Multi-Objective Test Problems, Linkages, and Evolutionary Methodologies

Kalyanmoy Deb, Ankur Sinha and Saku Kukkonen
Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
Kanpur, PIN 208016, India
{deb,ankursi,saku}@iitk.ac.in

KanGAL Report Number 2006001

Abstract

Existing test problems for multi-objective optimization are criticized for not having adequate linkages among variables. In most problems, the Pareto-optimal solutions correspond to a fixed value of certain variables and diversity of solutions comes mainly from a random variation of certain other variables. In this paper, we introduce explicit linkages among variables so as to develop difficult two and multi-objective test problems along the lines of ZDT and DTLZ problems. On a number of such test problems, this paper compares the performance of a number of EMO methodologies having (i) variable-wise versus vector-wise recombination operators and (ii) spatial versus unidirectional recombination operators. Interesting and useful conclusions on the use of above operators are made from the study.

1 introduction

There exists a number of test problems for multi-objective optimization in the evolutionary multi-objective evolutionary optimization (EMO) literature [1, 16, 6, 13, 8]. The reason for developing controllable yet challenging test problems for optimization and using them to test an optimization methodology is to investigate the problem difficulties, for which a method performs well and problem features, for which they do not perform so well. Identifying such problem features will enable developers and researchers to get a better insight to the working of different methodologies, a process which may help them develop better and more efficient algorithms. It has been criticized that many of the existing test problems for multi-objective optimization are either separable variable-wise, or possess linear functions of the variables. It is then argued that such test problems may not provide adequate difficulties to an EMO methodology and therefore the whole purpose of applying an EMO to these test problems for getting better insights about their working principles is lost. Thus, there is a need for developing more difficult yet controllable test problems for creating a more efficient EMO algorithm. Developed test problems can also then be used to compare different leading EMO methodologies to show the extent of difficulties provided by these test problems.

In this paper, we make a brief review of the existing multi-objective test problems and discuss the adequacy of these problems as *real* test problems. Thereafter, we propose a number of two and multi-objective test problems which allow an user to systematically introduce difficulties through variable linkages. Finally, through simulation results, we demonstrate that the EMO methodologies which use recombination operators capable of handling variable interactions are better able to solve these problems than those who do not use such an operator. For this purpose, we have used two EMO procedures using variable-wise recombination operators and two EMO procedures having vector-wise recombination and generation operators. Moreover, for both vector-wise EMO (PCX-based NSGA-II procedure and DE-based EMO procedure, GDE3), a parametric study is performed to find good parameter settings. The results of this paper are important for various reasons and should encourage readers to appreciate the need of linkage based EMO methodologies and simultaneously motivate them to use the proposed test problems and encourage them to develop and use more such test problems as benchmark problems before trying EMO methodologies to real-world problems.

2 Existing EMO Test Problems

David Van Veldhuizen, in his doctoral thesis [14], collated a number of multi-objective test problems (both constrained and unconstrained). These problems were explicit mathematical functions of a number of variables (mostly two or three) and involve a fixed number of objectives (mostly two and three). It was not clear what kind of difficulties these test problems would provide to an EMO methodology. Since the exact Pareto-optimal front were known to most of them (and for some problems they were clearly worked out from mathematical optimality conditions elsewhere [2]), these test problems could simply be used to investigate if an algorithm is able to find well-represented set of Pareto-optimal solutions or not. If they did, the applied EMO procedure might be considered to have overcome whatever difficulties these problems were providing. If they did not, it gets difficult to analyze why the applied methodology could not solve the problem and what can be done to improve the algorithm for solving the problem.

Thinking along these lines, Deb [1] introduced a procedure of designing two-objective unconstrained test problems with three explicit functionals f_1, g and h, which introduced a known and controllable difficulty to any EMO procedure. Although the number of objectives were fixed to two, the problems allowed users to set any number of variables. The variable set is partitioned into two non-overlapping sets $\mathbf{x} = (\mathbf{x}_I, \mathbf{x}_{II})$:

$$\begin{array}{ll} \text{Minimize} & f_1(\mathbf{x}) = f_1(\mathbf{x}_I), \\ \text{Minimize} & f_2(\mathbf{x}) = g(\mathbf{x}_{II}) \cdot h\left(f_1(\mathbf{x}_I), g(\mathbf{x}_{II})\right), \end{array} \tag{1}$$

By choosing appropriate functions for f_1, g and h, multi-objective problems having specific features were created as follows:

1. The shape and nature (convexity or discontinuity) of the Pareto-optimal front can be affected by choosing an appropriate h function.

2. Convergence to the true Pareto-optimal front can be affected by using a difficult g function.

3. Uniformity in the distribution of solutions on the Pareto-optimal front can be affected by choosing an appropriate f_1 function.

The functional h is chosen in such a manner that for a fixed value of f_1 (say for $\mathbf{x}_I^d$), the minimum of h corresponds to the minimum of g (say at $g^* = g(\mathbf{x}_{II}^*)$). Since f_1 and g do not involve any common variable, the above construction forces the following relationship to exist between the objectives for Pareto-optimal solutions:

$$f_2 = g^* \cdot h(f_1, g^*). \tag{2}$$

Thus by choosing an appropriate h functional, different shapes of the Pareto-optimal frontier can be developed. Since for any non-Pareto-optimal points $\mathbf{x}^d = (\mathbf{x}_I^d, \mathbf{x}_{II}^d)$, $g(\mathbf{x}_{II}^d) > g^*$, for a fixed f_1 value at $f_1(\mathbf{x}_i^d)$, the solution $\mathbf{x}^d$ gets dominated by the corresponding Pareto-optimal solution $(\mathbf{x}_I^d, \mathbf{x}_{II}^*)$. By choosing a multi-modal function for g, multiple optimal fronts can be introduced in a problem so that an EMO algorithm may have difficulties in converging to the true Pareto-optimal front. Similarly, by choosing a non-linear function for f_1, differential densities of solutions along the Pareto-optimal front can be introduced, thereby making an EMO to have difficulty in finding a uniformly distributed set of points.

Based on the above concept, a test-suite of six test problems were suggested elsewhere [16] by using a linear, single-variable function for f_1. In five of the six problems, $f_1(x_1) = x_1$ were used. These so-called ZDT (Zitzler-Deb-Thiele) test problems have been extensively used in many EMO studies in the recent past. Due to the simplicity in their construction, simultaneously these functions were also criticized for being too simple [13, 8].

The popularity of the six specific test problems have made researchers forget a couple of important matters related to the philosophy of constructing original test problems:

1. The functional f_1 can be chosen as a function of more than one variable and non-linear functionals can also be chosen.

2. The original study [1] also suggested a variable mapping strategy $\mathbf{y} = M \cdot \mathbf{x}$, in which an EMO solution vector ($\mathbf{x}$) is first mapped to another variable vector $\mathbf{y}$ by using a constant matrix M. The objective functions are then computed using equation 1 and replacing $\mathbf{x}$ with $\mathbf{y}$.

The mapping concept allows every Pareto-optimal solution to have a fixed $\mathbf{y}_{II}^d$ (which minimizes $g(\mathbf{y}_{II})$ function) and solutions differ with different values of $\mathbf{y}_I$. Representing the linear mapping as follows:

$$\begin{bmatrix} \mathbf{y}_I \\ \mathbf{y}_{II} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_I \\ \mathbf{x}_{II} \end{bmatrix}, \tag{3}$$

we obtain the following relationships between Pareto-optimal solution vectors, $\mathbf{x}$ and $\mathbf{y}$:

$$\mathbf{x}_I = (A - BD^{-1}C)^{-1} \left(\mathbf{y}_I - BD^{-1}\mathbf{y}_{II}^d \right), \tag{4}$$

$$\mathbf{x}_{II} = D^{-1}(\mathbf{y}_{II}^d - C\mathbf{x}_I). \tag{5}$$

Since different $\mathbf{y}_I$ vectors produce different Pareto-optimal solutions, it is likely that all Pareto-optimal solutions $\mathbf{x}$ will be different from each other. Thus, in every variable x_i, an EMO is expected to maintain a wide range of solutions in order to maintain a well-represented set of Pareto-optimal solutions. An EMO with a proper niching strategy must now be used to maintain a wide variety of solutions in the population. This is certainly a more difficult task compared to the case for which $M = I$, an identity matrix (with which ZDT problems were designed). In the ZDT problems, all Pareto-optimal solutions have a fixed value of $\mathbf{x}_{II}$, thereby making it easier to maintain a diverse set of solutions by simply maintaining a diversity in $\mathbf{x}_I$ variables. In ZDT problems, only one variable was used for $\mathbf{x}_I$, thereby making the task of generating a diverse set of Pareto-optimal solutions even easier.

Another study [3] constructed a test problem using the linear mapping approach discussed above and the NSGA-II procedure with a variable-wise SBX recombination operator was found to be difficult in finding a well-converged and well-distributed set of solutions.

Another study [11] suggested two-objective test problems by explicitly using three operations – deformation, rotation and shift – of the decision parameter space and suggested two test problems. Two EMO procedures were shown to have difficulties in finding the Pareto-optimal solutions in both problems. Although the study highlights the importance of linkages among variables, issues of difficulties in designing explicit mathematical operations for the above three tasks explicitly for a large number of objectives and variables were not addressed and remain as important task for extending the idea. Further follow-up studies are needed to establish the utility and ease of application of the procedure as a task of test problem development.

Later, Deb et al. [6] suggested and designed a set of scalable multi-objective test problems (so called DTLZ problems) which are scalable to any number of objectives ($M \geq 2$) and having any number of variables ($n \geq M$). The motivation behind these test problems is to first construct a Pareto-optimal surface in the objective space (either parametrically using decision variables $(f_1(\mathbf{x}_I), \ldots, f_M(\mathbf{x}_I)))$ or directly $f_M = f_M(f_1, \ldots, f_{M-1})$ and by choosing any functional relationships $f_j = f_j(\mathbf{x}_I)$). Thereafter, in this bottom-up approach, each objective function is multiplied by a term $g(\mathbf{x}_{II})$. By choosing a function g such that the minimum value of g is one, a multi-objective test problem is constructed with each Pareto-optimal solution to correspond to the minimum value of g. Based on different desired difficulties in DTLZ problems, nine such test problems were suggested [6]. These problems were attempted to be solved up to 30 objectives in a recent study [5]. However, in most of these problems, the variables are partitioned into two non-overlapping groups ($\mathbf{x}_I$ and $\mathbf{x}_{II}$) and moreover the variable in each group are also independent to each other. Although in DTLZ5, DTLZ8 and DTLZ9 problems, some linkages among some of the variables occur through explicit pair-wise dependencies or through constraint satisfaction, mostly the problems are inadequate to test an algorithm's ability to handle linkages among variables.

Recently, a couple of multi-objective test suites were also suggested [8, 13]. However, their construction procedure is similar to the above approaches and do not provide an adequate test for handling linkages among variables.

Thus, there is a need for a set of test problems having controllable linkages among variables so that EMO methodologies can be adequately tested. In the following sections, we modify the ZDT and DTLZ problems in a systematic manner for this purpose. Based on different levels of linkages, three different types of problems are developed here.

3 Modified ZDT Problems with Linkages of Type-1 (L_1-ZDT)

First, we introduce linkages among $\mathbf{x}_I$ and $\mathbf{x}_{II}$ variables individually, that is, there is no linkages set between two variables s and t, where $s \in \mathbf{x}_I$ and $t \in \mathbf{x}_{II}$, but linkages exist among variables within each group. Thus, we set B and C matrices to be zero matrices. Let us denote that there are k variables in $\mathbf{x}_I$ and $(n-k)$ variables in $\mathbf{x}_{II}$. Then A and D matrices are of size $k \times k$ and $(n-k) \times (n-k)$, respectively, and each element can take a value within $[-1, 1]$. The above matrix causes the function f_1 and the function g to be sheared, scaled and rotated independent of each other. Finally, in effect, the functions f_1 and g will be influenced in such a way that their variable separability will be lost, if they are separable. The sub-matrix $A_{k\times k}$ is responsible for bringing changes in the f_1 profile, while sub-matrix $D_{(n-k)\times(n-k)}$ is responsible for bringing changes in the g function. With the above mapping, the function f_1 is still a function of variable vector $\mathbf{x}_I$ and the function g is still a function of variable vector $\mathbf{x}_{II}$.

EMO procedures having variable-wise recombination operator (such as the SBX operator [2]) would not be able to perform well on such test problems, as the transformation makes the variables ($\mathbf{x}$) of the problem to be linked to each other group-wise. But still the entire front is decided by the minimum value of g. Using the approach, all ZDT problems can be modified and a test problem providing difficulties as they did in the original ZDT problems and an additional difficulty of variable linkage for g function can be created. To illustrate the procedure, we consider only one ZDT problem.

3.1 L_1-ZDT4

This is a $n = 10$ variable problem which is a modification to the original ZDT4 problem having a convex Pareto-optimal set:

$$\begin{aligned}
&f_1(y) = y_1^2,\\
&g(y) = 1 + 10(n-1) + \textstyle\sum_{i=1}^{n}(y_i^2 - 10\cos(4\pi y_i)),\\
&h(f_1, g) = 1 - \sqrt{f_1/g},\\
&x_1 \in [0,1], \quad x_i \in [-5,5],\ i \neq 1,\\
&\mathbf{y} = \begin{bmatrix} A & 0 \\ 0 & D \end{bmatrix}\mathbf{x},\\
&A = [a_{ij}], \quad a_{ij} \in R(-1,1),\\
&D = [d_{ij}], \quad d_{ij} \in R(-1,1),
\end{aligned} \tag{6}$$

where $R(-1,1)$ denotes a uniformly distributed random number in the range $[-1,1]$. Here, $k = 1$, $a_{11} = 1$, and the matrix D is a 9×9 matrix.

4 Modified ZDT Problems with Linkages of Type-2 (L_2-ZDT)

Next, we consider a transformation matrix which involves linkages among two variables, thereby making all four sub-matrices to be non-zero matrices. The matrix B is of size $k \times (n-k)$ and the matrix C is of size $(n-k) \times k$. In this type, each and every function is a function of all variables $(x_1, x_2, \ldots, , x_n)$. In this case it becomes difficult to decide the entire Pareto-optimal front, as the function g no longer remains constant on the Pareto-optimal front. Since the Pareto-optimal frontier may be placed anywhere on the real space by this mapping, we normalize the objective functions so that each normalized objective value for all Pareto-optimal solutions lies in [0,1]. The original functions are modified as follows:

$$\begin{aligned}
&\text{Min.} \quad F_1(\mathbf{y}_I) = \frac{f_1(\mathbf{y}_I) - f_1^{\min}}{f_1^{\max} - f_1^{\min}},\\
&\text{Min.} \quad F_2(\mathbf{y}) = \frac{\left[g(\mathbf{y}_{II}) - g(\mathbf{y}_{II}^{\text{max-I}})\right] h\left(f_1(\mathbf{y}_I), g(\mathbf{y}_{II})\right)}{\left[g(\mathbf{y}_{II}^{\text{min-I}}) - g(\mathbf{y}_{II}^{\text{max-I}})\right] h\left(f_1(\mathbf{y}_I^{\text{min-I}}), g(\mathbf{y}_{II}^{\text{min-I}})\right)}.
\end{aligned} \tag{7}$$

Above terminologies will be clear from the following description. The derived variable vectors can be written after transformation (equation 3), as follows:

$$\mathbf{y}_I = A\mathbf{x}_I + B\mathbf{x}_{II}, \quad \mathbf{y}_{II} = C\mathbf{x}_I + D\mathbf{x}_{II}.$$

Since f_1 is a function of $\mathbf{y}_I$ alone, its minimum and maximum values can be independently found and the function can be normalized to form the function F_1. Let us say that the minimum and maximum value of f_1 occurs at $\mathbf{y}_I^{\text{min-I}}$ and $\mathbf{y}_I^{\text{max-I}}$, respectively. Let us also say that these two values correspond to the following two solutions vectors:

$$\begin{aligned}
&\mathbf{y}_I^{\text{min-I}} : \mathbf{x}^{\text{min-I}} = (\mathbf{x}_I^{\text{min-I}}, \mathbf{x}_{II}^{\text{min-I}}),\\
&\mathbf{y}_I^{\text{max-I}} : \mathbf{x}^{\text{max-I}} = (\mathbf{x}_I^{\text{max-I}}, \mathbf{x}_{II}^{\text{max-I}}).
\end{aligned}$$

Now it is clear that at $\mathbf{x}^{\text{min-I}}$, the first normalized objective value is zero ($F_1 = 0$) and both denominator and numerator of the second normalized objective are equal, thereby having a value of one ($F_2 = 1$). On the other hand, at $\mathbf{x}^{\text{max-I}}$, the first normalized objective value is F_1 is one and corresponding F_2 value is zero. Since these two extreme solutions are likely to be singleton solutions (that is, there exist only one solution $\mathbf{x}^{\text{min-I}}$ and $\mathbf{x}^{\text{max-I}}$ each for minimum and maximum of f_1, these two solutions are never dominated (being extreme solutions) by any other solution in the search space. Thus, they form two extreme Pareto-optimal solutions for the modified normalized test problem. Based on the specific choice of f_1, g and h functions and the chosen transformation metric M, the other Pareto-optimal solutions get determined, but the exact location is difficult to determine.

It is interesting to note that a fixed value of g no longer defines the entire Pareto-optimal front. But still the function in the variable domain defining the Pareto-optimal front is linear and might be helpful to the linear operators.

4.1 Modified L$_2$-ZDT Problems

This problem is based on the $n = 30$ variable ZDT1 problem having a convex Pareto-optimal set:

$$\begin{array}{l} f_1(y) = y_1^2, \\ g(y) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} y_i^2, \\ h(f_1, g) = 1 - \sqrt{f_1/g}, \\ x_i \in [0,1], \quad \mathbf{y} = M\mathbf{x},\ M_{n\times n} = [m_{ij}], \quad m_{ij} \in R(-1,1). \end{array} \tag{8}$$

Similarly, we define L$_2$-ZDT2 to L$_2$-ZDT4 and L$_2$ZDT6 problems with following modified g functions:

$$\begin{array}{ll} \text{L}_2\text{-ZDT2}: & g(\mathbf{y}_{II}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} y_i^2, \\ \text{L}_2\text{-ZDT3}: & g(\mathbf{y}_{II}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} y_i^2, \\ \text{L}_2\text{-ZDT4}: & g(\mathbf{y}_{II}) = 1 + \frac{9}{n-1}\sum_{i=2}^{n}(y_i^2 - 10\cos(4\pi y_i)), \\ \text{L}_2\text{-ZDT6}: & g(\mathbf{y}_{II}) = 1 + 9[\sum_{i=2}^{n} y_i^2/9]^{0.25}. \end{array}$$

All other functions are the same as that in the original ZDT problems. Figure 1 shows the objective space of the above L$_2$-ZDT4 problem for $n = 2$ variable with 10^6 random solutions. Notice how the two extreme points are singleton points on the Pareto-optimal front, causing an algorithm difficulty in finding the end portions of the front.

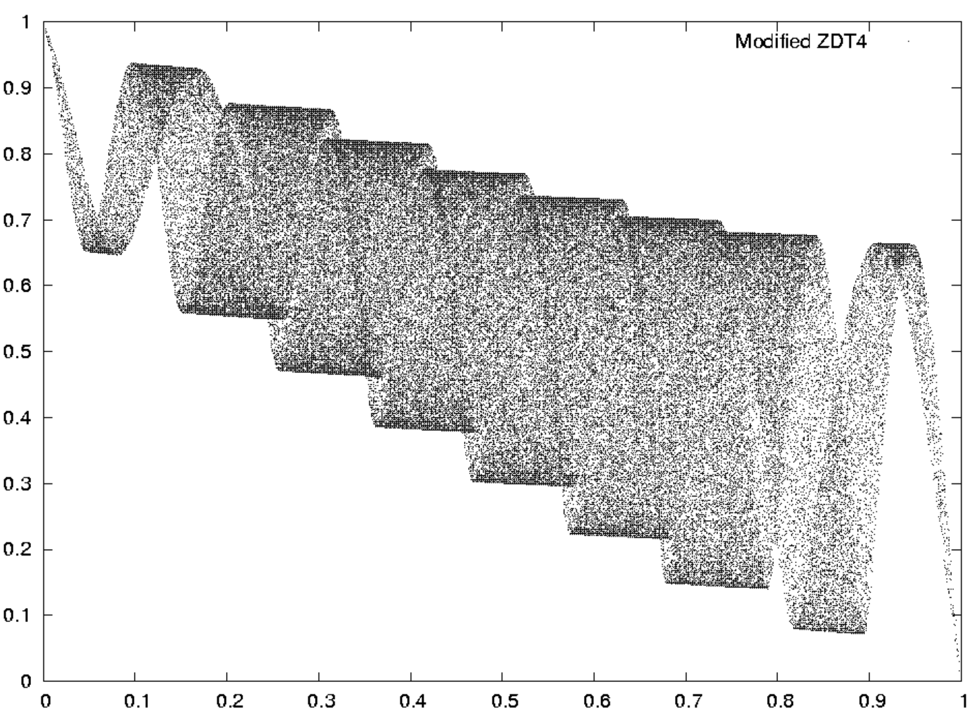

Figure 1: Objective space for a two-dimensional modified L$_2$-ZDT4 problem.

5 Modified ZDT Problems with Linkages of Type-3 (L_3-ZDT)

In both the above types of modification for linkages, there is a potential degeneracy which happens particularly for problems having a linear hyper-plane. In such cases, if a linear recombination operator, such as creating an offspring along a line joining two parents, is used, in both type of modifications the offspring created from two Pareto-optimal parent solutions will also lie on the Pareto-optimal front. This can be seen from equations 4 and 5. Since the variable transformation is a linear one, for linear hyper-plane problems, such a transformation does not provide adequate difficulty to an algorithm. To make the problems more difficult to be solved, we may use a non-linear mapping between $\mathbf{y}$ and $\mathbf{x}$. Here, we use the modified and normalized L_2-ZDT problems, but use the following mapping:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = M \cdot \begin{bmatrix} x_1^2 \\ \vdots \\ x_n^2 \end{bmatrix}. \tag{9}$$

This will not allow linear recombination operators to take any advantage of creating a Pareto-optimal solution from two Pareto-optimal solutions. Here also, we consider all five ZDT problems and create five L_3-ZDT problems. The three fundamental functionals are the same as in L_2-ZDT problems. Figure 2 shows the objective space of L_3-ZDT4 for $n = 2$ variable with 10^6 random solutions. Notice again that two extreme points are singleton points on the Pareto-optimal front and the Pareto-optimal front is a collection of a number of disconnected fronts.

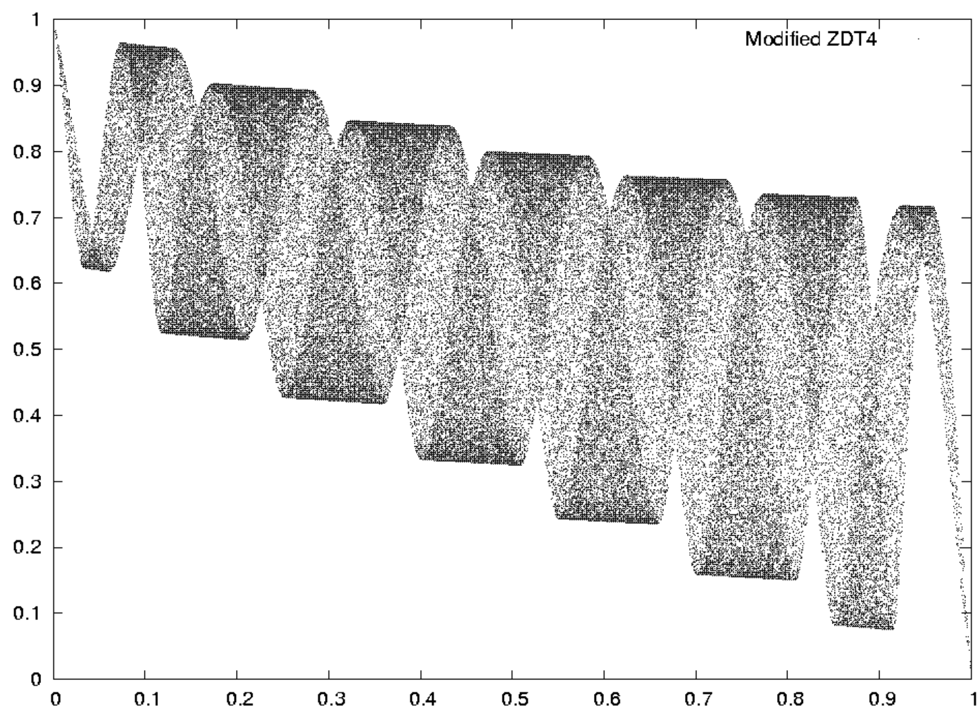

Figure 2: Search space for a two-dimensional modified L_3-ZDT4 problem.

6 Three EMO Procedures

To investigate its effect on different EMO procedures, we use three different procedures:

1. SBX-NSGA-II: The NSGA-II procedure is used with the variable-wise SBX recombination operator [2].

2. L-SBX-NSGA-II: The NSGA-II procedure is used with the SBX recombination operator applied along a line joining the two parents.

3. PCX-NSGA-II: The NSGA-II procedure is used with the vector-wise PCX recombination operator [4].

4. GDE3: The generalized differential evolution procedure is used [9].

SBX-NSGA-II is the original NSGA-II procedure. In L-SBX-NSGA-II, we replace variable-wise SBX operation by a unidirectional SBX along the line joining the two parent solutions. The PCX-NSGA-II and GDE3 procedures are more involved and we make a brief description of them next.

6.1 PCX-NSGA-II Procedure

Here, for the first time, we introduce the PCX recombination operator [4] in NSGA-II. The PCX operator involves two parameters σ_ζ and σ_η controlling the variances along the principal direction (centroid towards the index parent) and in each of the rest $(n-1)$-directions, respectively. Here, we assume both these parameters to be identical and represent them with σ. We we use four different σ values in this study. In the PCX-NSGA-II procedure, we choose three solutions using the usual binary tournament selection operator from the parent population. Each of them is, in turn, used as the index parent and an offspring solution is created by applying the PCX operator to three chosen solutions. Each of three offspring solutions are then operated by a polynomial mutation operator [2]. This operation is continued till a complete offspring population (of the same size as the parent population) is created. Thereafter, a non-dominated sorting of the combined population and a subsequent crowding distance operations are applied to create a population for the next generation, similar to that in the original NSGA-II procedure [3]. Thus, the only difference between SBX-NSGA-II (original NSGA-II) and PCX-NSGA-II procedures lies in the choice of the recombination operator.

6.2 New Generalized Differential Evolution (GDE3) Procedure

This procedure is again similar to the NSGA-II procedure, but the usual SBX recombination operator is replaced with a differential evolution operator. Generalized differential evolution 3 (GDE3) [9] is an extension of differential evolution (DE) for constrained multi-objective optimization. Evolutionary part of the algorithm is DE [12] and multi-objective part is from NSGA-II. In DE, an offspring of a parent is created using recombination of the parent with a mutated vector, which is created by a linear combination of three randomly selected members from the population in a way which makes DE self-adaptive. Control parameters CR and F are used to control recombination and mutation, respectively. After a generation, the combined parent and child population is then reduced back to original population size using sorting based on non-dominance and crowdedness same way as is in the original NSGA-II.

7 Simulation Results

In all simulations presented here, we perform 11 runs from different initial populations. For each simulation on two-objective problems, we use a fixed population of size 100 and for three objectives problems we use a population of size 200.

7.1 Type-1 Problems

To investigate the difficulties posed by Type-1 problems to four EMO methodologies, we use L_1-ZDT4 problem, described in equation 6. In this problem, the convergence of an algorithm can be determined by the g value obtained by a solution. Since different solutions in the final front may have different g values, we use the average-g value for all obtained non-dominated solutions. The true Pareto-optimal front corresponds to $g = 1$. Thus, the closer the value of average-g to one, the better is the converging ability of an EMO procedure.

First, we use PCX-NSGA-II with four different σ values – 0.01, 0.1, 0.4 and 0.7. To investigate the effect of linkages in variables, we systematically make the matrix D more dense by increasing the number of rows having randomly created elements (we refer to this number as the 'order of linkage' here). Thus, if the order of linkage is l, the D matrix has only l rows having randomly created elements and the diagonal entries of other $(n - k - l)$ rows are one. Figure 3, shows the performance of different PCX-NSGA-II procedures with the order of linkage. We have run the procedures till 2,000 generations to get a distribution of solutions close to the true Pareto-optimal front. For the polynomial mutation operator, we use $\eta_m = 20$. In each case, the best, average, and the worst average-g values of 11 runs are shown. We observe that the performance of all four PCX-NSGA-II procedures deteriorate with an increase in the order of linkage. However, all four procedures seem to have performed almost equally well, indicating that the σ value in the range of 0.01 to 0.7 does not make much of a difference to the performance of PCX-NSGA-II in this problem.

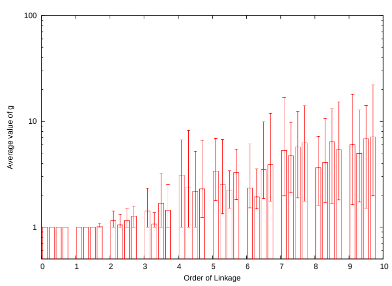

Figure 3: The bar represents average value of g from 11 runs along with the deviation for PCX(0.01), PCX(0.1), PCX(0.4) and PCX(0.7) respectively against the order of linkage.

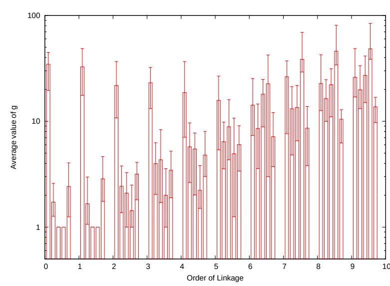

Figure 4: The bar represents average value of g from 11 runs along with the deviation for GDE3(1.0,1.0), GDE3(0.1,0.1), GDE3(0.5,1.0), GDE3(0.5,0.5) and GDE3(0.9,0.1) respectively against the order of linkage.

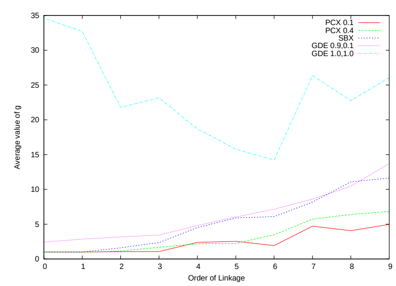

Figure 5: The graph represents average value of g from 11 runs for PCX(0.1), PCX(0.4), SBX, GDE3(0.9,0.1) and GDE3(1.0,1.0), respectively against randomness the order of linkage.

Next, we apply the GDE3 procedure for different values of CR and F values. Once again, 11 runs are made from different initial population and the best, average, and worst average-g values are shown in Figure 4. The performance is also seen to deteriorate with the order of linkage and the GDE3 with CR=0.9 and $F = 0.1$ is found to perform well on most situations.

Figure 5 shows the average values of front-average g value of 11 runs for five strategies: PCX-NSGA-II with $\sigma = 0.1$ and 0.4, SBX-NSGA-II with $\eta_c = 11$ and with $\eta_m = 20$ (which was found to be performing the best after some experimentations), and GDE3 with two different CR and F values. It is clear from the figure that PCX-NSGA-II procedures are able to get closer to the true Pareto-optimal front compared to SBX-NSGA-II and GDE3 procedures. However, it is quite evident that beyond the two-variable linkage none of the procedures is able to find the true Pareto-optimal front. Thus, the linkage among variables $\mathbf{x}_{II}$ is adequate to provide enough difficulty to all these EMO procedures.

7.2 Type-2 Problems

Here, we consider all five L_2-ZDT problems with a full M matrix with order of linkage same as n. As mentioned earlier, in these problems, although we know exactly the location of the two extreme Pareto-optimal solutions, the knowledge of other Pareto-optimal solutions is absent. To validate the obtained front, we use the normal-constraint (NC) method [10] to obtain 20 different well-dispersed Pareto-optimal solutions. We use a single-objective PCX-based GA procedure for this purpose.

Two performance criteria are used to measure converging and distributing ability of the chosen EMO procedures. The first approach uses the graphs showing the obtained frontier for 0% and 100% attainment values [7] of 11 runs. The second approach computes the hypervolume metric value [15] of the 0% attainment surface obtained with all 11 runs. Since in all these test problems, the extreme points are singleton solutions, we use a slightly different hypervolume measure. For all 11 runs obtained by an EMO procedure, we find the nadir point and use it as a reference point for computing the hypervolume measure. This way, different procedures will use different reference points. If a procedure does not find the complete Pareto-optimal front, the area covered by the front, as computed by the corresponding nadir point, may have a drastically small value.

7.2.1 L_2-ZDT1 Problem

Figure 6 shows the performance of PCX-NSGA-II, SBX-NSGA-II and GDE3. In each case,

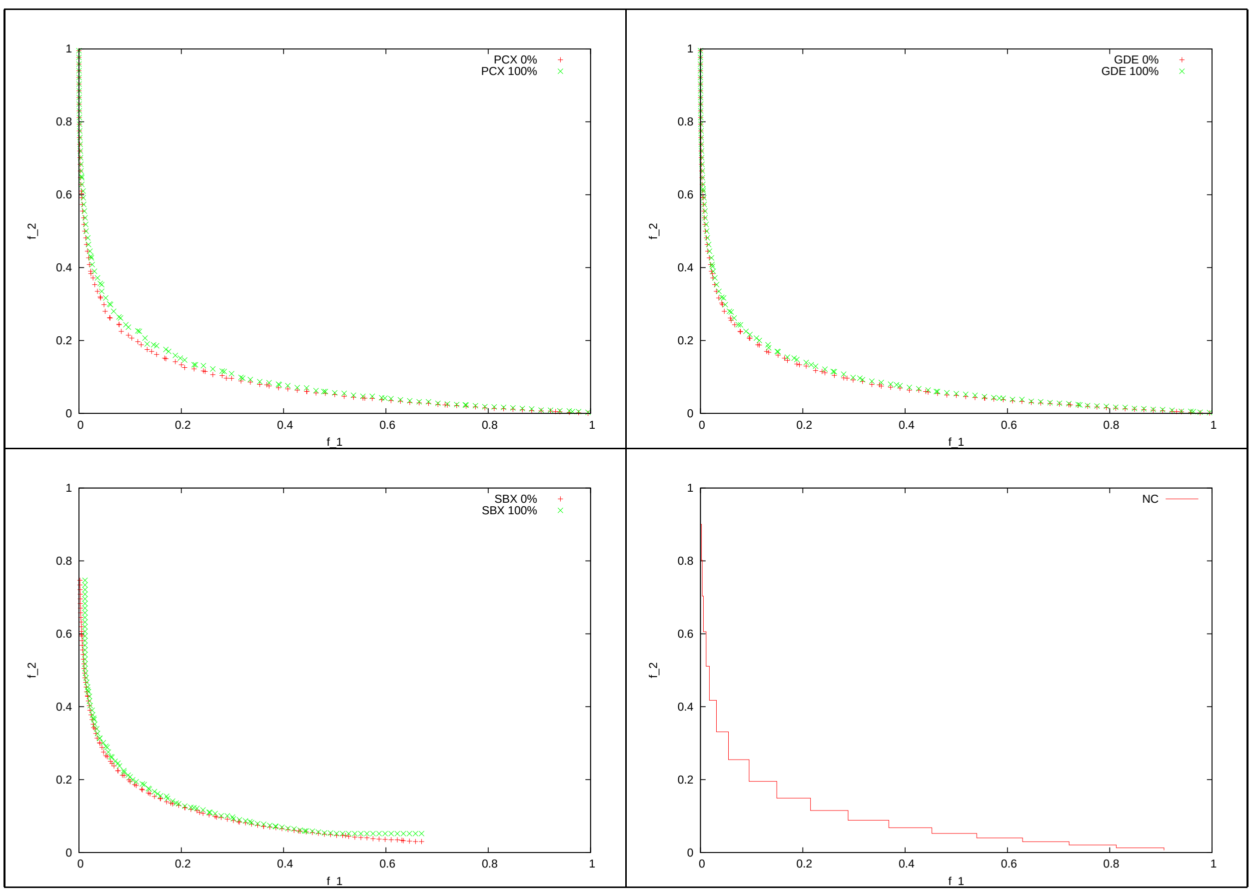

Figure 6: Pareto-optimal fronts obtained from PCX-NSGA-II(0.4), GDE3(1.0,1.0), SBX-NSGA-II, and NC methods for L_2-ZDT1.

the parameters which produced a consistently better performance in all test problems are used.

Table 1: Hypervolume metric for PCX(σ), SBX and GDE3(CR,F) for L_2-ZDT1.

Method	Hypervol.	
	T-2	T-3
PCX(0.01)	0.9057	0.9864
PCX(0.10)	0.9771	0.9797
PCX(0.40)	0.9757	0.9759
PCX(0.70)	0.9743	0.9750
SBX	0.4210	0.5705
L-SBX	0.3921	0.3847
GDE3(1.0, 1.0)	0.9783	0.9769
GDE3(0.5, 0.5)	0.7842	0.8957
GDE3(0.1, 0.1)	0.3318	0.3266
GDE3(0.9, 0.1)	0.7605	0.8689
GDE3(0.5, 1.0)	0.9786	0.9715

Table 2: Hypervolume metric for PCX(σ), SBX and GDE3(CR,F) for L_2-ZDT2.

Method	Hypervol.	
	T-2	T-3
PCX(0.01)	0.8611	0.8586
PCX(0.10)	0.9269	0.9277
PCX(0.40)	0.9254	0.9241
PCX(0.70)	0.9229	0.9239
SBX	0.4318	0.3243
L-SBX	0.3994	0.3043
GDE3(1.0, 1.0)	0.9284	0.9264
GDE3(0.5, 0.5)	0.8121	0.8238
GDE3(0.1, 0.1)	0.2789	0.2480
GDE3(0.9, 0.1)	0.7574	0.8236
GDE3(0.5, 1.0)	0.9272	0.9263

All EMO procedures are run for 500 generations. The figure shows clearly that both 0% and 100% attainment surfaces are close to each other, thereby indicating that the procedures reliably find non-dominated frontiers close to the true Pareto-optimal frontier (shown in the NC plot). However, the SBX-NSGA-II procedure cannot find the complete frontier, due to its limitation in handling linkages by using a variable-wise recombination operator. On the other hand, both PCX-NSGA-II and the GDE3 find the complete frontier.

Table 1 shows the hypervolume measures for all 11 algorithms. L-SBX is a line-SBX operator which creates two offspring on the line joining the two parents. The table shows that SBX-based NSGA-II runs are not able to find a good distribution, whereas all PCX-NSGA-II and some GDE3 procedures are able to find a good distribution. Due to restriction of solutions being created on a line joining the parents, the search power of L-SBX based NSGA-II is not as good as SBX-NSGA-II.

7.2.2 L_2-ZDT2 Problem

For brevity, here we do not show figures showing the distributions visually. Instead, we only present the hypervolume measures in Table 2. All parameters are set as that in L_2-ZDT1. Again, we observe a very similar trend in performance of the chosen EMO procedures.

7.2.3 L_2-ZDT3 Problem

Figure 7 shows the 0% and 100% attainment surfaces of 11 runs for three EMO procedures with a good parameter setting. As also evident from the Table 3, SBX-NSGA-II does not perform well compared to the PCX-NSGA-II and GDE3 procedures. It can be seen that in this problem PCX with a large σ works better. Like before, GDE3 with (CR,F) values of (1,1) and (0.5,1) consistently perform well.

7.2.4 L_2-ZDT4 Problem

Figure 8 shows that there is a significant difference in the 0% and 100% attainment surfaces for all three methodologies. This problem involves a number of local Pareto-optimal fronts and is comparatively harder to solve for optimality. For this problem, PCX with $\sigma = 0.4$ and 0.7 and GDE3 with CR=1 and $F = 1$ perform the best, as evident from Table 4.

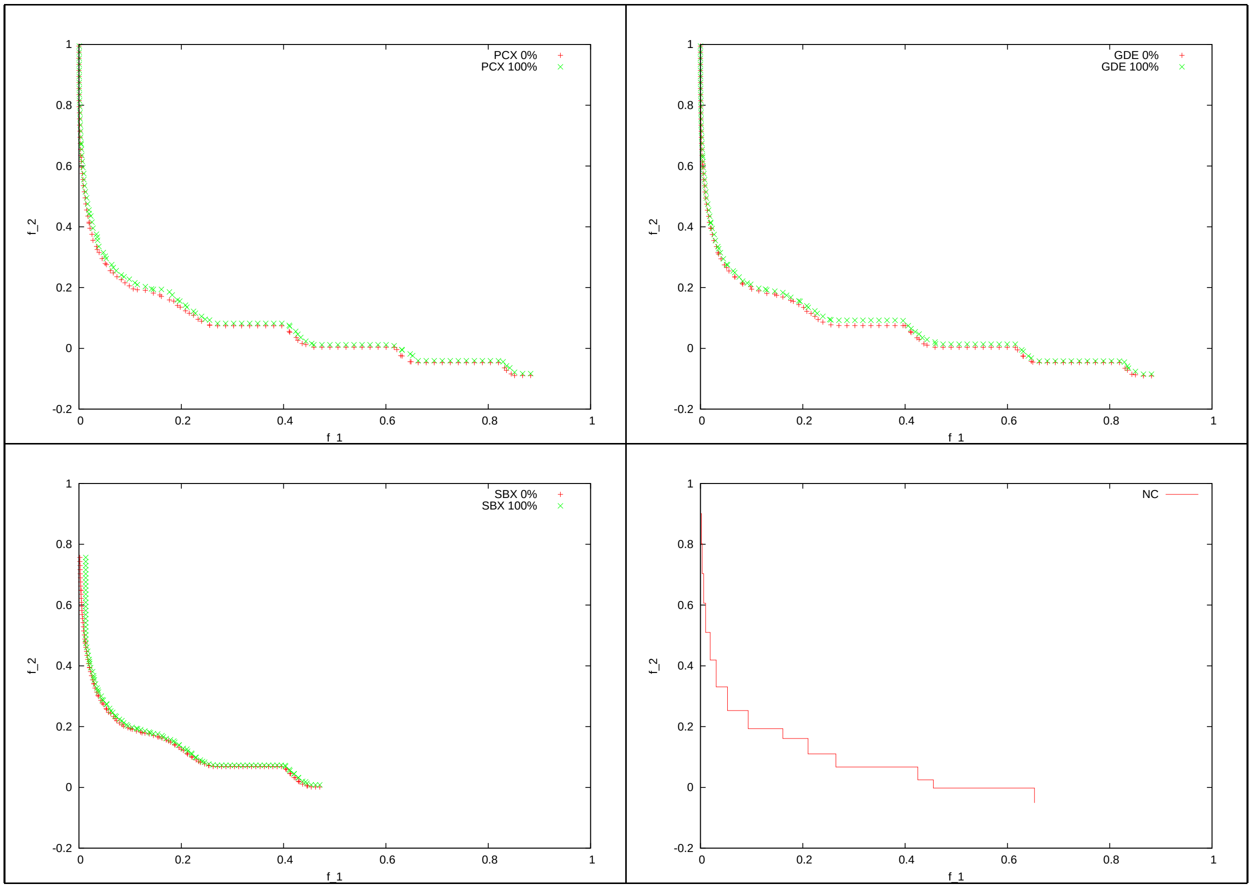

Figure 7: Pareto-optimal fronts obtained from PCX-NSGA-II(0.4), GDE3(1,1), SBX-NSGA-II, and NC methods for L_2-ZDT3.

7.2.5 L_2-ZDT6 Problem

Table 5 shows the hypervolume metric values. We observe that in this problem PCX-NSGA-II with larger σ values perform better than other methods, including the GDE3 procedures.

7.3 Type-3 Problems

These problems use a non-linear transformation rule, thereby causing difficulties to linear recombination operators. Since these EMO procedures except L-SBX-NSGA-II do not use a linear recombination operator, the performances of these algorithms do not get deteriorated by the non-linear transformation rule, as evident from the hypervolume metric values shown in Tables 1 to 5. However, the performance of NSGA-II having the line-SBX operator deteriorates in every Type-3 problem.

8 Three-Objective Modified DTLZ Problems

Like two-objective ZDT problems, we use the variable transformation method using linear (Type-2) and non-linear (Type-3) transformation rules. To investigate the difficulties of the chosen EMO procedures, here we choose 12-variable DTLZ2 and DTLZ3 problems. Both problems give rise to a three-dimensional, non-convex Pareto-optimal front. But DTLZ3 introduces multiple local Pareto-optimal fronts. In the modified versions (L_2-DTLZ and L_3-DTLZ problems), we have not

Table 3: Hypervolume metric for PCX(σ), SBX and GDE3(CR,F) for L_2-ZDT3.

Method	Hypervol.	
	T-2	T-3
PCX(0.01)	0.3935	0.3650
PCX(0.10)	0.6384	0.6578
PCX(0.40)	0.8556	0.8772
PCX(0.70)	0.9044	0.8749
SBX	0.2903	0.3156
L-SBX	0.2889	0.2723
GDE3(1.0, 1.0)	0.8561	0.8535
GDE3(0.5, 0.5)	0.5729	0.5928
GDE3(0.1, 0.1)	0.2089	0.2244
GDE3(0.9, 0.1)	0.4322	0.3572
GDE3(0.5, 1.0)	0.8550	0.8463

Table 4: Hypervolume metric for PCX(σ), SBX and GDE3(CR,F) for L_2-ZDT4.

Method	Hypervol.	
	T-2	T-3
PCX(0.01)	0.4352	0.5716
PCX(0.10)	0.5204	0.7443
PCX(0.40)	0.7539	0.7440
PCX(0.70)	0.7349	0.7420
SBX	0.4010	0.6138
L-SBX	0.4205	0.4113
GDE3(1.0, 1.0)	0.7353	0.7433
GDE3(0.5, 0.5)	0.5118	0.7104
GDE3(0.1, 0.1)	0.4275	0.6314
GDE3(0.9, 0.1)	0.4580	0.6217
GDE3(0.5, 1.0)	0.4206	0.7412

Table 5: Hypervolume Metric for PCX(σ), SBX and GDE3(CR,F) for L_2-ZDT6.

Method	Hypervolume	
	Type-2	Type-3
PCX(0.01)	0.0706	0.1699
PCX(0.10)	0.0822	0.1804
PCX(0.40)	0.1244	0.1599
PCX(0.70)	0.1574	0.2006
SBX	0.0982	0.1148
L-SBX	0.0811	0.0789
GDE3(1.0, 1.0)	0.1174	0.2301
GDE3(0.5, 0.5)	0.1127	0.1113
GDE3(0.1, 0.1)	0.1094	0.1476
GDE3(0.9, 0.1)	0.1088	0.1408
GDE3(0.5, 1.0)	0.1121	0.1111

normalized the objectives, as we did in the case of L_2-ZDT and L_3-ZDT problems. For these problems, we have used a population size of 200 and run each method for 500 generations.

8.1 DTLZ2 Problems with Linkages (L_2-DTLZ2 and L_3-DTLZ2)

Table 6 shows the hypervolume metric for PCX-NSGA-II, SBX-NSGA-II, L-SBX-NSGA-II and GDE3 methods. The performance of PCX-NSGA-II and GDE3 methods are much better than SBX-based NSGA-II procedures. Once again, it is evident that although the performance of SBX-NSGA-II procedure does not depend on the non-linear transformation, that of line-SBX based NSGA-II degrades when a non-linear transformation is used.

8.2 DTLZ3 Problems with Linkages (L_2-DTLZ3 and L_3-DTLZ3)

Table 7 shows the performance of all EMO methodologies considered in this paper on L_2-DTLZ3 (with linear transformation) and L_3-DTLZ3 (with non-linear transformation) problems. Due to the use of a multi-modal g function having a wide range of function values in these problems, the hypervolume metric values are larger here. However, as evident from the table, PCX-NSGA-II

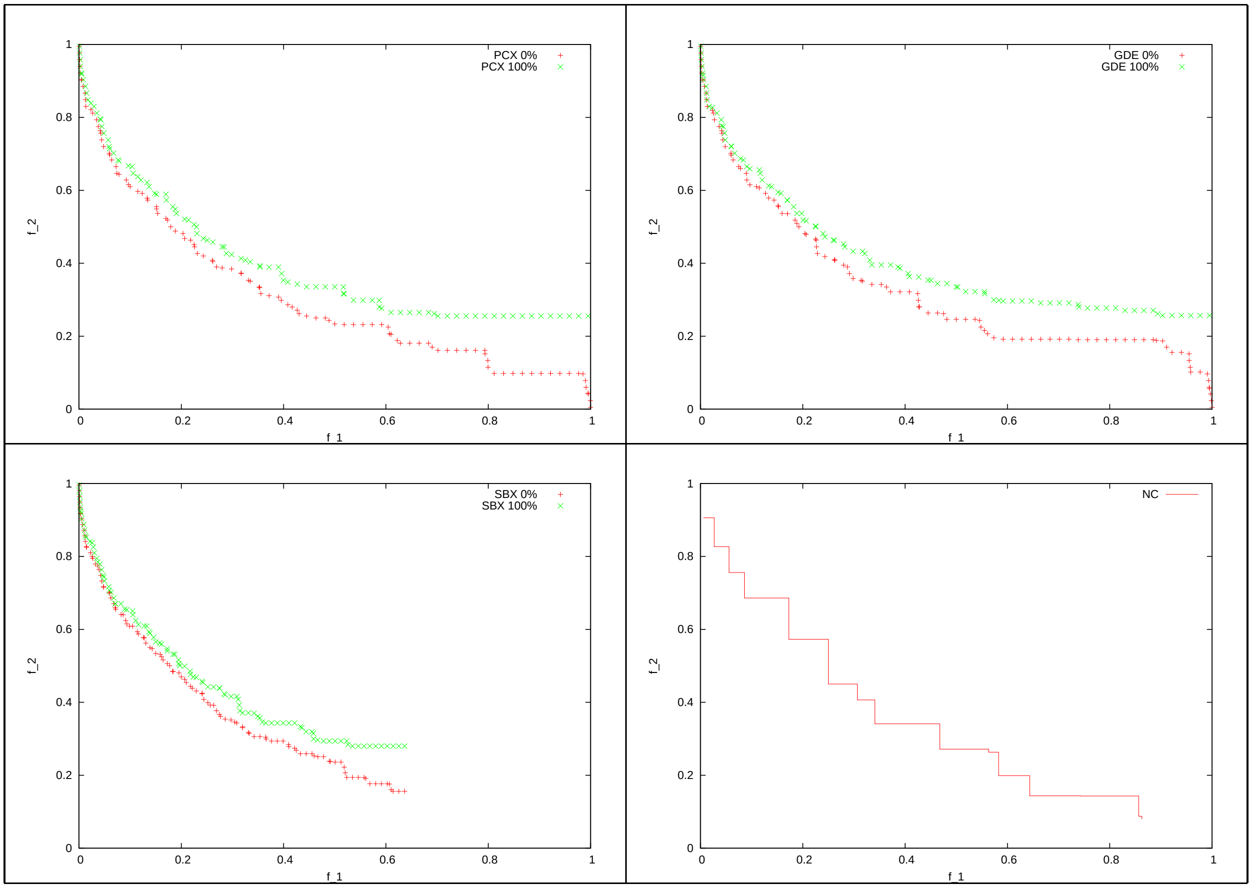

Figure 8: Pareto-optimal fronts obtained from PCX-NSGA-II(0.4), GDE3(1,1), SBX-NSGA-II, and NC methods for L_2-ZDT4.

and GDE3 procedures performed better than EMO with variable-wise recombination procedures. Again, the line-SBX operator seem to be get affected by the use of a non-linear transformation of the variables.

9 Conclusions

In this paper, we have addressed the issue developing multi-objective test problems which introduce controllable linkages among variables so that existing EMO methodologies can be provided with stringent tests. Several linkages are considered: (i) linkages among variables affecting either the convergence or diversity individually (ii) linkages among all variables causing a simultaneous effect in both convergence and maintenance of diversity among solutions and (iii) linkages which are non-linear, causing linear operators to face difficulty in preserving optimality of solutions. On a number of different test problems (which are modifications of existing two-objective ZDT and three-objective DTLZ problems), a few EMO methodologies have been tested: (i) NSGA-II with variable-wise recombination operators (SBX), (ii) NSGA-II with a variable-wise line-SBX operator, (iii) NSGA-II procedure with a vector-wise recombination operator (PCX), and (iv) a generalized multi-objective differential evolution procedure, GDE3 (having a vector-wise operation for creating offspring solutions). Several conclusions can be made from this extensive study:

1. An increase in order of linkage among variables cause a deterioration of the performance of the

Table 6: Hypervolume metric values for L$_2$-DTLZ2.

Method	Hypervol.	
	Type-2	Type-3
PCX(0.40)	1536.52	1533.12
SBX	1178.85	1176.87
L-SBX	1135.23	1003.45
GDE3(1,1)	1535.58	1533.63

Table 7: Hypervolume metric values for L$_2$-DTLZ3.

Method	Hypervol.	
	Type-2	Type-3
PCX(0.40)	3.2321E10	3.1014E10
SBX	2.6532E10	2.1745E10
L-SBX	2.4313E10	2.1012E10
GDE3	3.2287E10	3.0135E10

EMO methodologies.

2. EMO procedures with variable-wise recombination operators does not perform as well as those with vector-wise operators, due to their increased ability to handle linkages.

3. A non-linear transformation of variables does not affect the performance of SBX or PCX-based NSGA-II and GDE3 procedures, but deteriorates the performance of a line-SBX based NSGA-II.

Thus, vector-wise recombination operators are recommended for handling linkage-based multi-objective optimization problems.

Despite the existence and use of many test problems for multi-objective optimization, there seem to be a lack of test problems having controllable linkages among variables. This paper has hopefully demonstrated a procedure for creating such test problems and specifically suggested a set of two and multi-objective test problems for this purpose.

References

[1] K. Deb. Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation Journal*, 7(3):205–230, 1999.

[2] K. Deb. *Multi-objective optimization using evolutionary algorithms.* Chichester, UK: Wiley, 2001.

[3] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[4] K. Deb, A. Anand, and D. Joshi. A computationally efficient evolutionary algorithm for real-parameter optimization. *Evolutionary Computation Journal*, 10(4):371–395, 2002.

[5] K. Deb and D. Saxena. On finding pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems. Technical Report KanGAL Report No. 2005011, Department of Mechanical Engineering, IIT Kanpur, India, 2005.

[6] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham, L. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization*, pages 105–145. London: Springer-Verlag, 2005.

[7] C. M. Fonesca and P. J. Fleming. On the performance assessment and comparison of stochastic multiobjective optimizers. In *Proceedings of Parallel Problem Solving from Nature IV (PPSN-IV)*, pages 584–593, 1996.

[8] S. Huband, L. Barone, L. While, and P. Hingston. A scalable multi-objective test problem toolkit. In *Proceedings of the Evolutionary Multi-Criterion Optimization (EMO-2005).* Berlin, Germany: Springer, 2005.

[9] Saku Kukkonen and Jouni Lampinen. GDE3: The third evolution step of Generalized Differential Evolution. In *Proceedings of the 2005 Congress on Evolutionary Computation (CEC 2005)*, pages 443–450, Edinburgh, Scotland, Sept 2005.

[10] A. Messac and C. A. Mattson. Normal constraint method with guarantee of even representation of complete pareto frontier. *AIAA Journal*, in press.

[11] T. Okabe, Y. Jin, M. Olhofer, and B. Sendhoff. On test functions for evolutionary multi-objective optimization. In *Parallel Problem Solving from Nature (PPSN VIII)*, pages 792–802, 2004.

[12] Kenneth V. Price, Rainer Storn, and Jouni Lampinen. *Differential Evolution: A Practical Approach to Global Optimization*. Springer-Verlag, Berlin, 2005.

[13] A. Tiwari, R. Roy, G. Jared, and O. Munaux. Evolutionary-based techniques for real-life optimisation: Development and testing. *Applied Soft Computing*, 1:301–329, 2002.

[14] D. Van Veldhuizen. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. PhD thesis, Dayton, OH: Air Force Institute of Technology, 1999. Technical Report No. AFIT/DS/ENG/99-01.

[15] E. Zitzler. *Evolutionary Agorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, Zürich, Switzerland: Swiss Federal Institute of Technology ETH, 1999.

[16] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation Journal*, 8(2):125–148, 2000.