

Analysis of Differential Evolution by Experimenting on Critical Parameters

Rachna Biala and Mandeep Kaur

CSE Department
Guru Nanak Dev University Regional Campus
Jalandhar, INDIA
rachu_biala@yahoo.com, mandeep8@gmail.com

Abstract

DE is a topic of current interest in the optimization field. It is the most capable evolutionary algorithm based on biological theory of evolution because of its ease and competence in solving variety of problems, like multi-objective, multi-modal, dynamic optimization problems. But premature convergence or stagnation is a main problem with it. So In order to improve the performance of DE, significant number of DE variants has been proposed by many researchers over the last few decades. Mutation is one of the key tasks of DE. It appreciably influences the performance of DE. In this paper, DE variants with four different mutation techniques- DE/rand/1, DE/local-to-best, DE/either-or and MODE are studied and implemented. Comparison of DE having these mutation strategies is made for variety of dimension and population size and results shows that DE/local-to-best performs best on all the benchmark functions where as MODE also show significant performance.

Keywords: *Optimization, Differential Evolution, DE/rand/1, DE/either-or, Modified Differential Evolution.*

1. Introduction

In simple, Optimization means to enhance or we can say either increase or decrease the valuable features of any system. For example, for optimizing the performance of air conditioner, there is a need to decrease its power consumption and increase its cooling effect. In case of travelling from one city to another city, everybody's motive is to minimize the travelling time by choosing the shortest path. So there is some particular function or properties which needs to optimize, that function is called an optimization function. It is the goal of optimization. It is possible to have more than one optimization function just like in case of air conditioner.

2. Differential Evolution

“Differential Evolution can be defined as very simple and effective way to optimize the non-linear and non-differential problems”. It is a reliable optimizer introduced by K. Storn and R. Price in 1995[2]. It is a nature inspired algorithm that belongs to the category of evolutionary algorithms. The first written publication on DE appeared as a technical report in 1995 [2, 7]. Differential Evolution (DE) was derived from naturally mathematical (geometrical) arguments [7].

2.1 Outline of Differential Evolution

The required DE parameters are input [6, 7, 11].

- a. All the vector populations are initialized randomly in the limits specified for the decision variables.
- b. Each member of the population is evaluated for its fitness.
- c. Until the objective function is not achieved or maximum number of iteration is not reached, Mutation and crossover operations are performed on all the members of the population, i.e. for each parent P_i .
 - i. Distinct vectors are selected randomly from the current population other than the parent vector.
 - ii. The new mutation vector is calculated.
 - iii. The mutated vector is modified by binary crossover with the parent using crossover probability CR.
 - iv. The variables are restricted to their boundaries, if any variable is outside the lower or upper bound.
- d. Each member of the population is evaluated. Dominance with the parents is checked. If the candidate dominates the parent, it replaces the parent. If the parent dominates the candidate, it replaces the candidate.
- e. The latest solution vectors are taken.
- f. The generation counter is incremented, G to $G+1$ and termination criterion is checked. If the termination criterion is not met, then return to set c.

While implementing standard differential evolution, one must assign the values to parameters. DE requires mainly three control parameters: F (mutation rate), CR (crossover rate) and NP (population size). The other parameters may be number of generations, dimension of the problem, lower and upper boundary constraints that limits the search ability of optimal solution. Before starting optimization, the upper and lower limit of population is defined and population member's fitness is evaluated. Then values are assigned to the control parameters F [0, 1] and CR [0, 1].

2.2 Mutation Strategies

The performance of DE mostly depends on the choice of mutation technique and the related parameter values when the problem is complex [13]. Unsuitable choice of mutation and crossover strategies and related parameters may cause premature convergence, stagnation or wastage of computational resources [6, 11, 13, 14].

Considering the importance of mutation strategies in optimization, various researchers suggested different mutation strategies as mentioned below:

A. DE/rand/1

This technique was introduced by Storn and Price for mutating the vectors as one step to achieve optimization. It became popular for its better convergence properties and its structural and working simplicity [3]. In this, for each target vector, fixed mutation rate $F \in [0, 2]$ and fixed population size, three different vectors are selected randomly and mutant vector is generated according to equation:

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + F \cdot (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G}) \quad (1)$$

Here there is one condition that $NP > 4$ in order to use the technique. DE outperformed the minimizations approaches in terms of required number of function evaluations necessary to locate a global minimum of the test functions [2, 5].

B. DE/rand/2

DE/rand/2 mutation technique introduced by Storn and Price includes generation of mutant vector by selecting five random vectors. It is different from DE/rand/1 in the way that it utilizes two diverse weighting factors parameter vectors of current iteration. The mutant vector is generated as [2, 9, 10, 12].

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + F. (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G}) + F(\vec{X}_{r4,i,G} - \vec{X}_{r5,i,G}) \quad (2)$$

C. DE/best/1

This technique is similar to rand/1 but it considers the best vector of current iteration and two randomly selected vectors while finding donor vector for new iteration. It also uses one weighted difference of randomly selected two vectors [3, 9, 10, 12].

$$\vec{V}_{i,G} = \vec{X}_{best,G} + F. (\vec{X}_{r1,i,G} - \vec{X}_{r2,i,G}) \quad (3)$$

D. DE/best/2

In this technique, mutation is performed by taking the best vector of the current generation and two weighted difference vectors. It is different from DE/best/1 in terms of number of weighting difference vectors [3].

$$\vec{V}_{i,G} = \vec{X}_{i,G} + \lambda(\vec{X}_{best,G} - \vec{X}_{i,G}) + F. (\vec{X}_{r1,i,G} - \vec{X}_{r2,i,G}) \quad (4)$$

The term λ in (4) assumes a value in the range [0, 2].

E. DE/target-to-best

The DE variant known as DE/target-to-best/1 uses the best vector of the population and four random vectors to generate donor vectors. The scheme emphasizes on exploitation since all the vectors are attracted towards the same best position on the fitness landscape through iterations, resulting in faster convergence to that point but sometimes get trapped in local minima. On comparing this technique with the some variants of DE [12] has noted that DE/target-to-best/1 has poor performance and remains inefficient in exploring the search space, especially for multi-modal functions [6].

$$\vec{V}_{i,G} = \vec{X}_{i,G} + F. (\vec{X}_{best,G} - \vec{X}_{i,G}) + F. (\vec{X}_{r1,i,G} - \vec{X}_{r2,i,G}) \quad (5)$$

F. Either-or

In order to lessen the effort required to find the optimal solution, the new technique [7] having two different mutation strategies is used. The mutation strategy is chosen by the help of parameter which further depend on mutant factor by $K=0.5(F+1.0)$.

$$\begin{aligned} \text{if } (r < 0.5) \quad & \vec{V}_{i,G} = \vec{X}_{r1,i,G} + F. (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G}) \\ \text{else} \quad & \vec{V}_{i,G} = \vec{X}_{r1,i,G} + 0.5(F + 1.0). (\vec{X}_{r2,i,G} - 2\vec{X}_{r3,i,G}) \end{aligned} \quad (6)$$

G. Cauchy Mutation

The focus of this research was to introduce a mechanism which will not only keep a track of the progress of individuals but will also help the individuals to escape from the local basin by allowing them to jump to a new region. In order to keep a record of the success of individuals, a concept of ‘failure counter’ (FC) was used. The job of FC was to monitor the working of individuals in terms of fitness function value for a specified number of generations. If there is no improvement in fitness, then FC is increased by unity in each generation. This process is repeated until user-defined value of maximum failure counter (MFC) is achieved. Once MFC is attained, it is an indication that perturbation in the population is needed which will allow the individual to jump to a new position. It leads to less number of functions evaluations and convergence rate is fast [13].

H. Mutation with MDM technique: DE/best/1 with MDM

It is a mutation technique based on multiple-demes topology. To facilitate fast convergence speed and wide population diversity, the population is broken into subpopulations and each subpopulation run independently to generate various best individuals in the population.

$$V_{i,G}^K = X_{best,G}^{\gamma(k)} + F(X_{r1,G}^K - X_{r2,G}^K) \quad (7)$$

$$\text{where } \gamma(k) = \begin{cases} \text{rand}(1, M) \text{ not belongs to } K \text{ if} \\ \text{mod}(G, \zeta) = 0 \\ k \text{ otherwise} \end{cases}$$

Then occasionally they share their own exploration results with each other to generate new mutant vectors. The results show that this technique outperforms traditional DE in terms of quality of required solution [14].

I. DE/current-to-gr_best/1

It is a less exploitive and more explorative version of DE/current-to-best mutation scheme in which random vectors are grouped together into different groups corresponding to each target vector, each having size q% of total population size. The best member of this dynamic group is used to perturb the target vector. It avoids premature convergence as target solution get attracted towards different best vectors instead of single best vector in DE/current-to-best/1. [15]

$$\vec{V}_{i,G} = \vec{X}_{i,G} + F \cdot (\vec{X}_{gr_{best,G}} - \vec{X}_{i,G} + \vec{X}_{r1,i,G} - \vec{X}_{r2,i,G}) \quad (8)$$

Where $\vec{X}_{gr_{best,G}}$ is the best from q% vectors randomly chosen from current population and size of the subpopulation is taken to be q% of the NP. q=15% of five best results on the majority of the test fns.

J. Intersect mutation:

In this [16], the population is divided into two parts, best part and worst part. Then for each part, the mutant vectors are determined in two steps .The algorithm proves to be advantageous on standard DE.

$$\vec{V}_{i,G+1} = \vec{X}_{wr1,G} + F \cdot (\vec{X}_{br1,G} - \vec{X}_{br2,G}) \text{ where } br1 \neq br2 \neq wr1 \quad \vec{V}_{i,G+1} = \vec{X}_{br1,G} + F \cdot (\vec{X}_{wr1,G} - \vec{X}_{wr2,G}) \text{ where } br1 \neq wr1 \neq wr2 \quad (9)$$

K. MODE:

It [18] takes different mutation strategies to solve complex optimization problems. In order to choose particular mutation strategy, a new parameter MSCR is introduced with recommended range [0.9, 1]. It finds optimal or near optimal solution in less time.

$$\begin{aligned} \text{if } (r < 0.98) \quad \vec{V}_{i,G} &= \vec{X}_{r1,i,G} + F \cdot (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G}) \\ \text{else } \vec{V}_{i,G} &= \vec{X}_{r1,i,G} + F \cdot (\vec{X}_{best,G} - \vec{X}_{r2,i,G}) \end{aligned} \quad (10)$$

L. Restart Mutation:

It [17] is inspired from PSO [1] , where best and worst members from the population are analyzed. The new mutant vector follows the direction of best member and avoids the direction of worst member determined so far.

$$\vec{V}_{i,G} = \vec{X}_{r1} + F1 \cdot (\vec{X}_{best,G} - \vec{X}_{r1,G}) + F2 \cdot (\vec{X}_{r1,G} - \vec{X}_{worst,G}) \quad (11)$$

It has better search ability as it allow search in direction of best member determined so far in population. It leads to fast convergence.

3. Implementation Model

The algorithm of research work shows how the differential evolution with different mutation techniques is implemented [8].

a. Initialization of population:

For i=1, i≤NP, i++

For j=1, j≤D, j++

$$X_{i,j,G} = X_j^{lb} + rand(0,1). (X_j^{ub} - X_j^{lb})$$

End for

$$f(i) = f(X_{i,g})$$

End for

b. Mutation

For g=1, g≤gmax, g++

For i=1, i≤NP, i++

For j=1, j≤D, j++

If(DE/rand/1)

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + F. (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G})$$

If(DE/local-to-best)

$$\vec{V}_{i,G} = \vec{X}_{i,G} + F. (\vec{X}_{best,G} - \vec{X}_{i,G}) + (\vec{X}_{r1,i,G} - \vec{X}_{r2,i,G})$$

If(DE/either-or)

If (rand < 0.5)

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + F. (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G})$$

Else

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + 0.5(F + 1.0). (\vec{X}_{r2,i,G} + \vec{X}_{r3,i,G} - 2\vec{X}_{r4,i,G})$$

If(MODE)

If(rand<0.98)

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + F. (\vec{X}_{r2,i,G} - \vec{X}_{r3,i,G})$$

Else

$$\vec{V}_{i,G} = \vec{X}_{r1,i,G} + F. (\vec{X}_{best,i,G} - \vec{X}_{r2,i,G})$$

End for

End for

End for

c. Crossover

For g=1, g≤gmax, g++

For i=1, i≤NP, i++

For j=1, j≤D, j++

If (rand_j<CR)

$$U_{j,i,G} = V_{j,i,G}$$

```

        Else
             $U_{j,i,G} = V_{j,i,G}$ 
        End for
    End for
End for
d. Selection
For g=1 to gmax
    For i=1 to NP
        If  $f(U_{j,i}) < f(X_{j,i})$ 
            For j=1 to NP
                 $X_{j,i} = U_{j,i}$ 
            End for
        End for
    End for
     $f(i) = F(U_{i,g})$ 
End for
End for
    
```

4. Experimental Analysis

The performance of differential evolution with different mutation techniques and binomial crossover is evaluated on MATLAB software of version MatlabR2013a. The four variants of DE with different mutation and binomial crossover are analyzed for best cost, number of functions evaluated and convergence time on four benchmark functions.

4.1 Evaluation on the Basis of Minimum Cost

Here by varying the number of population and taking the constant values of dimension=2, mutant factor $F=0.8$ and crossover rate $CR=1.0$, the minimum cost for the DE variants are evaluated on different test functions.

Table 1. Minimum Cost Achieved by DE Variants on Different Test Functions for NP=20, 50

Function	Population size	Rand/1	Local-to-best	Either-or	Mode
Eggholder	Np=20	-25675.48	-29841.93	-18625.06	-16233.78
	Np=50	-26246.437	-19820.831	-17282.232	-26051.886
Michalewicz	Np=20	-1.507135	-1.738998	-1.827303	-1.769391
	Np=50	-1.764971	-1.739610	-1.738215	-1.633905
Peak	Np=20	-6.551131	-6.551133	-6.551124	-6.551130
	Np=50	-6.551133	-6.551133	-6.551133	-6.551133
Rastrigin	Np=20	0.264498	0.003741	0.349131	1.081168
	Np=50	0.81697	0.030962	0.059573	0.997120

4.2 Evaluation on the Basis of Convergence Time

In this, the DE variants are evaluated for time taken by them to converge while varying the number of population and taking the constant values of dimension=2, mutant factor $F=0.8$ and crossover rate $CR=1.0$

Table 2. Convergence Time of DE Variants on Different Test Functions for NP=20, 50

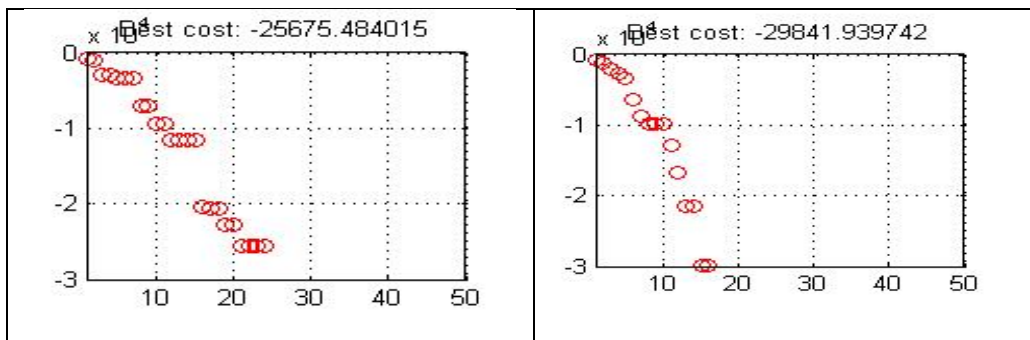
Function	Population size	Rand/1	Local-to-best	Either-or	mode
Eggholde-r	Np=20	92.094	56.763	61.174	67.454
	Np=50	181.979	146.202	137.459	175.416
Michalewicz	Np=20	220.839	223.152	219.307	218.095
	Np=50	2805.141	2340.557	4213.592	2501.593
Peak	Np=20	206.418	204.741	199.701	204.297
	Np=50	1984.188	2248.492	2266.599	2436.228
Rastrigin	Np=20	320.102	299.685	308.990	367.359
	Np=50	2993.578	2476.049	2422.400	4049.232

4.3 Evaluation on the Basis of Number of Functions Evaluated

In this, the DE variants are evaluated in terms of number of functions evaluated to reach the optimal value while varying the population and taking the constant values of dimension=2, mutant factor $F=0.8$ and crossover rate $CR=1.0$

Table 3. Number of Functions Evaluated by DE Variants on Different Test Functions for NP=20, 50

Function	Population size	Rand/1	Local-to-best	Either-or	mode
Eggholde-r	Np=20	500	340	360	380
	Np=50	800	650	600	800
Michalewicz	Np=20	1000	1000	1000	1000
	Np=50	2500	2500	2500	2500
Peak	Np=20	1000	1000	1000	1000
	Np=50	2500	2500	2500	2500
Rastrigin	Np=20	1000	1000	1000	1000
	Np=50	2500	2500	2500	2500



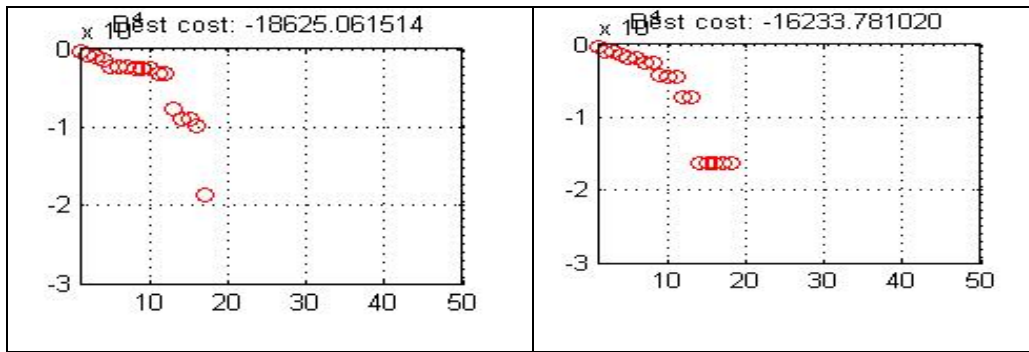


Figure 1. Graphs Showing Performance of DE Variants on NP=20 on Eggholder Function

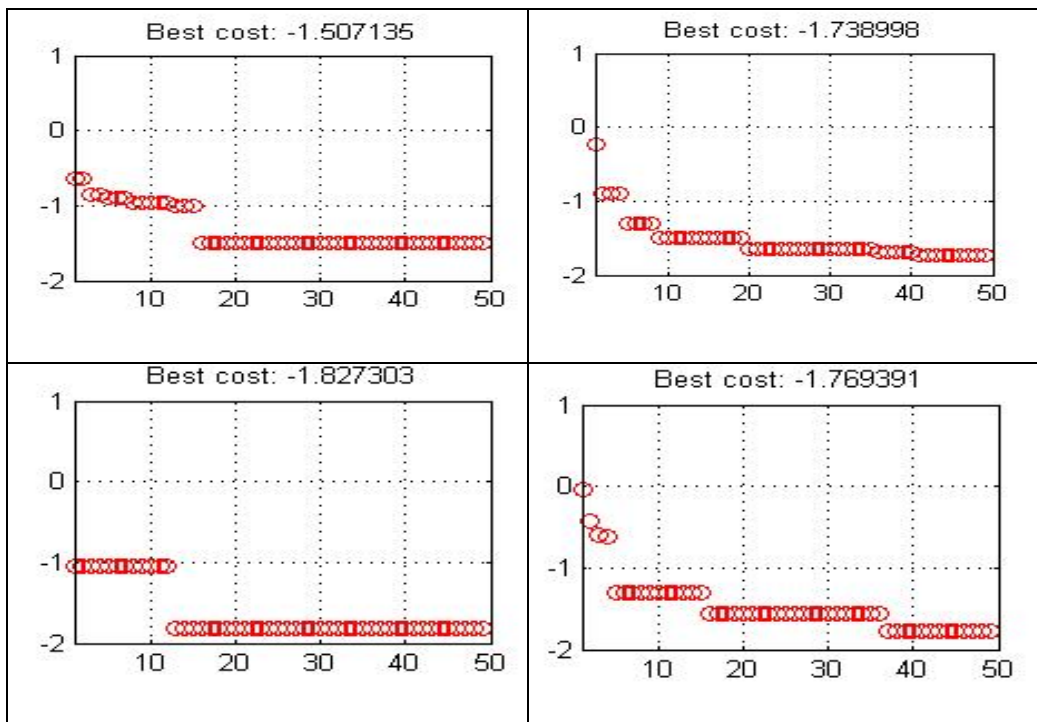
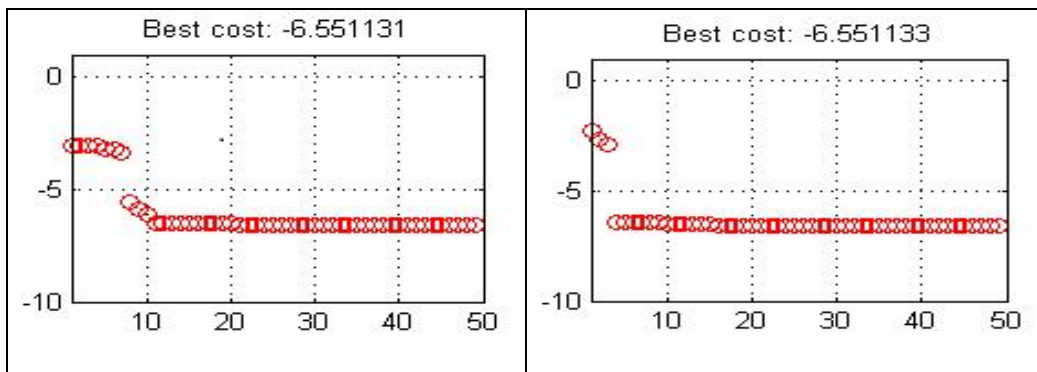


Figure 2. Graphs Showing Performance of DE Variants on NP=20 on Michalewicz Function



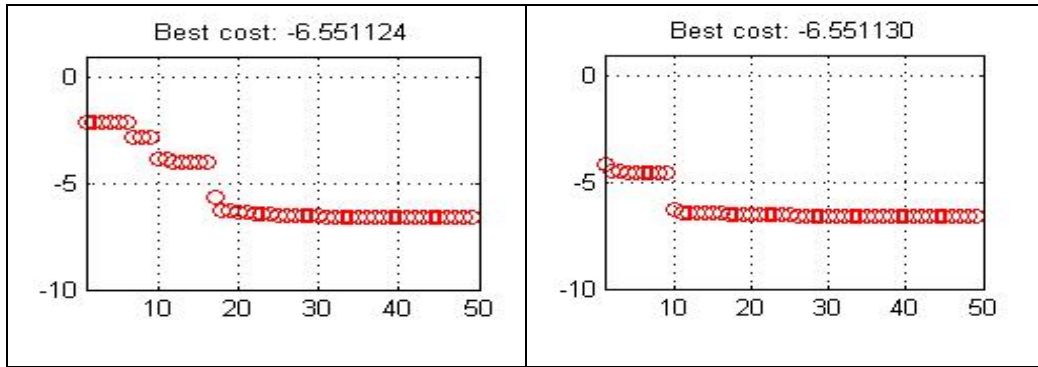


Figure 3. Graphs Showing Performance of DE Variants on NP=20 on Peak Function

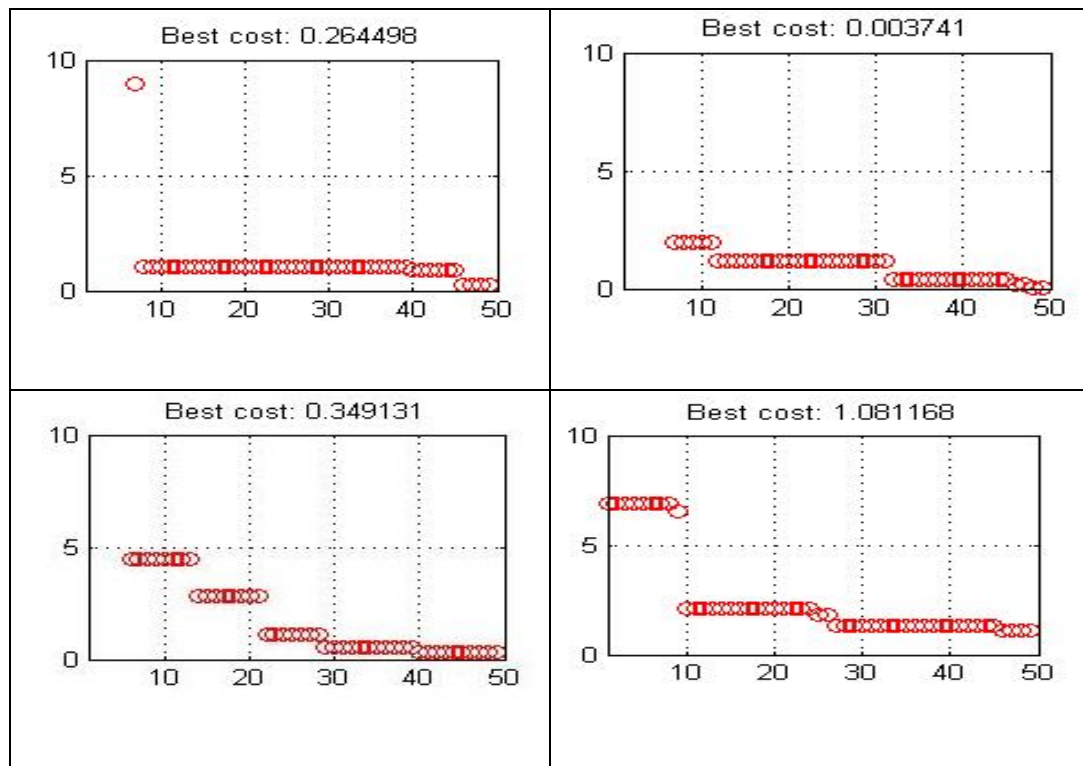
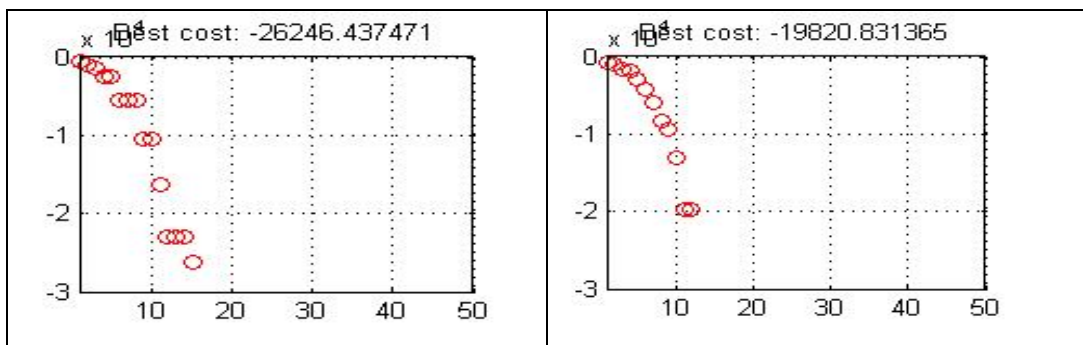


Figure 4. Graphs Showing Performance of DE Variants on NP=20 on Rastrigin Function



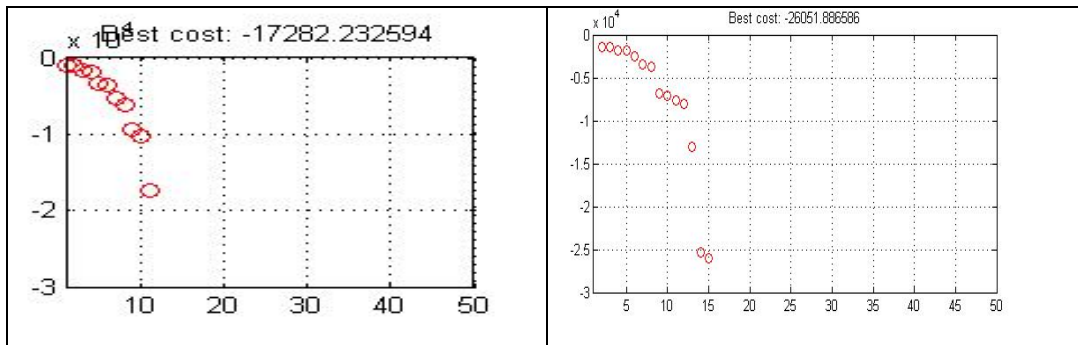


Figure 5. Graphs Showing Performance of DE Variants on NP=50 on Eggholder Function

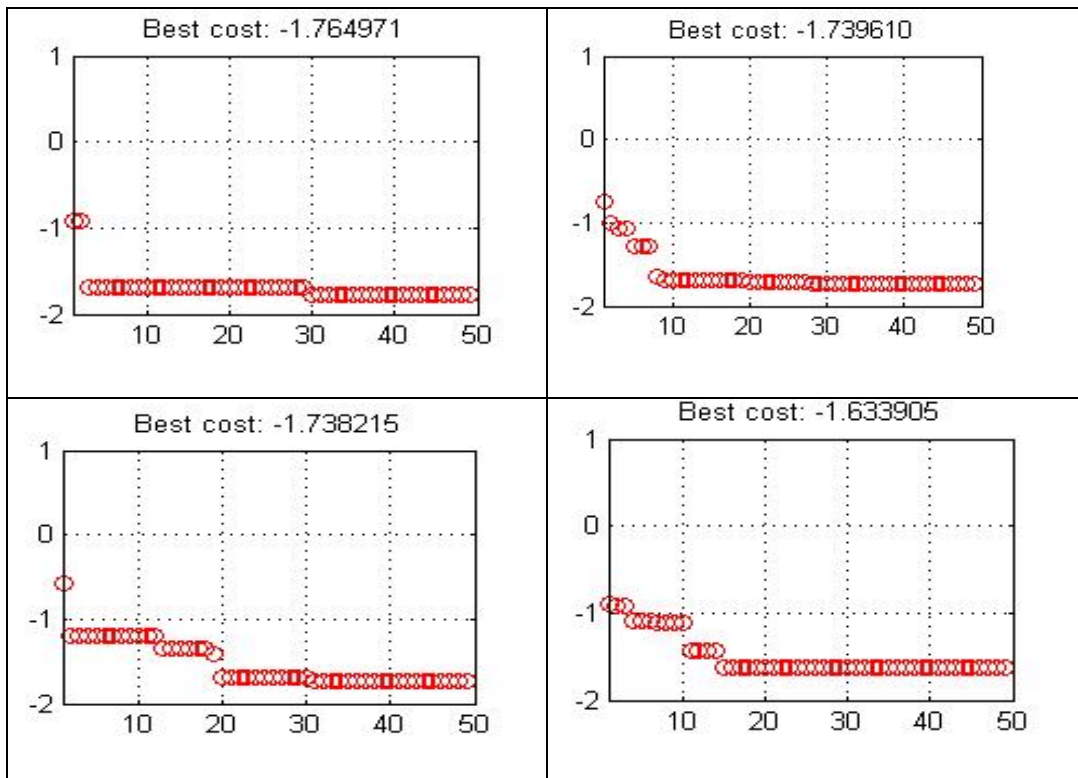
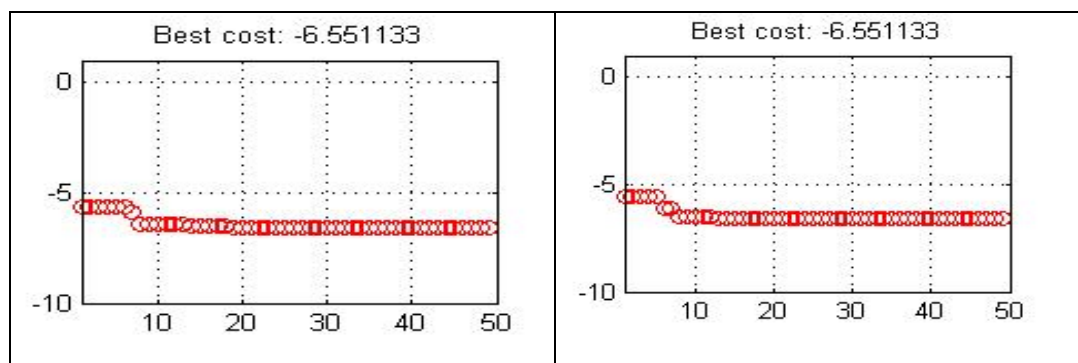


Figure 6. Graphs Showing Performance of DE Variants on NP=50 on Michalewicz Function



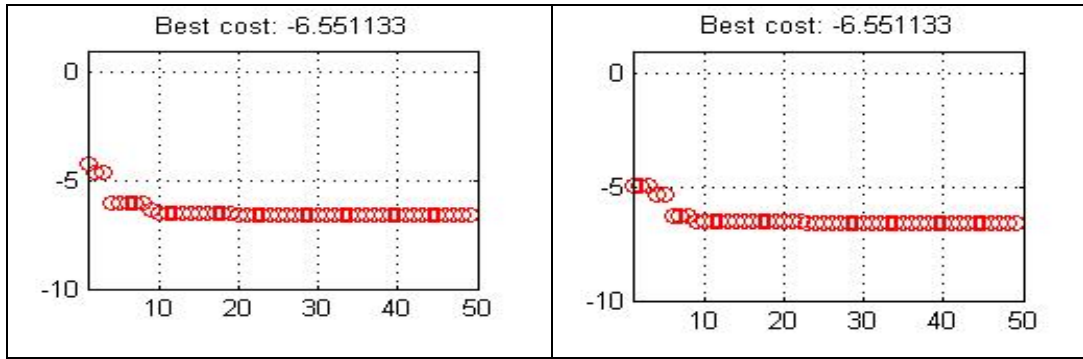


Figure 7. Graphs Showing Performance of DE Variants on NP=20 on Peak Function

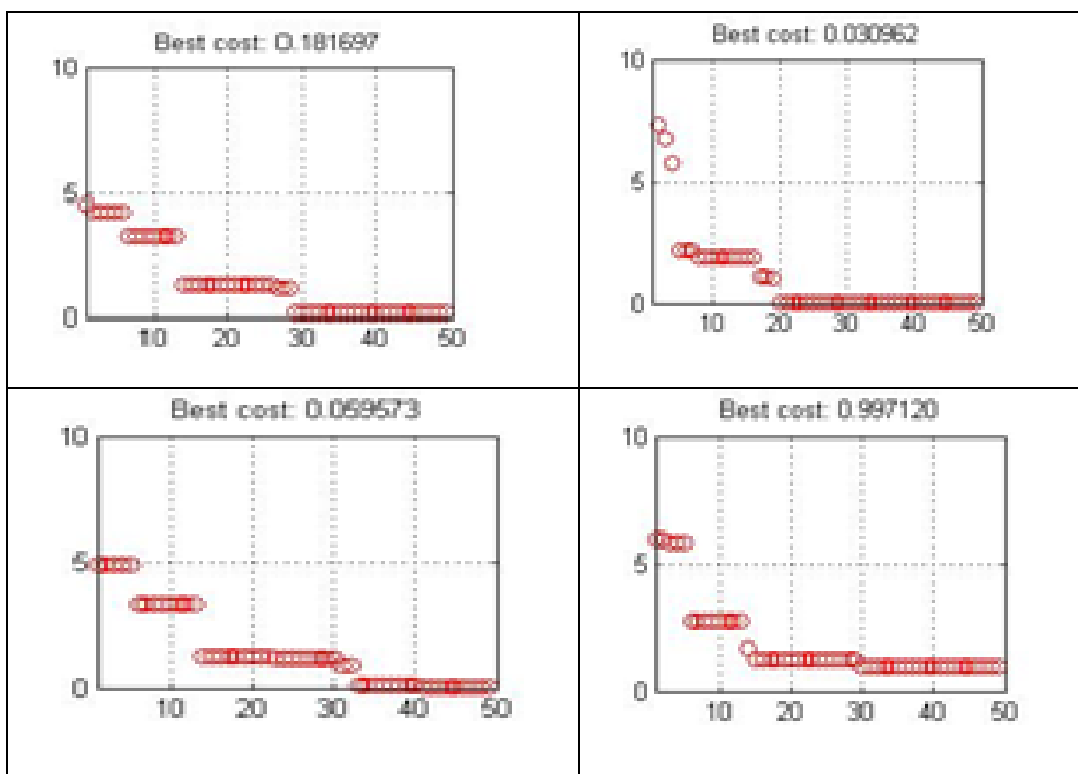


Figure 8. Graphs Showing Performance of DE Variants on NP=20 on Rastrigin Function

5. Conclusion and Future Scope

On statistical analysis of various DE variants on different test functions for NP=20, 50, F=0.8, CR=1.0 and D=2. It is found that DE/local-to-best technique approaches to minimum cost for all the test functions in both the cases. The standard DE/rand performance improves on increasing the population. MODE converges fast in most of the cases but does not give global minima for lower population but on increasing the population, it shows improvement in the performance. In terms of number of functions evaluated to attain objective function, MODE uses least number of functions evaluation for eggholder function whereas almost comparative number of functions are evaluated by all variants for other test functions. For the peak function, no variant is able to achieve global minima for 50 generations.

In future, lot of work can be done for improving the Differential Evolution approach by either using best mutation strategies in it. Further orthogonal crossover can be added. Mutation strategy selection criteria can also be enhanced. Comparative study of MODE can be done by varying the number of generations.

Acknowledgment

We would like to thank almighty for his constant blessings. Then we would like to thank our families and friends for helping and supporting us throughout the making of this paper.

References

- [1] J. Kennedy and R. C. Eberhart, "Particle swarm optimization", Proceedings of IEEE International Conference on Neural Networks, **(1995)**, pp. 1942-1948, Piscataway, New Jersey.
- [2] R. Storn and K. Price, "Differential Evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces", ICSI, Tech. Rep. Tr-95-012, **(1995)**, USA.
- [3] K. Price and R. Storn, "Minimizing the real functions of the ICEC'96 contest by Differential Evolution", IEEE International conference on evolutionary computation (ICEC'96), **(1996)**, pp. 842–844.
- [4] R. Storn, "On the usage of Differential Evolution for function optimization", Biennial conference of the North American fuzzy information processing society (NAFIPS), IEEE, **(1996)**, pp. 519-523, Berkeley.
- [5] K. Price and R. Storn, "Differential Evolution – A simple and efficient heuristic for global optimization over continuous spaces", Journal of Global Optimization, **(1997)**, pp. 341–359.
- [6] K. V. Price, R. M. Storn and J. A. Lampinen, "Differential Evolution: A practical approach to global optimization", Springer, **(2005)**, Berlin.
- [7] V. Feoktistov, "Differential Evolution", in Differential Evolution-In search of solution, **(2006)**, pp. 7-9, USA.
- [8] K. C. Uday, "Advances in differential evolution", **(2008)**.
- [9] S. Das, A. Abraham and A. Konar, "Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspectives", Studies in Computational Intelligence (SCI), vol. 116, **(2008)**, pp. 1-38.
- [10] S. Das, A. Abraham, U. K. Chakraborty and A. Konar, "Differential Evolution Using a Neighborhood-based Mutation Operator", IEEE transactions on evolutionary computations, vol. 3, no. 3, **(2009)**, pp. 526–553.
- [11] R. Mallipeddi and P. N. Suganthan, "Differential Evolution algorithm with ensemble of parameters and mutation and crossover strategies", Swarm, evolutionary, and memetic computing, lecture notes in computer science, vol. 6466, **(2010)**, pp. 71-78.
- [12] S. Das and P. N. Suganthan, "Differential Evolution: A Survey of the State-of-the-art", IEEE Transactions on Evolutionary Computation, vol. 15, no. 1, **(2011)**, pp. 4-31.
- [13] A. Musratt and M. Pant, "Improving the performance of differential evolution algorithm using cauchy mutation", soft computing, vol. 15, issue 5, **(2011)**, pp. 991-1007.
- [14] D. H. Lim, N. H. Luong and C. W. Ahn, "Novel Differential Evolution using multiple deme based mutation", International journal of innovative computing, information and control, vol. 8, no. 5, **(2012)**.
- [15] S. M. Islam and S. Das, "An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization", IEEE transactions on systems, man, and cybernetics—part B: Cybernetics, vol. 42, no. 2, **(2012)**.
- [16] Y. Zhou, L. Xinyu and G. Liang, "A Differential Evolution algorithm with intersect mutation operator", Applied soft computing vol.13, **(2013)**, pp. 390–401.
- [17] A. W. Mohamed, "RDEL: Restart Differential Evolution algorithm with local search mutation for global numerical optimization", Egyptian informatics journal, vol. 15, issue 3, **(2014)**, pp. 175–188.
- [18] O. Baskan and H. Ceylan, "Modified differential evolution algorithm for the continuous network design problem", procedia-social and behavioral sciences, vol. 111, no. 5, **(2014)**, pp. 48–57.