academicJournals

Vol. 8(9), pp. 315-326, 9 March, 2013
DOI: 10.5897/IJPS12.711
ISSN 1992-1950 © 2013 Academic Journals
http://www.academicjournals.org/IJPS

International Journal of Physical Sciences

Full Length Research Paper

Performance evaluation of multi objective differential evolution algorithm (MDEA) strategies

Fred Otieno and Josiah Adeyemo*

Durban University of Technology, P. O. Box 1334, Durban, 4000, South Africa.

Accepted 28 February, 2013

This study presents the results of comparison of the four strategies of multiobjective differential evolution algorithm (MDEA) namely, MDEA1, MDEA2, MDEA3 and MDEA4 in solving five test problems and an engineering design problem. The test problems are ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6. MDEA is a multiobjective algorithm developed from differential evolution, which is an evolutionary algorithm, to solve multiobjective optimization problems. The strategies were also compared with the results obtained from two other similar algorithms: NSGAII and DEMO. It was found that strategies of MDEA named MDEA1 and MDEA2 with binomial crossover method outperformed other two strategies named MDEA3 and MDEA4 with exponential crossover method in solving ZDT1 and ZDT4 while MDEA3 and MDEA4 outperformed MDEA1 and MDEA2 in solving ZDT2 and ZDT6. All the strategies performed equally in solving the engineering design problem presented. It was concluded that the strategies of MDEA are comparable/better than similar algorithms presented in solving the test problems and the engineering design problem.

Key words: Multiobjective differential evolution algorithm (MDEA), multiobjective, evolutionary algorithm, differential evolution.

INTRODUCTION

Differential evolution (DE) has grown over the years with several extensions to solve multiobjective problems (Abbass and Sarker, 2002; Angira and Babu, 2005; Babu and Jehan, 2003; Chakraborti et al., 2009; Madavan, 2002; Nariman-Zadeh et al., 2009). They are competing well with other multiobjective evolutionary algorithms (MOEAs) and they are producing good results (Angira and Babu, 2005; Ascia et al., 2012; Branke et al., 2009; Fan et al., 2006). However, more strategies of existing MOEAs and development of new ones are necessary to overcome some of the shortcomings and to develop more efficient and faster algorithms. The performance of different MOEAs based on differential evolution and their strategies are yet to be determined. This performance test will aid the modifications necessary to the existing algorithms and development of new ones. Several MOEAs have been developed over the past decade. They are based on different algorithms (Deb, 1999; Robic and Filipic, 2005; Sarker and Ray, 2009; Wang et al., 2009; Xue et al., 2003; Zitzler and Thiele, 1999). They have been applied to real world problems with success (Konstantinidis et al., 2012; Lee et al., 2008; Marcelloni and Vecchio, 2012). The advantages of MOEA in solving real world optimization problems cannot be overemphasized. They are better than classical methods in solving multiobjective problems. They are faster, easy to use and with few control parameters especially differential evolution. The objectives in design problems are increasing every day and they are usually conflicting with multiple constraints. This growth has necessitated the development of more powerful algorithms to find solutions to them. Researchers in MOEAs are rising up to

*Corresponding author. E-mail: josiaha@dut.ac.za, tintmodel@yahoo.com. Tel: +2731 373 2895. Fax: +27 31 373 2816.

Table 1. Formulations of the four different strategies of multi-objective differential evolution algorithm (MDEA).

Strategy	Description	Formulation
MDEA1	DE/rand/1/bin	$v(g,i,j) = x(g,r_3,j) + F * [x(g,r_1,j) - x(g,r_2,j)]$
MDEA2	DE/rand/2/bin	$v(g,i,j) = x(g,r_5,j) + F * [x(g,r_1,j) + x(g,r_2,j) - x(g,r_3,j) - x(g,r_4,j)]$
MDEA3	DE/rand/1/exp	$v(g,i,j) = x(g,r_3,j) + F * [x(g,r_1,j) - x(g,r_2,j)]$
MDEA4	DE/rand/2/exp	$v(g,i,j) = x(g,r_5,j) + F * [x(g,r_1,j) + x(g,r_2,j) - x(g,r_3,j) - x(g,r_4,j)]$

Source: Adeyemo and Otieno (2010).

this challenge with the development of more powerful algorithms. Researchers are not working in isolation. They are developing hybrid algorithms using the advantages of different algorithms to develop a hybrid algorithm better than the algorithms (parents) themselves. This phenomenon is similar to DE algorithms and at large evolutionary algorithms (EAs) where the offspring produced from the parents is usually better than the parents.

While the objective of single objective optimization problem is to come up with a solution to a problem, multiobjective presents many non-dominated solutions. The solutions are non-dominated in the sense that no solution is superior to another one in a set of non-dominated solutions when all the objectives are considered (Deb, 2001). A final solution is thus chosen based on other information available to a user.

Multiobjective differential evolution algorithm (MDEA) has been widely used to solve multiobjective problems with multiple constraints especially crop planning and optimum cropping patter determination (Adeyemo and Otieno, 2010). While MDEA was tested on test problems and engineering design problem, other strategies developed, MDEA2, MDEA3 AND MDEA4 are yet to be tested on test problems to determine their performance using different problems. The objective of this study was to determine the performance of the four different strategies of MDEA using different test problems and an engineering design problem. We wanted to find out the right parameter setting for each of the strategies for their optimum performance. Furthermore, we wanted to find out the best strategy for different types of problems and to find out any improvement needed for their optimum performance.

Strategies of multiobjective differential evolution algorithm (MDEA)

MDEA was proposed by Adeyemo and Otieno (2009). The algorithm is based on the first strategy of differential evolution which is DE/rand/1/bin which denotes that it uses binomial crossover and one vector for perturbation (Storn and Price, 1997). Later they proposed three other strategies and named them MDEA2, MDEA3 and MDEA4 and named the original one MDEA1 (Adeyemo and Otieno, 2010). MDEA2, MDEA3 and MDEA4 use the description DE/rand/2/bin, DE/rand/1/exp and DE/rand/2/exp as their formulations respectively (as presented in Table 1) from the ten strategies proposed by Storn and Price (1997).

Initially, practical advice by Price and Storn (2008) was followed in choosing crossover constant (CR) and scaling factor (F) but the values were later varied to get the best results from the strategies of the MDEA.

In MDEA, the vectors are randomly generated to create initial solutions to the problem. The generated solutions are allowed to undergo mutation, crossover and selection for the chosen number of generations just like the traditional differential evolution. The evolved solutions in the final generation are checked for domination and the dominated solutions are removed. Usually from previous experiment of this algorithm with crop planning problem and test problems, all the solutions turned out to converge to Pareto fronts (Adeyemo and Otieno, 2009, 2010). The trial solution survives to the next generation if its objective function is better or equal in all the objectives to the target solution. MDEA handles multiple constraints using penalty function proposed by Deb (2001). This algorithm combines the advantages of DEMO (Robic and Filipic, 2005) and the algorithm by Fan et al. (2006). However, MDEA runs faster with more quality Pareto optimal solutions than both algorithms (Adeyemo and Otieno, 2009).

RESULTS

The four strategies of MDEA namely MDEA1, MDEA2, MDEA3 and MDEA4 were tested using five of the six test problems proposed by Zitzler et al. (2000). The description of the test problems are given in Table 2. These are unconstrained test beds. They are labeled ZDT1, ZDT2, ZDT3, ZDT4 and ZDT6. They have two objectives to be minimized. The population size for all the problems was set at 300. The number of parameter, D was 30 in ZDT1, ZDT2 and ZDT3 while D was 10 for ZDT4 and ZDT6 and the maximum generation was 2000 at a step of 500. These steps are named convergence Stages A, B, C and D for 500, 1000, 1500 and 2000 iterations, respectively. The results were recorded at each stage (500 iterations). For ZDT1, CR and F were set at 0.2 and 0.3 respectively. These control parameters were set by trial and error to determine the best for each problem and each strategy. For ZDT1, all the 300 solutions converged to the Pareto optimal front within 500 iterations (Stage A) for MDEA1 and MDEA2. While the

Table 2. Description of test problems used in this study.

Problems	N	Variable bounds	Objective functions	Optimal solutions	Comments
ZDT1	30	[0,1]	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1-\sqrt{\left(\frac{x_1}{g(x)}\right)}\right]$ $g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$	$x \in [0,1]$ $x_i=0$ $i=2,\dots,n$	Convex
ZDT2	30	[0,1]	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1-\left[\frac{x_1}{g(x)}\right]^2\right]$ $g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$	$x \in [0,1]$ $x_i=0$ $i=2,\dots,n$	Nonconvex
ZDT3	30	[0,1]	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1-\left[\sqrt{\frac{x_1}{g(x)}}\right]-\frac{x_1}{g(x)}\sin(10\pi x_1)\right]$ $g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$	$x \in [0,1]$ $x_i=0$ $i=2,\dots,n$	convex, disconnected
ZDT4	10	$x_1 \in [0,1]$ $x_i \in [-5,5] = 2,\dots n$	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1-\sqrt{\left(\frac{x_1}{g(x)}\right)}\right]$ $g(x) = 1+10(n-1)+\sum_{i=2}^{n}\left(x_i^2-10\cos(4\pi x_i)\right)$	$x \in [0,1]$ $x_i=0$ $i=2,\dots,n$	Nonconvex
ZDT6	10	[0,1]	$f_1(x) = 1-e^{-4x_1}\sin^6(6\pi x_1)$ $f_2(x) = g(x)\left[1-\left(\frac{f_1(x)}{g(x)}\right)^2\right]$ $g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$	$x_1 \in [0,1]$ $x_i = 0$ $i = 2,3,\dots,n$	Nonconvex, Nonuniformly spaced solutions on Pareto front, Low density of solutions near Pareto front

convergence was maintained after 500 iterations for MDEA1, the solutions started diverging after 500 iterations for MDEA2 (Stage B). MDEA3 could not converge with CR = 0.2 and F = 0.3, therefore, we tried CR = 0.85 and F = 0.5. After 2000 iterations (Stage D), only 154 non-dominated solutions converged to the Pareto front. Divergence was noticed after 2000 iterations (Stage D). It was found that 71, 129, 160 and 154 non-dominated solutions were noticed at convergence stages A, B, C and D respectively. We had to experiment MDEA3 with different values of CR and F of 0.95 and 0.9 respectively to achieve the convergence of all the 300 solutions at convergence Stage D. We noticed a faster convergence of 278, 293, 298 and 300 at convergence Stages A, B, C and D respectively. For MDEA4, when we used CR = 0.2 and F = 0.3, only 87, 123, 194 and 185 solutions converged at convergence Stages A, B, C and D respectively. We could not achieve 100% convergence of non-dominated solutions (300) until we changed our CR and F to 0.95 and 0.9 respectively. It can be concluded that for test problem ZDT1, higher values of CR and F (0.95 and 0.9, respectively) could made the solutions converge to Pareto fronts for MDEA3 and MDEA4 while lower values of CR and F (0.2 and 0.3 respectively) achieved fast convergence for MDEA1 and MDEA2.

Performance measures

The common performance measures for multi-objective algorithms are employed in this study. Two performance metrics are used for adequately evaluating both goals of multi-objective optimization. One performance metric evaluates the progress towards the Pareto-optimal front and the other evaluates the spread of solutions. The descriptions of convergence metric Υ and diversity metric Δ used to evaluate strategies of MDEA are given below. They are:

Convergence metric Υ

This metric measures the distance between the obtained non-dominated front, Q and the set, P* of the Pareto-optimal solutions. It is defined as:

$$\Upsilon = \frac{\sum_{i=1}^{|Q|} d_i}{|Q|} \tag{1}$$

where Q is the number of non-dominated vectors found by the algorithm being analysed and d_i is the Euclidean distance (in the objective space) between the obtained non dominated front Q and the nearest member in the true Pareto front P.

Diversity metric Δ

This metric measures the extent of spread achieved using the non-dominated solutions. It is defined as:

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{|Q|-1} \left(d_i - \overline{d}\right)}{d_f + d_l + (|Q|-1)\overline{d}} \tag{3}$$

where d_i is the Euclidean distance (measured in the objective space) between consecutive solutions in the obtained non-dominated front Q, and $\overline{d}$ is the average of these distances. The parameter d_f and d_l are the Euclidean distances between the extreme solutions of the Pareto front P* and the boundary solution of the obtained front Q. Also an algorithm having a smaller value of diversity metric Δ is better.

All the 300 solutions converged to Pareto front at convergence Stage A for MDEA1 and MDEA2 for ZDT2 using CR =0.1 and F = 0.3. CR and F were increased to 0.95 and 0.9 respectively as with ZDT1 to achieve convergence of all the 300 solutions at convergence Stage C for MDEA3 and MDEA4. When we increased F to 0.95, all the solutions (300) converged to the Pareto front at convergence Stage B for MDEA4.

For ZDT3, CR = 0.5 and F=0.5 were used for MDEA1 and MDEA2 to achieve a convergence of 297 and 288 solutions respectively at Stage C. Only 280 and 287 solutions converged to the Pareto front at convergence Stages C and D respectively for MDEA3 and MDEA4 (Figure 1). For ZDT4, we experimented with CR = 0.2 and F=0.3 for MDEA1 and MDEA2 and all the 300 solutions converged to Pareto front at convergence Stage A. Only 168 and 164 solutions converged to Pareto front at Stage A using the same parameters for MDEA3 and MDEA4 respectively. The parameters were changed to CR = 0.95 and F = 0.9 for MDEA3 and MDEA4 for all the 300 solutions to converge to Pareto front at convergence Stage A.

We experimented ZDT6 with CR = 0.6 and F = 0.2 for MDEA1 and MDEA2 and all the 300 solutions converged to Pareto front at convergence Stage A. 205 solutions converged at convergence Stage A for MDEA3 and no convergence was noticed for MDEA4 using the same parameters. However, all the 300 solutions converged to Pareto front at Stage A for both MDEA3 and MDEA4 when the parameters were changed to higher values (CR = 0.95 and F = 0.9) Figure 2 and 3.

The four strategies were further tested on an engineering design problem of cantilever design (Deb et al., 2001). The description of this problem is given by Deb et al. (2001). It has two decision variables, two conflicting objectives to be minimized, two constraints and two bound constraints. The population size used for this problem was 100, CR = 0.8 and F = 0.3 were used for MDEA1 and MDEA2 with only 50 solutions converging to Pareto front at convergence Stage A. At higher values of CR = 0.95 and F = 0.9, 50 solutions also converged to Pareto front at Stage A for MDEA3 and MDEA4 (Figure 4).

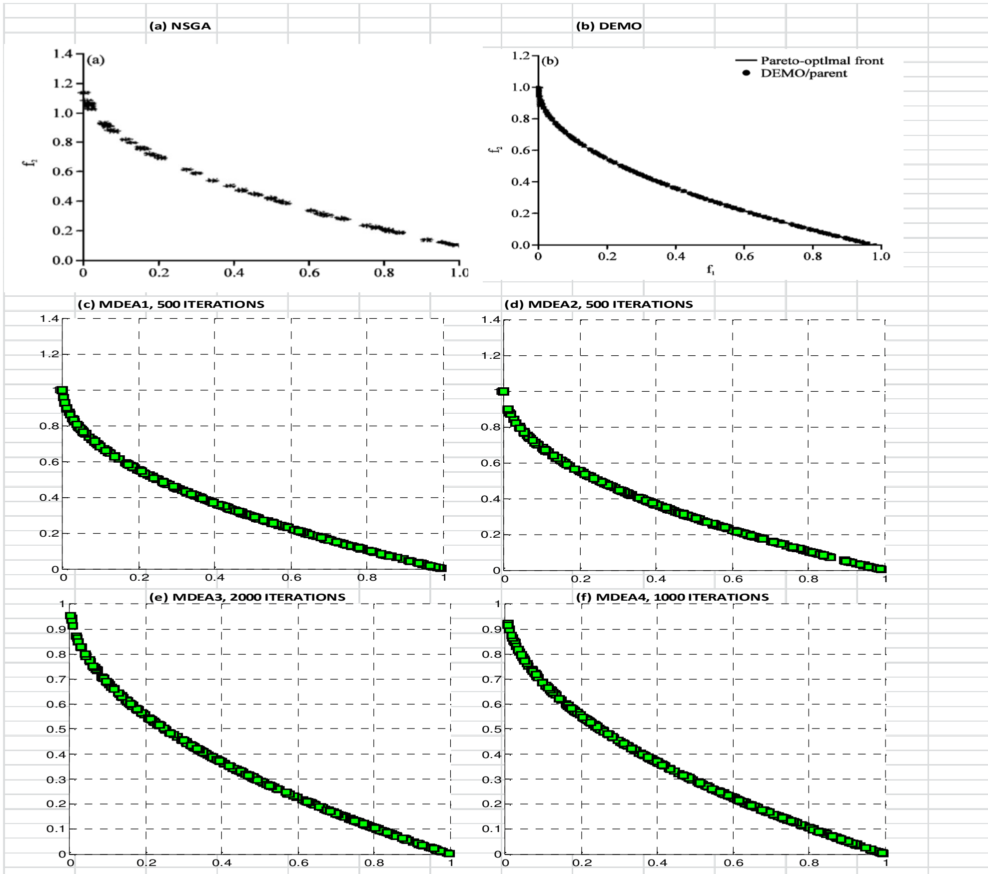

Figure 1. Results of different algorithms on test problem ZDT1.

Performance of the strategies

It was observed that MDEA1 and MDEA2 behave alike and MDEA3 and MDEA4 form another group. For convergence to be achieved, lower values of CR and F were suitable for MDEA1 and MDEA2 while higher values of CR and F (close to 1) favour the convergence of MDEA3 and MDEA4. It can be concluded that MDEA1 and MDEA2 with binomial crossover method perform better with lower values of CR and F and MDEA3 and MDEA4 with exponential crossover method perform better with higher values of CR and F (very close to 1) (Adeyemo and Otieno, 2010; Price and Storn, 2008) (Figure 5).

MDEA1 and MDEA2 converge within convergence Stage A (500 iterations) in all the test problems except ZDT3 which is a difficult optimization problem (Zitzler et al., 2000) and mostly all the solutions converged to the Pareto front except in cantilever problem where only half of them converged. In contrast, convergence to Pareto front only occurred at convergence Stages C and D for MDEA3 and MDEA4 except ZDT4 and ZDT6 and cantilever problem where it occurred at convergence Stage A. Also, not all the solutions converged to the Pareto front in all the problems using MDEA3 and MDEA4. In ZDT3 which is a difficult optimization, only 280 and 287 solutions converged to the Pareto front for MDEA3 and MDEA4 respectively. Half of the solutions in cantilever problem converged to the Pareto front like MDEA1 and MDEA2. Therefore, it can be concluded that MDEA1 and MDEA2 which use binomial crossover method, have been shown to show superior performance than MDEA3 and MDEA4 in the test problems except in the engineering design problem where they performed equally. This shows that all the strategies of MDEA are capable of solving the engineering problem with reasonable time (within 100 iterations). We also experimented with different values of CR and F for all the strategies when solving the engineering cantilever problem. It was found that all the strategies performed very well with any combination of values of CR and F. It further confirmed that all the strategies of MDEA are capable of solving the engineering design problem presented (Adeyemo and Otieno, 2010).

In Figure 1(a-f), the non-dominated solutions for NSGA, DEMO, MDEA1, MDEA2, MDEA3 and MDEA4 are presented. It is found that the non-dominated solutions for MDEA1 are of more quality than those of the other algorithms as can be deduced from convergence and diversity metrics in Tables 3 and 4. Though the generated Pareto optimal fronts are similar in all the algorithms, those of strategies of MDEA outperformed those of NSGA and DEMO (Tables 3 and 4). Also those of MDEA1 and MDEA2 outperform MDEA3 and MDEA4 on the same ZDT1 problem (Tables 3 and 4). We can conclude that MDEA is better in achieving quality Pareto front than the other algorithms.

Figure 2(a-f) shows the Pareto optimal fronts for test problem ZDT2 for all the algorithms. It was found that MDEA3 and MDEA4 produced Pareto fronts of higher quality than MDEA1 and MDEA2 in terms of diversity and convergence (Tables 3 and 4). MDEA2 produced the worst Pareto front in terms of convergence with solutions scattered on the Pareto front (Table 3). Though MDEA1 and MDEA2 converged faster at convergence Stage A while MDEA3 and MDEA4 converged at convergence Stage C, still their Pareto fronts are worse than MDEA3 and MDEA4 in convergence as can be seen in Table 3. MDEA1, MDEA3 and MDEA4 have superior Pareto fronts than the other algorithms (NSGAII and DEMO) but MDEA2 have an inferior Pareto front in test problem ZDT2 in diversity and convergence as in Tables 3 and 4.

Further observation on MDEA3 is presented in Figure 3. Figure 3a shows the non-dominated solutions' convergence at CR = 0.85 and F = 0.5. It was found that at convergence Stage A, only 219 solutions were non-dominated with scattered solutions on the Pareto front. At convergence Stage B (1000 iterations), only 268 solutions were non-dominated (Figure 3b). We therefore decided to increase CR to 0.95 and F to 0.9. It was found that 293 solutions were non-dominated at convergence Stage A (Figure 3c); 300 solutions at convergence Stage C, respectively (Figures 3d). The quality of the Pareto front in Figure 3d also improved. This shows that MDEA3 can only perform excellently with higher values of CR and F (close to 1) for a problem like ZDT2.

In Figure 4, ZDT3 optimization results are presented for all the algorithms. It was found that only MDEA1 and MDEA4 produced high quality non-dominated solutions interms of convergence and diversity in Tables 3 and 4. MDEA3 produced poorest non-dominated solutions in convergence and diversity as in Tables 3 and 4. It took MDEA4 2500 iterations to produce 287 non-dominated solutions and MDEA1 produced 297 non-dominated solutions in 1500 iterations (convergence Stage C). Though MDEA2 had more non-dominated solutions (288) than MDEA4, the non-dominated solutions of MDEA4 are said to be of better qualities because they spread evenly (that is, more diversity) (Table 4).

ZDT4 solutions are presented in Figure 5. It was observed that MDEA1 and MDEA2 have higher qualities of non-dominated solutions with more diversity than MDEA3 and MDEA4 though all the solutions are comparable/better than the ones by NSGAII and DEMO (Tables 3 and 4). The non-dominated solutions by the MDEAs were generated at the convergence Stage A (under 300 iterations).

ZDT6 test problem solutions are presented in Figure 6. It was found that all the strategies of MDEA produced 100% non-dominated solutions at the convergence Stage A (within 500 iterations). MDEA3 and MDEA4 have non-dominated solutions with higher diversity (Table 4). The solutions are well spread on the Pareto fronts unlike MDEA1 and MDEA2. All the non-dominated solutions are comparable/better in convergence and diversity than NSGAII and DEMO (Table 4). It was found that MDEA3 and MDEA4 are better than MDEA1 and MDEA2 in diversity and convergence in the test problems ZDT2 and ZDT6 which have similar concave shape Pareto fronts. Also MDEA1 and MDEA2 are better in convergence and diversity in ZDT1 and ZDT4 which have similar convex shape Pareto fronts (Tables 3 and 4). Therefore, the results agree with the findings of (Adeyemo and Otieno, 2010) that MDEA strategies with binomial crossover method (MDEA1 and MDEA2) can perform better than MDEA strategies with exponential crossover method (MDEA 3 and MDEA4) in some problems. Furthermore, we can now establish that MDEA3 and MDEA4 with exponential crossover method are also better in someproblems than MDEA1 and MDEA2 with binomial crossover method. Also, MDEA1 and MDEA4 are better than MDEA2 and MDEA3 in solving similar problems but MDEA1 has faster convergence than MDEA4 while MDEA4 has more diversity of non-dominated solutions than MDEA1 in similar problems.

We therefore proceeded to solving engineering cantilever problem with the algorithms. The results are presented in Figure 7. We set lower values of CR and F (CR = 0.8 and F = 0.3) for MDEA1 and MDEA2 as in the previous test problems. We set higher values of CR and F (CR = 0.95 and F = 0.9; close to 1) for MDEA3 and MDEA4. All the four strategies performed well in solving the problem and producing non-dominated solutions. The solutions further confirm our previous results that MDEA1 and MDEA2 outperform MDEA3 and MDEA4 in finding solutions on convex shaped Pareto fronts. The solutions of all the strategies of MDEA are comparable to the ones by DEMO and NSGAII. It is further confirmed that all the strategies of MDEA are capable of solving the engineering cantilever design problem presented with ease.

Conclusions

Strategies of multiobjective differential evolution algorithm (MDEA) are presented in this study. They were tested on five different test problems with different shapes of Pareto fronts. They were later tested on an engineering design problem. The strategies are based on the crossover methods and number of vectors used for the perturbation. MDEA1 and MDEA2 use binomial crossover method

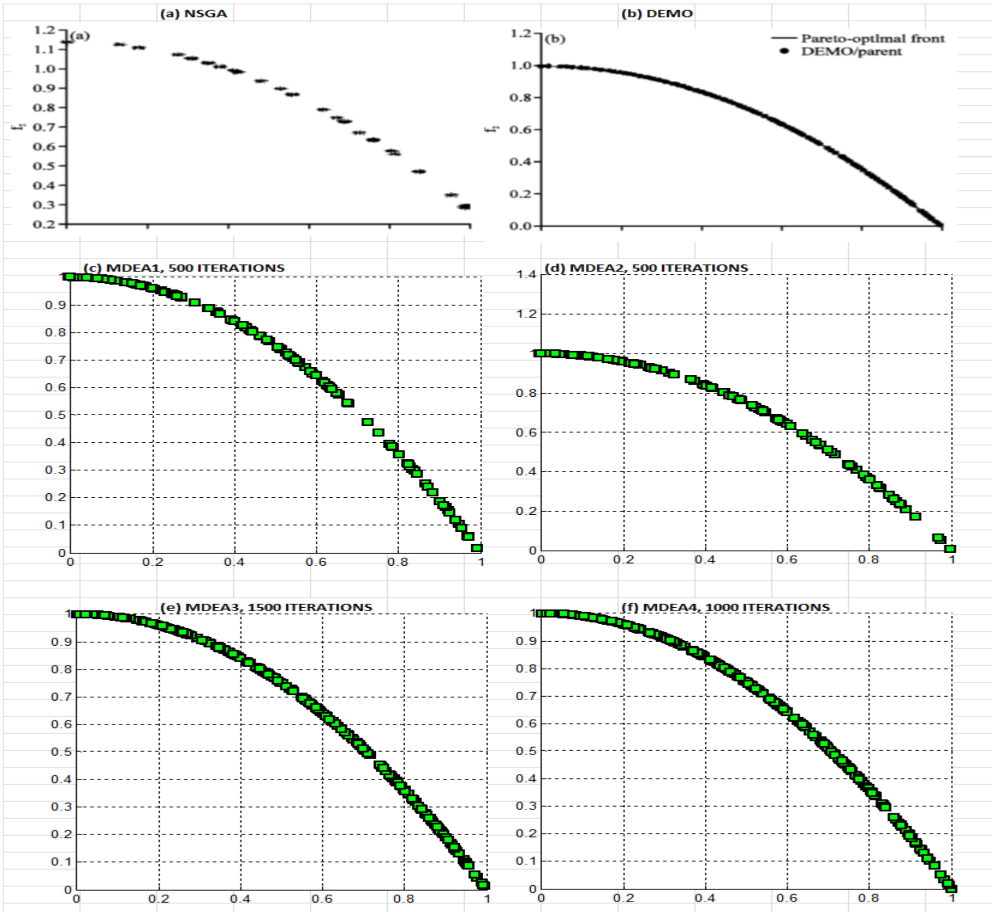

Figure 2. Results of different algorithms on test problem ZDT2.

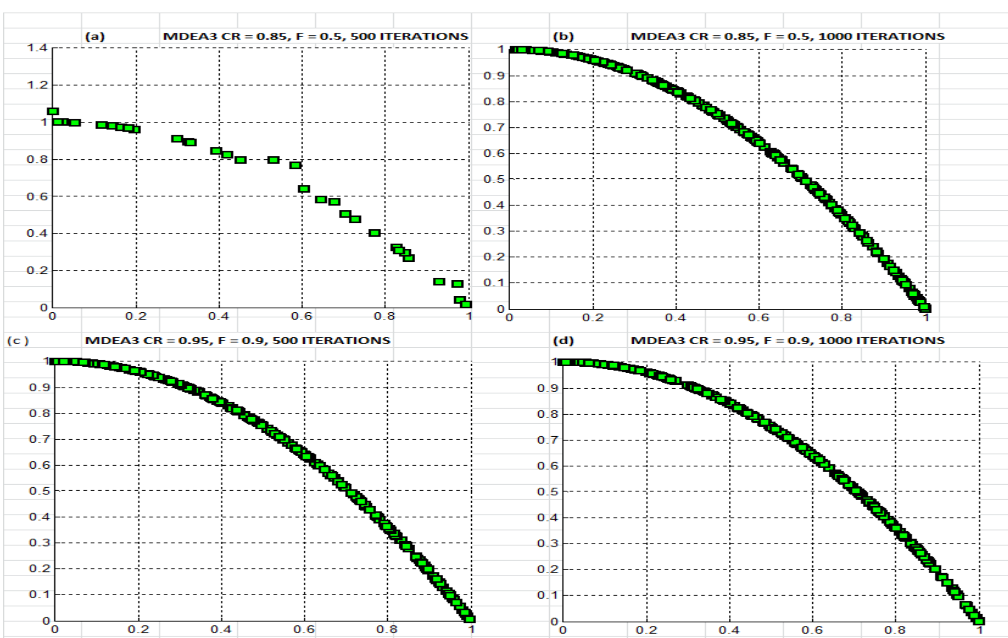

Figure 3. Experiment of the convergence on test problem ZDT2 by MDEA3.

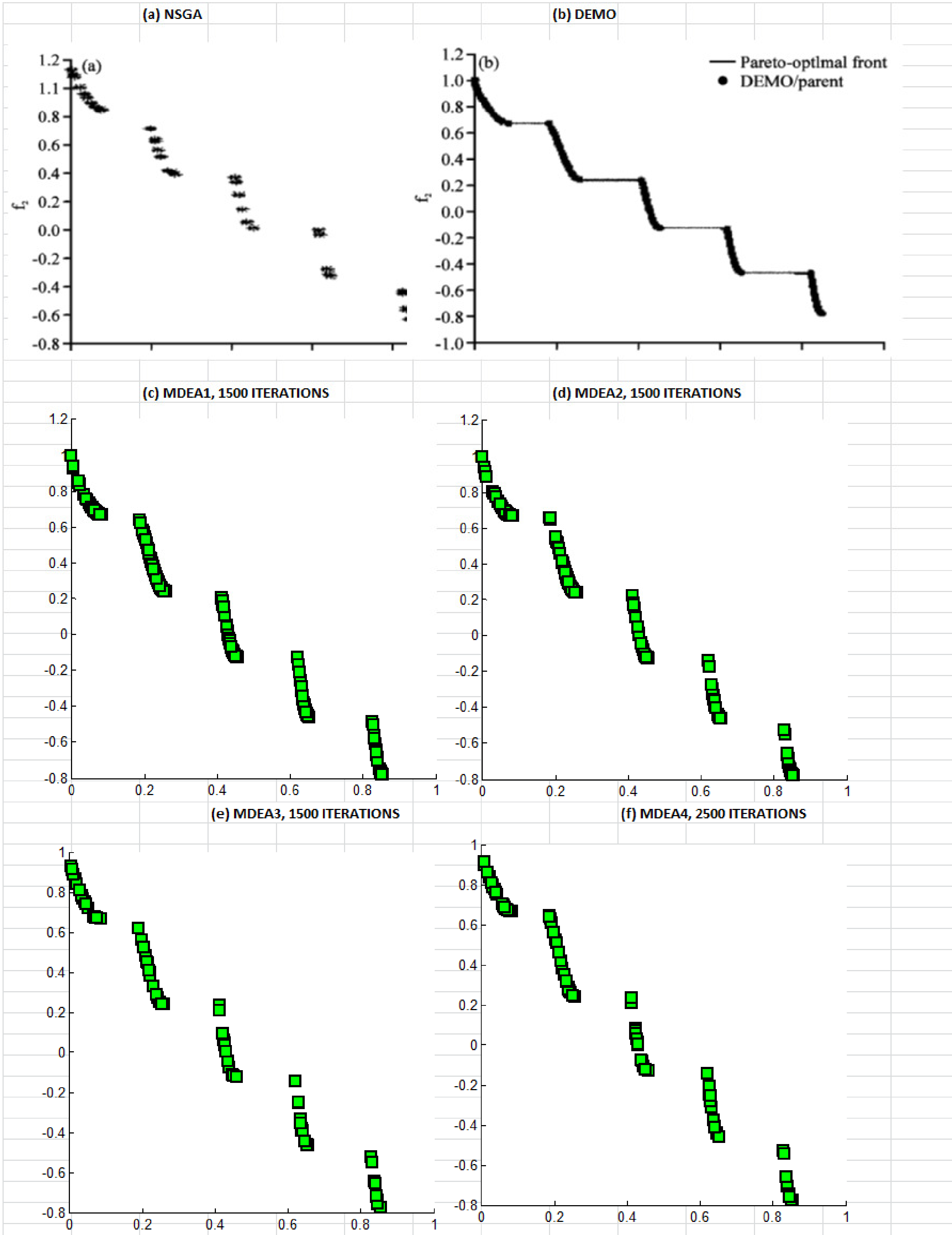

Figure 4. Results of different algorithms on test problem ZDT3.

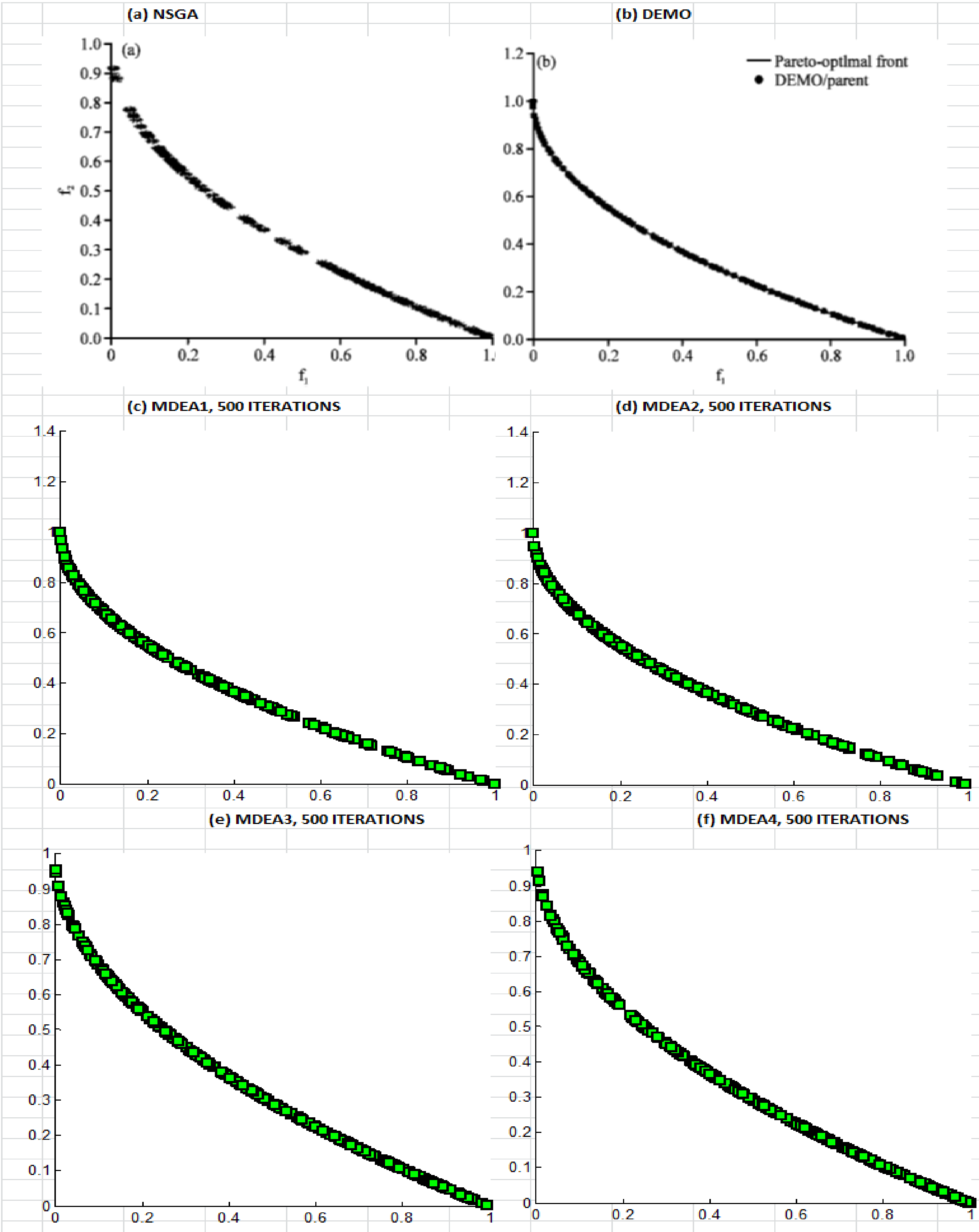

Figure 5. Results of different algorithms on test problem ZDT4.

Table 3. Statistics of the convergence metrics results on all the test problems.

Algorithm	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
NSGA-II	0.033482±0.004750	0.072391±0.031689	0.114500±0.004940	0.513053±0.118460	0.296564±0.013135
DEMO	0.001113±0.000134	0.000820±0.000042	0.001197±0.000091	0.001016±0.000091	0.000630±0.000021
MDEA1	0.000921±0.000005	0.000640±0.000000	0.001139±0.000024	0.048962±0.536358	0.000436±0.000055
MDEA2	0.000911±0.000010	0.080635±0.000841	0.001200±0.000000	0.041892±0.618792	0.000520±0.000111
MDEA3	0.000103±0.000008	0.000600±0.000016	0.019050±0.005011	0.215873±0.230000	0.000561±0.000212
MDEA4	0.000113±0.000015	0.000605±0.000051	0.001099±0.000028	0.221000±0.025893	0.000542±0.000252

Table 4. Statistics of the diversity metrics results on all the test problems.

Algorithms	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
NSGA-II	0.390307±0.001876	0.430776±0.004721	0.738540±0.019706	0.702612±0.064648	0.668025±0.009923
DEMO	0.319230±0.031350	0.335178±0.016985	0.324934±0.029648	0.359600±0.026977	0.461174±0.035289
MDEA1	0.283708±0.002938	0.450482±0.004211	0.299354±0.023309	0.406382±0.062308	0.305245±0.019407
MDEA2	0.280015±0.006014	0.460023±0.002102	0.3110000±0.000030	0.412000±0.000055	0.356000±0.001113
MDEA3	0.301500±0.000552	0.310560±0.005101	0.315100±0.000001	0.612000±0.000321	0.111000±0.000555
MDEA4	0.298012±0.000412	0.320050±.0002300	0.201000±0.003000	0.551000±0.000022	0.161000±0.000021

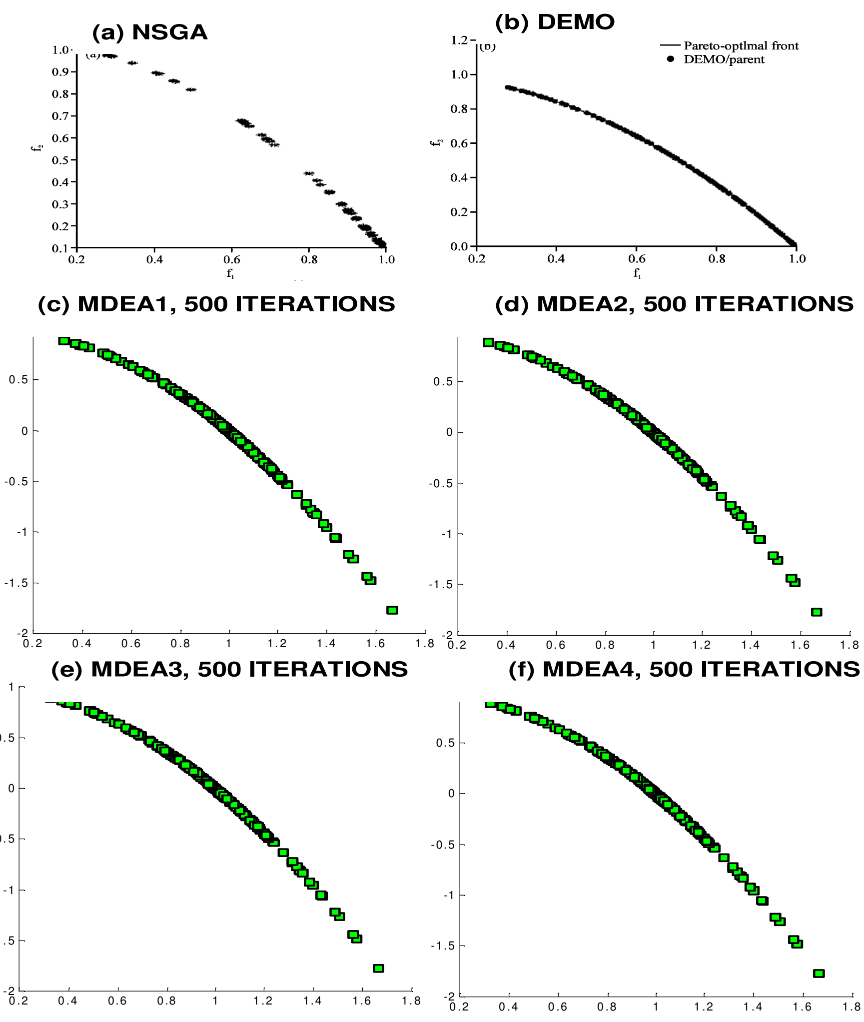

Figure 6. Results of different algorithms on test problem ZDT6.

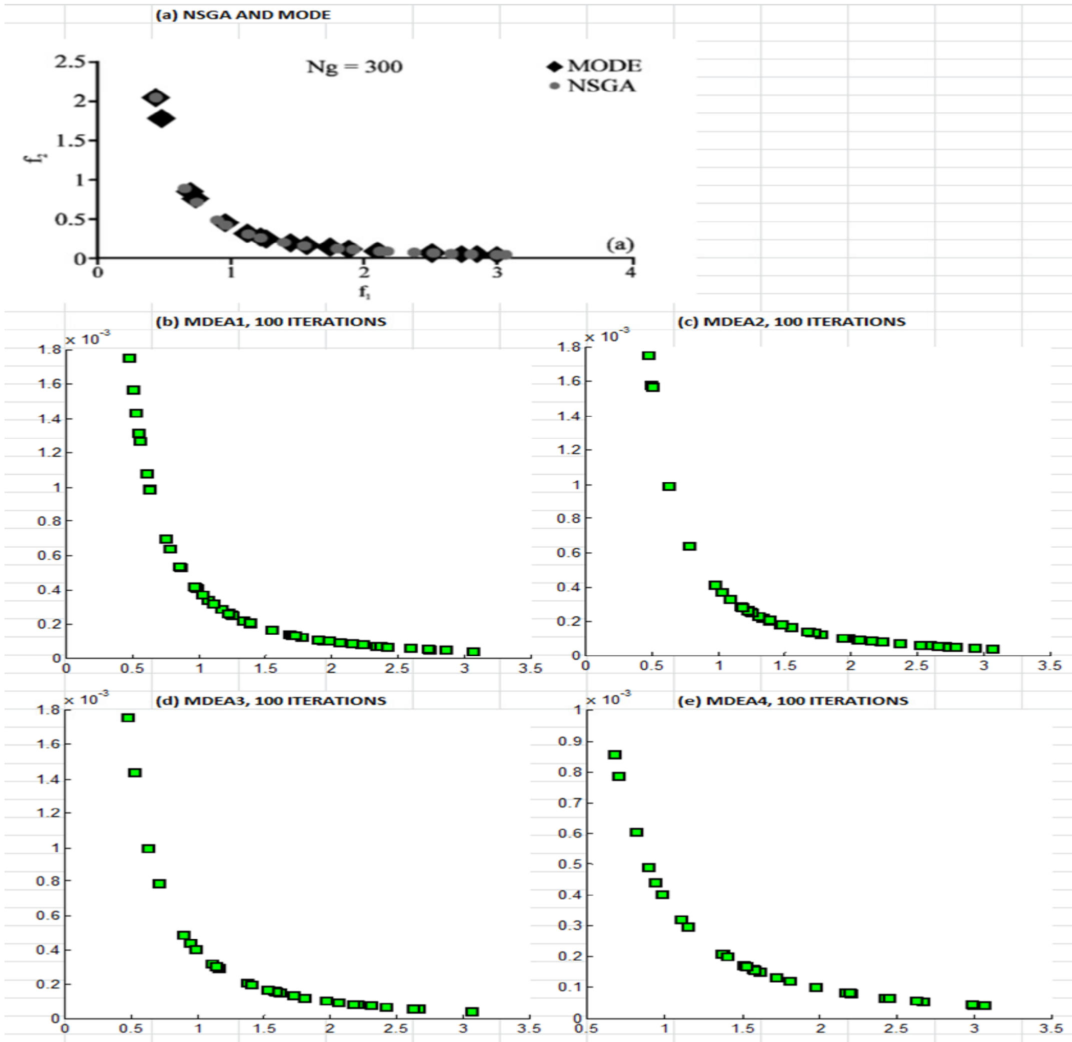

Figure 7. Results of different algorithms on cantilever design problem.

while MDEA3 and MDEA4 use exponential crossover method. MDEA1 is based on DE/rand/1/bin, MDEA2 on DE/rand/2/bin, MDEA3 on DE/rand/1/exp and MDEA4 on DE/rand/2/exp. MDEA2 and MDEA4 use two vectors for their perturbation while MDEA1 and MDEA3 use one vector for their perturbation. All the strategies of MDEA produced non-dominated solutions that converge to the Pareto fronts in the five different test problems. MDEA1 and MDEA2 outperformed MDEA3 and MDEA4 in finding solutions on convex shaped Pareto fronts that is, ZDT1 and ZDT4 while MDEA3 and MDEA4 outperformed MDEA1 and MDEA2 in finding solutions on concave shaped Pareto fronts i.e. ZDT2 and ZDT6. In all, it was found that MDEA1 and MDEA4 are better than MDEA2 and MDEA3 in solving all the test problems. It was however, found that MDEA4 produced non-dominated solutions of better quality and more diversity than MDEA1 though with higher iterations. In the engineering design problem, all the strategies of MDEA performed excellently with MDEA1 and MDEA2 outperforming MDEA3 and MDEA4. It was concluded that all the strategies of MDEA are comparable/better than the other two algorithms compared (NSGAII and DEMO) in solving the test problems and the engineering cantilever design problem presented. It is suggested that modifications should be made to the strategies of MDEA to make them perform better in terms of diversity especially, MDEA2 and MDEA3.

REFERENCES

Abbass HA, Sarker R (2002). The Pareto differential evolution algorithm. Int. J. Artif. Intell. Tools 11(4):531-552.

Adeyemo J, Otieno F (2010). Differential evolution algorithm for solving multi-objective crop planning model. Agric. Water Manag. 97(6):848-856.

Adeyemo JA, Otieno FAO (2009). Multi-objective differential evolution

algorithm (MDEA) for solving engineering problems. J. Appl. Sci. 9(20):3652-3661.
Angira R, Babu BV (2005). Non-dominated sorting differential evolution for multi-objective optimization. 2nd Indian Int. Conf. Artif. Intell. (IICAI-05):1428-1443.
Ascia G, Catania V, Di Nuovo AG, Palesi M, Patti D (2012). Performance evaluation of efficient multi-objective evolutionary algorithms for design space exploration of embedded computer systems. Appl. Soft Comput. 11(1):382-398.
Babu BV, Jehan MM (2003). Differential evolution for multi-objective optimization. Paper presented at the Congress on Evolutionary Computation (CEC-2003), Canberra, Australia. pp. 2696-2703.
Branke J, Scheckenbach B, Stein M, Deb K, Schmeck H (2009). Portfolio optimization with an envelope-based multi-objective evolutionary algorithm. Eur. J. Oper. Res. 199(3):684-693.
Chakraborti N, Sreevathsan R, Jayakanth R, Bhattacharya B (2009). Tailor-made material design: An evolutionary approach using multi-objective genetic algorithms. Comput. Mater. Sci. 45(1):1-7.
Deb K (1999). Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems. Evol. Comput. 7(3):205-230.
Deb K (2001). Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons, Chichester, UK. pp. 22-25.
Deb K, Agrawal S, Pratap A, Meyarivan T (2001). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. Kanpur Genetic Algorithms Laboratory (KanGAL) Indian Inst. Technol. Kanpur. pp. 182-197.
Fan H-Y, Lampinen J, Levy Y (2006). An easy-to-implement differential evolution approach for multi-objective optimizations. Engineering Computations: Int. J. Comput-Aided Eng. Softw. 23(2):124-138.
Konstantinidis A, Yang K, Zhang Q, Zeinalipour-Yazti D (2012). A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks. Comput. Netw. 54(6):960-976.
Lee LH, Chew EP, Teng S, Chen Y (2008). Multi-objective simulation-based evolutionary algorithm for an aircraft spare parts allocation problem. Eur. J. Oper. Res. 189(2):476-491.
Madavan NK (2002). Multi-objective optimization using a Pareto differential evolution approach. Paper presented at the Congress on Evolutionary Computation (CEC '2002). pp. 1145-1150.
Marcelloni F, Vecchio M (2012). Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization. Inf. Sci. 180(10):1924-1941.
Nariman-Zadeh N, Felezi M, Jamali A, Ganji M (2009). Pareto optimal synthesis of four-bar mechanisms for path generation. Mech. Mach. Theory 44(1):180-191.
Price K, Storn R (2008). Differential evolution homepage (web site of Price and Storn) as at 2008. http:\\www.ICSI.Berkeley.edu/~storn/code.html, vol 2008.
Robic T, Filipic B (2005). DEMO: Differential evolution for multiobjective optimization. Paper presented at the 3rd Int. Conf. on Evolutionary Multi-Criterion Optimization, EMO 2005. pp. 520-533.
Sarker R, Ray T (2009). An improved evolutionary algorithm for solving multi-objective crop planning models. Comput. Electron. Agric. 68(2):191-199.
Storn R, Price K (1997). Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. 11:341-359.
Wang J-Y, Chang T-P, Chen J-S (2009). An enhanced genetic algorithm for bi-objective pump scheduling in water supply. Expert Syst. Appl. 36(7):10249-10258.
Xue F, Sanderson AC, Graves RJ (2003). Pareto-based multi-objective evolution. Paper presented at the Congress on Evolutionary Computation, Canberra, Australia. 2:862-869.
Zitzler E, Deb K, Thiele L (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. Evol. Comput. 8(2):173-195.
Zitzler E, Thiele L (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Trans. Evol. Comput. 3(1999):257-271.