Improved Strategies of Multi-objective Differential Evolution (MODE) for Multi-objective Optimization

Ashish M Gujarathi[1], B V Babu[*]

Chemical Engineering Department
Birla Institute of Tehnology and Science
Pilani – 333031 Rajasthan India
[*]Corresponding author: Dean –Educational Hardware Division (EHD) and Professor,
Phone: +91-1596-245073 Ext. 259 / 212; Fax: +91-1596-244183;
Email: bvbabu@bits-pilani.ac.in
[1]Email: ashishg@bits-pilani.ac.in (Ashish Gujarathi)

Abstract. Multi-objective optimization using an evolutionary computation technique is used extensively for solving conflicting multi-objective optimization problems. In this work, an improved strategy of multi-objective differential evolution (MODE) where the mutation strategy is changed to a trigonometric mutation approach is proposed. The proposed strategy along with other well known strategies of MODE is used to compare the performance metrics (such as convergence and divergence) with other evolutionary algorithms from the literature. The Pareto optimal solutions are obtained for benchmark test functions and are compared using several strategies of MODE. Improved strategies of MODE show a competitive performance when compared with other evolutionary multi-objective optimization algorithms (EMOAs).

Keywords: Multi-objective optimization; evolutionary algorithms, trigonometric mutation; differential evolution; optimization

1 Introduction

Evolutionary Multi-objective optimization (EMO) is an improved approach over the deterministic methods to solve mainly nonlinear, complex and real world problems. Few of the known advantages of EMO include the robustness, their ability to give number of solutions in a single run, their ability to handle the global search space, ease to handle multiple objectives simultaneously, and ease to handle constraints efficiently, etc. Due to these advantages and due to the complexity involved in test problems and real world problems, evolutionary algorithms have gained popularity in solving multi-objective optimization (MOO) problems. Due to the conflicting nature of objectives, it is nearly impossible to get the optimum solution in case of a multi-objective optimization problem. Therefore, compromised solutions are preferred and such solutions are called as the non-dominated solutions rather than the optimum solution. A set of such non-dominated or compromised solutions is called as a Pareto optimal set [1].

Several modifications in the existing EMOs are reported in the literature. Jumping gene adaptation is one of the improvements applied to non-dominated sorting genetic algorithm II (NSGA - II) and the simulated annealing [2, 3]. Elitist strategy of NSGA was proposed to improve the speed of convergence of NSGA [4]. NSGA-II uses the concept of elitism, in which better chromosomes from current generation are copied to the next generation. But due to the elitism mechanism (to preserve better chromosomes) the diversity of the algorithm decreases. Kasat et al. [2] introduced jumping genes (JG) into NSGA-II. This improved strategy of NSGA-II (NSGA-II-JG and NSGA-II-aJG) performed much better than NSGA for various test problems and real world industrial applications. Jumping gene adaptation was also applied to Simulated Annealing algorithm to improve the performance of Simulated Annealing algorithm [3].

Multi-objective differential evolution (MODE) algorithm is an extension of differential evolution algorithm to solve multi-objective optimization problems [5, 6]. MODE algorithm is successfully applied on several test problems and on real life industrial applications [7-12]. Two improved strategies of MODE, i.e., MODE II and MODE III were proposed by Babu et al. [13]. MODE II was further improved by incorporating the concept of elitism [14]. Hybrid strategy of MODE was proposed to improve the convergence of MODE III algorithm [15]. In this work we propose yet another strategy of MODE, where trigonometric mutation operation [16] is applied on MODE III algorithm. The developed algorithm is successfully tested on several benchmark test problems (both constrained and unconstrained). The Pareto front obtained using newly developed algorithm is compared with the Pareto front obtained using MODE III, E-MODE, and MODE algorithms. The performance of Trigonometric MODE algorithm is compared (for convergence and divergence metrics) with other well known algorithms (such as NSGA-II (both binary and real coded), Strength Pareto evolutionary algorithms (SPEA) and Pareto achieved evolutionary algorithm (PAES) for two selected test problems [1, 4]. In the following sections we present the working principles of MODE III and trigonometric MODE followed by the results and discussion.

2 Improved Strategies of Multi-objective differential evolution (MODE)

Multi-objective differential evolution is developed to solve multi-objective optimization problems. MODE involves non-dominance sorting of current population members after each generation. The non-dominance sorting is applied to remove the dominated population points and to give a better direction to the algorithm towards the Pareto front. However, repetitive non-dominance sorting may reduce the size of population points and may result in minimum number of points on the Pareto front. MODE III algorithm was proposed by Babu et al. [13] to remove the problem of reduction of population size. MODE III algorithm is simple extension of differential evolution algorithm with modified selection scheme. In MODE III algorithm only the non-dominated points may enter in the current population. Thus the selection scheme itself ensures that dominated and worst points are discarded from the current population and only the better and non-dominated points enter in the population. Thus the population points are updated and Pareto front is obtained. The non-dominated sorting is invoked at the end of the algorithm to ensure that the converged front does not contain any dominated point. The pseudo-code of MODE III algorithm may be represented as below:

```
**Pseudo-code for MODE - III
Initialize the Crossover Constant (CR), Maxgen, Size of Population (NP), Number
of Dimensions (D)
for i=1:NP
        for j=1:D
                X_ij= Lower (j) + (Upper (j) –Lower (j)) *rand (0,1);
End for
Evaluate Cost(i)
End for
%Perform mutation, crossover, selection and evaluation of the objective function for
%trial and target vector for a specified number of generations.
For gen=1:Maxgen  **Generation Loop**
For i:1:NP                        **Population Loop**
        Select Target vector X_i
        Select three distinct vectors X_a, X_b and X_c other than X_i
        do
        {
                r_1=round (rand*NP)
                r_2=round (rand*NP)
                r_3=round (rand*NP)
        }
while [(r_2== i) || (r_1==i) ||(r_3== i) ||(r_1== r_2 ) ||(r_1== r_3) ||(r_2== r_3)]

J_rand=int[rand(0,1)*D] + 1
for q=1:D
{
if (p<CR || q==j_rand)
X_t,i=Xr_1 + F (Xr_2-Xr_3)              %Trial vector
else
X_t,i=X_i,j                             %Trial vector
}
Evaluate Trial vector
%Perform selection for each target vector, X_i by comparing its function value with
%that of the trial vector, X_t,i . If X_t,i dominates X_i then select X_t,i  otherwise select X_i
%for the %next generation population.
If (X_t,i dominates X_i)
Replace Target with current X_t,i
else   Retain current X_i as a Target vector
```

*end for **population loop end***
*end for **generation loop end***

Trigonometric Mutation operation for MODE III. As shown in the pseudo-code above, the mutation operation is carried out in MODE and MODE -III algorithm using three distinct vectors. These vectors are perturbed by applying a scale factor so that a new and hopefully efficient vector (noisy random vector) is created [16]. However, the noisy random vector thus created does not get any direction towards the better function value. This is achieved by applying trigonometric mutation operation to the selected vectors. Earlier trigonometric mutation operation was applied to differential evolution algorithm to solve single objective optimization problems [17]. In this work we apply trigonometric mutation operation to MODE III algorithm to solve multi-objective optimization problems. The trigonometric mutation operation for multi-objective optimization problems is given by the following pseudo-code:

```
**Pseudo-code for trigonometric mutation operation for MOO
Select three distinct vectors Xa, Xb and Xc other than Xi
      do
      {
            r1=round (rand*NP)
            r2=round (rand*NP)
            r3=round (rand*NP)
      } while [(r2== i) || (r1==i) ||(r3== i) ||(r1== r2 ) ||(r1== r3) ||(r2== r3)]
If (rand(0,1) >Mt)
      Tempp1=f(Xr1);
      Tempp2=f(Xr2);;
      Tempp3=f(Xr3);
      Sum= |Tempp1|+| Tempp1|+| Tempp1|
      p1= |Tempp1|/sum;
      p2= |Tempp2|/sum;
      p3= |Tempp3|/sum;
      for j=1:D
      Xt,i,j = (Xr1(1,j)+X r2(1,j) + X r3(1,j))/3 + abs(p2-p1)*(X r1 (1,j) - X r2 (1,j))
      + abs(p3-p2)*(X r2(1,j) - Xr3(j)) + abs(p1-p3)*(X r3 (1,j)-X r1(1,j));
      end for
else
      Xt,i=Xr1 + F (Xr2-Xr3)                    %Trial vector
End
```

Simple mutation operation involves the random selection of first individual vector out of randomly selected three vectors. The scaled difference is added to this randomly selected individual. In case of trigonometric mutation operation, the center point of the hyper geometric point is taken as the vector to be perturbed. As seen from the pseudo-code of trigonometric mutation operation for MOO, the perturbation in the trigonometric mutation operation is contributed together by three vertices of the triangle defined by three randomly selected vectors. The weights applied to the vectors differentials i.e. (p_2-p_1), (p_3-p_2), and (p_1-p_3) ensures that the new point moves in the direction of improved objective function value. In case of single objective optimization, $Tempp_1$, $Tempp_2$ and $Tempp_3$ (as given in pseudo-code above) are the variables which contain a single objective function value.

However, in case of multi-objective optimization $Tempp_1$, $Tempp_2$ and $Tempp_3$ are the vectors which contain the values of evaluated multiple objective functions. The sum is also a vector which contains the sum of individual objective functions. The mean value of all the objectives is calculated as shown by p_1, p_2 and p_3 variables. The noisy random vector is now created by using the p_1, p_2 and p_3 variables as weight as given in above pseudo-code. It is ensured that the noisy random vector moves in the better direction where there is an improvement in the objective function value. The mutation probability used in this algorithm is 0.5%, i.e. if the random number generated is greater than 0.5 then trigonometric mutation operation is carried out otherwise simple mutation operation as given in MODE III algorithm is used.

Table 1 gives the test problems, number of variables and its bounds and nature of Pareto front. 5 test problems are used in this study to compare the performance of trigonometric mutation MODE Pareto front with the Pareto front obtained using other algorithms. Following section deals with the results and discussion of present study.

Table 1. Multi-objective optimization problems, number of variables, their bounds and nature of Pareto front considered in the present study

Problem	Description	n	Bounds	Pareto front
Schaffers Study 1 (SCH1) [18]	$f_1(x) = x^2$ $f_2(x) = (x-2)^2$	1	$[-10^{-3}, 10^3]$	Convex
Fonseca's and Fleming's Study (FON) [1]	$f_1(x) = 1 - \exp\left(-\sum_{i=1}^{n}\left(x_i - \frac{1}{\sqrt{n}}\right)^2\right)$ $f_2(x) = 1 - \exp\left(-\sum_{i=1}^{n}\left(x_i + \frac{1}{\sqrt{n}}\right)^2\right)$	3	$[-4,4]$	Nonconvex
Schaffers Study 2 (SCH2) [18]	$f_1(x) = \begin{cases} -x & \text{if } x \le 1 \\ x-2 & \text{if } 1 < x \le 3 \\ 4-x & \text{if } 3 < x \le 4 \\ x-4 & \text{if } x > 4 \end{cases}$ $f_2(x) = (x-5)^2$	1	$[-5,10]$	Nonconvex, Disconnected
Poloni's Study (POL) [19]	$f_1(x) = \left[1 + (A_1 - B_1)^2 + (A_2 - B_2)^2\right]$ $f_2(x) = \left[(x_1+3)^2 + (x_2+1)^2\right]$ $A_1 = 0.5\sin 1 - 2\cos 1 + \sin 2 - 1.5\cos 2,$ $A_2 = 1.5\sin 1 - \cos 1 + 2\sin 2 - 0.5\cos 2$ $B_1 = 0.5\sin x_1 - 2\cos x_1 + \sin x_2 - 1.5\cos x_2$ $B_2 = 1.5\sin x_1 - \cos x_1 + 2\sin x_2 - 0.5\cos x_2$	2	$[-\pi, \pi]$	Nonconvex, Disconnected
Binh and Korn Study (BNH) [20]	$f_1(x) = 4x_1^2 + 4x_2^2$ $f_2(x) = (x_1-5)^2 + (x_2-5)^2$ $Sub\ to\ C_1(x) \equiv (x_1-5)^2 + x_2^2 \le 25$ $C_2(x) \equiv (x_1-8)^2 + (x_2+3)^2 \ge 7.7$	2	$x_1 \in [0,5]$ $x_2 \in [0,3]$	Convex, Constrained

3 Results and Discussion

The trigonometric mutation MODE code (developed using Matlab 7.0 routine) is tested for various test problems as given in Table 1. The parameters used in the present study are same as that reported in the literature [1]. The crossover constant is assumed to be 0.9, maximum numbers of generation is 250, and the scaling factor is generated randomly. Initial population size is kept to be 100 for all the algorithms. Two widely used performance metrics are used for comparing the performance of trigonometric MODE with other algorithms reported in the literature. The first metric is the convergence metric (γ) in which the average distance between the currently obtained non-dominated solutions and the true Pareto front is obtained. Another metric used is the divergence metric (Δ). This metric uses the diversity of currently obtained non-dominated solutions among themselves [1]. The average and the variance (square of standard deviation) for 10 different runs for the convergence and the divergence metric are reported in table 2 and table 3 respectively.

Table 2. Performance metric (Convergence) comparison of several algorithms on selected test problems

Algorithm		SCH	FON
Real coded NSGA-II˜	γ	0.003391	**0.001931**
	σ^2_γ	**0**	**0**
NSGA-II Binary˜	γ	0.002833	0.002571
	σ^2_γ	0.000001	**0**
SPEA˜	γ	0.003465	0.010611
	σ^2_γ	**0**	0.000005
PAES˜	γ	**0.001313**	0.151263
	σ^2_γ	0.000003	0.000905
MODE*	γ	0.0021	0.02554
	σ^2_γ	**0**	0.00063
MODE III*	γ	0.002236	0.003381
	σ^2_γ	**0**	**0**
E-MODE*	γ	*0.001948*	*0.002119*
	σ^2_γ	**0**	**0**
T-MODE*	γ	0.002035	0.003151
	σ^2_γ	**0**	**0**

*Values obtained in the present study, ˜Values taken from ref. [1, 4]
First Best: **Bold** Font; Second Best: *Italic* font

Performance metrics for two test functions (namely SCH and FON) are reported in this study. Table 2 shows that PAES converged to the greatest extent for SCH test problem. All the algorithms except binary coded NSGA-II show the variance value as zero. Elitist strategy of MODE performed second best in case of convergence for SCH test problem. For FON test problem, real coded NSGA-II outperformed other algorithms. Binary coded NSGA-II and T-MODE algorithms performed better than other algorithms for SCH test problem for average value of diversity metric and the variance value respectively. Real coded NSGA-II is second best for average diversity aspect; however MODE III is second best for variance value. FOR FON test problem, in terms of diversity the real coded NSGA-II performed better than other algorithms. Both the average and the variance values of real coded NSGA-II algorithm are better than other algorithms. NSGA-II (binary coded) is second best for average value of diversity metric whereas MODE III is second best for the variance value.

Fig. 1 shows the objective space and the Pareto optimal solutions obtained using several algorithms. The objective space is convex in nature. Because SCH is a single variable, unconstrained multi-objective optimization test problem, it is relatively easy to solve in comparison to other complex test problems. Fig. 2 shows the Pareto optimal front obtained using different strategies of MODE algorithm. Fig. 2 shows that Pareto optimal solutions lie in the range of $x \in [0,2]$. Both f_1 and f_2 objective functions covers the range [0, 4].

Table 3. Performance metric (divergence) comparison of several algorithms on selected test problems

Algorithm		SCH	FON
Real coded NSGA-II˜	Δ	*0.477899*	**0.378065**
	σ^2_Δ	0.003471	**0.000639**
NSGA-II Binary˜	Δ	**0.449265**	*0.395131*
	σ^2_Δ	0.002062	0.001314
SPEA˜	Δ	0.818346	0.804113
	σ^2_Δ	0.004497	0.002961
PAES˜	Δ	1.063288	1.162528
	σ^2_Δ	0.002868	0.008945
MODE*	Δ	0.67099	0.70069
	σ^2_Δ	0.01332	0.03397
MODE III*	Δ	0.59953	0.620052
	σ^2_Δ	*0.00155*	*0.00095*
E-MODE*	Δ	0.571475	0.700227
	σ^2_Δ	0.006496	0.018964
T-MODE*	Δ	0.596636	0.612571
	σ^2_Δ	**0.000926**	0.002879

*Values obtained in the present study, ˜[1, 4]
First Best: **Bold** Font; Second Best: *Italic* font

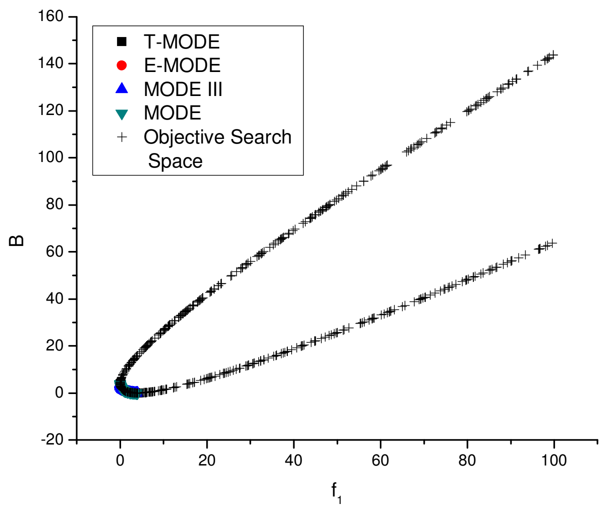

Fig. 1. Objective space and the Pareto optimal front for SCH 2 test problem

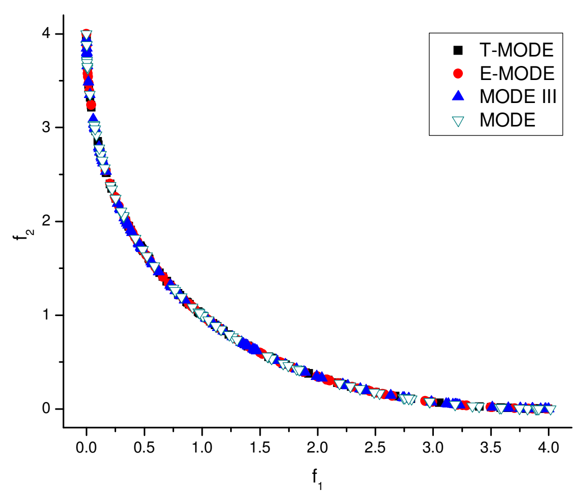

Fig. 2. Pareto optimal solutions for SCH 1 test problem

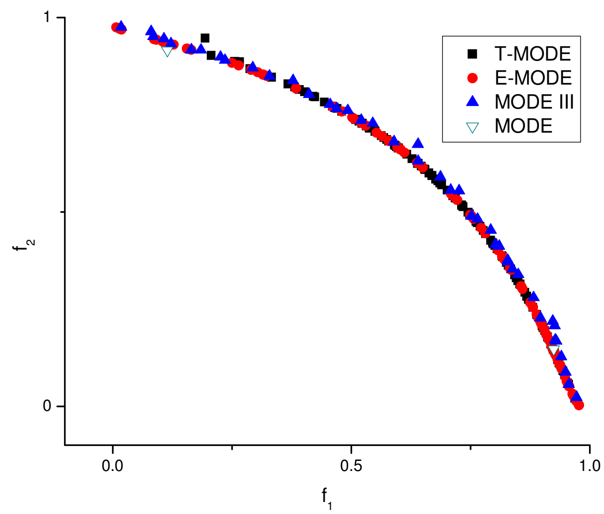

Fig. 3. Comparison of Pareto optimal solutions for FON test problem

The Pareto fronts obtained using FON test problem is shown in fig. 3. FON test problem is nonconvex in nature and therefore it is difficult for any algorithm to converge to the true Pareto front. The Pareto optimal solutions correspond to $x_i^* = -1/\sqrt{3}$. The MODE algorithm resulted in only 2 numbers of points on the Pareto front (out of initial population of 100) after a specified numbers of generations.

SCH2 is a single variable bi-objective optimization problem with two discontinuous regions as shown in Fig. 4. The objective space is nonconvex in nature. The dominated region and the objective space boundary are shown in Fig. 4. The Pareto optimal front is shown in the region marked with box. The Pareto front lies in the range of decision variable $x^* \in [1,2] \cup [4,5]$. Region A-B (in fig. 4) corresponds to f_1 value in the range of [-1 0] and region C-D corresponds to f_1 value in the range of [0 1] satisfying the x^* value as given above. The Pareto fronts obtained using several strategies of MODE algorithm is shown in Fig. 5. All the strategies of MODE converged to the true Pareto optimal front with numbers of solutions as shown in Table 3.

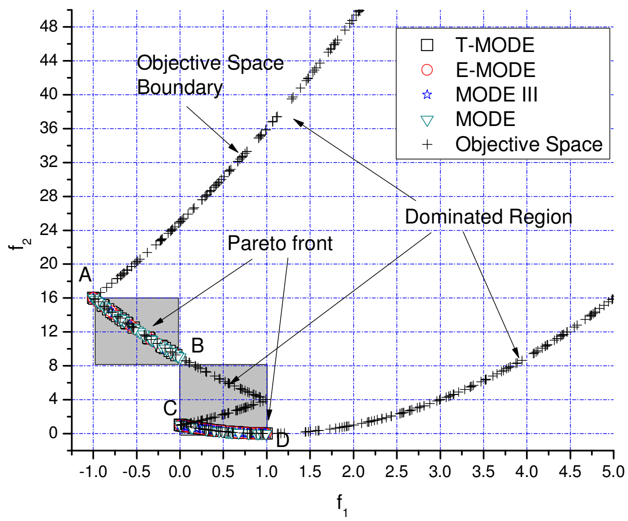

Fig. 4. Search space and the disconnected Pareto front for POL test problem

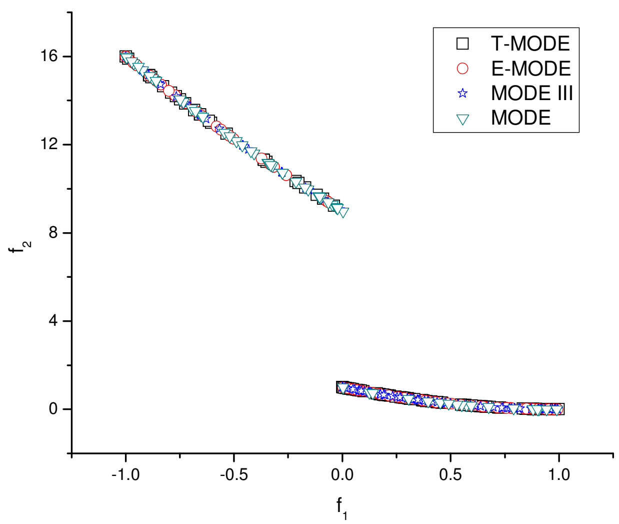

Fig. 5. Pareto front using several algorithms for POL test problem

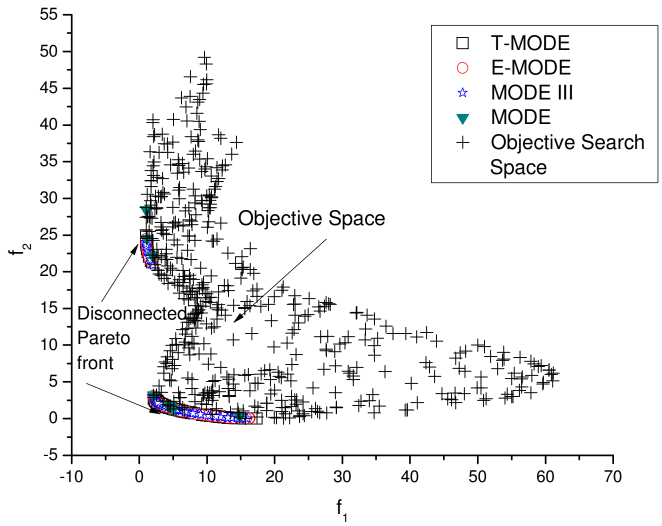

Fig. 6. Search space and the disconnected Pareto front for POL test problem

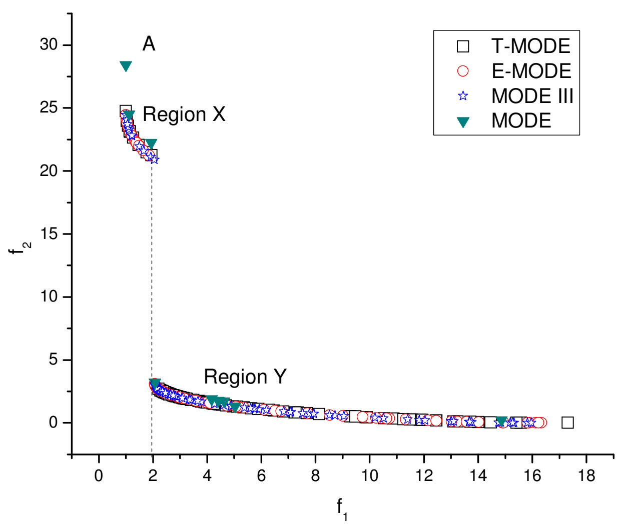

Fig. 7. Pareto front for POL test problem

The nonconvex objective space and the disconnected Pareto fronts for POL test problem is shown in Fig. 6. Region X and region Y totally depend on the bounds of the decision variables. Region X is obtained when the decision variable x_1 converged to its lower bound. Fig. 7 shows that MODE algorithm converged to the local Pareto front in region X. Point A $[(f_1,f_2) = (1.00, 28.422)]$ is the dominated point when the algorithm gets converged to the global Pareto front. Thus the local Pareto optimal front may not only shrunk but also may widen the local Pareto front in the given search space. The value of $f_1 < 2$ is obtained when x_1 approaches its lower bound of $-\pi$.

BNH is a bi-objective constrained problem having a convex objective search space. Due to the convex nature of search space, all algorithms are able to reach the global Pareto front. Due to the simple convex nature of the search space all the algorithms converged to the Pareto front with 100% convergence ratio of initial points. The effect of constraints on the Pareto front is shown in Fig. 9. Trigonometric mutation algorithm is run individually after relaxing constraint C_1, C_2 and then both C_1 and C_2. Fig. 9 shows that constraint C_2 is redundant in nature. The constraint C_1 and C_2 does not change the nature of Pareto front as shown in Fig. 9. The Pareto optimal solutions are due to variables in the range of $x_1{}^* = x_2{}^* \in [0,3]$ *and* $x_1{}^* \in [3.5]$, $x_2{}^* = 3$ Both x_1 and x_2 values increase from 0 to 3 linearly till x_2=3. After x_2=3, with further increase in x_1 variable value, the value of variable x_2 remains almost constant.

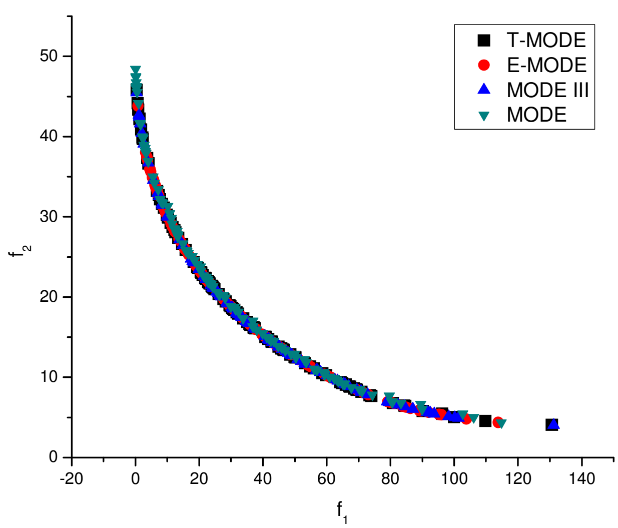

Fig. 8. Pareto optimal solutions for BNH test problem

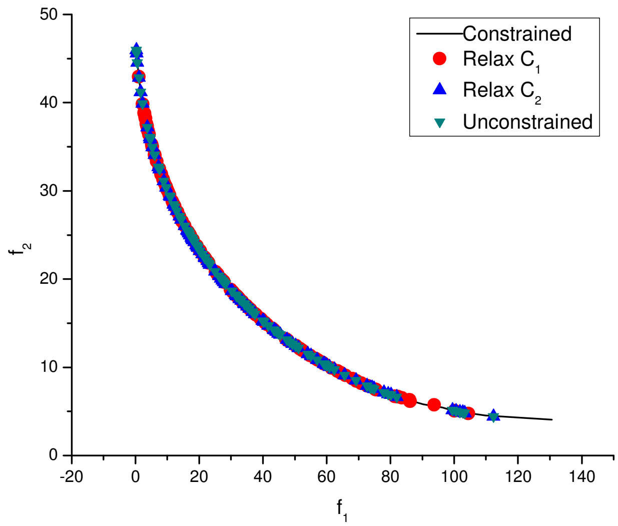

Fig. 9. Effect of constraints on BNH test problem Pareto front

Table 3. Number of points converged to Pareto optimal front

Test problem	T-MODE	E-MODE	MODE- III	MODE
SCH1	100	100	100	69
SCH 2	100	100	100	60
FON	99	100	43	02
POL	98	100	100	9
BNH	100	100	100	100

4 Conclusions

Trigonometric mutation operation is successfully applied to an improved strategy of multi-objective differential evolution (MODE). The proposed trigonometric mutation multi-objective differential evolution (T-MODE) algorithm is tested for several benchmark test problems. The performance metrics such as convergence metric and the divergence metric are calculated for T-MODE, E-MODE, MODE III and MODE algorithms for two test functions (SCH and FON) and are compared with other well known algorithms from the literature. The proposed algorithm is found to be competitive one for solving multi-objective optimization problems. The Pareto optimal solutions are obtained and are compared using improved strategies of MODE for several benchmark test functions (such as SCH1, FON, SCH2, POL and BNH). The proposed algorithm can be used to solve the complex test problems and the real world engineering problems.

References

1. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons Limited, New York (2001)
2. Kasat, R. B., Gupta, S. K.: Multi-objective optimization of an industrial fluidized-bed catalytic cracking unit (FCCU) using genetic algorithm (GA) with the jumping genes operator, Compu. Che. Engng., 27, 1785--1800 (2003)
3. Sankarrao, B., Gupta, S. K.: Multi-objective optimization of an industrial fluidized bed catalytic cracking unit (FCCU) using two jumping gene adaptations of simulated annealing, Comput. Che. Eng., 31, 1496--1515 (2007)
4. Deb, K., Agarwal, S., Pratap, S., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II, In Proceedings of the parallel problem solving from nature VI (PPSN-VI), 849--858 (2000)
5. Price, K.V., Storn, R.,: Differential evolution – a simple evolution strategy for fast optimization,. Dr. Dobb's J., 22, 18--24 (1997)
6. Babu, B. V.: Process Plant Simulation, New York: Oxford University Press (2004)
7. Babu, B. V., Chakole, P. G., Mubeen, J.H.S.: Multi-objective Differential Evolution (MODE) for Optimization of Adiabatic Styrene Reactor, Che. Eng. Sci., 60, 4822--4837 (2005)

8. Babu, B. V., Gujarathi, A. M.: Multi-Objective Differential Evolution (MODE) for Optimization of Supply Chain Planning and Management. In *Proc. IEEE Cong. E.C. (CEC-2007)*, 2732
9. Angira, R., Babu, B. V.: Multi-Objective Optimization using Modified Differential Evolution (MDE). Int. J. Math. Sci.: Spe. Issue Rec. Trends Comp. Math. App, 5, 371--387 (2006)
10. Babu, B. V., Jehan, M. M. L.: Differential Evolution for Multi-Objective Optimization, Proceedings of 2003 Congress on Evolutionary Computation (CEC-2003), Canberra. 2696--2703 (2007)
11. Babu, B. V., Mubeen, J.H.S., Chakole, P. G.: Multiobjective Optimization Using Differential Evolution, TechGenesis- J. Info. Tech., 2, 4--12 (2005)
12. Gujarathi, A. M., Babu, B. V.: Improved Multi-objective Differential Evolution (MODE) Approach for Purified Terephthalic Acid (PTA) Oxidation Process. Mat. Manuf. Processes. 24, 303-- Gujarathi, Ashish M.; Babu, B. V. Improved Multi-objective Differential Evolution (MODE) Approach for Purified Terephthalic Acid (PTA) Oxidation Process. Mat. Manuf. Processes, 24, 303--319 (2009)
13. Babu, B. V., Gujarathi, A. M., Katla, P., Laxmi, V. B.: Strategies of Multi-Objective Differential Evolution (MODE) for Optimization of Adiabatic Styrene Reactor,. Proceedings of International Conference on 'Emerging Mechanical Technology-Macro to Nano, BITS Pilani, 243--250 (2007)
14. Babu B. V., Gujarathi, A. M.: *Elitist*-Multi-objective Differential Evolution (E-MODE) Algorithm for Multi-objective Optimization, International conference on Artificial intelligence, 441--456 (2007)
15. Gujarathi, A.M., Babu, B. V.: Hybrid Multi-objective Differential Evolution (H-MODE) for optimization of Polyethylene Terephthalate (PET) reactor, Int. J. Bio-insp. Compu. Accepted (In Press).
16. Onwubolu G. C. and Babu, B. V.: New Optimization Techniques in Engineering, Springer-Verlag, Heidelberg, Germany (2004)
17. Fan, H. Y., Lampinen, J.: A trigonometric mutation operation to differential evolution, J. Glob. Opt., 27, 105--129 (2003)
18. Schaffer, J. D.: Some experiments in Machine Learning using Vector evaluated Genetic Algorithms, Ph.D Thesis, (1984), Nashville, TN: Vanderbilt University
19. Poloni, D., Giurgevich, A., Onesti, L., Pediroda, V.: Hybridization of multi-objective genetic algorithm, a nueral netwerok and a classical optimizer for complex design problem in fluid dynamics, Compu. Meth. App. Mech. Engng., 186, 424--430 (2000)
20. Binh, T.T., Korn, U.: MOBES: A multi-objective evolution strategy for constrained optimization problems, In Third International Conference on Genetic Algorithms, 176--182 (1997)