Punjab University Journal of Mathematics (ISSN 1016-2526)
Vol. 49(2)(2017) pp. 89-99

Derivative Based Hybrid Genetic Algorithm: A Preliminary Experimental Results

Muhammad Asim
Department of Mathematics,
Kohat University of Science Technology,
Khyber Pakhtunkhwa-Pakistan,

Wali Khan Mashwani
Department of Mathematics,
Kohat University of Science & Technology,
Khyber Pakhtunkhwa-Pakistan,
Email: mashwanigr8@gmail.com
(Corresponding Author)

Muhammad Asif Jan
Department of Mathematics,
Kohat University of Science Technology,
Khyber Pakhtunkhwa-Pakistan,

Javed Iqbal
Department of Business Adminstration,
Bahauddin Zakariya University, Multan-Pakistan,
Email: asimuop@gmail.com

Received: 04 October, 2016 / Accepted: 09 November, 2016 / Published online: 02 June, 2017

Abstract. Global Optimization has become an important branch of mathematical analysis and numerical analysis in the recent years. Practical example of the optimization problems including the design and optimization of electrical circuit in electrical engineering, object packing problems, the Gibbs free energy in chemical engineering and the Protein structure prediction problems. Genetic algorithm (GA) is one of the most popular population based and stochastic nature based techniques in the field of evolutionary computation (EC). GA mimics the process of natural evolution and provides the maximum or minimum objective function value in a single simulation run unlike traditional optimization methods. This paradigm has great ability to efficiently locate the region in which the global optimum of the test problems exists. However, sometime, it has difficulties and spends much time to find the exact local optimum in the search space of the given test suites and complicated real world optimization problems.

In such a situation, local search (LS) techniques are very good tools to handle these issues by incorporating them in the framework of evolutionary algorithms in order to improve further their global search process. In this paper, we have incorporated the Broyden-Fletcher-Goldfarb-Shanno (BFGS) as local search optimizer in GA framework with a hope to alleviate the issues related to optimality and convergence of the original GA. The performance of the suggested hybrid GA (HGA) have been examined by selecting eight test problems from the widely used benchmark functions. The suggested HGA have shown promising results for dealing with most of the test problems compared to simple GA by implementing them in a Matlab 2013 environment.

Keywords: Global optimization, Evolutionary Computation, Evolutionary Algorithms, Genetic Algorithm, Hybridization, BFGS.

1. INTRODUCTION

Optimization problems have extreme importance in both mathematical analysis and evolutionary computation (EC) having had wide applications in various disciplines of science and engineering. In general, optimization can be divided into two categories depending on whether the variables are continuous or discrete. A problems with discrete variables are called combinatorial optimization problems. Continuous Optimization Problems having had floating point variables. Combinatorial optimization problems have wide application in the area of airline scheduling, production planning, location and distribution management, internet routing and many others. However, Continuous optimization problems have wide application in almost all engineering disciplines. A general optimization problem can be expressed as a minimization (without loss of generality) problem as follows

$$\text{Minimize } F(x) = f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}) \ldots, f_m(\mathbf{x}) \tag{1. 1}$$

$$\text{subject to} \begin{cases} g_j(x) \leq 0, j = 1, 2, \ldots, p, \\ h_i(x) \leq 0, i = 1, 2, \ldots, q. \end{cases}$$

Where $\mathbf{x} = (x_1, x_2, \ldots, x_m) \in R$ is an n-dimensional vector of optimization/decision variables, p is the number of inequality constraints and q is the number of equality constraints. Moreover $L_i \leq x_i \leq U_i$, $j = 1, 2, \ldots, n$, L_i and U_i are the lower and upper bounds of parametric space S and the function $f(x)$ is called an objective /fitness function. A solution that optimizes this objective function approximately well is called an optimal solution of the problem (1. 1). If $m = 1$, then the problem (1. 1) is called single objective optimization problem (SOP) in which we focus on the decision space i.e on the convergence of the solution towards an optimal solution. If $m \geq 2$, then problem (1. 1) is called multi objective problem(MOP). In multi objective optimization, we focus both decision space as well as on the objective space. However, in single objective optimization, our focus is only on decision space.

Cauchy was first mathematician who applied the gradient based optimization method to solve unconstrained optimization problems in 1847. G.Dantzig introduced its Simplex based method in 1947. In 1984, N.Karmarkar's polynomial time algorithm begins a boom of interior point optimization methods. Since then different optimization methods were developed as reviewed in [23]. Among them the Sequential Unconstrained Minimization Techniques (SUMT), Sequential Linear Programming (SLP) algorithm, the Modified Method of Feasible Directions (MMFD) algorithm, and the Sequential Quadratic Programming (SQP), Quadratic interpolation method, Interior point methods are applied on various types of optimization and real-world problems.

The Fletcher-Reeves [5] and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) [2], Powell's method [18] and Nelder-Mead simplex algorithm [16], golden-section search [19] and many others are well-known and existing local search optimizers for solving diverse unconstrained optimization problems. The Fletcher-Reeves method makes use of conjugate search directions to reach the optimum. BFGS methods are known as variable metric methods that make use of information gained from the previous n iterations to find a new search direction.

The Nelder-Mead algorithm [16] makes use of a simplex and a set of simple rules that reflects the worst vertex through the centroid of the simplex. All the aforementioned have certain limitations and advantages. They have handled different test suite of optimization problems and real-world problems. However, in general there no guarantee that these algorithm may not provide a global set of optimal solutions in single simulation. Therefore, it make sense to employ them in global search algorithms for exploitation purpose with aim at to develop various advanced optimization techniques.

Evolutionary algorithms (EAs) are well established population based global search optimization methods since its origination. They solve the given problem (1. 1) automatically without requiring the user to know or specify any form or structure of the problem in advance. EAs have successfully tackled various optimization and search problems [**?**, 4, 12, 20, 24, 13, 14, **?**, 15]. Mutation, crossover and selection are their basic operators as shown in the Figure 1. These operators mainly evolve the stochastic nature population of EAs.

In general, classical EAs can be divided into four paradigms, namely, Genetic Algorithms (GAs) [9], Evolution Strategies (ES) [1], Evolutionary Programming (EP) [6] and Genetic Programming (GP) [10, 11].

The first evolutionary-based optimization technique was the genetic algorithm (GA). It was first proposed by John Holland in 1975 inspired by Darwin's principle of survival of the fittest [9]. They do not need any kind of prior knowledge related to the problem to be solved as required to gradient based optimization methods. They have the ability of handling functions with noises and many complexities. However, being a global search optimizer, sometime, it trap in local basin of attraction while solving complicated optimization and search problems.These limitations and drawbacks of GA can be overcome with its hybridization with less time consuming local search techniques [17].

Variation (i.e,crossover and mutation) and selection are their important intrinsic operators. GA use a set of solutions called population to perform evolutionary process to get a set of optimal solutions to a problem (1. 1) in single simulation unlike classical optimization techniques. The three genetic operators like selection, crossover and mutation operators have their own advantages at different stages of population evolutions in order to push forward new population to the next generation of the process. This process continues until the optimal solutions are not found or number of function evaluations are not terminated as outlined in the algorithm1.

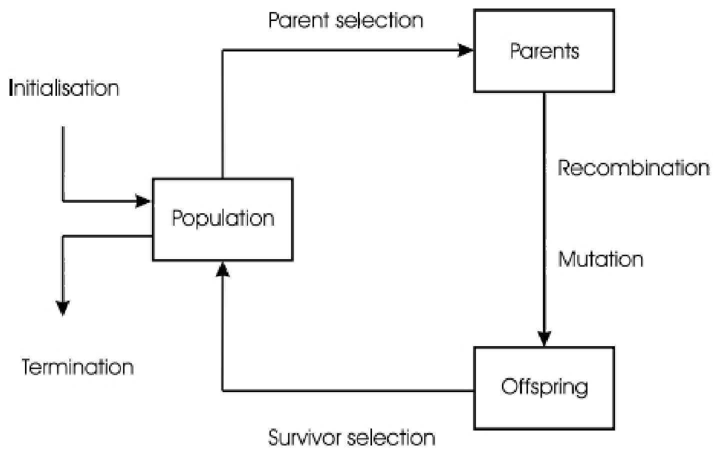

FIGURE 1. Flowchart of Evolutionary Algorithm (EA) [4]

.

Algorithm 1 Framework of the Genetic Algorithm (GA)

- $t = 0$, n: dimension of Decision Space;
- $\{x^1, \ldots, x^N\}^T \leftarrow$ **Initialize-Population**(N, n);
- $\{f(x^1), \ldots, f(x^N)\} \leftarrow$ **Evaluate**$(\{x^1, \ldots, x^N\}^T)$;

while $t < Max_t$ **do**
  for $i \leftarrow 1 : N$ **do**
    $x^a, x^b \leftarrow$ **Select-parents**$(\{x^1, \ldots, x^N\}^T, 2)$;
    $y^a, y^b \leftarrow$ **Xovers**(x^a, x^b);
    $z^a, z^b \leftarrow$ **Mutation**(y^a, y^b);
    $f^a, f^b \leftarrow$ **Evaluate**(z^a, z^b);
    $i = i + 1$;
    P(t+1)=**build next generation from** (pc(t) , p(t));
  end for
  t=t+1;
end while

The rest of the paper is arranged as under. Section 2 provides the framework of the derivative based genetic algorithm (HGA). Section 3 describes the mathematical formulation of the used benchmark functions. Section 4 devoted to the discussion related to the numerical results. Section 5 finally concludes this paper with some future work plan.

2. HYBRIDIZATION OF GA WITH LOCAL SEARCH

In general, the combined use of GAs and different efficient local search methods are typically considered to be a good idea for locating local optima with high accuracy and to capture a global view of the search space of the complicated optimization and search problems. Due to fast convergence behaviors, Broyden Fletcher Goldfarb Shanno (BFGS) [2] is well known and most affective hill-climbing local search method. BFGS method was proposed by Broyden [3], Fletcher [5], Goldfarb [8], and Shanno [21]. It has been frequently applied to different unconstrained nonlinear global optimization problems.

In this paper, BFGS [2] has been employed as local search optimizer in GA framework and as resultant hybrid algorithm called HGA developed to tackle test functions given in Table 1. The algorithmic structure of BFGS is hereby explained in the algorithm 3. BFGS has been embedded in step 5 of the HGA Framework. BFGS is an iterative based line search that does not demand for the Hessian matrix explicitly. Instead, it needs a positive definite matrix (i.e., initially identity matrix) which is further updated at each iteration by using the gradient information found in both current and previous iterations

Algorithm 2 Framework of the Hybrid Genetic Algorithm (HGA)

1: $G = 0$, N, n,Max_G;
2: $X = \{x^1, \dots, x^N\}^T \leftarrow$ **Initialize-Population**(N, n);
3: $F = \{f(x^1), \dots, f(x^N)\} \leftarrow$ **Evaluate**$(\{x^1, \dots, x^N\}^T)$;
4: **while** $G < MAX_G$ **do**
5:   **if** $rand < 0.2$ **then**
6:     $[X^C, F^C] = BFGS(X, F, tol, G)$;
7:   **else**
8:     $X^P, F^P \leftarrow$ **Select-parents**(X, F);
9:     $X^q, F^P \leftarrow$ **Xovers**(X^p, F);
10:     $X^C, F^P \leftarrow$ **Mutation**(X^q, F);
11:     $X^C, F^C \leftarrow$ **Evaluate**(X^c, F);
12:   **end if**
13:   $X \leftarrow X^C$; $F \leftarrow F^C$
14:   $G = G + 1$;
15: **end while**

3. Tested Functions

Due to the flurry of evolutionary algorithms recently developed, their performance is mainly measured on different test suites. In the last few years, several test suites of unconstrained (i.e. bound constrained) and constrained problems have been designed and presented in special sessions of IEEE Conference of Evolutionary Computation (IEEE-CEC) [22].In study of this paper, we have used twenty different unconstrained test functions as formulated in the Table 2.

In the carried out experiments, we have examined the performance of our suggested derivative based hybrid genetic algorithm by selecting nine single-objective optimization problems of the test suite designed for the special session of the 2005 IEEE conference of evolutionary computing (CEC'05) [22]. The characteristics of the used benchmark functions are listed in the Table 1.

Algorithm 3 Broyden-Fletcher-Goldfarb-Shanno (BFGS)

$K = 0$, old_x, old_g,Max_K,$error$;
$old_x = \{x^1, \ldots, x^N\}^T \leftarrow$ **Initialize-Population**;
$F = \{f(x^1), \ldots, f(x^N)\} \leftarrow$ **Evaluate**$(\{x^1, \ldots, x^N\}^T)$;
while $((\|(new_x - old_x)\| > error) or (k < 1) and (k < Max_K)$ **do**
  $new_g \leftarrow$ **Gradient**$(fun - name, new_x, error)$
  $new_p \leftarrow$ **New point**$(new_x - old_x)$
  $new_q \leftarrow$ **Change in gradient**$(new_g - old_g)$
  $old_x = new_x$
  $old_g = new_g$
  $Old_H \leftarrow$ **Update hessian Matrix**
  $old_d \leftarrow$ **old direction**$(old_H * old_g)$
  $alpha \leftarrow$ **Step size using golden section search**$(fun_name, old_x, old_d, error)$
  $\{x_n^1, \ldots, x_n^N\}^T \leftarrow$ **Update new solutions**$(old_x + alpha * old_d)$
  $K = K + 1$;
end while
$NewFittnes \leftarrow$ **Evaluate using Child Chrom**$(\{x_n^1, \ldots, x_n^N\}^T)$

TABLE 1. Tested functions

Functions	Problem Names	Search Range	Dimension	Optimal Value
f01	Ackley	$[-65.536, 65.536]$	10	0
f02	Beale	$[-4.5, 4.5]$	10	0
f03	Axis par: hyp: elip: fun:	$[-5.12, 5.12]$	10	0
f04	Branian	$[-65.536, 65.536]$	10	0
f05	Restrigen	$[-5.12, 5.12]$	10	0
f06	Schwefel Double Sum	$[-5.12, 5.12]$	10	0
f07	Sphere	$[-5.12, 5.12]$	10	0
f08	Sum of Difference	$[-65.536, 65.536]$	10	0
f09	Booth	$[-10, 10]$	10	0

4. PARAMETER SETTING AND DISCUSSION

The experiments were carried out with following computing platform:

- Operating System: Windows 7 Professional;
- Programming language of the algorithms: Matlab;
- CPU: Core 2 Quad 2.4 GHz;
- RAM: 4 GB DDR2 1066 MHz;
- Execution: 25 times each algorithm with different random seeds.

The suggested hybrid Genetic Algorithm (HGA) has been tested by using 9 simple benchmark functions with search space having dimensions $D = 10$, as explained in Table 1.

All experiments were carried out by using $D * 100$ function evaluations to handle each used tested function. The population size of the randomly generated set of solutions were restricted to $N = 100$. To establish a fair comparison between HGA and GA, the same

TABLE 2. Mathematical Formulation of the f01-f09 functions

Problem	formulation
f01	$f(x) = -20\exp(-0.2\sqrt{(1/n\sum(x^2))} - \exp(1/n\sum(cos(2\pi x))) + 20 + e$
f02	$f(x) = (1.5 - x_1(1-x_2))^2 + (2.25 - x_1(1-x_2^2))^2 + (2.625 - x_1)(1 - x_2^3))^2$
f03	$f(x) = \sum_{i=1}^{n}(ix^2)$
f04	$f(x) = (x_2 - (5.1/(4\pi^2))x_1^2 + 5x_1/\pi - 6)^2 + 10(1 - 1/(8\pi))cos(x_1) + 10$
f05	$f(x) = 10n + \sum((x^2) - 10cos(2\pi x))$
f06	$f(x) = \sum(\sum x^2)$
f07	$f(x) = \sum_{i=1}^{n} x_i^2$
f08	$f(x) = \sum_{i=1}^{n}(\lvert x^i + 1\rvert)$
f09	$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2;$

TABLE 3. Numerical Results of the functions f01-f09 provided by HGA

Problem	Best	Worst	Median	Mean	St.Dev
f01	17.0999	19.9975	19.9141	19.2546	1.0371
f02	0.000000	0.972884	0.087345	0.218618	0.310503
f03	0.006867	23.428381	1.388550	4.264540	7.187118
f03	0.000007	13.374590	4.857686	5.751813	5.448957
f04	0.397889	5.336386	0.410679	1.598965	1.983369
f05	404.390515	430.520457	415.232016	416.259603	7.833741
f06	0.000249	20.712455	1.727706	5.420115	7.222197
f07	0.000000	0.000010	0.000003	0.000004	0.000003
f08	159.836169	244.810847	207.107158	207.394831	25.873420
f09	0.000002	0.115610	0.001806	0.024288	0.042625

number of function evaluations and other related parameters were kept to execute both algorithms 25times independently in order to solve each employed test problem. The solution qualities are summarized in the form of the minimum, median, mean, standard deviation and maximum of the objective values obtained as gathered in the Tables 3 and Table 4. respectively.

The numerical results of the HGA provided by HGA in the Table 3. Table 5 provides a comparative numerical results in terms of best and average of both simple GA and HGA. These table clearly indicates that HGA has efficiently solved most of the used test functions f02-f07, f09 as compared to simple GA. However, test functions f02 and f08 are tackled by simple GA in an efficient manner as can see from the figures plotted below. The figures 2 display the average variation in the function values of the selected test functions generated by HGA versus GA.

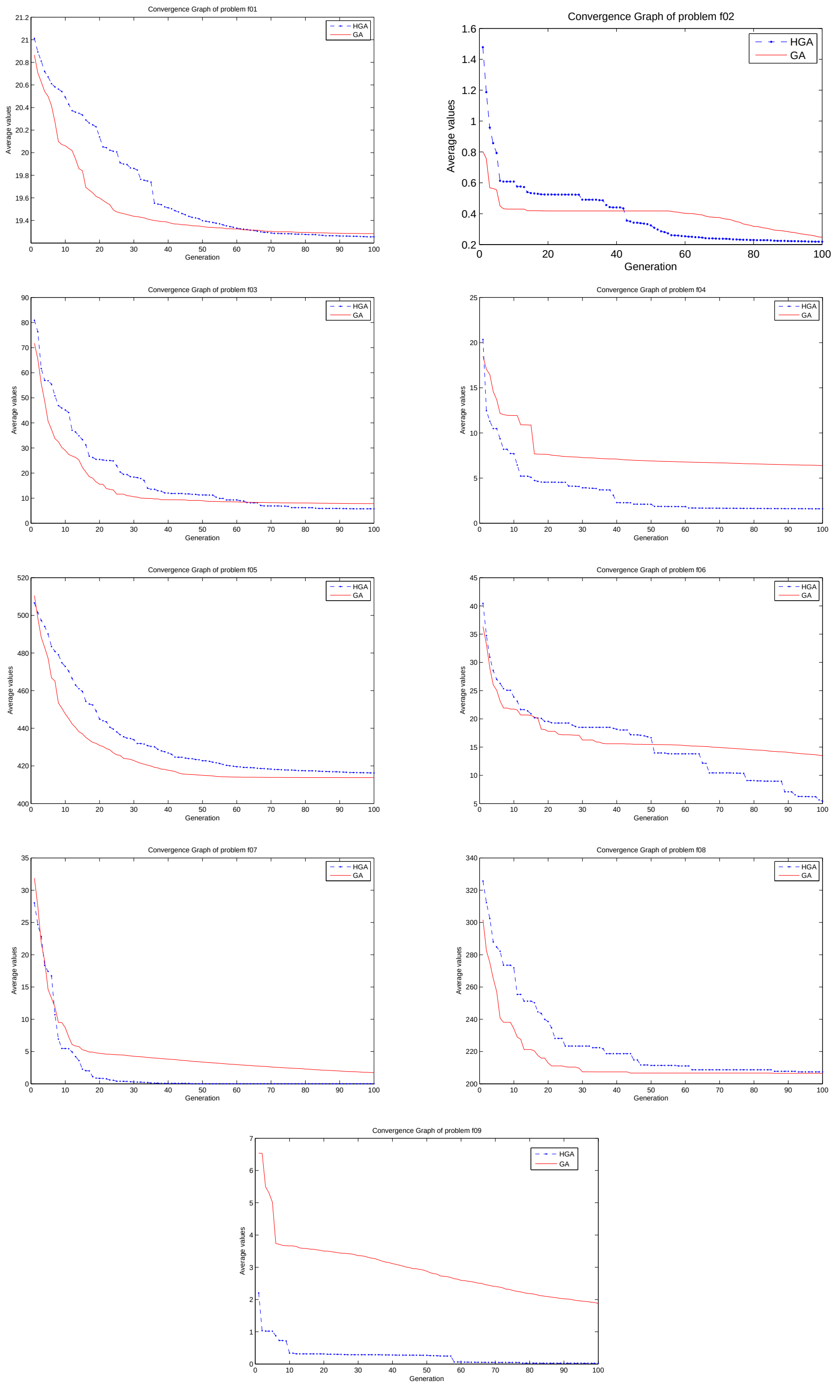

FIGURE 2. Plots of the Convergence Graph by HGA versus GA in 25 indepent runs for dealing with nine CEC'05 [22].

TABLE 4. Numerical Results of functions f01-f09 provided by GA.

Problem	Best	Worst	Median	Mean	St.Dev
f01	15.0118	20.0002	19.9236	19.2835	1.5352
f02	0.000011	0.987225	0.126604	0.247287	0.307781
f03	1.452044	13.283570	7.432468	7.817041	3.852394
f04	0.734682	18.821125	3.218115	6.379849	6.673731
f05	407.106963	423.974239	413.119060	413.778980	4.839356
f06	9.397204	18.239409	12.707327	13.480023	3.658149
f07	0.185642	3.826202	1.370506	1.719613	1.296216
f08	138.136093	235.388925	222.537187	206.439345	33.942018
f09	0.000011	7.788831	0.794659	1.877028	2.658563

TABLE 5. The Comparison of GA versus HGA to cope with f01-f09.

Genetic Algorithm VS Hybrid Genetic Algorithm			
problems	Best	Mean	Algorithm
f01	**15.0118**	**19.2835**	**GA**
	17.0999	19.2546	*HGA*
f02	0.000011	0.247287	GA
	0.000000	**0.218618**	***HGA***
f03	1.452044	7.817041	GA
	0.006867	**5.751813**	***HGA***
f04	0.734682	6.379849	GA
	0.397889	**1.598965**	***HGA***
f05	407.106963	413.778980	GA
	404.390515	**416.259603**	***HGA***
f06	9.397204	13.480023	GA
	0.000249	**5.420115**	***HGA***
f07	0.185642	1.719613	GA
	0.000000	**0.000004**	***HGA***
f08	**138.136093**	**206.439345**	**GA**
	159.836169	207.394831	*HGA*
f09	0.000011	1.877028	GA
	0.000002	**0.024288**	***HGA***

5. CONCLUSION

Slow convergence and other related to search direction issues regarding simple GA can address by employing different search techniques in GA framework. This is because GA has ability of finding global optima to abide trapping in the local basin of attraction of the problem to solve. The use of local search methods is a good choice to tackle the issue of GA's trapping. BFGS method is very efficient technique having had fast convergence behavior. However, it is quite sensitive to the starting point but still perform well over non-convex problems.

In this paper, we proposed a new hybrid HGA of GA and BFGS strategy for the optimization of single objective functions. We have used nine problems to examine the performance of the HGA. The suggested hybrid GA have offered best results for most test problems compared to GA in terms of convergence speed. In future, we intend to examine the algorithmic behavior of the suggested algorithm over latest test suites which are regularly designed for the special session in the IEEE Conference of Evolutionary Computation (IEEE-CEC) available on the link: **http://www.ntu.edu.sg/home/epnsugan/**

ACKNOWLEDGMENT

All authors have equal contribution and declare hereby no conflict of interest to publish this paper.

REFERENCES

[1] T. Back, F. Hoff Meister and H.P. Schwefel, *A Survey of Evolution Strategies*, Proceeding of the Fourth International Conference on Genetic Algorithms, San Mateo, CA: Morgan Kauffman, 2-9.
[2] R. Battiti and F. Masulli, *BFGS Optimization for Faster and Automated Supervised Learning*, Internatonal conference on Neural Network (INCC 90), Paris, 757-760.
[3] C. G. Broyden, *The convergence of a class of double rank minimization algorithms: 2: The new algorithm*, J. Inst. Math. Appl. **5**, 222-231.
[4] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, ed. $\mathbf{1^{st}}$, Springer-Verlag, Berlin, Germany.
[5] R. Fletcher, *A new approach to variable metric algorithms*, Computer J. **13**, 317-322.
[6] L. J. Fogel, A. J. Owens and M. J. Walsh, *Artificial Intelligence through Simulated Evolution*,ed. $\mathbf{1^{st}}$, John Wiley & Sons.
[7] D. E. Goldberge, *Genetic Algorithms in Search, Optimization and Machine Learning*, ed. $\mathbf{1^{st}}$, Addison-Wesely, Reading, Massachusetts.
[8] D. Goldfarb, *A family of variable metric methods derived by variational means*, Math. Comp. **24**, 23-26.
[9] J. H. Holland, *Genetic Algorithms and the Optimal Allocation of Trials*, J. SIAM Compute, **2**, 88-105.
[10] J. R. Koza, *Genetic Programming on the Programming of Computers by means of Natural Selection*, ed. $\mathbf{1^{st}}$, MIT Press.
[11] J. R. Koza), *Genetic programming II Autonomous Discovery of Reusable Programe*, ed. $\mathbf{1^{st}}$, MIT Press.
[12] Wali Khan Mashwani, *Enhanced versions of Differential Evolution: State-of-the-art Survey*, International Journal Computing Sciences and Mathematics (IJCSM), **5**, No. 2 107-126.
[13] R. A. Khanum, Nasser Tairan, M. A. Jan, W. K. Mashwani and A. Salhi,*Reflected Adaptive Differential Evolution with Two External Archives for Large-Scale Global Optimization*, International Journal of Advanced Computer Science and Applications, **7**, No. 2 675-683.
[14] Rashida Adeeb Khanum, Nasser Mansoor Tairan, Muhammad Asif Jan, and Wali Khan Mashwani, *Hybridization of Adaptive Differential Evolution with an Expensive Local Search Method*, Journal of Optimization, Volume 2016, Article ID 3260940, 14 pages.
[15] R. A. Khanum, Islam Zari, M. A. Jan and W. K. Mashwani, *Reproductive Nelder-Mead Algorithm for Unconstrained Optimization Problems,* Science International Journal, **7**, No. 2 65-72.
[16] J. A. Nelder and R. Mead, *A simplex method for function minimization* Computer Journal **7**, (1965) 308313.
[17] Parry Gowher Majeed and Santosh Kumar, *Genetic Algorithms in Intrusion Detection Systems: A Survey*, International Journal of Innovation and Applied Studies, **5**, 233-240.
[18] M. J. D. Powell, *An efficient method for finding the minimum of a function of several variables without calculating derivatives*, Computer Journal **7**, No. 2 (1964) 155-162.
[19] Press, WH; Teukolsky, SA; Vetterling, WT; Flannery, BP (2007), ”Section 10.2. Golden Section Search in One Dimension”, Numerical Recipes: The Art of Scientific Computing (3rd ed.), New York: Cambridge University Press.
[20] Tayyaba Shah, Muhammad Asif JAN, Wali Khan Mashwani and Hamza Wazir, *Adaptive Differential Evolution for Constrained Optimization Problems*, Sci.Int.(Lahore) **28**, No. 3 2313-2320.

[21] D. F. Shanno, *Conditioning of quasi-Newton Methods for function minimization*, Math. Comp. **24**, 647-650.

[22] P. N. Suganthan, J. J. Liang, B. Y. Qu and Q. Chen, *Problem Definitions and Evaluation Criteria for the CEC' 2005 Special Session on Real-Parameter Optimization*, Kanpur Genetic Algorithms Laboratory, IIT Kanpur, KanGAL Report Number 2005005.

[23] G. Venter, *Review of Optimization Techniques*, Encyclopedia of Aerospace Engineering. Edited by Richard Blockley and Wei Shyy John Wiley & Sons, Ltd.

[24] Hamza Wazir, Muhammad Asif JAN, Wali Khan Mashwani and Tayyaba Shah, *A Penalty Function Based Differential Evolution Algorithm for Constrained Optimization*, the Nucleus Journal, **53**, No. 1 155-161.