Particle Swarm Optimization: A Study of Variants and Their Applications

Ashok Kumar
Deptt. of CSE,
Dr. APJ Abdul Kalam Technical University, Lucknow,
India

Brajesh Kumar Singh
Deptt. of CSE,
RBS Engineering Technical Campus,
Agra, India

B. D. K. Patro
Deptt. of CSE,
RBS Engineering Technical Campus,
Agra, India

ABSTRACT

In order to improve the performance of PSO algorithm, number of its variants has been made. This paper presents detail overview of the basic concepts of PSO and its variants. Many variants of PSO have been developed due to improved speed of convergence and quality of solution found by Researchers. The Applications of PSO in Complex Environments is discussed. Modifications, both those already developed, and promising future application areas are reviewed. Observation and review of 117 related studies in the period between 1995 and 2015 on different variants of PSO algorithms are discussed along with their advantages and disadvantages.

Keywords

Particle Swarm Optimization (PSO), Basic PSO, Modification PSO, Bird Flocking Evolutionary Optimization, Biologically Inspired Computational Search.

1. INTRODUCTION

Particle Swarm Optimization (PSO) is a computational search and optimization method. The behavior of this method is biologically inspired. It is developed by Eberhart and Kennedy [1, 2] in 1995. Researchers are using various optimization methods [49]. Theory of particle swarm optimization (PSO) has been growing rapidly too. Many Researchers are using PSO in their optimization work. PSO has been used by many applications of several problems. The algorithm of PSO was developed by identifying the behavior of animal's societies which work in a group of swarm, such as bird flocking and fish schooling. Generally, a flock of animals or birds has no leader. They find their food by random follow one of the members of the group. These members should be in closest position with a food source. This closest position is also known as potential solution for that flock. The members of flocks communicate to the members who already have a better situation to achieve their best condition in terms of position. Members those having better condition inform to other members of flocks. And the other members move immediately to that place. This happens repeatedly until the best conditions or a food source discovered. Study on the development of PSO [50] is necessary to know how far its development, advantages and disadvantages and how to settle a problem [3, 4] by using this method. This paper describes modifications, advantages and disadvantages. Each modification of PSO makes a conclusion from those. In **section 2** describes about basic PSO Algorithm, basic Parameters of PSO, and its modifications. **Section 3** discussed various variants of PSO. **Section 4** focused on applications of PSO in complex environments. In **section 5** describes observation and Analysis and the last section describe about Conclusion and future work.

2. REVIEW OF LITERATURE

2.1 Basic PSO Algorithm

Search algorithms have the ability of exploration. Through exploration different regions of the search space are identified to locate optimum solution. Next need to refine a candidate solution[5] is explored .the exploitation has the ability to refine solutions and concentrate the search around a promising area .due to exploration and exploitation, the particles of the swarm fly through hyperspace and have two essential reasoning capabilities. First capability of particle is memory of their own best position also known as ***local best (pbest)*** and second capability is the global or their neighborhood's best also known as *global* ***best (gbest)***.Position of the particle is influenced by velocity.

The process for implementing the PSO is as follows [6]:

Step 1: Initialize a population of particles with random positions and velocities on ***d-*** dimensions in the problem space.

Step 2: evaluate the desired optimization fitness function for each particle in ***d-*** variables.

Step 3: Compare particle's pbest with particle's fitness evaluated value. This current value may be better than pbest in this case set pbest value equal to the current value and pbest location equal to the current location in search space.

Step 4: Compare population's overall previous best with the fitness evaluated value. If this current value is better than gbest, reset gbest to the current particle's value.

Step 5: Change the velocity and position of the particle according to equations (1) and (2) respectively:

$V_{id} = V_{id} + c_1 * \text{rand}(\,) * (P_{id} - X_{id}) + c_2 * \text{rand}(\,) * (P_{gd} - X_{id})$ (1)

$X_{id} = X_{id} + V_{id}$

(2)

Step 6: Loop to **step2** until a criterion is met, usually a sufficiently good fitness or a maximum number of iterations.

Where, V_{id} = Velocity of ith Particle in d-dimension.

P_{id} = local best position of ith Particle in d-dimension.

X_{id} = Current position of ith Particle in d-dimension.

P_{gd} = global best position of ith Particle in d-dimension.

c_1, c_2 are acceleration constants and function rand ( ) generates random numbers between 0 and 1 at each iteration.

PSO has many advantages and disadvantages [7]. Advantages of the basic particle swarm optimization algorithm: PSO is based on the intelligence. It can be applied into scientific, research and engineering use. PSO has no overlapping and mutation calculation. Search can be carried out by the speed of the particle. during the development of several generations, only the most optimist particle can send information to the other particles. And then speed of the re-searching become very fast. The calculation in PSO is very simple. The PSO adopts the real number code and decided by the solution.

The number of constants of the solution is equal to dimension. The Disadvantages of the basic PSO algorithm are the method easily suffers from the partial optimism. This causes the less exact at the regulation of its speed and direction. This basic PSO method does not produced good results on problems related to scattering and non-coordinated systems. The examples of such problems are as the solution to the energy field and the moving rules of the particles in the dynamic environment. These are also called complex environments.

2.2 Basic Parameters of PSO

The basic PSO has many control parameters. These control parameters are the dimension of the problem, the number of particles, acceleration coefficients known as c_1 and c_2, random vector rand (), inertia weight (w), neighborhood size, number of iteration, and the random values. These random values are used to scale the contribution of the cognitive and social components in the PSO. Basic parameters of PSO are as:

2.2.1 Velocity Clamping

Velocities of Particles on each dimension are clamped to a maximum velocity Vmax. If Vmax is too high, particles might fly past good solutions. If Vmax is very small, particles may not explore sufficiently beyond locally good regions. Particles velocity should be clamped into a reasonable interval. Here the new constant vmax is defined which represent the maximum value of velocity:

If $v > v_{max}$, then $v = v_{max}$ (3)

If $v_{ij} < - v_{max}$, then $v = -v_{max}$ (4)

Research work from Fan Huiyuan [8] shows that an appropriate dynamically changing Vmax can improve the performance of the PSO.

2.2.2 Inertia Weight

Shi and Eberhart [9] proposed a new parameter w for the PSO, named inertia weight. in order to better control the scope of the search, inertia weight multiplies the velocity at the previous time step, i.e., v_{id} . Inertia weight, w is used to improve the performance in many applications. This parameter can be interpreted as " inertia constant". Equation (1) is now updated as below:

$V_{id} = \omega * V_{id} + c_1 * \text{rand}(\,) * (P_{id} - X_{id}) + c_2 * \text{rand}(\,) * (P_{gd} - X_{id})$ (5)

The value of inertia weight depends upon the nature of system.The value of inertia weight remains constant for static type of systems and the value of inertia weight changed for a dynamic systems. If w = 1; equation (5) turns back to the original form (1).

2.2.3 Acceleration constants

The acceleration constants $\mathbf{c_1}$ and $\mathbf{c_2}$ in equation (1) represent the weighting of the stochastic acceleration terms .these terms are used to pull each particle towards pbest and gbest positions. Set the acceleration constants each equal to 2.0 for almost all applications. After several experiments for a single particle in a one-dimensional problem space, Ozcan and Mohan [10] proposed few suggestions on how to choose the value of c_1 and c_2. In such a case, the two acceleration constants are considered as a single acceleration constant $c = c_1 + c_2$.

2.2.4 Constriction Factor

When the particle swarm optimization algorithm is run without controlling the velocity in this case the system explodes after a few iterations. In order to control the convergence properties of a particle swarm system, Clerc and Kennedy [11] introduced a constriction coefficient called x. A very simple model using a constriction factor is shown below:

$v_i(t+1)=X(v_i(t)+c_1r_1(pbest(t)-x_i(t))+c_2r_2(gbest(t)-x_i(t)))$ *(6)*

Where $X=\frac{2}{2-c-\sqrt{c^{(2-4c)}}}$ (7)

$c_1 + c_2 = c > 4.0$

3. VARIANTS OF PSO ALGORITHM

The prominent variants of PSO are following:

3.1 Discrete PSO

Discrete PSO algorithm [12-15] is applied to discrete and combinatorial optimization problems over discrete-valued search space. In this algorithm, particle's positions [16] are discrete values. The velocity and position equation are developed for real values and updated in each iteration. Discrete PSO has a high success rate in solving integer programming problems as compare with other methods, such as branch-and-bound fail [17]. It has a quick convergence and better performance results [18].

In binary PSO [19-22] explains that population has a set of particles. Each individual particle represents a binary decision. This decision can be represents by either YES/TRUE = 1 or NO/FALSE = 0. All particles represent their positions through binary values which are 0 or 1. Each particles value is changed from 1 to 0 or vice versa. In this algorithm velocity must be [23] restricted (normalized) within the range {0,1}. The velocity vector equation and position vector equation are defined as:

$v_i^n(t+1) = \frac{1}{1+e-v_i^n(t)}$ (8)

$x_i^n(t+1) = \begin{cases} 1\ ifr < v_i^n \\ 0\ otherwise \end{cases}$ (9)

3.2 Guaranteed Convergence PSO (GCPSO)

The basic idea of Guaranteed Convergence PSO [24] is to include an additional particle. This particle is used to search the region around the current global best position, i.e. its local best position. This local best position is equal to the current global best position. In this manner, the current global best particle is considered as a member of the swarm (e.g. particle).

The update formula for this particle is seen below:

$v^r(t+1) = x^r(t)+gbest(t)+wv^r(t)+\rho(t)(1-2r)$ (10)

It is noted that this variant is so far only applied to the Gbest mode. The other particles in the swarm continuously use the normal velocity update formula. Here the term $- x^r(t) + gbest(t)$ looks like the global cognitive part. Here r is a vector randomly generated in the domain [0, 1] and ρ (t) is the diameter of the search area and dynamically adapted based on the behavior of the swarm. it means that the swarm always moves towards a better position than the current global best position in consecutive iterations. The search diameter will become larger. when the swarm could not find a better position than the current global best position in consecutive iterations, the search diameter will become smaller. The update formula of the diameter is as follows:

$$\rho(t+1) = \begin{cases} 2\rho(t)\ if\ \#successes > sc \\ 0.5\,\rho(t)\ \ if\ \#failures > fc \\ \rho(t)\ otherwise \end{cases} \qquad (11)$$

Where, terms #successes and #failures are used to denote as the number of successes or failures in the consecutive respectively. The threshold parameters sc and fc are defined empirically. Compared with the original PSO, it also has a faster convergence on uni-modal functions. This approach is a good supplement to the original PSO.

3.3 Neighborhood GCPSO

Neighborhoods GCPSO model the structure of social networks [25]. Implementing neighborhoods in the standard PSO requires replacing the velocity update with the following:

$$v_i(t+1)=X(v_i(t)+c_1r_1(pbest(t)-x_i(t))+c_2r_2(gbest(t)-x_i(t))) \qquad (12)$$

Where, gbest is equal to the best position found in the neighborhood of the i^{th} particle. The purpose of neighborhood is to preserve the diversity within the swarm by impending the flow of information through the network. Neighborhood topologies such as lbest [26] with k = 2 provide a very slow flow of information via a long directed path. This allows particle to explore larger areas of search space without being immediately drawn to the global best particle.

3.4 Niche PSO

PSO and its variants have been shown to effectively solve uni-modal optimization problems. These PSO algorithms are not well suited for solving multiple problems because of the way in which they socially exchange information regarding a good global solution. So this concept was first induced to PSO in [27] in order to heighten its ability to handle more complex optimization problems that can search for multiple solutions in parallel. The work flow of this approach is shown below:

(a) Initialize main particle swarm.

(b) Train main swarm particles using one iteration of the individual memory only model.

(c) Update fitness of each main swarm particle.

(d) For each sub-swarm: use GCPSO to update each particles position and then update swarm radius

(e) If possible, merge sub-swarms

(f) Allow all sub-swarms to include particles from the main swarm that moved into it.

(g) Search main swarm for any particle that meets the partitioning criteria. If any particle found than create a new sub-swarm to its closest neighbor.

(h) Repeat from 2 until stopping criteria are met.

The niche concept is also applied in many variants of PSO, such as Adaptive Sequential Niche Particle Swarm Optimization (ASNPSO) [31], PVPSO (parallel vector-based particle swarm optimizer) [26]. There are many enhancements for niche based PSO such as Enhancing the Niche PSO [28], adaptively choosing nicking parameters in a PSO [29].

3.5 Regrouping PSO

Regrouping PSO [30] presents an approach to deal with the stagnation problem by building a mechanism into the PSO algorithm. This mechanism is capable of automatically triggering swarm regrouping when premature convergence is detected. The regrouping factor is evaluated as given below:

$$\rho= 6/5(\epsilon) \qquad (13)$$

in case of premature convergence is detected than the range in which particles are to be regrouped about the global best is calculated per dimension as the minimum of (i) the starting range of the search space on dimension j and (ii) the product of the re-grouping factor having maximum distance along dimension j of any particle from global best:

$$\mathrm{range}_j(\Omega\|x_{i,j}^{r-1} - g_j^{r-1}\|) \qquad (14)$$

The swarm is then regrouped by re-initializing particles position as:

$$x_i = g^{r-1}+r^{'}.\mathrm{range}(\Omega^r) - 0.5\mathrm{range}(\Omega^r) \qquad (15)$$

Where range (Ω^r) = [range$_1(\Omega^r)$, ……., range$_n(\Omega^r)$] which utilizes a random vector r’ to randomize particles within the implicitly defined search space. The maximum velocity is recalculated with each regrouping according to the new range per dimension as

$$v_j^{max,r} = \lambda\ .\mathrm{range}\ (\Omega^r) \qquad (16)$$

Where λ is the velocity clamping percentage.

3.6 NSPSO

A new PSO algorithm called PSO with neighborhood search strategies (NSPSO), utilizes one local and two global neighborhood search strategies. This strategy is defined by Yao, Jingzheng et al. [32]. The NSPSO includes two operations. (i) For each particle, three trial particles are generated by these search strategies respectively. (ii) The best one among the three trial particles and the current particle is select or set as the new current particle.

3.7 IEPSO

In Immunity-enhanced particle swarm optimization IEPSO [33], a population of particles is sampled randomly in the feasible space. The population of particles is used to execute PSO or its variants having the updated values of position and velocity. After that, it executes receptor editing operator also known as non- uniform mutation according to a certain probability (pr), and vaccination operator according to probability (pv). The new generation is obtained by the selection operator after the flying of particles and two immune operators receptor editing and vaccination.

3.8 PSClass algorithm

Particle swarm classifier known as PSClass algorithm [34] has two necessary steps to construct the classifier. In the first step, a number of similar types of prototypes are selected. These prototypes have some density of data are positioned in an unsupervised way on regions of the input space. for this, the Particle Swarm Clustering (PSC) algorithm is used.

in second step, the prototypes are adjusted by an LVQ1 method in order to minimize the percentage of misclassification.

PSClass algorithm is used to eliminate the need for the users to define the swarm size a priori. A critical parameter required for many data classification algorithms.

3.9 Quantum-Behaved PSO algorithm

Quantum-behaved particle swarm optimization algorithm in [35]. Implicit space decomposition is adopted i.e. the whole swarm is divided into several sub-swarms which search the whole space respectively. Particles in different sub-swarms will locate in different areas and evolve in different directions which prevents rapid decline in diversity of the whole swarm effectively.

3.10 Multi-objective optimization (MPSO)

In current years, multi-objective optimization has been a very active research area to the researchers. In multi-objective optimization (MO) problems, objective function may be optimized separately from each other and the best solution may be found for each objective. This results in there being a group of alternative solutions. The relevance of each objective relative to the others are considered equivalent in the absence of concern information. The group of alternative solutions is known as Pareto optimal set or Pareto front. The selection of social and cognitive leaders (nBest and pBest) is the key point of MO-PSO algorithms [36, 37, 38-45].

3.11 Hybrid PSO

A hybrid PSO algorithm [12,13] can be achieved by incorporating different methods. these methods have already been tested in other evolutionary computation techniques. Many authors have incorporated selection, mutation and crossover into their PSO algorithms. As a result, hybrid versions of PSO have been created to test in different applications [14]. The most common ones include hybrid of Genetic Algorithm and PSO (GA-PSO) [13, 15].

4. PSO IN COMPLEX ENVIRONMENTS

4.1 Constraint Optimization

There are a lot of constraints in optimization field. To deal with these constraints, the original PSO method needs to be modified. For constraint optimization problems, Hu and Eberhart [46] introduced a method to solve the constraint optimization problems. For dealing with constraints, a preserving feasibility strategy is employed. Two modifications were made to PSO algorithms.

1. During updating the memories, all the particles keep feasible solutions in their memory.
2. During the initialization process, each particle is begun with feasible solutions.

4.2 PSO in dynamic Environments

Dynamic systems change their state frequently or continuously. Many real world systems involve in dynamic Environments. Initial work in tracking dynamic systems with particle swarm optimization was reported in paper written by Eberhart and Shi [47]. Paper [48] introduces an adaptive PSO (APSO) in which automatically tracks found having various changes in a dynamic system.

5. OBSERVATION AND ANALYSIS

On the basis of Research articles reviewed during year 1995 to 2015, certain observation and corresponding analysis produced better results in optimization field compared with traditional methods and found that velocity clamping, inertia weight, and acceleration constants are the basic variations of PSO. These basic variants of PSO are used to improve speed of convergence and quality of solution. **Figure 1** presents that inertia weight and velocity clamping have the largest number of literatures between 1995 and 2015. It is observed that originally, particle swarm optimization was used to solve statics problem. For solving other types of problems, many researchers have developed different variants of PSO such as: Discrete PSO, Niche PSO, Guaranteed Convergence PSO, Multi-objective PSO, Neighborhood PSO, Regrouping PSO, Neighborhood Search Strategies PSO, and Immunity-Enhanced PSO, Particle swarm Classifier, Quantum-Behaved PSO and Hybrid PSO. Every variant of PSO has different form and function. Each variant has different methods to solve their optimization problems.

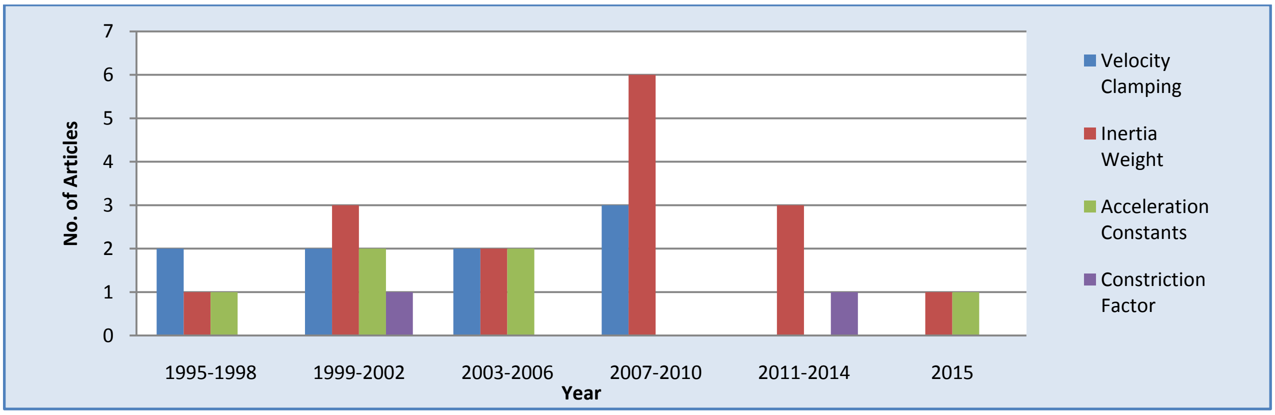

Figure 1: Distribution of Articles for Basic Parameters of PSO(Data Collected Till Date for Year 2015)

Figure 2 presents distribution of articles in terms of Variants of Particle Swarm Optimization. The largest number of papers using MPSO, Discrete PSO, PSClass and HPSO published in 2006, 2007, 2011, and 2012 respectively. Niche PSO, QBPSO, GCPSO, NGCPSO, and IEPSO are only used by some of researchers. From the figure, multi-objective PSO and Hybrid PSO are considered in large numbers by various researchers between 1995 and 2015. PSO has a characteristic of modification. Due to this characteristic, different variants of PSO can easily be developed. These variants are developed according to their applications and usages in various optimization problems. The optimization can be classified into different application areas such as scheduling, searching, forecasting, feature selection, classification. The method for modification in PSO is very conducive to create a new method of variation PSO .there are others area that can be implemented by modification of PSO.

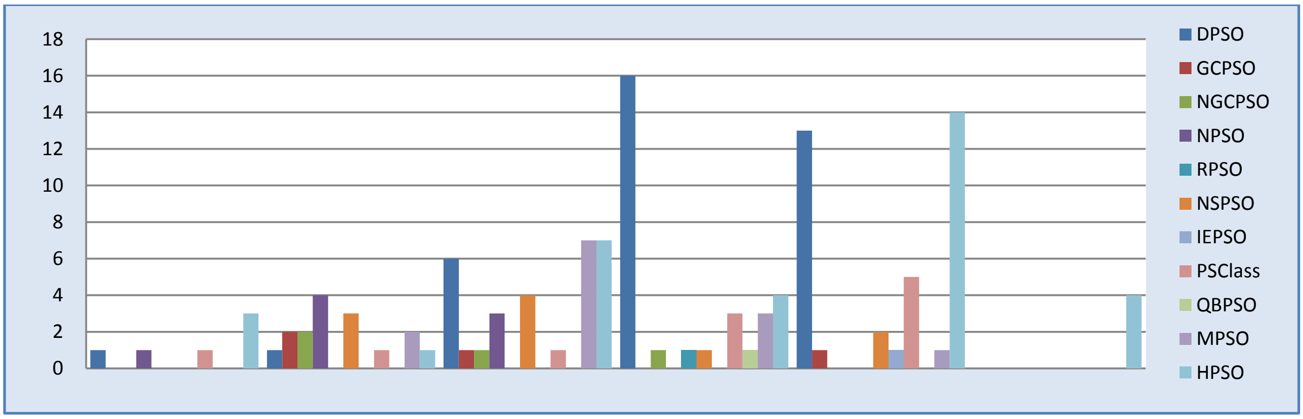

Figure 2: Distribution of Articles in term of PSO Variants(Data Collected Till Date for year 2015)

Figure 3 presents Aggregation chart for Application of the PSO over different Years. Regarding on this the use of PSO increase drastically from 2002 in various applications.

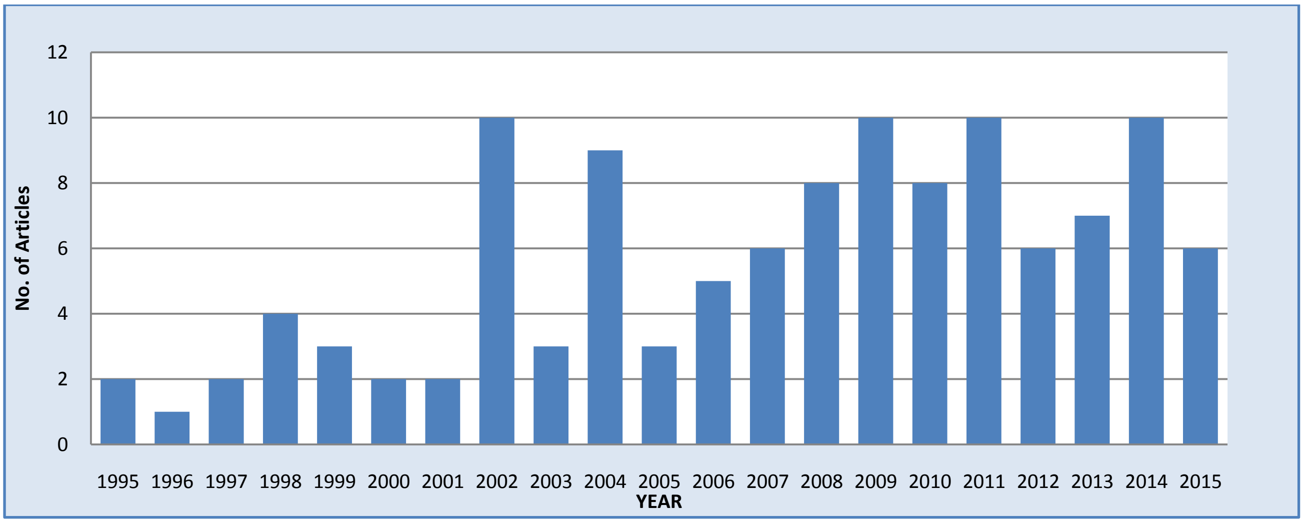

Figure 3: Aggregation chart for Application of the PSO over different Year

(Data Collected Till Date for Year 2015)

6. CONCLUSION

The process of PSO algorithm is used to find optimal values in a search area. the process follows the work of an animal society which has no leader. Particle swarm optimization consist swarm of particles. These particles represent a potential solution called better condition. Particle will move through a multidimensional search space to find the best position in that space.

In this paper, different methods of PSO algorithm are reviewed. Basic particle swarm optimization has many advantages and disadvantages. There are several basic Parameters of PSO as mentioned in this paper. These parameters are used to control the velocity and for stable convergence. On the other hands, modified variants of PSO help the PSO to process other conditions. These conditions are very difficult to optimize and cannot be solved by the basic PSO. The observation and review is made to show the absolute function of PSO, advantages, disadvantages of PSO, Modification of PSO and their applications in complex environment. The different applications of PSO can show which one is better between the modified or variant.

7. REFERENCES

[1] R. C. Eberhart and J. Kennedy, *“A new optimizer using particle swarm theory,”* Proceedings of the 6th International Symp. on Micromachine and Human Science., Nagoya, Japan, pp. 39–43,1995.

[2] J. Kennedy and R. Eberhart, *"Particle Swarm Optimization"*, Proceedings of IEEE International Conference on Neural Networks, vol.4, pp. 1942-1948, Piscataway, 1995.

[3] A.P.Engelbrecht, *“Fundamental of Computational Swarm Inteligent,”* First ed. The atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England: John Wiley & Sons Ltd, 2005.

[4] B. Santosa, *"Tutorial Particle Swarm Optimization,"* 2006.

[5] M. B. Ghalia, *"Particle Swarm Optimization with an Improved Exploration-Exploitation Balance,"* Proceedings of 51st Midwest symposium on circuits and systems(MWSCAS'08),pp.759-762,USA,August 2008.

[6] Eberhart,R.C.and Shi,Y.*"Particle swarm optimization: developments, applications and reseources.*" Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2001),vol.1, pp. 81-86,seoul,south korea,May2001.

[7] Qinghai. Bai, *"Analysis of Particle Swarm Optimization Algorithm,"* Computer and Information Science, vol. 3,no.1, pp.180, Februari 2010.

[8] H.Y.Fan, *"A modification to particle swarm optimization algorithm,"* Engineering Computations vol.19,no.8, pp. 970-989,2002.

[9] Y. Shi and R. C. Eberhart, *"A modified particle swarm optimizer,"* Proceedings of IEEE International conference on world Congress on Evolutionary Computation(ICEC'98),., pp. 69–73, USA,May 1998.

[10] Ozcan, Ender, and Chilukuri K. Mohan. *"Partial shape matching using genetic algorithms.*" Pattern Recognition Letters vol.18,no.10, pp. 987-992,1997.

[11] M. Clerc and J. Kennedy, *"The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional Complex Space"*, IEEE Transactions on Evolutionary Computation, vol. 6,no.1, pp. 58-73, 2002.

[12] Wang, Hui, Yong Liu, and SanyouZeng. *"A hybrid particle swarm algorithm with Cauchy mutation."* Proceedings of IEEE Swarm Intelligence Symposium(SIS'07),pp. 356-360,Honolulu,Hawaii,USA,April 2007.

[13] Settles, Matthew, and Terence Soule. *"Breeding swarms: a GA-PSO hybrid*," Proceedings of the 2005 conference on Genetic and evolutionary computation, ACM, 2005.

[14] R.Thangaraj, M.Pant,A.Abraham,P.Bouvry, *"Particle swarm optimization: Hybridization perspectives and experimental illustrations".* Applied Mathematics and Computation ,vol. 217,pp. 5208-5226, 2011.

[15] Premalatha. K., and A. M. Natarajan. *"Hybrid PSO and GA for global maximization*," International Journal Open Problems Compt. Math., vol.2,no.4, pp. 597-608,December 2009.

[16] K. Parsopoulos and M. Vrahatis, *"Recent approaches to global optimization problems through particle swarm optimization"*, Natural Computing, vol. 1, pp. 235-306, May 2002.

[17] E. Laskari, K. Parsopoulos, and M. Vrahatis*, "Particle swarm optimization for integerprogramming",* in Proceedings IEEE Congr.Evol.Comput., vol. 2, pp. 1582-1587, 2002.

[18] H. Shayeghi, M. Mahdavi, A. Bagheri, *"Discrete PSO algorithm based optimization of transmission lines loading in TNEP problem, Energy Conversion and Management,*" vol. 51, no. 1, pp. 112-121, January 2010.

[19] Li-Yeh Chuang, Hsueh-Wei Chang, Chung-JuiTu, Cheng-Hong Yang, *"Improved binary PSO for feature selection using gene expression data,*" Computational Biology and Chemistry, vol. 32, no. 1, pp.29-38, 2008.

[20] M. FatihTaşgetiren, 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, 2003.

[21] Tao Gong, Andrew L Tuson, *"Binary particle swarm optimization: a formal analysis approach,"* Proceedings of the 9th annual conference on Genetic and evolutionary computation (GECCO '07), pp. 172-172, ACM New York, USA, 2007.

[22] Mohamad, M.S., Omatu, S., Deris, S., yoshioka,M., *"A modified binary particle swarm optimization for selecting the small subset of informative genes from gene expression data,"* IEEE Trans. on Information Technoogy.in Biomedicine,vol.15,no.6,pp.813-822, 2011.

[23] Kennedy, J and Eberhart, R.C. *"A discrete binary version of the particle swarm algorithm",* IEEE International Conference on Systems, Man, and Cybernetics, vol.5,pp. 4104-4108, 1997.

[24] F.Van den Bergh. ,*"An Analysis of Particle Swarm Optimizers,"* Department of Computer Science, PhD thesis, University of Pretoria, South Africa, 2002.

[25] Narinder Singh, S.B. Singh, *"Personal best position particle swarm opti-mization,"* Journal of Applied Computer Science and Mathematics, vol. 12 ,no.6,pp.69-76, 2012.

[26] http://en.wikipedia.org/wiki/Particle swarm optimization.

[27] Brits, Riaan, Andries P. Engelbrecht, and F. Van den Bergh. *"A niching particle swarm optimizer."* Proceedings of the 4th Asia-Pacific conference on simulated evolution and learning. Vol. 2, pp. 692-696,Singapore, 2002.

[28] A.P Engelbrecht and L.N.H. van Loggerenberg, *"Enhancing the Niche PSO,"* Proceedings of IEEE Congress on Evolutionary Computation (CEC'07), pp. 2297-2302,Singapore,September 2007.

[29] S.Bird, X. Li., *"Adaptively choosing niching parameters in a PSO,"* Proceedings of the 8th annual conference on Genetic and evolutionary computation(GECCO'06), ACM, pp.3-9,July 2006.

[30] Evers, George I., and M. Ben Ghalia, *"Regrouping particle swarm optimization: a new global optimization algorithm with improved performance consistency across benchmarks"*, Proceedings of IEEE International Conference on Systems, Man and Cybernetics *(*SMC'09),pp.3901-3908 October 2009.

[31] M. Dorigo, T. Stützle, *"Ant Colony Optimization,"* MIT Press, Cambridge, MA, 2004.

[32] Yao, Jingzheng, and Duanfeng Han,*"Improved barebones particle swarm optimization with neighborhood search and its application on ship design,"* Mathematical Problems in Engineering, 2013.

[33] Kang, Fei, Junjie Li, and Sheng Liu. *"Combined data with particle swarm optimization for structural damage detection,"* Mathematical Problems in Engineering, 2013.

[34] Szabo, Alexandre, and Leandro Nunes de Castro. *"A Constructive Data Classification Version of the Particle Swarm Optimization Algorithm,"* Mathematical Problems in Engineering, 2013.

[35] Zhou, Di, Jun Sun, and WenboXu., *"An advanced quantum-behaved particle swarm optimization algorithm utilizing cooperative strategy."* Proceedings of IEEE Third International Workshop on Advanced Computational Intelligence(IWAC'10),pp. 344-349,Suzhou,Jiangsu,August 2010.

[36] Coello Coello,C.A. and Lechuga,M.S, *"MOPSO: a proposal for multiple objective particle swarm optimization,"* Proceedings of the IEEE Congress on Evolutionary Computation(CEC'02),pp.1051-1056,vol.2, Honolulu, Hawaii USA,May 2002.

[37] Hu,X .and Eberhart, R.C., *"Multiobjective optimization using dynamic neighborhood particle swarm optimization,"* Proceedings of the IEEE Congress on Evolutionary Computation(CEC'02),vol. 2, pp. 1677-1681,Honolulu, Hawaii USA, May 2002.

[38] Fieldsend, J.E. and Singh, S.A, *"multi-objective algorithm based upon particle swarm optimization, an efficient data structure and turbulence,"* Proceedings of the Workshop on Computational Intelligence,. pp. 37-44, Birmingham, UK, 2002.

[39] Hu,X., Eberhart,R.C., and Shi,Y., *"Particle swarm with extended memory for multiobjective optimization,"*Proceedings of the IEEE Swarm Intelligence Symposium(SIS'03), pp. 193-197, Indianapolis, Indian, USA.,April 2003.

[40] Li,X., *"A non-dominated sorting particle swarm optimizer for multiobjective optimization,"* Lecture Notes in Computer Science, Proceedings of the Genetic and Evolutionary Computation Conference, pp.37-48, USA ,2003.

[41] Yen,G.G, Lu,H.,*"Dynamic population strategy assisted particle swarm optimization in multiobjective evolutionary algorithm design,"* Proceedings of the IEEE International Symposium on Intelligent Control (ISIC'03),pp.697-702, Houston,Texas,USA, October 2003.

[42] Moore, J. and Chapman, R., *"Application of particle swarm to multi-objective optimization,"* Department of Computer Science and Software Engineering, Auburn University,1999.

[43] Mostaghim,S. and Teich, J., *"Strategies for finding local guides in multi-objective particle swarm optimization (MOPSO),"* Proceedings of the IEEE Swarm Intelligence Symposium(SIS'03), pp. 193-197 Indianapolis,IN,USA,April 2003.

[44] K.E.Parsopoulos and M.N.Vrahatis,*"Particle swarm optimization method in multiobjective problems,"* Proceedings of the ACM Symposium on Applied Computing, pp. 603-607, 2002.

[45] Ray,T. and Liew,K.M, *" A swarm metaphor for multiobjective design optimization,"* Engineering Optimization, vol. 34 ,no. 2, pp. 141-153, 2002.

[46] Hu, X., Eberhart, R.C.,*"Solving constrained nonlinear optimization problems with particle swarm optimization,"* Proceedings of the Sixth World Multiconference on Systemics Cybernetics and Informatics, Orlando,USA, 2002.

[47] Y. Shi , R. C.and Eberhart, "*Tracking and optimizing dynamic systems with particle swarms,"* Proceedings of the IEEE Congresson Evolutionary Computation ,pp. 94-97, Seoul,Korea 2001.

[48] Hu, X., Eberhart, R.C.,*"Adaptive particle swarm optimization:detection and response to dynamic systems,"* Proceedings of the IEEE Congress on Evolutionary Computation, pp. 1666-1670, Honolulu,Hawaii USA, 2002.

[49] Brajesh Kumar Singh, A. K. Misra, *"Software Effort Estimation by Genetic Algorithm Tuned Parameters of Modified Constructive Cost Model for NASA Software Projects,"* International Journal of Computer Applications, Volume 59 No.9,pp.22-26,December2012.

[50] Rini,D.P., Shamsuddin.S.M. and Yuhaniz,S.S.,"Particle Swarm Optimization: Technique,System and Challenges", International Journal of Computer Applications, Volume 14, No.14,pp.19-27, January 2011.