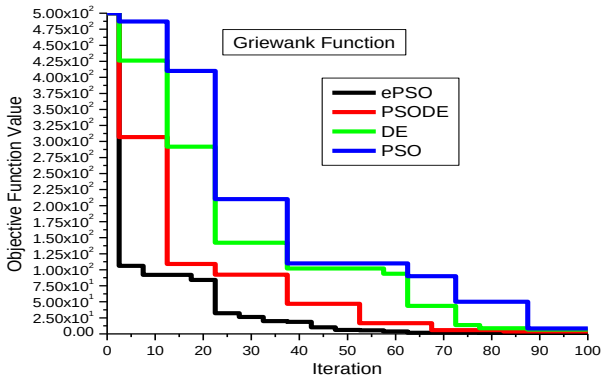


Fig. 2(d). Convergence of ePSO with others for Griewank

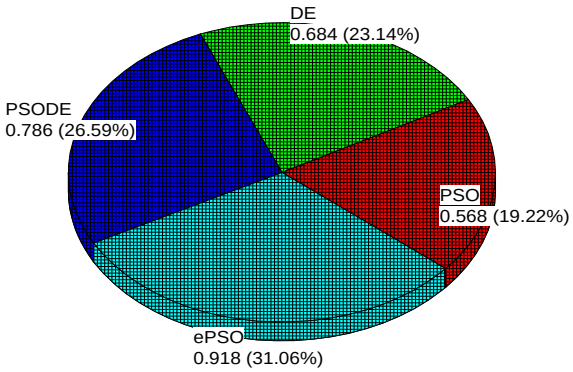


Fig.3. Performance evaluation on ePSO with other methods

B. On four unconstrained real life problems

Here in this section the four unconstrained real life problems (URLPs) as listed above are solved by applying proposed ePSO algorithm. The stopping criteria, independent runs and size of population kept absolutely same as compared algorithms for the sake of fair competition.

The experimental outcomes produced on URLPs i.e. URLP1- URLP4 by ePSO are given in Table 2-5 respectively and it is also compared with 13 others classical optimization algorithm namely DMSPSO [41], F-PSO [42], OLPSO [43], SLPSO [44], PSODDS [45], SL-PSO [46], HCLPSO [47], SSS-APSO [48], SopPSO [49], JADE [50], SaDE [51], CoDE [52] and CMA-ES [53]. The Tables 2 to 5 comprise best and mean values more than 30 independent runs with ranking i.e. RankB and RankM respectively, whereas in the Table 6 the average value of RankB and RankM of the respective algorithms is mentioned.

It is very obvious in Table 2 to 5 with respect of best value and mean value of the objective function the proposed ePSO algorithm produces either better of comparable results on URLPs., The average value of RankB (ranking on best value) and RankM (ranking on mean value) mentioned moreover the ranked followed by CoDE, ePSO, DMSPSO, PSODDS, SL-PSO, SopPSO, HCLPSO, CMA-ES, SSS-APSO, JADE, SaDE, SLPSO, OLPSO and F-PSO all have been illustrated in Table 6. On summing up all observation and calculations we can say that proposed ePSO algorithm play very prominent role to solve complicated and complex real life problems.

Table 2. Comparative results algorithms on URLP1

Algorithm	Best	RankB	Mean	RankM
ePSO	***0.00e+00***	***1***	4.03e+00	5
DMSPSO	***0.00e+00***	***1***	7.63e+00	6
F-PSO	8.42e+00	14	1.21e+01	11
OLPSO	1.45e−20	10	1.31e+01	12
SLPSO	4.74e−11	12	9.66e+00	8
PSODDS	***0.00e+00***	***1***	1.91e+01	12
SL-PSO	***0.00e+00***	***1***	1.05e+01	9

HCLPSO	***0.00e+00***	***1***	3.32e+00	4
SSS-APSO	***0.00e+00***	***1***	1.14e+01	10
SopPSO	***0.00e+00***	***1***	8.74e+00	7
JADE	1.05e−13	11	1.87e−01	2
SaDE	***0.00e+00***	***1***	***0.00e+00***	***1***
CoDE	***0.00e+00***	***1***	1.02e+00	3
CMA-ES	8.42e+00	13	2.03e+01	14

Table 3. Comparative results algorithms on URLP2

Algorithm	Best	RankB	Mean	RankM
ePSO	***0.00e+00***	***1***	***0.00e+00***	***1***
DMSPSO	***0.00e+00***	***1***	***0.00e+00***	***1***
F-PSO	7.51e−17	14	2.85e−14	14
OLPSO	***0.00e+00***	***1***	2.85e−21	8
SLPSO	***0.00e+00***	***1***	1.86e−29	6
PSODDS	***0.00e+00***	***1***	***0.00e+00***	***1***
SL-PSO	***0.00e+00***	***1***	***0.00e+00***	***1***
HCLPSO	4.75e−21	13	1.49e−15	12
SSS-APSO	1.32e−22	12	9.45e−15	13
SopPSO	***0.00e+00***	***1***	5.18e−25	7
JADE	***0.00e+00***	***1***	2.39e−17	11
SaDE	2.39e−29	11	7.67e−18	9
CoDE	***0.00e+00***	***1***	1.94e−17	10
CMA-ES	***0.00e+00***	***1***	5.91e-34	5

Table 4. Comparative results algorithms on URLP3

Algorithm	Best	RankB	Mean	RankM
ePSO	7. 99e−01	7	1.19e+0−0	8
DMSPSO	1.01e+00	9	1.24e+00	9
F-PSO	1.12e+00	13	1.59e+00	13
OLPSO	1.21e+00	14	1.69e+00	14
SLPSO	1.07e+00	11	1.26e+00	10
PSODDS	7.89e-01	6	1.14e+00	6
SL-PSO	5.57e-01	4	8.01e-01	2
HCLPSO	7.59e-01	5	1.08e+00	5
SSS-APSO	5.20e-01	2	9.37e-01	4
SopPSO	9.92e-01	8	1.39e+00	12
JADE	1.02e+00	10	1.17e+00	7
SaDE	1.11e+00	12	1.31e+00	11
CoDE	***5.00e-01***	***1***	***6.73e-01***	***1***
CMA-ES	5.36e-01	3	8.03e-01	3

Table 5. Comparative results algorithms on URLP4

Algorithm	Best	RankB	Mean	RankM
ePSO	−2.77e+01	6	***−2.69e+01***	***1***
DMSPSO	−2.84e+01	2	−2.52e+01	6
F-PSO	−1.47e+01	13	−1.34e+01	12
OLPSO	−1.42e+01	14	−1.00e+01	14
SLPSO	−2.73e+01	11	−1.80e+01	11
PSODDS	−2.72e+01	5	−1.99e+01	8
SL-PSO	−2.75e+01	10	−1.02e+01	13
HCLPSO	−2.84e+01	2	−2.67e+01	2
SSS-APSO	−2.84e+01	7	−2.64e+01	4
SopPSO	−2.84e+01	2	−2.58e+01	5
JADE	−2.53e+01	12	−2.30e+01	7
SaDE	−2.84e+01	9	−2.18e+01	8
CoDE	−2.84e+01	7	−2.66e+01	3
CMA-ES	***−2.84e+01***	***1***	−1.99e+01	10

Table 6. Average values of RankB and RankM on URLPs

ePSO	DMSPSO	F-PSO	OLPSO	SLPSO
3.75	4.375	13	10.875	8.75
PSODDS	SL-PSO	HCLPSO	SSS-APSO	SopPSO
5	5.125	5.5	6.625	5.375
JADE	SaDE	CoDE	CMA-ES	
7.625	7.75	***3.375***	6.25	

V. RESULT AND DISCUSSION

Since traditional PSO suffer from some major drawbacks therefore to overcome such drawbacks and to improve the ability of PSO, an enriched PSO (namely ePSO) introduced in this research paper based on decreasing and/or increasing parameters. The ePSO algorithm proposed novel gradually decreased inertia weight, gradually decreased cognitive and gradually increased acceleration coefficients in place of a constant value which used to be fixed. To evaluate effectiveness of the proposed ePSO, functions have been simulated by four typical unconstrained benchmark and the corresponding results compared with standard PSO, traditional DE and PSODE (hybridization of PSO & DE). Moreover, four real life problems are also solved by applying ePSO algorithm and therefore compared with 13 classical optimization algorithms to verify ePSO practicability on real life problems.

It is very clear by analysis of numerical and graphical results that ePSO is more accurately, reliably, effectively, efficiently and has the ability to avoid fall into local optimal solution is far better as compared to other algorithms. Concretely, the proposed ePSO based on gradually varying (increased and/or decreased) controlled parameters enhance the convergence speed as well as global/local search ability. It is benefitted for various optimization problems due to simple and easy implementation. The proposed novel controlled parameters (w, c_1 and c_2) behaviours construct the ePSO algorithm more random which may assist to neglect the problem of trapping into local optima of particle. The proposed algorithm ePSO can apply in solving multi-objective real world problem optimization problems in future.

REFERENCES

1. Kennedy, J. & Eberhart, R. (1995) "Particle swarm optimization", in proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4, pp. 1942-1948.

2. Karaboga, D. & Basturk, B. (2007) "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", J. global opt., vol. 39, pp. 459-471.
3. Dorigo, M., Maniezzo, V. & Colorni, A. (1996) "Ant system: optimization by a colony of cooperating agents", IEEE Trans. Sys. (Man, and Cybernetics, Part B Cybernetics), vol. 26, pp. 29-41.
4. Yang, X. S. (2009) "Firefly algorithms for multimodal optimization", International symposium on stochastic algorithms: Springer, pp. 169-78.
5. Gandomi, A. H. & Alavi, A. H. (2012) "Krill herd: a new bio-inspired optimization algorithm", Comm. in Nonlinear Sci. & Num. Simu., vol. 17, pp. 4831-4845.
6. Yang, X. S. (2010) "A new metaheuristic bat-inspired algorithm", Nature inspired cooperative strategies for optimization: Springer, pp. 65-74.
7. Gandomi, A. H., Yang, X. S. & Alavi, A. H. (2013) "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems", Eng. with comp., vol. 29, pp.17-35.
8. Uymaz, S. A., Tezel, G. & Yel, E. (2015) "Artificial algae algorithm (AAA) for nonlinear global optimization", Appl. Soft Comput., vol. 31, pp.153-171.
9. Kiran, M. S. (2015) "TSA: Tree-seed algorithm for continuous optimization", Exp. Syst. with App., vol. 42, pp. 6686-6698.
10. Mirjalili, S., Mirjalili, S. M. & Lewis, A. (2014) "Grey wolf optimizer", Adv. in eng. soft., vol. 69, pp. 46-61.
11. James, J. & Li, V. O. (2015) "A social spider algorithm for global optimization", App. Soft Comp., vol. 30, pp. 614-627.
12. Mirjalili, S. (2015) "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm", Know. Based Syst., vol. 89, pp. 228-249.
13. Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H. & Mirjalili, S. M. (2017) "Salp Swarm Algorithm: A bioinspired optimizer for engineering design problems", Advan. in Eng. Soft., vol. 114, pp. 163-191.
14. Mirjalili, S. & Lewis, A. (2016) "The whale optimization algorithm", Advan. in Eng. Soft., vol. 95, pp. 51-67.
15. Kaveh, A. & Farhoudi, N. (2013) "A new optimization method: dolphin echolocation", Advan. in Eng. Soft., vol. 59, pp. 53-70.
16. Chu, S. C., Tsai, P. W. & Pan, J. S. (2006) "Cat swarm optimization", Pacific Rim Internatonal Conference on Artificial Intelligence: Springer, pp. 854-858.
17. Yazdani, M. & Jolai, F. (2016) "Lion optimization algorithm (LOA): a nature inspired metaheuristic algorithm", J. of comput. design & eng., vol. 3, pp. 24-36.
18. Parouha, R. P. & Das, K. N. (2018) "Economic load dispatch using memory based differential evolution", Int. J. of Bio Ins. Comput., vol.11, pp.159-170.
19. Parouha, R. P. & Das, K. N. (2016) "DPD: An intelligent parallel hybrid algorithm for economic load dispatch problems with various practical constraints", Exp. Syst. with App., vol. 63, pp. 295–309.
20. Parouha, R. P. & Das, K. N. (2016) "A Robust Memory Based Hybrid Differential Evolution for Continuous Optimization Problem", Know. Based Syst., vol. 103, pp. 118-131.
21. Parouha, R. P. & Das, K. N. (2016) "A novel hybrid optimizer for solving Economic Load Dispatch Problem", Int. J. of Elec. Power & Energy Syst., vol. 78, pp. 108-126.
22. Parouha, R. P. & Das, K. N. (2015) "An Efficient Hybrid Technique for Numerical Optimization and Applications", Com. & Indus. Eng., vol. 83, pp. 193-216.
23. Das, K. N. & Parouha, R. P. (2015) "An ideal tri-population approach for unconstrained optimization and applications", App. Math. & Comput., vol. 256, pp. 666-701.
24. Parouha, R. P. & Das, K. N. (2014) "An Effective Hybrid DE with PSO for Constrained Engineering Design Problem", J. of Int. Acad. of Phy. Sci., vol. 18, pp. 01-15.
25. Shi, Y. & Eberhart, R. (1998) "A modified particle swarm optimizer", in: proceedings of IEEE International Conference on Evolutionary Computation, Anchorage, AK, pp. 69-73.
26. Liang, J. J., Qin, A. K., Suganthan, P. N. & Baskar, S. (2006) "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions", IEEE Trans. Evolut. Comput., vol. 10, pp. 281-295.
27. Zhan, Z. H., Zhang, J., Li, Y. & Shi, Y. H. (2011) "Orthogonal learning particle swarm optimization", IEEE Trans. Evolut. Comput., vol. 15, pp. 832-846.
28. Xinchao, Z. (2010) "A perturbed particle swarm algorithm for numerical optimization", Appl. Soft Comput., vol. 10, pp. 119-124.
29. Wang, Y., Li, B., Weise, T., Wang, J., Yuan, B. & Tian, Q. (2011) "Self adaptive learning based particle swarm optimization", Inf. Sci., vol. 181, pp. 4515-4538.
30. Tsoulos, I. G. & Stavrakoudis, A. (2010) "Enhancing PSO methods for global optimization", Appl. Math. Comput., vol. 216, pp. 2988-3001.
31. Ratnaweera, A., Halgamuge, S. K. & Watson, H. C. (2004) "Self organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients", IEEE Trans. Evolut. Comput., vol 8, pp. 240-255.
32. Yang, X., Yuan, J., Yuan, J. & Mao, H. (2007) "A modified particle swarm optimizer with dynamic adaptation", Appl. Math. Comput., vol. 189, pp. 1205-1213.
33. Mendes, R., Kennedy, J. & Neves, J. (2004) "The fully informed particle swarm: simpler, may be better", IEEE Trans. Evolut. Comput., vol. 8, pp. 204-210.
34. Kennedy, J. & Mendes, R. (2002) "Population structure and particle swarm performance", in: IEEE Congr. Evolut. Comput., Honolulu, pp. 1671–1676.
35. Liang, J. J. & Suganthan, P. N. (2005) "Dynamic multi-swarm particle swarm optimizer", in: Swarm Intelligence Symposium, California, pp. 124-129.
36. Beheshti, Z., Shamsuddin, S. M. & Sulaiman, S. (2014) "Fusion global local topology particle swarm optimization for global optimization problems", Math. Probl. Eng., vol. 14, pp. 1-19.
37. Parouha, R. P. (2019) "Non-convex/non-smooth economic load dispatch using modified time varying particle swarm optimization", Comput. Intell., https://doi.org/10.1111/coin.12210.
38. Huynh, D. C. & Dunnigan, M. W. (2010) "Parameter estimation of an induction machine using advanced particle swarm optimization algorithms", IET J. Elec. Power App., vol. 4, pp. 748-760.
39. Niu, B. & Li, L. (2008) "A novel PSO-DE-based hybrid algorithm for global optimization", in: Huang, D.-S., Wunsch II, D.C., Levine, D.S., Jo, K.-H. (eds.) ICIC 2008. LNCS (LNAI), vol. 5227, pp. 156–163.
40. Parouha, R. P. & Das, K. N. (2016) "A memory based differential evolution algorithm for unconstrained optimization", App. Soft Comput., vol. 38, pp. 501-517.
41. Zhao, S. Z., Liang, J. J., Suganthan, P. N. & Tasgetiren, M. F. (2008) "Dynamic multi-swarm particle optimizer with local search for large scale global optimization", proc.of Congr. on Evol. Comput., vol. 08, pp. 3845-3852.
42. Marco, A., Montes, d. O., Thomas, S., Mauro, B. & Marco, D. (2009) "Frankenstein's PSO: a composite particle swarm optimization algorithm", IEEE Trans. Evol. Comput., vol. 13, pp. 1120-1132.
43. Zhan, Z. H., Li, Y. & Shi, Y. H. (2011) "Orthogonal learning particle swarm optimization", IEEE Trans. Evol. Comput., vol. 16, pp. 832-847.
44. Li, C., Yang, S. & Nguyen, T. T. (2012) "A self-learning particle swarm optimizer for global optimization problems", IEEE Trans. Syst. Man Cybern. B: Cybern, vol. 42, pp. 627-646.
45. Jin, X., Liang, Y. Q., Tian, D. P. & Zhuang, F. Z. (2013) "Particle swarm optimization using dimension selection methods", Appl. Math. Comput., vol. 219, pp. 5185-5197.
46. Cheng, R. & Jin, Y. (2015) "A social learning particle swarm optimization algorithm for scalable optimization", Inf. Sci., vol. 291, pp. 43–60.
47. Lynn, N. & Suganthan, P. N. (2015) "Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation", Swarm & Evol. Comput., vol. 24, pp. 11–24.
48. Pornsing, C., Sodhi, M. S. & Lamond, B. F. (2016) "Novel self-adaptive particle swarm optimization methods", Soft Comput., vol. 20, pp. 3579–3593.
49. Xia, X. W., Xie, C.W., Wei, B., Hu, Z. B., Wang, B. J. & Jin, C. (2017) "Particle swarm optimization using multi-level adaptation and purposeful detection operators", Inf. Sci., vol. 385, pp. 174-195.
50. Zhang, J. & Sanderson, A. C. (2009) "JADE: adaptive differential evolution with optional external archive", IEEE Trans. Evol. Comput., vol. 13, pp. 945-958.
51. Qin, A. K., Huang, V. L. & Suganthan, P. N. (2009) "Differential evolution algorithm with strategy adaptation for global numerical optimization", IEEE Trans. Evol.Comput., vol. 13, pp. 398-417.
52. Wang,Y., Cai, Z. Z. & Zhang, Q. F. (2011) "Differential evolution with composite trial vector generation strategies and control parameters", IEEE Trans. Evol.Comput., vol. 15, pp. 55-66.
53. Hansen, N. & Ostermeier, A. (2001) "Completely derandomized self-adaptation in evolution strategies", Evol. Comput., vol. 9, pp. 159–195.

AUTHORS PROFILE

Pooja Verma, is currently research scholar at Department of Mathematics, Indira Gandhi National Tribal University, (A Central University Established by an Act of Parliament) Lalpur, Amarkantak, Dist.-Anuppur Madhya Pradesh - 484887 India. She did B. Sc. in electronics from Pt. Sambhunath Shukla University Shahdol (M.P.) in 2014 and M. Sc. In mathematics from St. Aloysius College (A+) Autonomous Jabalpur (M.P.) in the year of 2016. Currently she is perusing Ph. D. under the supervision of Dr. Raghav Prasad Parouha, in the area of Evolutionary Algorithms (EAs) like Differential Evolution (DE), Genetic Algorithm (GA) and Nature Inspired (NI) Techniques such as Particle Swarm Optimization Techniques (PSO), Artificial Bee Colony (ABC) etc. as well as Hybrid Algorithms.
Email Id: pooja.v0903@gmail.com

Raghav Prasad Parouha, is currently Assistant Professor at Department of Mathematics, Indira Gandhi National Tribal University, (A Central University Established by an Act of Parliament) Lalpur, Amarkantak, Dist.-Anuppur Madhya Pradesh - 484887 India. He did Ph. D. from National Institute of Technology (NIT), Silchar, Assam India in August 2015 and M. Sc. from Govt. Model Science College (Auto.) 'A+', Jabalpur, (M.P.) His area of research interests related to Computational Intelligence including Algorithm Design, Real World Application etc. He has received Young Scientist Award (YSA) by International Academy of Physical Sciences (IAPS), Allahabad (U.P.), India. He has about 25 repute national and international publications and attended about 30 and organized around 05 workshops/seminars etc.
Email Id: rparouha@gmail.com
https://orcid.org/0000-0002-8076-454X
https://scholar.google.co.in/citations?user=talhHfQAAAJ&hl=en