Sindh Univ. Res. Jour. (Sci. Ser.) Vol.49 (004) 899-906 (2017)

http://doi.org/10.26692/sujo/2017.12.0079

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Hybrid Genetic Firefly Algorithm for Global Optimization Problems

M. ASIM, W. K. MASHWANI++, M. A. JAN

Kohat University of Science & Technology, KPK, Pakistan

Received 24th January 2016 and Revised 19th October 2017

Abstract: Global Optimization is an active area of research for the variety of optimization problems that are frequently arising in network design and operation, finance, supply chain management, scheduling, and many other areas. In the last few years, different types of evolutionary algorithms (EAs) have been proposed for solving and analyzing the properties of diverse types of optimization problems. EAs work with a set of random solutions called population and find a set of optimal solutions for the problems at hand in a single simulation run opponent to traditional optimization methods. Among the stochastic based algorithms, genetic algorithm (GA) is one of the most popular and frequently used stochastic based meta-heuristic inspired by natural evolution. The premature convergence, genetic drift and trapping in the local basin attraction are their major drawbacks. These issues can be overcome by hybridizing GA with some efficient local search optimizers and different search operators. In this paper, we have proposed hybrid GA by employing the Firefly Algorithm (FA) as search operator aiming at to improve the searching ability of the baseline GA. The performance of the suggested hybrid genetic firefly algorithm (HGFA) is hereby evaluated by using 24 benchmark functions which was designed for the special session of the 2005 IEEE Congress on Evolutionary Computation (CEC'05). The numerical results provided by HGFA are summarized in the numerical form such as best, mean and standard deviation by executing 25 times independently with different random seeds to solve each test problem. The suggested HGFA have tackled most of the used test problems with good convergence speed as compared to the stand alone Genetic Algorithm.

Keywords: Global Optimization, Single objective optimization, Evolutionary Algorithms, Genetic Algorithm, Hybridization, Firefly Algorithm.

1. INTRODUCTION

In general, the optimization problems can conventionally categorize into two groups: continuous, discrete according to the value range of their variables. They can be generally formulated as follows:

$$Minimize \quad f(x) \qquad (1)$$
$$subject\ to \begin{cases} g_j(x) \leq 0, j = 1,2,\dots p, \\ h_k(x) \leq 0, k = 1,2,\dots q, \end{cases}$$

Where $x = [x_1, x_2, \dots, x_n]^T \in R^n$ is an n-dimensional solution, p is the number of inequality constraint functions, and q is the number of equality constraint functions. Moreover, $L_i \leq x_i \leq U_i, i = 1,2,..n, L_i \& U_i$ are the lower and upper bounds of parametric space, S and the function $f(x)$ is called an objective/fitness function.

Evolutionary algorithms (EAs) are well established stochastic nature of optimization methods. They have attracted much attention due population based nature across the globe for solving both real world problems and benchmark functions of IEEE-CEC. The existing literature of ECare witnessed that most of the existing EAs have successfully tackled various types of optimization and search problems (Eiben and James, 2015). EAs employ evolutionary operators such as mutation, crossover and selection simultaneously for population evolution. In general, classical EAs can be divided into four main paradigms, namely, Genetic Algorithms (GAs) (Holland, 1973), Evolution Strategies (ES) (Bäck, *et al.*, 1991), Evolutionary Programming (EP) and Genetic Programming (GP) (Fogel, *et al.*, 1966). The algorithmic behavior of all these of EA's paradigms are mainly determined with exploitation and exploration relationship kept throughout the all simulation runs. These issues can efficiently tackle with hybridization of EAs with local search optimizers for the purposes to improve the performance of the basic evolutionary approach in terms of exploration and exploitation (Mashwani, 2014).

Recently, various local search techniques and search operators have been incorporated in the framework of the EAs and as resultant many Hybrid EAs are developed (El-Mihoub, 2006). They are successfully applied on several test suites of optimization problems and different real world problems having complicated search spaces, noisy environment, imprecision, uncertainty, and vagueness. (Grosan, *et al.*, 2007).

This paper proposes a hybrid version of GA by employing firefly algorithm (FA) (Yang, 2008), (Zhang, *et al.*, 2016) as search operator to evolve the population to improve the slower convergence speed of

++Corresponding author: mashwanigr8@gmail.com

the basic GA on test suite of 2005 IEEE Congress on Evolutionary Computation (IEEE-CEC05) (Suganthan, *et al.*, 2005). The suggested hybrid genetic firefly algorithm (HGFA) has tackled most of the test problems

The primary purpose for a firefly's flash is to act as a signal system to attract other fireflies. Xin-She Yang formulated this firefly algorithm as explained in the Algorithm 2. with better convergence speed as compared to basic basic GA.

The rest of the paper is organized as under. Section 2 explains the algorithmic structure of the proposed HGFA. Section 3presents the discussion regarding the obtained experimental results. Section 4 finally concludes this paper with brief future plan.

2. HYBRID GENETIC FIREFLY ALGORITHM (HGFA)

In this paper, we have implanted FA in the original framework of GA as given in Algorithm 1 and as resultant hybrid Genetic Firefly Algorithm denoted by HGFA is developed as explained in the Algorithm 3. The proposed algorithm HGFA engages FA at generation of multiple five to perform population evolution and search process in combination with basic GA at step 5 of the main Algorithm1.

Algorithm1: Framework of the HGFA

```
1: DefineG N, n, MaxG; Total function evaluations.
2: x= {x^1,...,x^N}^T ← Initialize Pop(N, n);
3: f(x) = {f(x^1),...,f(x^N)}←Eval({x^1,...,x^N}^T);
4: while G < MAX_G do
5:     if rem(G, 5) = 0 then
6:         [x^C, f(x^c)] = FA(x, f(x), tol, G);
7:     else
8: for j ← 1: N(all i fireflies) do
9:             if I_i>I_j then
10:                x_j ← x_j move firefly i towards j;
11:            end if
12:            x^P, f^P← Select-parents(x, f);
13:            x^q, f^P← Xovers(x^p, f);
14:            x^C, f^P← Mutation(x^q, f);
15:            x^C, f^C← Eval(x^c, f);
16:    end if
17:        x← x^C;f← f^C
18:      G = G + 1;
19: end while
```

2.1 Firefly Algorithm (FA)

FA was first developed by Xin-She Yang inspired by the flashing patterns and behavior of the fireflies (Yang, 2008).

Algorithm2: Framework of FA

```
x = [x^1, x^2..., x^N];
    f(x) = [f(x^1), f(x^2) ..., f(x^N)];
.I = f(x) is the light intensity.
γ is the absorption coefficient.
while (t <FES) do
        for i ← 1: N (all N fireflies) do
            for j ← 1: N(all i fireflies) do
                if I_i>I_j then
                    x_j ← x_j move firefly i towards j;
            end if
Attractiveness varies with distance r via exp(−γr), where γ ⟶ 0.
Evaluate new solutions and update light intensity I
        end for j
    end for i
Rank the fireflies and find the current best
    t = t + 1;
end while
```

The proposed algorithm HGFA calls FA in 5th multiple of generation in combination with GA that can see at step 5 of the Algorithm 1. The main update formula of FA for any pair of two fireflies x_i and x_j is

$$x_i^{t+1} = x_i^t + \beta \exp[-\gamma r_{ij}^2](x_j^t - x_i^t) + \alpha_t \mathcal{E}_t$$

Where$\alpha_t$is a parameter controlling the step size, while$\mathcal{E}_t$is a vector drawn from a Gaussian or other distribution (Yang, 2008).

3. EXPERIMENTAL RESULTS AND DISCUSSION

In this paper, the experiments were carried out in the platform as under:

- Operating system: Windows XP Professional;
- Programming language : Matlab;
- CPU: Core 2 Quad 2.4 GHz;
- RAM: 4 GB DDR2 1066 MHz;
- Execution: 25 times.

Due to the flurry of EAs recently developed, their performance is mainly analyzed by using different test suites of optimization and search problems. For this purpose, several test suites unconstrained (i.e. bound constrained) and constrained problems have been designed. In this paper, we have carried out all experiments by using the CEC2005 problems (Suganthan, *et al.*, 2005) in which f01-f05 are unimodal and all others are multimodal. The functions f01, f09 and f15 are separable and the rest of the functions arenon-separable. Moreover, f01-f14 are shifted functions and the remaining are non-shifted one, f07, f08, f10, f11, f16, f17,f18, f21 and f24 are rotated functions. Furthermore, most of these CEC2005 test problems (Suganthan, *et al.*, 2005) are consist of scalable functions. The suggested hybrid algorithm HGFA have applied on each test problem with number of decision variables such as $n = 2, 10, 20, 30,\ 40,\ 50$ and size of population $N = 100$to carried out our experiments. However, due to page restrictions in this paper, we have included only the numerical results obtained with n=10 and 30 for each test problem as given in the **(Table 1 and Table 2)** respectively. Most of the used test problems have been tackled by the suggested HGFA in terms of mean and standard values as compared to the stand alone GA as shown in Table 1 and Table 2. Furthermore, the better results obtained by both suggested hybrid GA and simple genetic algorithm (GA) are highlighted bold for the sack of fare comparison and differentiation among both used algorithms in the study of this paper.

Table 1: Comparison of HGFA (A) and GA (B) by using IEEE-CEC 05 (Suganthan, *et al.*, 2005) with n = 10.

Problems	Best	Mean	Std. deviation	Algorithm
f01	**0.000000**	**0.000000**	**0.800391**	**A**
	0.000001	0.000001	0.521016	B
f02	**0.000000**	**0.000000**	**0.031437**	**A**
	0.000000	0.000000	0.198374	B
f03	**0.000000**	**0.000000**	**100.797254**	**A**
	0.000000	0.000000	637.666968	B
f04	0.000000	0.000000	0.596816	A
	0.000000	0.000000	0.327159	B
f05	**0.000000**	**0.000000**	**23.526747**	**A**
	0.000000	0.000000	32.036672	B
f06	**0.000000**	**0.000000**	**9.341659**	**A**
	0.000000	0.000000	13.781630	B
f07	**0.000003**	**0.071089**	**3.504889**	**A**
	0.000009	3.557594	2.587509	B
f08	0.000000	0.000000	0.189494	A
	0.000000	0.000000	0.076302	B
f09	0.000000	0.000000	0.170180	A
	0.000000	0.000000	0.311911	B
f10	0.001453	0.017168	0.059705	A
	0.000061	**0.007804**	**0.103723**	**B**
f11	0.000000	0.000000	0.430164	A
	0.000000	0.000000	0.206919	B

f12	**0.000000**	**0.000000**	**0.000828**	**A**
	0.000000	0.000000	0.012976	B
f13	**0.000002**	**0.000015**	**0.020031**	**A**
	0.000006	0.000768	0.040584	B
f14	**1257.215466**	**1257.216296**	**3.802231**	**A**
	1457.215466	1457.215806	4.142761	B
f15	1024.920243	1024.957685	13.828212	A
	1024.920243	1024.957612	22.599572	B
f16	1018.756033	1018.800076	19.758873	A
	1018.756033	1018.792758	14.263865	B
f17	**827.203421**	**827.213144**	**11.472861**	**A**
	827.203398	827.205103	19.246432	B
f18	**1250.690337**	**1250.751380**	**9.268012**	**A**
	1250.690437	1250.806352	6.944744	B
f19	**883.234637**	**918.239453**	**67.039984**	**A**
	885.666535	916.629538	67.349620	B
f20	1341.214724	1341.214724	0.000000	A
	1341.214724	1341.214724	0.000000	B
f21	**1180.000079**	**1184.108599**	**13.549518**	**A**
	1180.095532	1185.909739	16.297264	B

Table 2: Comparison of HGFA (A) and (B) GA with search dimension n=30 for CEC'05 (Suganthan*, et al.*, 2005)`

Problems	Best	Mean	Std. deviation	Algorithm
f01	**0.000000**	**0.000000**	**567.150498**	**A**
	0.000000	0.000000	605.382503	B
f02	**0.000000**	**0.000917**	**794.296418**	**A**
	0.000003	0.039364	942.982813	B
f03	**2510.3194**	**5750.9726**	**3452856.5533**	**A**
	315225.984858	539100.620758	5887328.994022	B

	0.000009	**0.005786**	**1066.696958**	**A**
f04	0.000040	0.149584	1154.242033	B
	0.000000	**0.000002**	**1301.207326**	**A**
f05	0.000000	0.000003	1366.160967	B
	0.000590	**0.062728**	**27009161.9265**	**A**
f06	1.231123	4.107561	47669318.286299	B
	20.141028	**20.141028**	**0.036944**	**A**
f07	20.267766	20.279503	0.015343	B
	0.000000	**0.340904**	**9.395620**	**A**
f08	0:000000	3:422413	10:595200	B
	0.000000	**4.344671**	**12.483695**	**A**
f09	17.676048	17.676048	8.873088	B
	6.716940	7.967402	0.376060	A
f10	**6.364467**	**7.146139**	**0.286591**	**B**
	0.060091	**1.813182**	**5384.056593**	**A**
f11	12:600003	40:466356	3909:201355	B
	0.784527	**1.015096**	**0.876470**	**A**
f12	0.895147	1.349706	1.421402	B
	1.370662	**1.949312**	**0.268969**	**A**
f13	3.103262	3.103262	0.068194	B
	1360.491337	**1360.5335**	**16.6058**	**A**
f14	1360.491281	1360.554486	26.255421	B
	1264.390559	**1267.651894**	**22.213818**	**A**
f15	1287.077900	1297.099559	22.730716	B
	1264.6407	**1273.3474**	**21.8662**	**A**
f16	1290.781525	1296.891223	20.723626	B
	1254.6118	**1256.9636**	**12.8326**	**A**
f17	1255.746668	1260.787053	14.183325	B
	1332.0312	**1336.1445**	**28.6197**	**A**

f18	1342.330076	1379.304890	29.825873	B
	1152.2058	**1157.2058**	**24.6993**	**A**
f19	1153.289503	1211.524349	25.341379	B
	1386.1486	**1386.1927**	**18.4524**	**A**
f20	1391.379431	1400.300400	27.610007	B
	1333.6833	**1348.1435**	**12.0010**	**A**
f21	1356.522833	1365.801357	12.111518	B

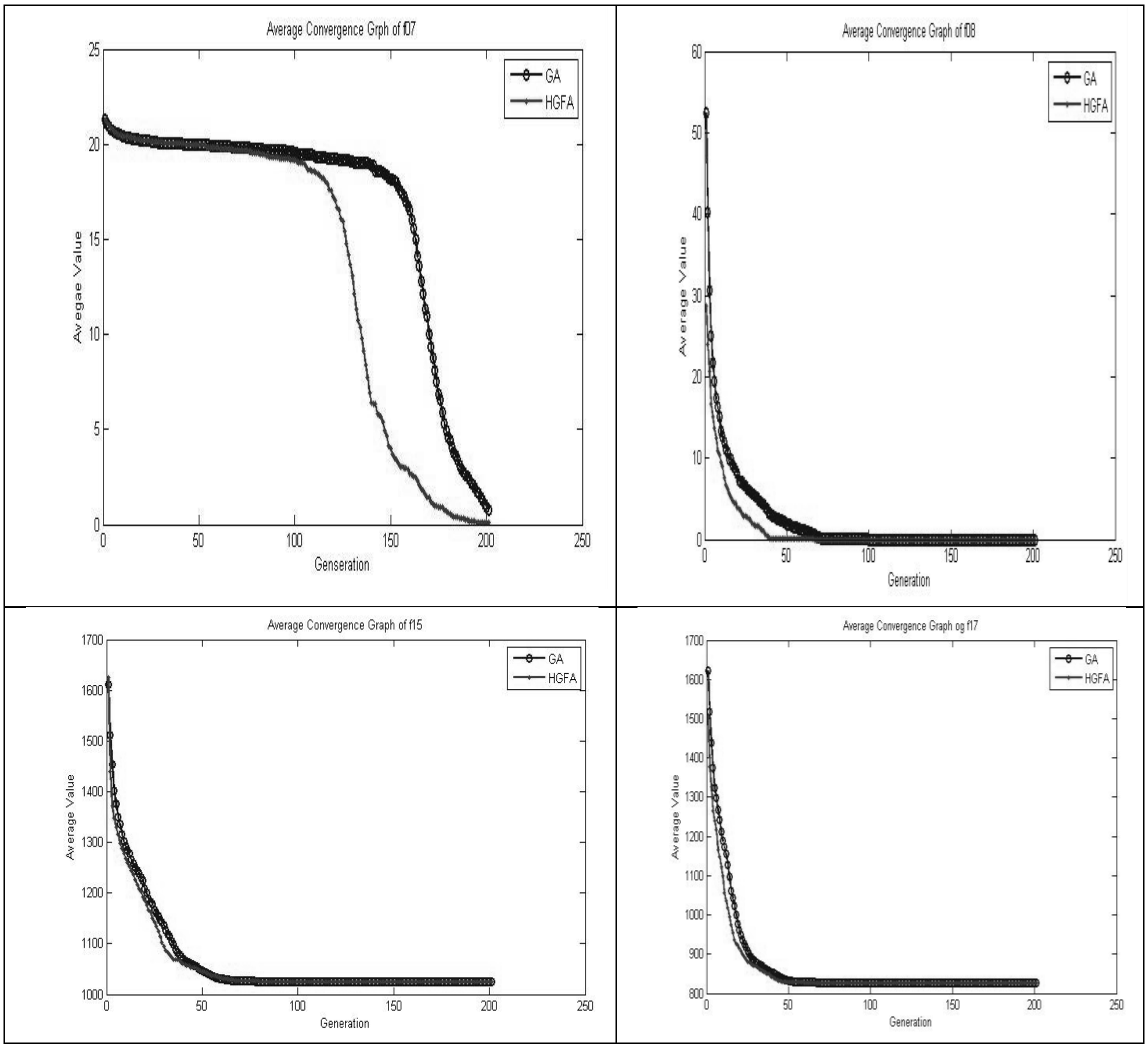

Fig.1: Convergence Graphs of the CEC 05 Benchmark Functions (Suganthan, *et al.,* 2005).

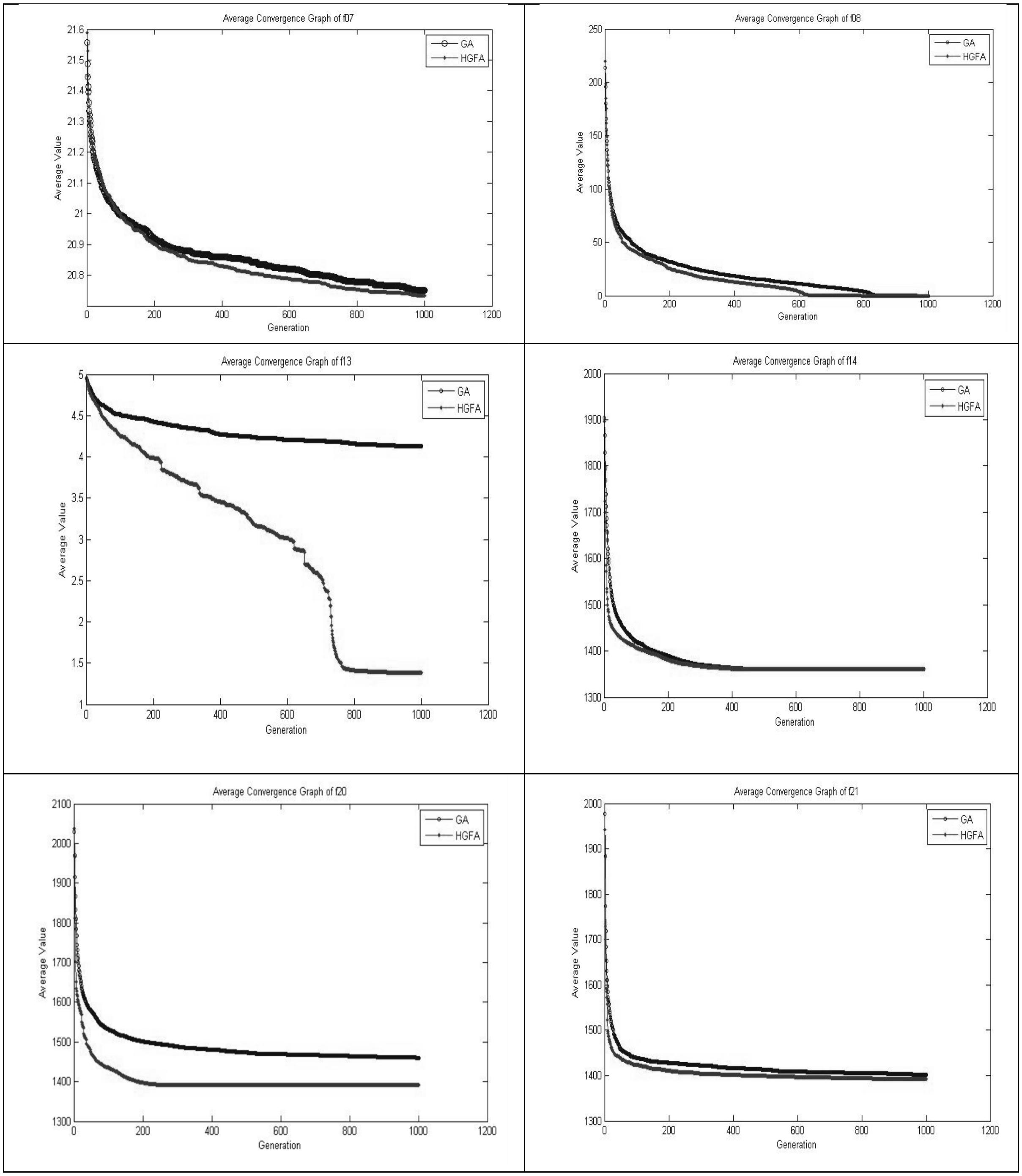

Fig.2: Convergence Graphs of the CEC 05 Benchmark Functions (Suganthan, *et al.*, 2005)

In single objective optimization, most of the suggested algorithm aiming at to find an optimum solution with fast convergence speed toward the known optimal solution of the problem at hand. **(Fig. 1)** demonstrates the convergence graph provided by HGFA with faster speed for dealing with f_{07}, f_{08}, f_{15} and f_{17} against original GA with$n = 10$. Similarly, HGFA perform better than GA on the problems f_{07}, f_{08}, f_{13}, f_{14}, f_{20}-f_{21} while converging near toward their respective optimal solution that can see from (**Fig. 2).** The minimum the values of the best, mean and standard deviation result in better approximated result for the corresponding each used test function.

4. CONCLUSION

Genetic algorithm (GA) is one of the commonly used met heuristic since its inception inspired by biological process of natural selection and variation (crossover and mutation). GA belongs to the larger class of evolutionary algorithms (EA) of the evolutionary computation (EC) field. The last two decades are witnessed for the huge amount GAs application to various test suites of optimization problems and real-world application including Airlines revenue management problems, Traveling salesman problems (TSPs), Vehicle routing problems (VRPs), RNA structure prediction, Multiple Sequence Alignment problems and many others.

In this research paper, we have employed Firefly Algorithm as search operator in combination with GAas resultant hybrid Genetic Firefly Algorithm (HGFA) developed. The main objective of proposed algorithm was to further improve the global search ability of the basic GA while coping with CEC'05 test suite (Suganthan, *et al.*, 2005). The suggested hybrid version of GA has tackled most of the test problems and validated its effectiveness against GA in terms of convergence toward the known optimal solutions of the benchmark functions.

In future, we will study impact of the use of recently suggested β-Hill climbing local search optimizer (Al-Betar, 2016) in the framework to solve the latest test suites of IEEE-CEC series[1] and real-world problems.

REFERENCES:

Al-Betar, M. A. (2016). β-Hill climbing: an exploratory local search. Neural Computing and Applications, 28, 1-16.

Bäck, T., F. Hoffmeister, and H.P. Schwefel, (1991). A Survey of Evolution Strategies. Proceedings of the Fourth International Conference on Genetic Algorithms 2-9. Morgan Kaufmann.

Eiben, A. E., and S. E. James, (2015). Introduction to Evolutionary Computing. New Delhi, India: Springer-Verlag Berlin Heidelberg.

El-Mihoub, T. A. (2006). Hybrid Genetic Algorithms: A Review. Engineering Letters, 124-137.

Fogel, L. J., A. J. Owens, and J. Walsh) .1966 .(Artificial Intelligence Through Simulated Evolution. New York, NY, USA: John Wiley & Sons.

Grosan, C., A. Abraham, and H. Ishibuchi, (2007). Hybrid Evolutionary Algorithms.Springer-Verlag Berlin Heidelberg.

Holland, J. H. (1973). Genetic Algorithms and the Optimal Allocation of Trials. SIAM J. Comput., 88-105.

Mashwani, W. K. (2014). Enhanced versions of Differential Evolution: State-of-the-art Survey. International Journal Computing Sciences and Mathematics, 5(2), 107-126.

Suganthan, P. N., N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A., Auger, and S. Tiwari, (2005). Problem Definitions and Evaluation Criteria for the CEC Special Session on Real-Parameter Optimization. Nanyang Technological University. Singapore: Technical Report. Retrieved from http://www.ntu.edu.sg/home/EPNSugan/

Yang, X.-S. (2008). Nature-Inspired Metaheuristic Algorithms. Luniver Press.

Zhang, L., L. Liu, X. S. Yang, and Y. Dai, (2016). A Novel Hybrid Firefly Algorithm for Global Optimization. (B. U. Wen-Bo Du, Ed.) journal.pone, 1-17.