MJFAS Malaysian Journal of Fundamental and Applied Sciences
ISSN:2289-5981 e-ISSN:2289-599X
UTM UNIVERSITI TEKNOLOGI MALAYSIA

Performance evaluation of Black Hole Algorithm, Gravitational Search Algorithm and Particle Swarm Optimization

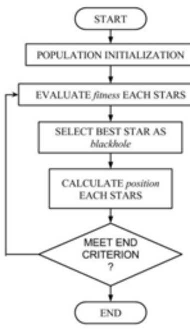
Mohamad Nizam Aliman[1], Zuwairie Ibrahim[1], Fardila Naim[1], Sophan Wahyudi Nawawi[2], Shahdan Sudin[2]

[1]*Engineering Optimization Research Group, Faculty of Electronics & Electrical Engineering, Universiti Malaysia Pahang, 26300 Kuantan, Pahang, Malaysia.*
[2]*Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia*

**Corresponding Author: mohamad.aliman@gmail.com*

Article history :
Received 3 September 2014
Accepted 16 February 2015

GRAPHICAL ABSTRACT



ABSTRACT

Particle Swarm Optimization (PSO) and Gravitational Search Algorithm are a well-known population-based heuristic optimization techniques. PSO is inspired from a motion flock of birds searching for a food. In PSO, a bird adjusts its position according to its own ‘‘experience’’ as well as the experience of other birds. Tracking and memorizing the best position encountered build bird’s experience which will leads to optimal solution. GSA is based on the Newtonian gravity and motion laws between several masses. In GSA, the heaviest mass presents an optimum solution in the search space. Other agents inside the population are attracted to heaviest mass and will finally converge to produce best solution. Black Hole Algorithm (BH) is one of the optimization technique recently proposed for data clustering problem. BH algorithm is inspired by the natural universe phenomenon called "black hole”. In BH algorithm, the best solution is selected to be the black hole and the rest of candidates which are called stars will be drawn towards the black hole. In this paper, performance of BH algorithm will be analyzed and reviewed for continuous search space using CEC2014 benchmark dataset against Gravitational Search Algorithm (GSA) and Particle Swarm Optimization (PSO). CEC2014 benchmark dataset contains 4 unimodal, 7 multimodal and 6 hybrid functions. Several common parameters has been chosen to make an equal comparison between these algorithm such as size of population is 30, 1000 iteration, 30 dimension and 30 times of experiment.

Keyword : black hole algorithm, nature inspire metaheuristic

http://dx.doi.org/xx.xxx/xxx.xxx.xxx |

1. INTRODUCTION

Optimization is an applied science which explores the best values of the parameters of a problem that may take under specified conditions [1][2]. The design of an optimization problem generally starts with the design of an objective function [3][4][5][6]. The metaheuristic optimization algorithms use two basic strategies while searching for the global optimum; exploration and exploitation [3].

There are numerous metaheuristic optimization algorithms to date. Those algorithms are Ant Colony Algorithm [7], Firefly Algorithm [8], Artificial Bee Colony [4], Cuckoo Search Algorithm [9], Harmony Search Algorithm [10]. However, in this study, swarm intelligence algorithms, which are part of metaheuristic optimization algorithms, are studied. In particular, the performance of particle swarm optimization (PSO), gravitational search algorithm (GSA), and the most recent black hole algorithm (BH) are evaluated based on the latest benchmark functions called CEC2014 benchmark functions [11]. The purpose of this study is to explore the capability of PSO, GSA, and BH algorithms and to obtain a general conclusion regarding which one is the best among others.

The paper is organized as follows: Section 2 present a brief introduction to all algorithms involved; PSO, GSA and BH. Section 3 describe about benchmark functions, common setting and parameter will be used in the experiment. The experimental result and discussion are provided in Section 4. Finally, Section 5 concludes the work.

2. ALGORITHM

2.1 Particle Swarm Optimization (PSO)

PSO is a stochastic global optimization technique inspired by social behaviour of bird flocking or fish schooling [12]. PSO uses simple mechanism observed from swarm behaviour to guide particles in search for a global optimal solution. In PSO, each particle moving inside search space with a velocity dynamically adjusted according to its own previous best position and its neighbourhood best position. Hence, every particle is representing as a potential optimal solution for the problem. Initially, each particle is randomly placed inside of *d*-dimensional search space. The *i*th particle is represented as $X_i = \left(x_i^1 \ldots x_i^d \ldots x_i^n\right)$.

At the specific time “*t*”, the velocity for *i*th particle is calculated using below formula:

$$v_i^d(t+1) = \omega(t)v_i^d(t) + c_1 rand_{i1}\left(pbest_i^d - x_i^d(t)\right) + c_2 rand_{i2}\left(gbest^d - x_i^d(t)\right) \quad (1)$$

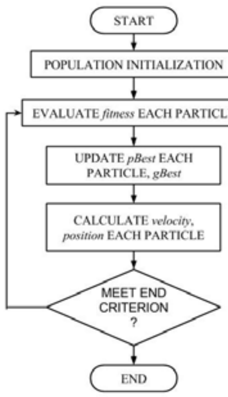


Figure 1. General principle of PSO

Where, $pbest_i$ represent best previous position of the *i*th particle and $gbest$ represent best previous position among all the particles in the population. Particle position for the next iteration is calculated as follow:

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (2)$$

The general principle of PSO is shown in Figure 1.

2.2 Gravitational Search Algorithm (GSA)

GSA has been inspired from physical phenomenon of interaction between objects in the universe. It is defined by Newton as, "*Every particle in the universe attract every other particle with a force that is directly proportional to the square of the distance between them*". This definition is known as gravitational force and is defined as follow:

$$F = \frac{GM_1M_2}{R^2} \quad (3)$$

In GSA, agents are considered as objects and their performance are expressed by their masses [3] value which calculated from specific fitness function. The population will be initialized by placing the agent at randomly position inside search space. Assuming gravitational and inertia mass is equal, agents masses are calculated using following equations:

$$best(t) = \max_{j\in\{1,\ldots,N\}} fit_j(t) \quad (4)$$
$$worst(t) = \min_{j\in\{1,\ldots,N\}} fit_j(t) \quad (5)$$

$$M_{ai} = M_{pi} = M_{ii} \text{ where } M_i, i = 1,2,3 \ldots N \quad (6)$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (7)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \quad (8)$$

So, at specific time "*t*", the gravitational force acting on agent "*i*" from agent "*j*" can be represent as following:

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t)\times M_{aj}(t)}{R_{ij}(t)+\varepsilon}\left(x_j^d(t) - x_i^d(t)\right) \quad (9)$$

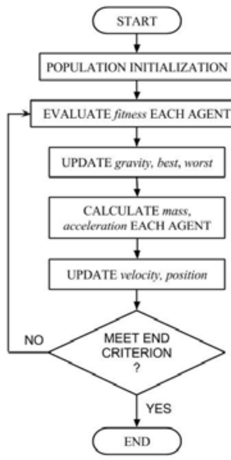


Figure 2. General principle of GSA

The Euclidian distance between two agents is:

$$R_{ij}(t) = \left\|X_i(t), X_j(t)\right\|_2 \quad (10)$$

The gravitational coefficient $G(t)$ will be reduced with time to control the search accuracy.

$$G(t) = G(G_0, t) \quad (11)$$

Following formulas has been used to determine the "*i*"th agent acceleration:

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_i F_{ij}^d(t) \quad (12)$$

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (13)$$

Then, the agent new velocity and position are calculated using these equations:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (14)$$
$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (15)$$

The general principle of GSA is shown in Figure 2.

2.3 Black Hole Algorithm (BH)

BH was created from a black hole phenomenon. It was first intended to be use as an alternative for clustering problem. Black hole phenomenon has been named by John Wheeler an American Physicist in 1967. It is a space having a huge gravitational power in which anything crosses the boundary will be swallowed even the light. A black hole in space is what forms when a star of massive size collapses [13].

Imagined an *n*-population of stars with initially placed inside of *d*-dimensional search space as follow:

$$X_i = \left(x_i^1 \ldots x_i^d \ldots x_i^n\right) \quad (16)$$

Then, each star will be evaluated using chosen fitness function. Star having the best fitness value will be selected to become a *black hole*. The new position for all stars will be calculated using this formula:

$$x_i^d(t+1) = x_i^d(t) + rand \times \left(x_{BH}^d - x_i^d(t)\right) \tag{17}$$

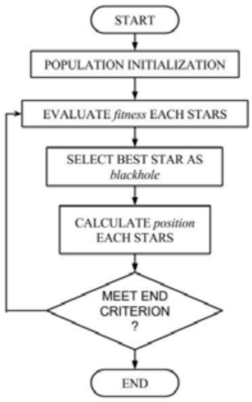


Figure 3. Standard BH flowchart

The new star will be created at random location every time it cross *event horizon*. *Event horizon* is a radius boundary around black hole. The value is determined by the following formula:

$$R = \frac{fitness_{BH}}{\sum_{i=1}^{N} fitness_i} \; where\ N = population\ size \tag{18}$$

The general principle of BH is shown in Figure 3.

3. EXPERIMENTS, RESULTS AND DISCUSSION

The parameter setting of PSO, GSA, and BH are shown in Table 1. Number of run, number of iteration, population size, and number of dimension are general parameter applied to all algorithms. Inertia weight (ω), cognitive (c_1) and social (c_2) parameters are exclusive to PSO. On the other hand, initial gravitation constant (G_0) and alpha (α), are exclusive to GSA.

The experiments were based on the recently published benchmark functions called CEC2014 benchmark functions [11]. The formulation of the CEC2014 benchmark functions are listed in Table 6 and Table 7. The benchmark functions are divided into unimodal, multimodal, and hybrid functions with optimal value are also included.

Table 1 Parameter setting

	Parameter	Value
General	Number of run per experiment	30
	Number of iteration per run	1000
	Population size	30
	Number of dimension	30
PSO	Inertia Weight, ω	0.9 to 0.4
	Coefficient Factor, c_1,c_2	2, 2
GSA	Initial Gravity, G_0	100
	Alpha, α	-20

The average value, standard deviation value, minimum value, and maximum value from experimental result were recorded and tabulated in Table 2, Table 3, Table 4 and Table 5. The average value is being used as comparison between algorithms and value written in bold indicates the best result among them.

Table 2 Unimodal function result

Function	Measure	BH	GSA	PSO
F1	AVERAGE	88570364	**74838854**	96722332
	STDDEV	31842233	23787602	71277779
	MIN	25102312	38704969	25102312
	MAX	1.51E+08	1.24E+08	4.06E+08
F2	AVERAGE	1.36E+09	3.29E+08	**57729922**
	STDDEV	1.89E+09	4.34E+08	96226073
	MIN	226617.4	226617.4	1005150
	MAX	8.00E+09	2.13E+09	4.08E+08
F3	AVERAGE	43136.29	76896.34	**14178.49**
	STDDEV	27926.54	5996.396	14777.49
	MIN	1150.474	64485.01	1150.474
	MAX	48712.46	87213.57	59125.39

Table 3 Multimodal function result.

Function	Measure	BH	GSA	PSO
F4	AVERAGE	1046.042	**976.161**	1031.017
	STDDEV	277.3205	289.4972	262.6738
	MIN	578.9538	732.3658	578.9538
	MAX	2136.106	2177.511	1513.875
F5	AVERAGE	520.306	**519.999**	520.9736
	STDDEV	0.453085	0.000556	0.101111
	MIN	519.9975	519.9975	520.6635
	MAX	520.0637	519.9997	521.0941
F6	AVERAGE	624.9259	624.7608	**617.3393**
	STDDEV	6.816158	2.449066	4.123768
	MIN	610.2717	621.1697	610.2717
	MAX	638.7539	629.5551	626.4682
F7	AVERAGE	720.3244	**706.2988**	710.0631
	STDDEV	19.22478	3.606268	11.96609
	MIN	701.0355	701.0355	701.2862
	MAX	767.6485	714.0935	747.8384

For unimodal functions, GSA is better than PSO and BH for F1 function. However, PSO is better than GSA and BH for F2 and F3 functions while BH was not able to outperform PSO and GSA in all cases. The examples of boxplot and convergence curves for F1, F2, and F3 functions are shown in Figure 4, Figure 5, and Figure 6 respectively.

Similar to unimodal functions, multimodal functions also shows that BH was not able to outperform PSO and GSA in all cases. Further comparisons of PSO and GSA show that GSA outperformed PSO in 7 cases (F4, F5, F6, F7, F8, F9, and F10). On the other hand, PSO outperformed GSA in 6 cases (F11, F12, F13, F14, F15, and F16). The examples of boxplot and convergence curves for F4 to F16 are shown in Figure 7 to Figure 18 respectively.

Table 4 Multimodal function result.

Function	Measure	BH	GSA	PSO
F8	AVERAGE	913.3048	947.4192	**854.8536**
	STDDEV	42.41577	9.564984	13.18151
	MIN	834.6769	929.3442	834.6769
	MAX	971.3446	963.1726	882.0581
F9	AVERAGE	1110.987	1077.898	**1013.347**
	STDDEV	97.44744	20.3977	39.0376
	MIN	959.6918	1043.274	959.6918
	MAX	1268.172	1127.844	1095.535
F10	AVERAGE	3481.428	4617.347	**2027.867**
	STDDEV	1132.673	425.2253	371.118
	MIN	1262.792	3669.495	1262.792
	MAX	4331.487	5230.704	2759.996
F11	AVERAGE	5870.741	**5350.104**	7166.539
	STDDEV	1196.285	352.6554	1081.404
	MIN	3860.466	4685.935	3984.792
	MAX	6526.11	6189.264	8707.647
F12	AVERAGE	1201.036	**1200.016**	1202.343
	STDDEV	1.03915	0.008515	0.536511
	MIN	1200.005	1200.005	1201.291
	MAX	1201.644	1200.044	1203.383
F13	AVERAGE	1300.821	**1300.515**	1300.987
	STDDEV	0.633971	0.373776	0.708315
	MIN	1300.203	1300.226	1300.203
	MAX	1302.185	1302.164	1302.704
F14	AVERAGE	1412.875	**1404.689**	1419.276
	STDDEV	13.07767	9.784895	16.72074
	MIN	1400.201	1400.201	1400.207
	MAX	1439.243	1435.619	1453.36
F15	AVERAGE	1574.77	1554.118	**1520.718**
	STDDEV	62.15823	22.23178	5.288553
	MIN	1508.884	1522.382	1508.884
	MAX	1811.993	1597.114	1534.195
F16	AVERAGE	1612.951	1613.61	**1612.529**
	STDDEV	0.635518	0.225436	0.45425
	MIN	1611.25	1613.185	1611.37
	MAX	1613.535	1613.975	1613.24

Table 5 Hybrid function result.

Function	Measure	BH	GSA	PSO
F17	AVERAGE	30784257	55283638	**19729686**
	STDDEV	44065650	67306551	20705286
	MIN	1335720	3167296	1335720
	MAX	25736003	3.28E+08	1.04E+08
F18	AVERAGE	6.38E+08	1.76E+09	**32246294**
	STDDEV	1.19E+09	1.51E+09	1.45E+08
	MIN	2075.141	2519.887	2075.141
	MAX	1.43E+08	5.06E+09	7.88E+08
F19	AVERAGE	2045.871	2105.174	**2021.973**
	STDDEV	63.20811	42.23975	64.39642
	MIN	1922.153	2001.32	1922.153
	MAX	2042.365	2222.969	2184.441
F20	AVERAGE	57585.45	91579.14	**41567.73**
	STDDEV	30545.26	24091.65	16684.78
	MIN	18416.79	52543.42	20306.05
	MAX	65022.72	153681.4	96529.06
F21	AVERAGE	3824788	4337785	**3445418**
	STDDEV	2321736	1783216	3219689
	MIN	612435.2	1888554	612435.2
	MAX	7217596	8681296	17397691
F22	AVERAGE	4341.088	4544.53	**4169.702**
	STDDEV	537.9198	462.2675	629.633
	MIN	2967.647	3832.037	2967.647
	MAX	5000.246	6206.148	5370.399

As for hybrid cases, the result show that PSO is superior to GSA and BH in all cases. The examples of boxplot and convergence curves for F17, F18, F19, F20, F21, and F22 are shown in Figure 19 to Figure 24, respectively.

4. CONCLUSIONS

This study considers three different swarm intelligence algorithms, namely PSO, GSA, and BH. The purpose of this study is to evaluate the superiority of these algorithms when finding the optimal solution based on CEC2014 benchmark functions.

By observing the results produced based on the unimodal, multimodal, and hybrid functions of CEC2014, it can be concluded that briefly, both PSO and GSA perform well in solving unimodal and multimodal optimization problems. However, for the case of hybrid optimization problem, PSO is superior to GSA and BH for all cases. The next step of this research are to re-execute similar experiments for high-dimensional optimization problem and to perform a detailed statistical analysis in order to obtain a more concrete conclusion as well as to further understand the behaviour of the PSO, GSA, and BH algorithms.

ACKNOWLEDGMENT

This work is financially supported by the RAGS (RDU121403) awarded by the Ministry of Higher Education (MOHE) to Universiti Malaysia Pahang (UMP). The first author is thankful to Universiti Malaysia Pahang (UMP) for granting him an opportunity to further his study in postgraduate program in UMP.

References

[1] D. Corne, M. Dorigo, and F. Glover, *New Ideas in Optimization*. McGraw-Hill, 1999.

[2] R. Horst, P. M. Pardalos, and N. V. Thoai, *Introduction to global optimization*, 2nd ed. Dordrecht, The Netherland: Kluwer Academic Publishers, 2000.

[3] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, “GSA: A Gravitational Search Algorithm,” *Inf. Sci. (Ny).*, vol. 179, no. 13, pp. 2232–2248, 2009.

[4] D. Karaboga and B. Akay, “A comparative study of Artificial Bee Colony algorithm,” *Appl. Math. Comput.*, vol. 214, no. 1, pp. 108–132, 2009.

[5] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. G. Harley, “Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems,” *Evol. Comput. IEEE Trans.*, vol. 12, no. 2, pp. 171–195, 2008.

[6] R. Storn and K. Price, “Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces,” *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[7] M. Dorigo, M. Birattari, and T. St, “Ant Colony Optimization,” *IEEE Computational Intelligence Magazine*, no. November, pp. 28–39, 2006.

[8] X. Yang, “Firefly Algorithms for Multimodal Optimization,” in *Stochastic Algorithms: Foundations and Applications*, 2009, pp. 169–178.

[9] R. Rajabioun, “Cuckoo Optimization Algorithm,” *Appl. Soft Comput.*, vol. 11, no. 8, pp. 5508–5518, Dec. 2011.

[10] G. V. Loganathan, “A New Heuristic Optimization Algorithm: Harmony Search,” *Simulation*, vol. 76, no. 2, pp. 60–68, Feb. 2001.

[11] J. J. Liang, B. Y. Qu, and P. N. Suganthan, *Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization*, no. December 2013. 2013.

[12] J. Kennedy and R. Eberhart, “Particle Swarm Optimization,” *IEEE Int. Conf. Neuron Networks*, pp. 1942–1948, 1995.

[13] A. Hatamlou, “Black hole: A new heuristic optimization approach for data clustering,” *Inf. Sci. (Ny).*, vol. 222, pp. 175–184, Feb. 2013.

Table 6 CEC2014 basic function.

Function	Equation
High Conditioned Elliptic	$f_1(\mathbf{x})=\sum_{i=1}^{D}(10^6)^{\frac{i-1}{D-1}}x_i^2$
Bent Cigar	$f_2(\mathbf{x})=x_1^2+10^6\sum_{i=2}^{D}x_i^2$
Discuss	$f_3(\mathbf{x})=10^6x_1^2+\sum_{i=2}^{D}x_i^2$
Rosenbrock	$f_4(\mathbf{x})=\sum_{i=1}^{D-1}(100(x_i^2-x_{i+1})^2+(x_i-1)^2)$
Ackley	$f_5(\mathbf{x})=-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2})-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i))+20+e$
Weiestrass	$f_6(\mathbf{x})=\sum_{i=1}^{D}(\sum_{k=0}^{k\max}[a^k\cos(2\pi b^k(x_i+0.5))])-D\sum_{k=0}^{k\max}[a^k\cos(2\pi b^k\cdot 0.5)]$
Griewank	$f_7(\mathbf{x})=\sum_{i=1}^{D}\frac{x_i^2}{4000}-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})+1$
Rastrigin	$f_8(\mathbf{x})=\sum_{i=1}^{D}(x_i^2-10\cos(2\pi x_i)+10)$
Schwefel	$f_9(\mathbf{x})=418.9829\times D-\sum_{i=1}^{D}g(z_i),\quad z_i=x_i+4.209687462275036\text{e}+002$ $g(z_i)=\begin{cases} z_i\sin(\lvert z_i\rvert^{1/2}) & \text{if } \lvert z_i\rvert\le 500 \\ (500-\text{mod}(z_i,500))\sin(\sqrt{\lvert 500-\text{mod}(z_i,500)\rvert})-\frac{(z_i-500)^2}{10000D} & \text{if } z_i>500 \\ (\text{mod}(\lvert z_i\rvert,500)-500)\sin(\sqrt{\lvert\text{mod}(\lvert z_i\rvert,500)-500\rvert})-\frac{(z_i+500)^2}{10000D} & \text{if } z_i<-500 \end{cases}$
Katsuura	$f_{10}(\mathbf{x})=\frac{10}{D^2}\prod_{i=1}^{D}(1+i\sum_{j=1}^{32}\frac{\lvert 2^jx_i-round(2^jx_i)\rvert}{2^j})^{\frac{10}{D^{1.2}}}-\frac{10}{D^2}$
HappyCat	$f_{11}(\mathbf{x})=\left\lvert\sum_{i=1}^{D}x_i^2-D\right\rvert^{1/4}+(0.5\sum_{i=1}^{D}x_i^2+\sum_{i=1}^{D}x_i)/D+0.5$
HGBat	$f_{12}(\mathbf{x})=\left\lvert(\sum_{i=1}^{D}x_i^2)^2-(\sum_{i=1}^{D}x_i)^2\right\rvert^{1/2}+(0.5\sum_{i=1}^{D}x_i^2+\sum_{i=1}^{D}x_i)/D+0.5$
Expanded Griewank plus Ronsenbrock	$f_{13}(\mathbf{x})=f_7(f_4(x_1,x_2))+f_7(f_4(x_2,x_3))+...+f_7(f_4(x_{D-1},x_D))+f_7(f_4(x_D,x_1))$
Expanded Scaffer's F6	$f_{14}(\mathbf{x})=g(x_1,x_2)+g(x_2,x_3)+...+g(x_{D-1},x_D)+g(x_D,x_1)$ $g(x,y)=0.5+\frac{(\sin^2(\sqrt{x^2+y^2})-0.5)}{(1+0.001(x^2+y^2))^2}$
Hybrid	$F(\mathbf{x})=g_1(\mathbf{M}_1z_1)+g_2(\mathbf{M}_2z_2)+...+g_N(\mathbf{M}_Nz_N)+F^*(\mathbf{x})$ $F(\mathbf{x})$: hybrid function $g_i(\mathbf{x})$: i^{th} basic function used to construct the hybrid function N: number of basic functions $\mathbf{z}=[\mathbf{z}_1,\mathbf{z}_2,...,\mathbf{z}_N]$ $\mathbf{y}=\mathbf{x}-\mathbf{o}_i, S=randperm(1:D)$ $\mathbf{z}_1=[\mathbf{y}_{S_1},\mathbf{y}_{S_2},...,\mathbf{y}_{S_{n_1}}],\mathbf{z}_2=[\mathbf{y}_{S_{n_1+1}},\mathbf{y}_{S_{n_1+2}},...,\mathbf{y}_{S_{n_1+n_2}}],...,\mathbf{z}_N=[\mathbf{y}_{S_{\sum_{i=1}^{N-1}n_i+1}},\mathbf{y}_{S_{\sum_{k=1}^{N-1}n_i+2}},...,\mathbf{y}_{S_D}]$ $n_1=\lceil p_1D\rceil,n_2=\lceil p_2D\rceil,...,n_{N-1}=\lceil p_{N-1}D\rceil,n_N=D-\sum_{i=1}^{N-1}n_i$ p_i: used to control the percentage of $g_i(\mathbf{x})$ n_i: dimension for each basic function $\sum_{i=1}^{N}n_i=D$

Table 7 CEC2014 benchmarking test function.

Type	Function	Equation	Optimal
Unimodal	Rotated High Conditioned Elliptic	$F_1(\boldsymbol{x}) = f_1(\mathbf{M}(\boldsymbol{x}-\boldsymbol{o}_1)) + F_1{*}$	100
	Rotated Bent Cigar	$F_2(\boldsymbol{x}) = f_2(\mathbf{M}(\boldsymbol{x}-\boldsymbol{o}_2)) + F_2{*}$	200
	Rotated Discuss	$F_3(\boldsymbol{x}) = f_3(\mathbf{M}(\boldsymbol{x}-\boldsymbol{o}_3)) + F_3{*}$	300
Multimodal	Shifted and rotated Rosenbrock	$F_4(\boldsymbol{x}) = f_4(\mathbf{M}(\frac{2.048(x-o_4)}{100})+1) + F_4{*}$	400
	Shifted and rotated Ackley	$F_5(\boldsymbol{x}) = f_5(\mathbf{M}(\boldsymbol{x}-\boldsymbol{o}_5)) + F_8{*}$	500
	Shifted and rotated Weiestrass	$F_6(\boldsymbol{x}) = f_6(\mathbf{M}(\frac{0.5(\boldsymbol{x}-\boldsymbol{o}_6)}{100})) + F_6{*}$	600
	Shifted and rotated Griewank	$F_7(\boldsymbol{x}) = f_7(\mathbf{M}(\frac{600(\boldsymbol{x}-\boldsymbol{o}_7)}{100})) + F_7{*}$	700
	Shifted Rastrigin	$F_8(\boldsymbol{x}) = f_8(\frac{5.12(\boldsymbol{x}-\boldsymbol{o}_8)}{100}) + F_7{*}$	800
	Shifted and rotated Rastrigin	$F_9(\boldsymbol{x}) = f_8(\mathbf{M}(\frac{5.12(\boldsymbol{x}-\boldsymbol{o}_9)}{100})) + F_9{*}$	900
	Shifted Schwefel	$F_{10}(\boldsymbol{x}) = f_9(\frac{1000(\boldsymbol{x}-\boldsymbol{o}_{10})}{100}) + F_{10}{*}$	1000
	Shifted and rotated Schwefel	$F_{11}(\boldsymbol{x}) = f_9(\mathbf{M}(\frac{1000(\boldsymbol{x}-\boldsymbol{o}_{11})}{100})) + F_{11}{*}$	1100
	Shifted and rotated Katsuura	$F_{12}(\boldsymbol{x}) = f_{10}(\mathbf{M}(\frac{5(\boldsymbol{x}-\boldsymbol{o}_{12})}{100})) + F_{12}{*}$	1200
	Shifted and rotated Happycat	$F_{13}(\boldsymbol{x}) = f_{11}(\mathbf{M}(\frac{5(\boldsymbol{x}-\boldsymbol{o}_{13})}{100})) + F_{13}{*}$	1300
	Shifted and rotated HgBat	$F_{14}(\boldsymbol{x}) = f_{12}(\mathbf{M}(\frac{5(\boldsymbol{x}-\boldsymbol{o}_{14})}{100})) + F_{14}{*}$	1400
	Shifted and rotated Expanded Griewank-Rosenbrock	$F_{15}(\boldsymbol{x}) = f_{13}(\mathbf{M}(\frac{5(\boldsymbol{x}-\boldsymbol{o}_{15})}{100})+1) + F_{15}{*}$	1500
	Shifted and rotated Expanded Scaffer F6	$F_{16}(\boldsymbol{x}) = f_{14}(\mathbf{M}(\boldsymbol{x}-\boldsymbol{o}_{16})+1) + F_{16}{*}$	1600
Hybrid	Hybrid 1 (N=3)	$p = [0.3,0.3,0.4]$ g_1: Modified Schwefel's Function f_9 g_2: Rastrigin's Function f_8 g_3: High Conditioned Elliptic Function f_1	1700
	Hybrid 2 (N=3)	$p = [0.3, 0.3, 0.4]$ g_1: Bent Cigar Function f_2 g_2: HGBat Function f_{12} g_3: Rastrigin's Function f_8	1800
	Hybrid 3 (N=4)	$p = [0.2, 0.2, 0.3, 0.3]$ g_1: Griewank's Function f_7 g_2: Weierstrass Function f_6 g_3: Rosenbrock's Function f_4 g_4: Scaffer's F6 Function: f_{14}	1900
	Hybrid 4 (N=4)	$p = [0.2, 0.2, 0.3, 0.3]$ g_1: HGBat Function f_{12} g_2: Discus Function f_3 g_3: Expanded Griewank's plus Rosenbrock's Function f_{13} g_4: Rastrigin's Function f_8	2000
	Hybrid 5 (N=5)	$p = [0.1, 0.2, 0.2, 0.2, 0.3]$ g_1: Scaffer's F6 Function: f_{14} g_2: HGBat Function f_{12} g_3: Rosenbrock's Function f_4 g_4: Modified Schwefel's Function f_9 g_5: High Conditioned Elliptic Function f_1	2100
	Hybrid 6 (N=5)	$p = [0.1, 0.2, 0.2, 0.2, 0.3]$ g_1: Katsuura Function f_{10} g_2: HappyCat Function f_{11} g_3: Expanded Griewank's plus Rosenbrock's Function f_{13} g_4: Modified Schwefel's Function f_9 g_5: Ackley's Function f_5	2200

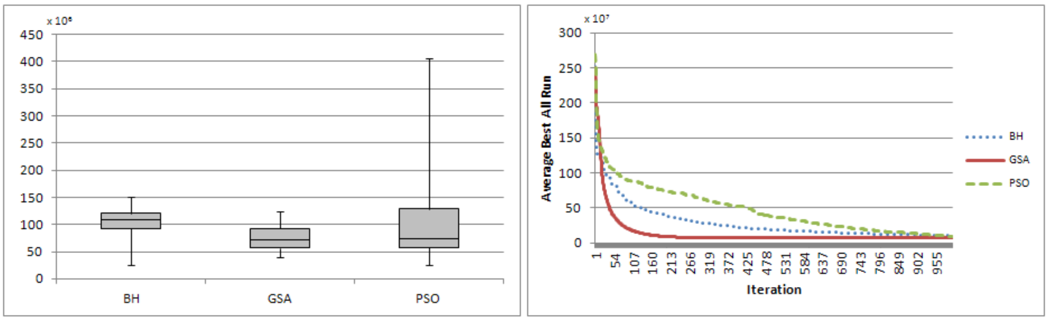


Figure 4 Rotated High Conditioned Elliptic function

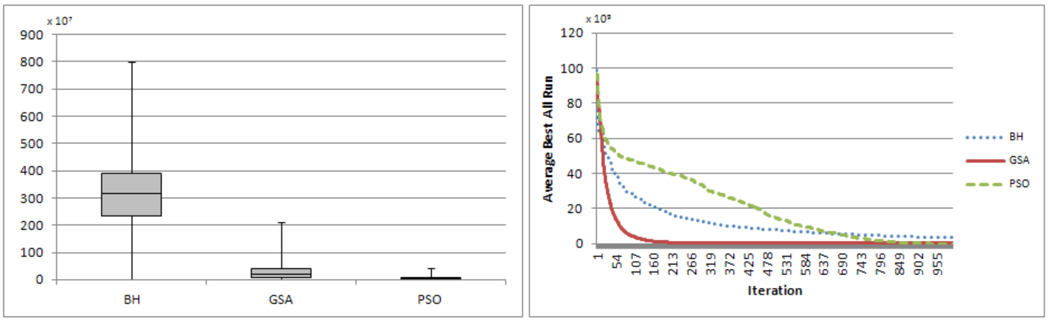


Figure 5 Rotated Bent Cigar function

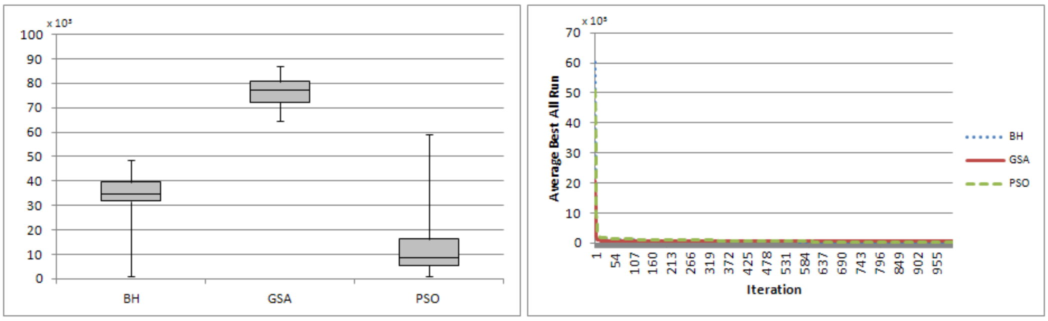


Figure 6: Rotated Discuss function

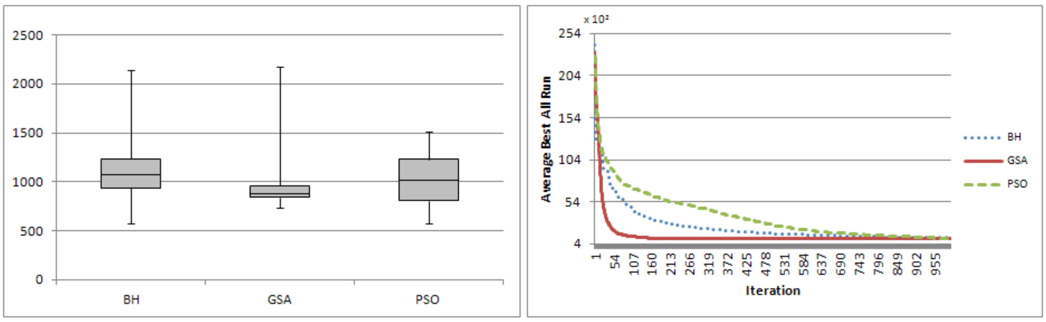


Figure 7 Shifted and rotated Rosenbrock function

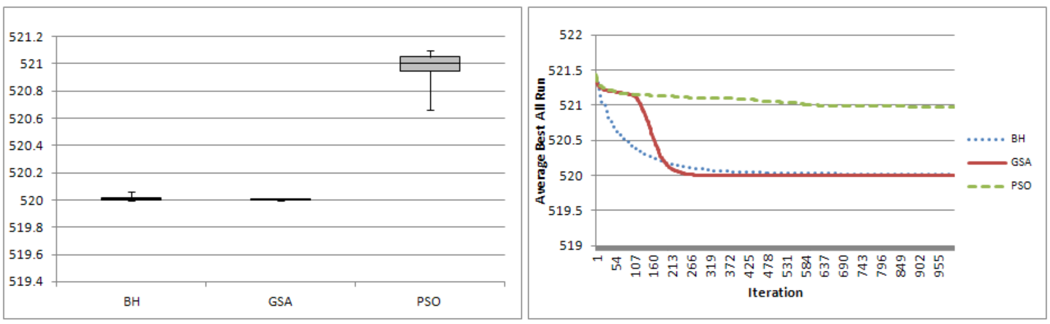


Figure 8 Shifted and rotated Ackley function

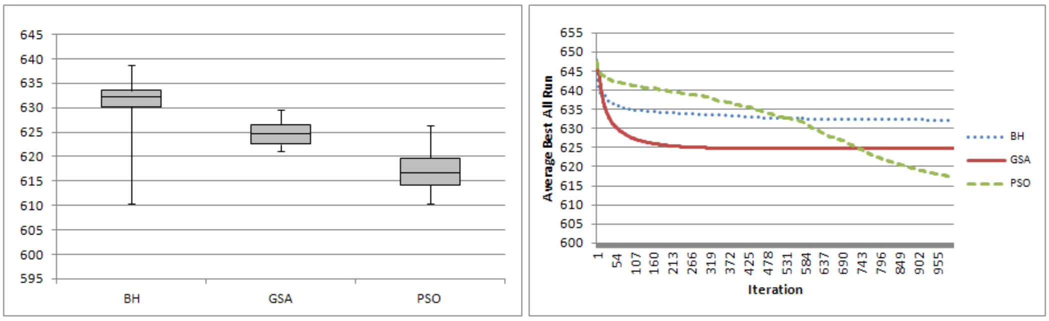


Figure 9 Shifted and rotated Weiestrass function

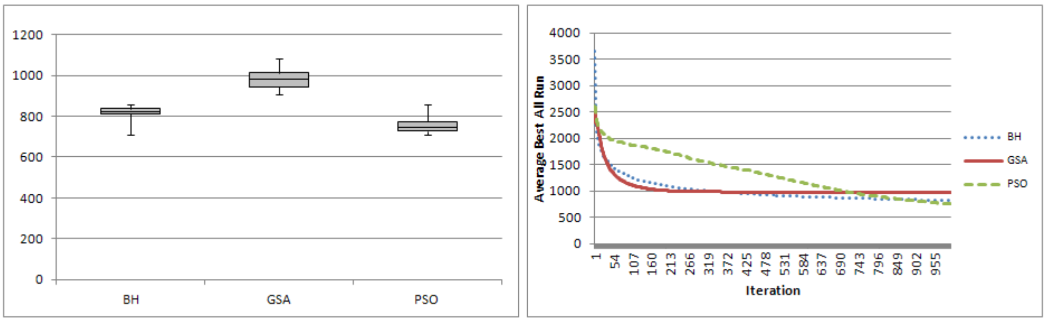


Figure 10 Shifted and rotated Griewank function

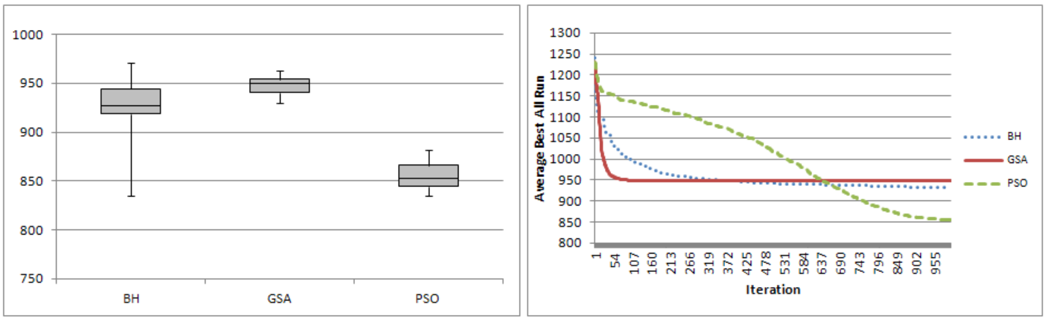


Figure 11 Shifted Rastrigin function

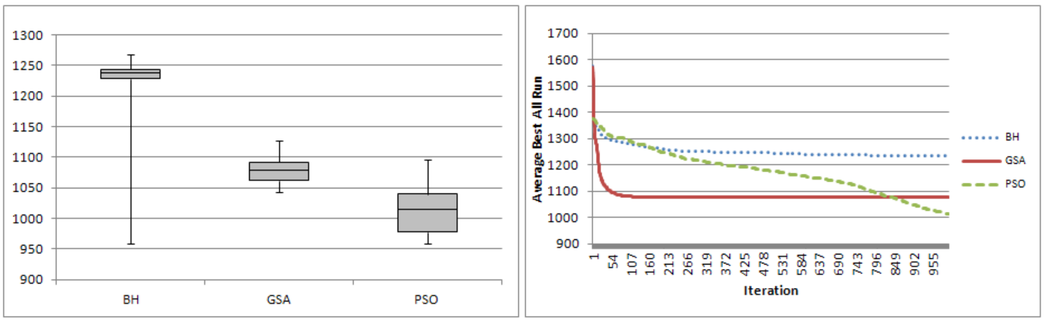


Figure 12 Shifted and rotated Rastrigin function

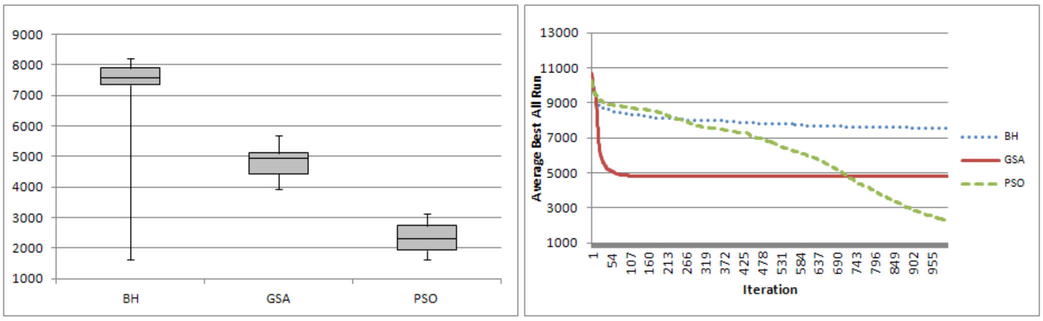


Figure 13 Shifted Schwefel function

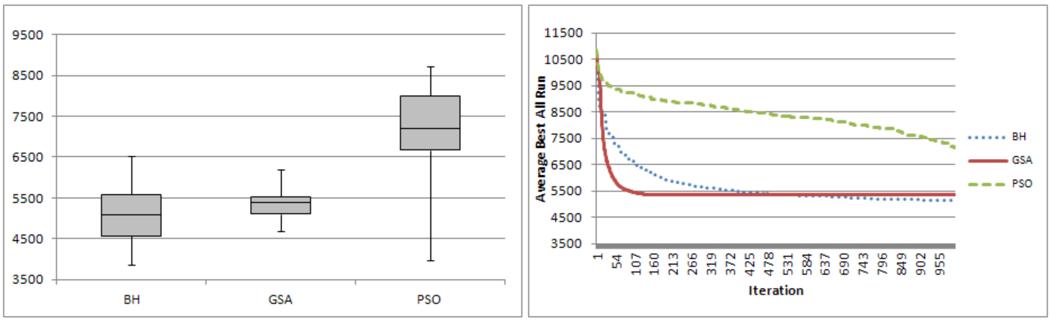


Figure 14 Shifted and rotated Schwefel function

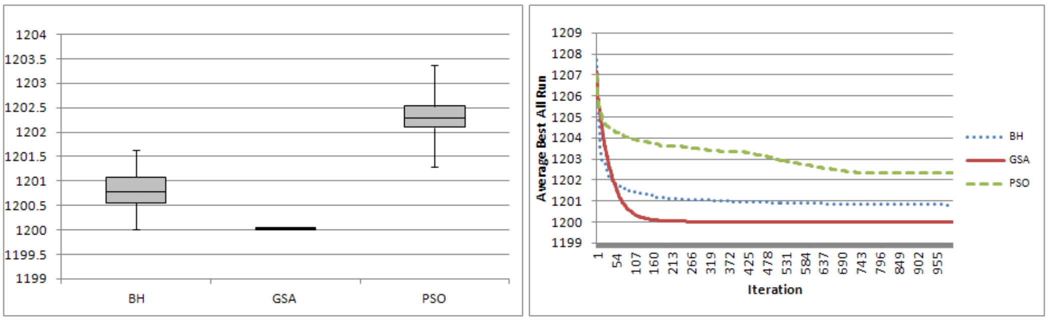


Figure 15 Shifted and rotated Katsuura function

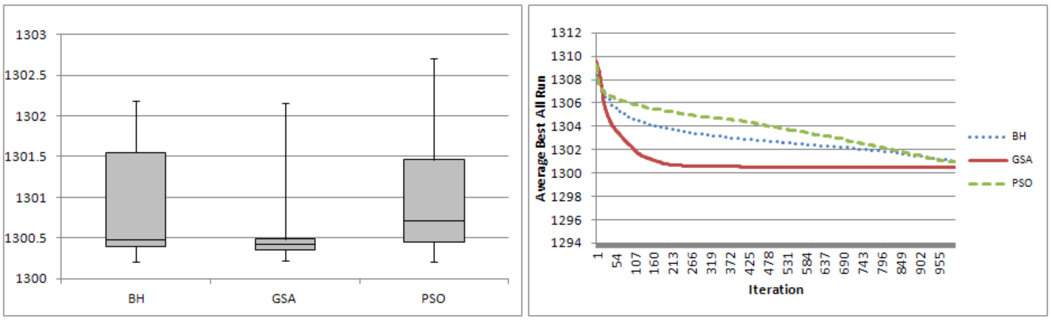


Figure 16 Shifted and rotated HappyCat function

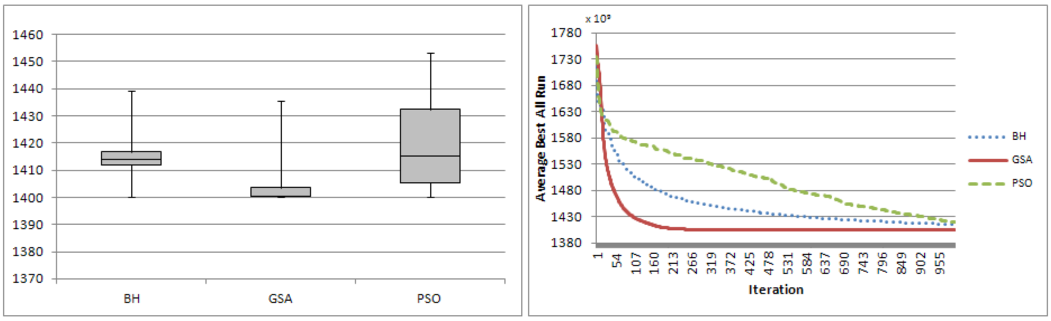


Figure 17 Shifted and rotated HgBat function

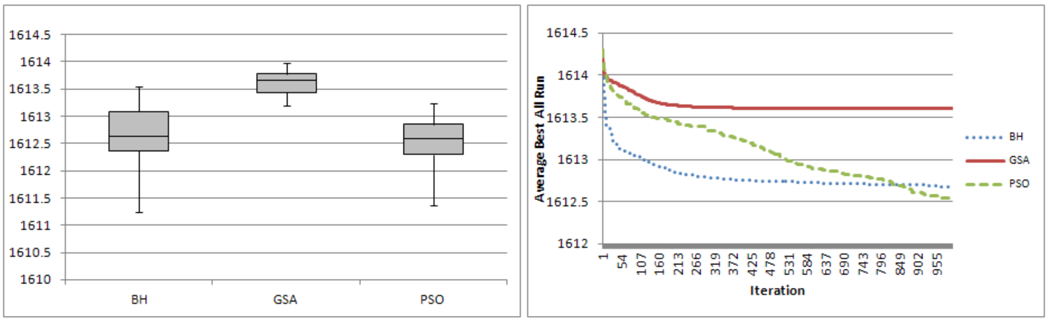


Figure 18 Shifted and rotated Expanded Scaffer F6 function

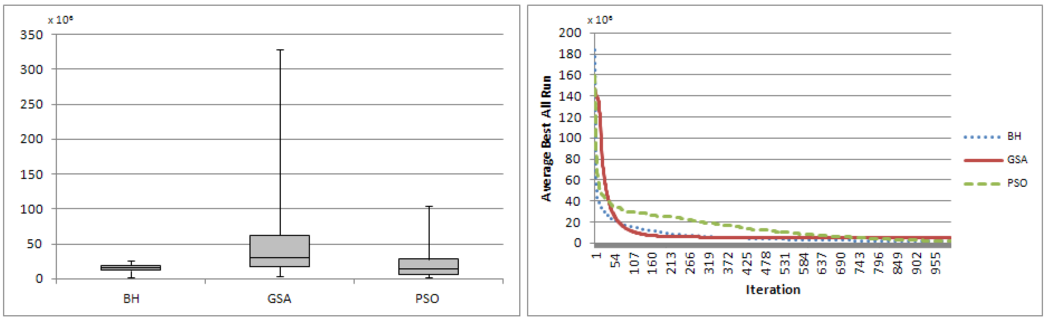


Figure 19 Hybrid Function 1 (N=3)

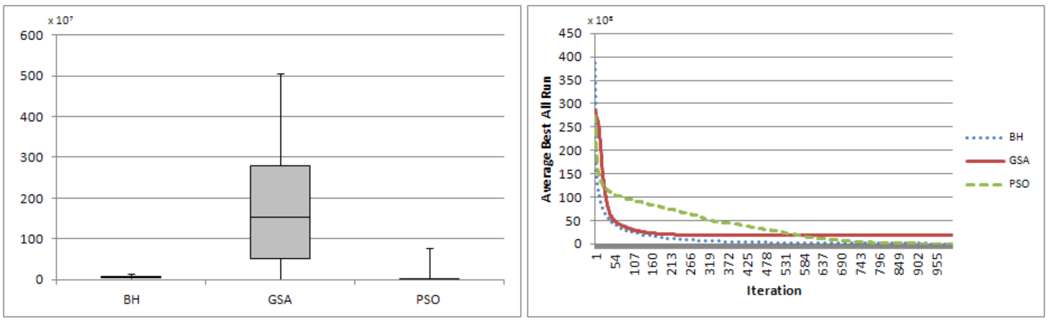


Figure 20 Hybrid Function 2 (N=3)

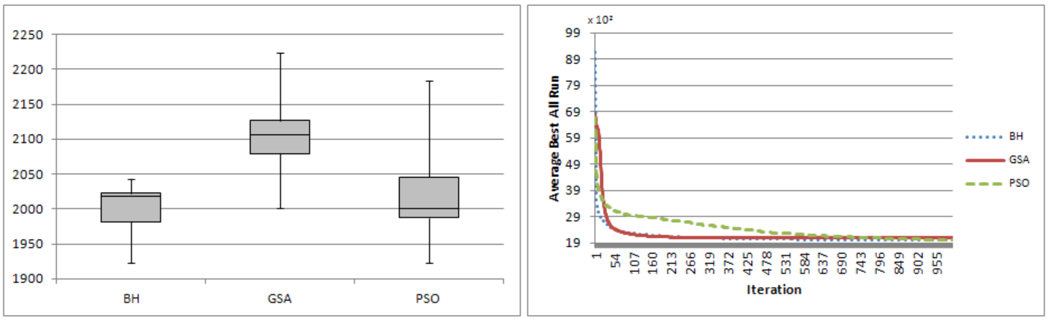


Figure 21 Hybrid Function 3 (N=4)

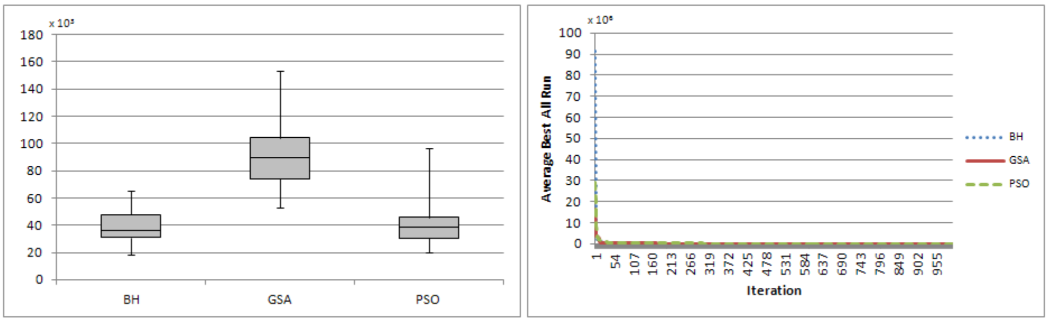


Figure 22 Hybrid Function 4 (N=4)

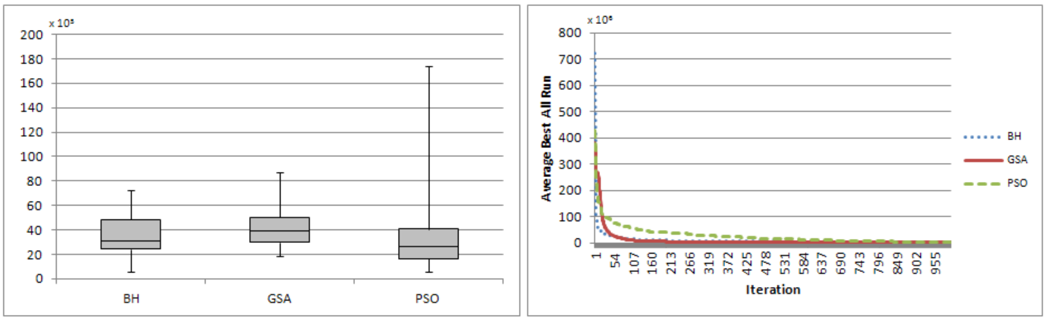


Figure 23 Hybrid Function 5 (N=5)

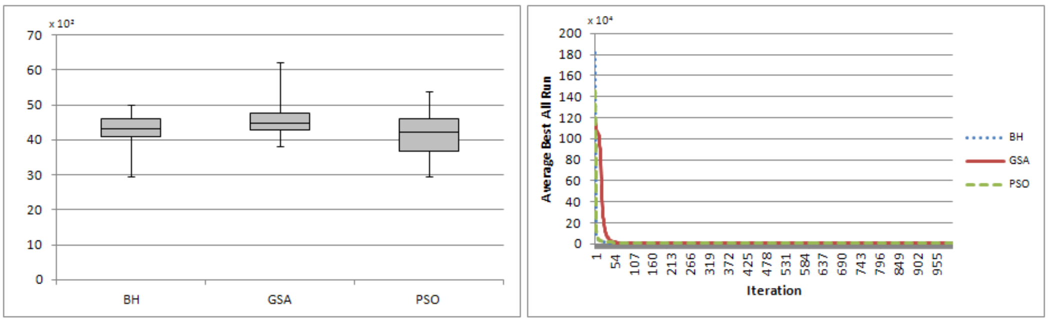


Figure 24 Hybrid Function 6 (N=5)