Novelty Journals

ISSN 2394-9651

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 3, Issue 2, pp: (32-40), Month: May - August 2016, Available at: www.noveltyjournals.com

A Conditionally Distributed Particle Swarm Optimization Approach

Sharandeep Singh

Department of Mathematics, Punjabi University, Patiala

Abstract: **A modified version of SPSO known as CHF-PSO has been presented in this paper. In this version has been improved the convergence rate of the each member of the entire swarm. The location of each member has been updated by two probability conditions if one condition is fail to search the global optima then position of the each member of the population is updated by other existing condition of a probability. The performance of this modified approach has been tested on different types of classical/benchmark functions.**

Keywords: **HF-Hazard Function, CHF-Cumulative Hazard Function, Velocity and Position.**

1. INTRODUCTION

Firstly in 1995, the young scientists James Kennedy and Russell C. Eberhart have been presented the new idea of computational intelligence known as PSO algorithm. In which this version each particle is performed to search best location in the global search area. Each location of particle is changed with the help of self and global confidence. The movement of particle is updated by following equations:

$$v_{ij}^{t+1} = w \times v_{ij}^{t} + c_1 r_{1j}^{t}\left(p_{best,i}^{t+1} - x_{ij}^{t}\right) + c_2 r_{2j}^{t}\left(G_{best} - x_{ij}^{t}\right) \quad (1)$$

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

2. RELATED WORK

Kennedy & Eberhart (1997) has been originated a Binary-PSO. In which this version particle's position is discrete but its velocity is continuous. Krohling & Renato (2004) presented a novel PSO based on the Gaussian probability distribution. The help of this approach has been improving the convergence ability of PSO. The execution of this version has been tested on several types classical functions

Cai, Zang, Cui & Tan (2007) have originated a novel velocity threshold automation strategy by incorporating a Levy probability distribution. On the behalf of experiential has been represented that the existing version is efficient and effective.

Lu & Qiu (2010) have presented a probabilistic PSO approach. In which this version have been implement of distinct types of probability conditions by this conditions improved the movement of each particle in the global search space.

Singh & Singh (2013) has been presented HF-PSO. In which this version the position of each particle is updated by DED-HF condition. The execution of this version has been compared by SPSO technique. By obtaining results has been shown that presented version provided the best quality of solutions outperform than SPSO.

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 3, Issue 2, pp: (32-40), Month: May - August 2016, Available at: www.noveltyjournals.com

3. HAZARD AND CUMULATIVE HAZARD FUNCTION

The HF-hazard function (also known as the failure rate) is the ratio of the PDF (probability density function) to the SF (survival function).

The cumulative hazard function is the integral of the hazard function. The formula for the cumulative hazard function of the double exponential distribution is

$$H(x) = \begin{cases} -\ln(1-\frac{e^x}{2}) & \text{for } x < 0 \\ x+\ln(2) & \text{otherwise} \end{cases} \quad (3)$$

4. CHF-PSO ALGORITHM

In which SPSO algorithm the position of each particle is updated by previous and present execution of the particle in the search space.

But, on the other side, the modified approach CHFPSO is depending on two probability conditions. In which one condition is fail to search the global optimal point, then we find the global optimal point by other existing probability condition. Finally we update the position of each particle in the entire swarm by using the following equations:

$$x_{ij}^{t+1} = \begin{cases} -\ln(1-\frac{e^{v_{ij}^{t+1}}}{2}) & if\ v_{ij}^{t+1} < U(0,1) \\ x_{ij}^{t}+\ln(2) & if\ v_{ij}^{t+1} \geq U(0,1) \end{cases} \quad (4)$$

Update Velocity equation

$$\begin{aligned} v_{ij}^{t+1} = w \times v_{ij}^{t} + c_1 r_{1j}^{t}\left(p_{best,i}^{t+1} - x_{ij}^{t}\right) \\ + c_2 r_{2j}^{t}\left(G_{best} - x_{ij}^{t}\right) \end{aligned} \quad (5)$$

In our version we replace equation (2) by (4) remaining same. The idea is to replace the conventional generation of x_{ij}^{k+1} using cumulative hazard function. The rest of the operations are same as in SPSO.

The pseudo code of CHFPSO is shown below:

Initialize the crowd // initialize all members of the crowd

For every member in the crowd do

If $f(x_i) < f(p_{best,i})$

Then

$x_i = p_{best,i}$

End if

If $f(p_{best,i}) < f(g_{best})$ then

$g_{best} = p_{best,i}$

End if

End for

// update member of the crowd velocity and position

For every member in the crowd do

For every j -dimension in D do

$$v_{ij}^{t+1} = w \times v_{ij}^{t} + c_1 r_{1j}^{t} \left(p_{best,i}^{t+1} - x_{ij}^{t} \right) + c_2 r_{2j}^{t} \left(G_{best} - x_{ij}^{t} \right) \quad (6)$$

$$x_{ij}^{t+1} = \begin{cases} -\ln(1 - \frac{e^{v_{ij}^{t+1}}}{2}) & if\ v_{ij}^{t+1} < U(0,1) \\ x_{ij}^{t} + \ln(2) & if\ v_{ij}^{t+1} \geq U(0,1) \end{cases} \quad (7)$$

End for

End for

Iteration = Iteration + 1

Until > Max_Iterations

5. UPDATED DIRECTION

The scientists, engineers and researcher are presented, lot of a number of modified approach of particle swarm optimization algorithm for the purpose of improving the convergence rate of SPSO like that BPSO (Kennedy & Eberhart (1997)), NPSO (Krohling & Renato (2004)), NVTAS (Lai, Zeng, Cui & Tan (2007)), NPPSO (Wang, Wang, Fu & Zhen (2008)), PPSO (Lu & Qiu (2010)) and HPSO (Singh & Singh (2012)), HFPSO (Singh & Singh (2013)) and so on. On the behalf of these algorithms the researcher has been modified the position of each particle in the search space and some other researcher has been modified the position of each particle and improve the execution of standard version of particle swarm optimization algorithm with the help of probability condition based algorithm of PSO. Similarly, in which article has been also presented the modified version of PSO. A modified existing approach depends on probability conditions of a cumulative hazard function. On the behalf of this existing approach, we change the position of each particle in the search space as comparison to PSO algorithm particle position. CHFPSO version of SPSO algorithm may be more fundamental due to simple changes in coding. The floating-point technique is operated through a stochastic change in the rate of change of position. The original update position equation of SPSO is

$$x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1} \quad (8)$$

In which SPSO, the rate of change of position of each particle is updated by previous and newly execution of the particle. But, CHFPSO technique seem very distinct from

$$x_{ij}^{t+1} = \begin{cases} -\log(1 - \frac{e^{v_{ij}^{t+1}}}{2}) & if\ v_{ij}^{t+1} < U(0,1) \\ x_{ij}^{t} + \log(2) & if\ v_{ij}^{t+1} \geq U(0,1) \end{cases} \quad (9)$$

In which newly approach the rate of change of position of each particle is updated by two probability conditions of cumulative hazard function.

6. TESTING

CHFPSO approach has been tested on several types of classical functions i.e. Scalable and Non-Scalable functions (more see in Appendix).

7. ANALYSIS

On the behalf of newly existing approach has been providing the best position of each particle in the search space as comparison to standard version of PSO (SPSO) and HFPSO. With the helping of this version the each particle in the entire swarm moving in positive direction and all particles achieve the superior global optimal values without wasting of time. The execution of CHFPSO, SPSO and HFPSO has been tested on many types of classical functions in which one or more existing dimension of the swarm in the search space. The quality of these algorithms has been testing in the terms of efficiency, reliability, accuracy and stability and its execution compare with the help of obtaining results (Table 1 to 4) and plotting by graphs (Figure 1 to 5) in following sections.

The difference between PSO, HFPSO and CHFPSO lies in their defined searching spaces. In SPSO, moving in the space means a change in the value of position coordinates in one or more of existing dimensions. However, in the CHFPSO moving in the spaces means a change in the probability condition of cumulative hazard function of the fact that the value of position coordinate is updated by:

$$x_{ij}^{t+1} = \begin{cases} -\ln(1-\dfrac{e^{v_{ij}^{t+1}}}{2}) & if\ v_{ij}^{t+1} < U(0,1) \\ x_{ij}^{t} + \ln(2) & if\ v_{ij}^{t+1} \geq U(0,1) \end{cases} \qquad (10)$$

Firstly, we compute the minimum function value by testing on several classical functions of CHFPSO, SPSO and HFPSO algorithm. On the behalf of obtaining results, we observing that a modified approach is superior perform outperform than SPSO and HFPSO.

Secondly, the movement of each particle in the entire swarm testing by SPSO, HFPSO and CHFPSO algorithms on 15 classical functions in one or more existing dimensions and movement of each particle is plotted by the figure 2 to figure 5. All the obtaining results by Table 3 to 4 shown that each particle moves in best position as comparison to SPSO and HFPSO particle movement in the search space. The execution of each particle in search space is comparing and plotted by the graph (see figure 2 to figure 5).

8. DISCUSSIONS

The modified version has been easily implemented on my system. SPSO, HFPSO and CHFPSO algorithms execution has been tested on several classical functions. The results reported in the preceding section represented that the CHFPSO approach is capable of solving these various functions very rapidly and provide the superior location of each particle in search space outperform than SPSO and HFPSO algorithm in one or more existing dimension. An existing approach improved the rate of convergence of SPSO approach.

The fifteen functions were implemented in a SPSO, HFPSO and CHFPSO code. All other aspects of the code, including parameter values: minimum function value, inertia factor value, range of function, swarm size, acceleration value, and error value ran identically on the various functions. Thus it appears that the existing algorithm is extremely flexible and robust. It controls the exploration and exploitation of each particle of the swarm and provided a superior location of each particle in global search space in which one or more existing dimension as comparison to SPSO and HFPSO approach.

Finally, based on numerical and experimental results has been observing that a modified version is most suitable of solving these various functions very rapidly outperform than SPSO, HFPSO and other PSO algorithms.

Table.1: CHFPSO comparison with SPSO and HFPSO in the terms of Minimum Objective Function Values and Mean Value of the Function

Problem No's	Dim	Min Value	Minimum Values	Objective	Function	Mean Value of the Function		
			CHFPSO	SPSO			HFPSO	CHFPSO
1.	D = 12	0	**2343.009**	**0.113987**	**2.**	D = 12	0	**2343.009**
3.	D = 12	0	**22.08655**	**0.000000**	**4.**	D = 12	0	**22.08655**
5.	D = 12	0	0.000000	0.002440	**6.**	D = 12	0	0.000000
7.	D = 12	0	0.001899	0.002349	**8.**	D = 12	0	0.001899
9.	D = 12	0	**0.000037**	**0.000000**	**10.**	D = 12	0	**0.000037**
11.	D = 12	0	**0.000002**	**0.000000**	**12.**	D = 12	0	**0.000002**
13.	D = 12	0	**416.0038**	**0.037272**	**14.**	D = 12	0	**416.0038**
15.	D = 12	0	**0.020693**	**0.000000**	**16.**	D = 12	0	**0.020693**
17.	D = 12	0	0.000489	0.005152	**18.**	D = 12	0	0.000489
19.	D = 12	0	**0.000000**	**0.000000**	**20.**	D = 12	0	**0.000000**
21.	D = 02	0	**0.009179**	0.009326	**22.**	D = 02	0	**0.009179**
23.	D = 02	0.9	0.480465	0.480474	**24.**	D = 02	0.9	0.480465
25.	D = 04	0	**0.149216**	**0.000000**	**26.**	D = 04	0	**0.149216**
27.	D = 02	-0.3523	**0.016554**	**0.016554**	**28.**	D = 02	-0.3523	**0.016554**

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 3, Issue 2, pp: (32-40), Month: May - August 2016, Available at: www.noveltyjournals.com

Table.2: CHFPSO comparison with SPSO and HFPSO in the term of Standard Deviation (S.D.) Values

Problem No's	Dim	Min Value	Standard Deviation (S.D)		
			SPSO	HFPSO	CHFPSO
1.	D = 12	0	2071.21648	0.185918	0.446666
2.	D = 12	0	18.993111	0.075549	5.907242
3.	D = 12	0	0.251207	0.241859	0.246096
4.	D = 12	0	0.055918	0.115841	0.049622
5.	D = 12	0	0.173598	0.000026	0.000019
6.	D = 12	0	0.146718	0.000009	0.000007
7.	D = 12	0	3081.156	0.224407	0.264876
8.	D = 12	0	0.264615	0.000000	0.000000
9.	D = 12	0	0.142786	0.180001	0.174789
10.	D = 12	0	0.007893	0.000000	0.000000
11.	D = 02	0	0.263413	0.434515	0.255275
12.	D = 02	0.9	0.028292	0.036256	0.032660
13.	D = 04	0	9.504959	0.252133	0.165410
14.	D = 02	-0.3523	0.228200	0.037575	0.124286
15.	D = 02	0	0.222921	0.004829	0.001720

Table.3: The position of each particles of CHFPSO algorithm comparison with SPSO and HFPSO testing on Parabola and Griewank Problems

Each Particle Position ⇩	Parabola (Sphere)			Griewank		
	SPSO	**HFSO**	**CHFPSO**	**SPSO**	**HFPSO**	**CHFPSO**
P = 1	1.349769	0.522806	0.385261	8.098611	0.000000	0.00000
P = 2	19.518050	0.153947	0.289698	117.10830	0.000000	0.000000
P = 3	25.686748	0.134687	0.284230	154.12048	1.000000	0.002852
P = 4	3.000642	0.133664	0.205585	18.003850	0.104978	0.000000
P = 5	21.955533	0.173660	0.349502	131.73319	0.000000	0.000000
P = 6	48.957666	0.000048	0.507799	293.74599	0.000000	0.000000
P = 7	25.664124	0.000001	0.304279	153.98474	0.000000	0.000000
P = 8	10.365079	0.256405	0.625287	62.190473	1.000000	0.512244
P = 9	25.171103	0.060502	0.171426	151.02661	0.000000	0.000000
P = 10	22.533826	0.131981	0.393202	135.20295	0.000000	0.000000
P = 11	16.132464	0.177015	0.251264	96.794786	0.000000	0.000000
P = 12	34.858597	0.000366	0.374004	209.15158	0.000000	0.000000

Table.4: The position of each particles of CHFPSO algorithm comparison with SPSO and HFPSO testing on Step and Krishna Kumar Problems

Each Particle Position ⇩	Step			Krishna Kumar		
	SPSO	**HFPSO**	**CHFPSO**	**SPSO**	**HFPSO**	**CHFPSO**
P = 1	0.948935	2.457731	0.012657	7.616504	-5.934935	-9.182337
P = 2	2.646487	0.637988	-0.768305	-2.228767	-4.167913	2.576444
P = 3	-3.809022	-1.921465	1.728647	-9.786981	0.422681	1.663774
P = 4	3.184003	3.601829	2.926496	1.332743	9.700919	-5.133752
P = 5	-2.280226	1.076439	-4.128720	1.582385	-7.257607	0.241149
P = 6	-1.241757	1.884276	3.438386	-3.570971	7.005524	-8.828323
P = 7	1.046751	4.556233	-0.409231	2.569353	-3.522752	0.017844
P = 8	5.029372	-4.912494	0.203912	2.800073	3.948790	0.055216
P = 9	-3.803397	-3.409948	-3.974653	-0.008850	1.606189	0.001314
P = 10	-2.613674	-4.731863	1.602393	4.250313	7.789239	0.038930
P = 11	0.047033	3.257131	-3.525576	-9.940184	-9.567858	1.049715
P = 12	-0.219225	-2.971497	3.375572	4.431593	-7.999817	0.258120

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 3, Issue 2, pp: (32-40), Month: May - August 2016, Available at: www.noveltyjournals.com

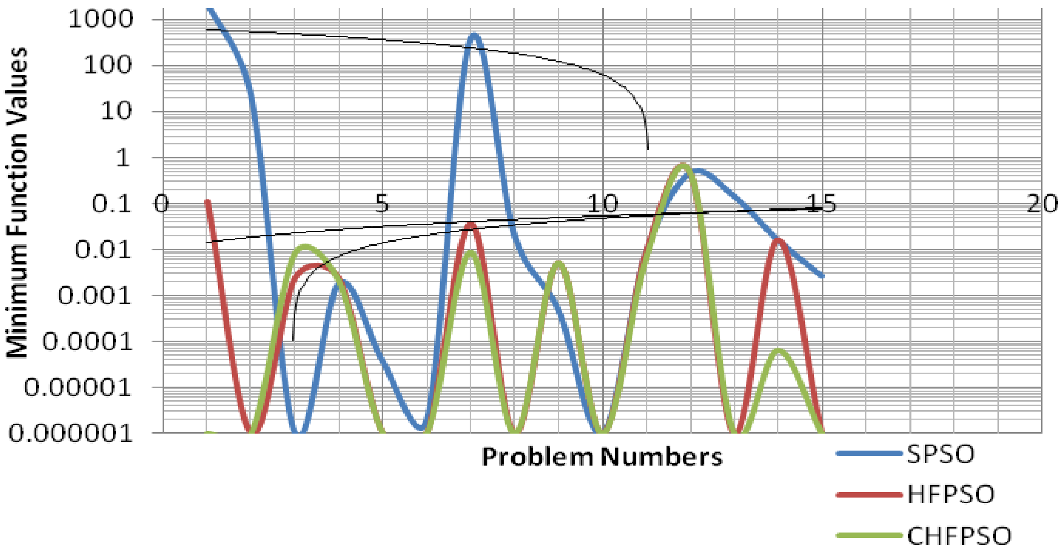

Figure.1: Comparison of SPSO, HFPSO and CHFPSO optimal point performance by minimum function value

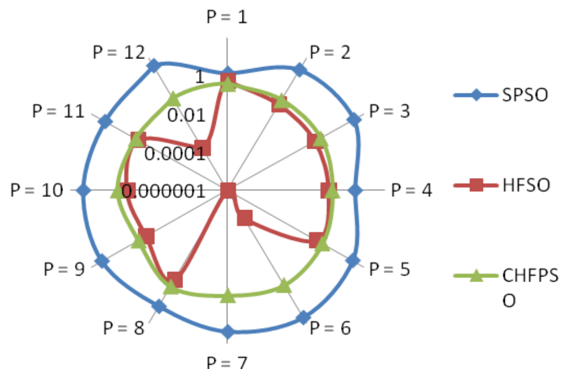

Figure.2: Comparison of each particle position SPSO, HFPSO and CHFPSO tested on Parabola (Sphere) Function

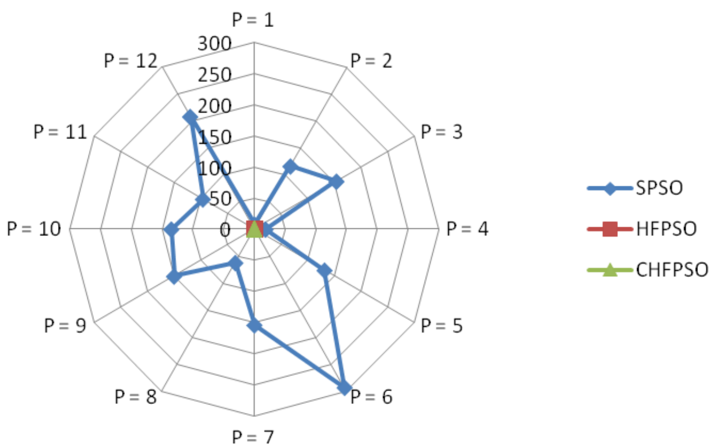

Figure.3: Comparison of each particle position SPSO, HFPSO and CHFPSO tested on Griewank Function

International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 3, Issue 2, pp: (32-40), Month: May - August 2016, Available at: www.noveltyjournals.com

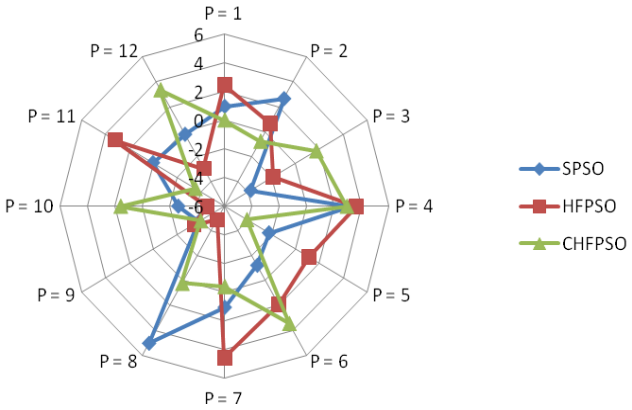

Figure.4: Comparison of each particle position SPSO, HFPSO and CHFPSO tested on Step Function

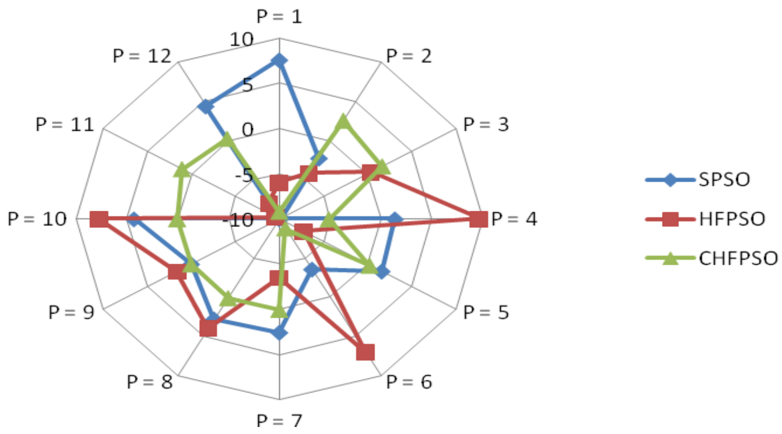

Figure.5: Comparison of each particle position SPSO, HFPSO and CHFPSO tested on Krishna Kumar Function

9. CONCLUSION

In this article has been presented the probability based technique is known as CHF-PSO. In this version has been improved the convergence rate of each member of the population in the global search space. The position of each member of the crowd has been updated by two probability condition if one condition is fail to search the global optima then position of the each member of the population is updated by other existing condition of a probability. The execution of this version has been tested on several types of classical functions and after study the obtaining results has been examines that modified version is superior to HPSO and SPSO technique.

REFERENCES

[1] Kennedy, J. & Eberhart, R.C. (1995). Particle swarm optimization. *Proceeding of IEEE International Conference on Neural Networks, Piscataway, NJ.*, 1942-1948.

[2] Kennedy, J. & Eberhart, R.C.(1997). A discrete binary version of the particle swarm algorithm. *In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, IEEE Press, Piscataway, NJ*, 4104-4108.

[3] Shi, Y. & Eberhart, R.C. (1998). A Modified Particle Swarm Optimizer. *In Proceedings of the IEEE Congress on Evolutionary Computation*, 69–73.

[4] Eberhart, R.C. & Shi, Y. (2000). Comparing inertia weigthts and constriction factors in particle swarm optimization. *In Proceedings of IEEE Congress on Evolutionary Computation, San Diego, CA*, 84-88.

[5] Clerc M. & Kennedy J. (2002). The Particle Swarm: Explosion, Stability, and Convergence in a Multi-dimensional Complex Space. *IEEE Transactions on Evolutionary Computation*,6(1), 58-73.

[6] Krohling & Renato, A. (2004). Gaussian swarm: a novel particle swarm optimization algorithm. *Proceeding in Cybernetics and Intelligent systems IEEE*, 1(1), 372 – 376.

[7] Bratton, D. & Kennedy, J. (2007). Defining a Standard for Particle Swarm Optimization. *Proceedings of the IEEE Swarm Intelligence Symposium, SIS*.

[8] Cai, X., Zeng, J., Cui, Z. & Tan, Y. (2007). Particle Swarm Optimization Using *Lévy* Probability Distribution. *Advances in Computation and Intelligence Lecture Notes in Computer Science* , 4683(2), 353-361.

[9] Wang, L. Wang, X., Fu, J. & Zhen, L. (2008). A Novel Probability Binary Particle Swarm Optimization Algorithm and Its Application. *Journal of Software*, 3(9), 28-35.

[10] Rao, S.S. (2009). Engineering Optimization Theory and Practice. *4th edition, Ed.:John Wiley and Sons*.

[11] Talbi, E.G. (2009). Metaheuristics Form Design to Implementation. *John Wiley and Sons*.

[12] Lu,Q. & Qiu, X. (2010). An Improved Probability Particle Swarm Optimization Algorithm. *Advances in Swarm Intelligence Lecture Notes in Computer Science* , 6145(1), 102-109.

[13] Singh, N. & Singh, S.B. (2013). HFPSO: Hazard Function Particle Swarm Optimization Algorithm", *Journal of computation intelligence studies* (Communicated).

[14] Singh, N., Singh, S. and Singh, S.B. (2012). HPSO: A New Version of Particle Swarm Optimization. *Journal of Artificial Intelligence*, ISSN: 2229-3965, 3(3), 123-134.

APPENDIX: CLASSICAL FUNCTIONS

Problem No.	Problems Name	Problems	Range of the Problems
1.	**Sphere**	$Min\ f(x) = \sum_{i=1}^{n} x_i^2$	Search space range $-5.12 \le x_i \le 5.12$ and Min Value is 0.
2.	**Griewank**	$Min\ f(x) = 1 + \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}})$	Search space range $-600 \le x_i \le 600$ and Min Value is 0.
3.	**Step Function**	$Min\ f(x) = 6 \times \sum_{i=1}^{n} \lvert x_i \rvert$	Search space range $-5.12 \le x_i \le 5.12$ and Min Value is 0.
4.	**Krishna Kumar**	$Min\ f(x) = \sin(x + x^2) + \sin(\frac{2}{3}x^3)$	Search space range $-5.12 \le x_i \le 5.12$ and Min Value is 0.
5.	**Zakharov's**	$Min\ f(x) = \sum_{i=1}^{n} x_i^2 + [\sum_{i=1}^{n} (\frac{i}{2})x_i]^2 + [\sum_{i=1}^{n} (\frac{i}{2})x_i]^4$	Search space range $-5.12 \le x_i \le 5.12$ and Min Value is 0.
6.	**Axis parallel hyper ellipsoid**	$Min\ f(x) = \sum_{i=1}^{n} i x_i^2$	Search space range $-5.12 \le x_i \le 5.12$ and Min Value is 0.
7.	**Dejong**	$Min\ f(x) = \sum_{i=1}^{n} (x_i^4 + rand(0,1))$	Search space range $-10 \le x_i \le 10$ and Min Value is 0.

ISSN 2394-9651
International Journal of Novel Research in Physics Chemistry & Mathematics
Vol. 3, Issue 2, pp: (32-40), Month: May - August 2016, Available at: www.noveltyjournals.com

8.	**Schwefel**	$Min\ f(x) = Max\{\|x_i\|, 1 \le i \le n\}$	Search space range $-100 \le x_i \le 100$ and Min Value is 0.
9.	Brown '3'	$Min\ f(x) = \sum_{i=1}^{n-1} [(x_i^2)(x_{i+1}^2+1) + (x_{i+1}^2)(x_i^2+1)]$	Search space range $-1 \le x_i \le 4$ and Min Value is 0.
10.	**Function '15'**	$Min\ f(x) = \sum_{i=1}^{n} (0.2x_i^2 + 0.1x_i^2 \sin 2x_i)$	Search space range $-10 \le x_i \le 10$ and Min Value is 0.
11.	**Eggcrate**	$Min\ f(x) = x_1^2 + x_2^2 + 25(\sin^2 x_1 + \sin^2 x_2)$	Search space range $-2\pi \le x_i \le 2\pi$ and Min Value is 0.
12.	**Periodic**	$Min\ f(x) = -1 + \sin^2 x_1 + \sin^2 x_2 - 0.1\exp(-x_1^2 - x_2^2)$	Search space range $-10 \le x_i \le 10$ and Min Value is 0.9
13.	**Camel back-3**	$Min\ f(x) = 2x_1^2 + 1.05x_1^4 + \frac{1}{6}x_1^6 + x_1x_2 + x_2^2$	Search space range $-5 \le x_1, x_2 \le 5$ and Min Value is 0
14.	**Aluffi-Pentini's**	$Min\ f(x) = 0.25x_1^4 - 0.5x_1^2 + 0.1x_1 + 0.5x_2^2$	Search space range $-10 \le x_i \le 10$ and Min Value is - 0.352
15.	**Powell's**	$Min\ f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$	Search space range $-10 \le x_i \le 10$ and Min Value is 0