On Finding Multiple Pareto-Optimal Solutions Using Classical and Evolutionary Generating Methods

Pradyumn Kumar Shukla and Kalyanmoy Deb
Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
Kanpur, PIN 208016, India
Email: {shukla,deb}@iitk.ac.in
http://www.iitk.ac.in/kangal/pub.htm

KanGAL Report Number 2005006

Abstract

In solving multi-objective optimization problems, evolutionary methods have been adequately applied to demonstrate that multiple Pareto-optimal solutions can be found in a single simulation run. In this paper, we discuss and put together various different classical generating methods which are either quite well-known or are in oblivion due to lack available resources and some of which were even suggested before the inception of evolutionary methodologies. These generating methods specialize either in finding multiple Pareto-optimal solutions in a single simulation run or specialize in maintaining a good diversity by systematically solving a number of scalarizing problems. Most classical generating methodologies are classified into four groups mainly based on their working principles and one representative method from each group is chosen in the present study for a detailed discussion and for its performance comparison with a state-of-the-art evolutionary method. On visual comparisons of the efficient frontiers obtained for a number of two and three-objective test problems, the results bring out interesting insights about the strengths and weaknesses of these approaches. The results should enable researchers to design hybrid multi-objective optimization algorithms which may be better than each of the individual methods.

1 Introduction

Multi-objective optimization is a rapidly growing area of research and application in modern-day optimization. There exist a plethora of methods and algorithms for solving multi-objective optimization problems, see for example [32, 13, 8, 4]. The methods can be divided in two categories: (i) classical methods which use direct or gradient-based methods following some mathematical principles and (ii) non-classical methods which follow some natural or physical principles. Of them, the evolutionary multi-objective optimization (EMO) has been getting growing attention over the past decade. The classification is also appropriate from two other perspectives. The classical approaches usually use deterministic transition rules, whereas non-classical approaches usually use stochastic rules. They are also different from each other in another vital aspect. Classical methods mostly attempt to scalarize multiple objectives and perform repeated applications to find a set of Pareto-optimal solutions. On the other hand, an EMO method attempts to find multiple Pareto-optimal solutions in a single simulation run, providing the entire range of solutions (with ideal and nadir points) and the shape of the Pareto-optimal front.

However, there exist a few classical generating methods (stochastic and deterministic) which either attempt to find multiple Pareto-optimal solutions in a single simulation run, or attempt to solve multiple scalarized problems such that a good diversity among resulting solutions is maintained. In terms of overall goal of the optimization task, these classical generating methods are similar to the way EMO methods work. In this paper, we present some of these classical algorithms, provide a systematic comparison of four different such algorithms by showing simulation results on a number of two and three-objective optimization problems of varying complexity. We also compare their performances with an EMO methodology and unveil the problem classes where the classical generating methods are better and the problem classes where the EMO methodologies have their niche. The importance of the study stems from the fact although there has been many comparative studies of evolutionary algorithms [39, 16, 9] that there has been only one such (limited) comparative study with classical generating methods by the authors [34]. This study reveals important insights about the working of the algorithms, which can be combined together in a hybrid manner to develop even a better algorithm than individual algorithms.

This paper is divided into five sections of which this is the first. The next section present an overview of various classical generating methods, while the third section describe the chosen algorithms. Comparison with an EMO methodology (the elitist non-dominated sorting GA or NSGA-II [9]) is done in the fourth section. A discussion on the simulation results as well as extensions which emanated from this comparative study are presented in the end of this contribution.

2 Classical Generating Methods

The main goal of multi-objective optimization is to seek Pareto-optimal solutions. Over the years there have been various approaches toward fulfillment of this goal. It has been observed that *convergence* and *diversity* are two conflicting criteria which must be balanced in trying to generate the entire efficient front [8]. Clearly, there are two different possible principles for generating a set of solutions representing the entire Pareto-optimal front:

- One-at-a-time strategy, and
- Simultaneous strategy

In the former method, a multi-objective optimizer may be applied one at a time with the goal of finding one single Pareto-optimal solution. Most classical generating multi-objective optimization methods use such an iterative scalarization scheme of standard procedures, such as weighted-sum or ε-constraint method [13, 32]. The main criticism of most of these approaches is that although there are results for convergence, diversity among obtained Pareto-optimal solutions is hard to maintain in the objective space. Moreover, a careful thought suggests that a systematic variation of weight vectors or ε parameters in these scalarization techniques does not guarantee a good diversity in the solution sets [5, 8]. Another important matter is that independent applications of a single-objective optimization algorithm to find different Pareto-optimal solutions one-at-a-time do not make an efficient search and the search effort required to *solve* the problem to optimality this way needs to be found in every single time the algorithm is applied.

In the simultaneous approach, multiple Pareto-optimal solutions are found in a single simulation run, thereby not requiring multiple applications of an optimizer. These procedures are usually population or archive-based and often introduce a *parallel* search by sharing and exchanging important information among population or archive members. There exist some classical population-based generating methods which attempt to find multiple Pareto-optimal solutions in a single simulation run. On the other hand, there are also some classical point-by-point generating methods which involve archives and make an attempt to use a scalarization scheme so as to generate a diverse set of solutions in a single simulation. However, all evolutionary multi-objective optimization (EMO) methods use either an archive or a population and find multiple Pareto-optimal solutions in a single simulation. Although classifying these algorithms according to a single criterion is a difficult task, an attempt is made in the following section to group similar methods together. It may be noted that specialized methodologies exist for solving special class of multi-objective optimization problems, such as linear problems [2], bi-objective problems [14, 23] or convex problems [15]. However, we shall concentrate here on methods that are general and can solve problems with any number of objectives and without any convexity assumptions.

2.1 Stochastic Point-By-Point Spreading Algorithms

These algorithms use one point in each iteration, but use an archive to store the non-dominated solutions obtained thus far. Thus, once a Pareto-optimal solution is found, the search moves from one Pareto-optimal point to another, thereby capturing a well distributed set of solutions on the efficient frontier. According to our knowledge, there are two such methods suggested in the literature.

The earlier method in this class of algorithms is the *adaptive search method* by Benson and Meisel [1]. This method has also been presented by Hwang and Masud [19] and in a comprehensive survey by Marler and Arora [28]. The method uses a stochastic search together with the gradient information of the objective functions. The individual function minimum solutions are taken as the starting points. Thereafter, a search is made along a combined gradient direction by an amount controlled by a *step size* parameter which is changed adaptively from one iteration to another. For a detailed discussion of this method the reader is referred to the comprehensive survey work by Marler and Arora [28].

A recent method by Schäffler et. al. [33] also belongs to this category. It is a trajectory based method which uses function gradient information for a local search and a Brownian motion based global search. This method is chosen a representative algorithm of this class in this study and is described in detail in Section 3.

2.2 Stochastic Population Based Algorithms

These algorithms work with a population of points and iteratively progress toward the efficient frontier. Since they work with a population, similar to an EMO methodology, many Pareto-optimal solutions are found simultaneously in a single simulation run. To our knowledge, there are two methods in this category suggested so far. Surprisingly, both methods were developed in 1980s (even before the suggestion of EMO methodologies) and used ideas similar to the EMO algorithms.

The first algorithm in this category is the *sampling-search-clustering* method suggested by Törn [37]. In this algorithm, first, the initial points are sampled from the feasible region. After the initial points are obtained in this way, they are *pushed* toward the efficient frontier by a local optimizer. The local optimizer suggested is a random search algorithm, which also has been successfully used in single objective global optimization problems [12]. Since many such single-objective minimum solutions may be closer to each other, a clustering is performed, similar to the clustering techniques used in EMO algorithms [38, 8]. This process is iteratively repeated and finally the results are presented to a decision maker. A similar such EMO methodology was suggested elsewhere [20].

Another similar algorithm is the one suggested by Timmel [35, 36]. This algorithm also starts from a randomly sampled initial population. From each point (parent), another solution (child) is created by traversing along a gradient direction. Thereafter, the child and parent populations are combined together and a non-domination check is done to remove all dominated points. We have chosen this algorithm as a representative one of this class and a detail description is given in Section 3.

2.3 Direction-Based algorithms

These algorithms use scalarizing schemes that provide a good diversity among solutions in the objective space. For this purpose, most of these methods start from carefully chosen equidistant points on the ideal plane, that is, the plane passing through individual function minimizers (some authors call it as the utopia plane) and then proceed along certain directions. The progress along the search direction is measured using an auxiliary real variable. There are many methods which belong to this category.

The first algorithm we shall discuss in this category is the *normal boundary intersection* or the NBI method, developed by Das and Dennis [6]. Their study was aimed at getting a good diversity of solutions on the efficient frontier by starting from normal directions to the ideal plane. The study used an equality constraint formulation of the subproblems. A modified version of the NBI approach (called the recursive knee approach) was developed elsewhere [7] for convex problems. Better formulations were also developed elsewhere [27, 24]. Since this algorithm is used in this study, we shall discuss the NBI in more detail in the next section.

Benson and Sayin [3] developed a *global shooting procedure* to seek points on the efficient frontier. In this method, first, a simplex is obtained that contains the feasible region of the problem in the objective space. Next, representations of the efficient frontier are obtained by shooting rays toward certain facets of this simplex.

Kim and Weck [22] developed the *adaptive weighted sum method* for multi-objective optimization. Initially, the efficient frontier is approximated by employing a single-objective optimization algorithm with the weighted-sum approach many times. Efficient front patches are then identified and further refined by using additional equality constraints.

In Karasakal and Köksalan's [21] algorithm, first, the nadir point is estimated and a few points are found on the efficient frontier using the constrained Tchebycheff method. Next a weighted L_p hyper-surface is fitted and then equidistant points on this surface are obtained. Finally, these points are projected on to the efficient frontier.

2.4 Other Generating Algorithms

Messac and Mattson [31] developed the *normal constraint* method for getting an even representation of the efficient frontier through points. In the normal constraint method, there is a sequential reduction of the feasible space by hyper-planes passing through a point on the ideal plane. The method is described in detail in next section, as we use this method in this study. Messac also developed the *physical programming* method [29] and then presented a different method [30] for generating the entire efficient frontier using the physical programming approach.

Klamroth et. al. [25] developed a method for approximation of the efficient frontier. It generates a piecewise linear approximation of the efficient frontier. Two separate methods – *inner approximation* and *outer approximation* – are presented. In these methods, the current approximation is used as a distance measure. Also the approximation adds the next point which is currently the worst approximated or where the approximation error is maximal.

Hillermeier [17, 18] developed a *generalized homotopy method* for multi-objective optimization. In this method, the Pareto-optimal set is considered as a differentiable manifold and local charts are constructed and evaluated numerically for obtaining new Pareto-optimal solutions.

3 Representative Algorithms

For the comparative study here, we select Schäffler's [33] method among the stochastic point by point spreading algorithms, while Timmel's [35, 36] algorithm is chosen among the stochastic population based algorithms. The NBI [6] method is chosen among the direction-based algorithms, while the normal constraint [31] method is picked from the other algorithms. In the following subsections, we describe these methods in somewhat more detail.

3.1 Schäffler's Stochastic Method (SSM)

A stochastic method for the solution of unconstrained multi-objective optimization problems was proposed by Schäffler et. al. [33] in 2002. The method is based on the solution of a set of stochastic differential equations. This method requires the objective functions to be twice continuously-differentiable. It may be used for the computation of all or a large number of Pareto-optimal solutions. In each iteration, a trace of non-dominated points is constructed by calculating at each point $\mathbf{x}$ a direction $(-q(\mathbf{x}))$ in the decision space which is a direction of descent for *all* objective functions. The direction of descent is obtained by solving a quadratic subproblem. The following initial value problem (IVP) for a multi-objective optimization problem is then set up:

$$\dot{\mathbf{x}} = -q(\mathbf{x}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0,$$

where x_0 is a starting point. The numerical solution of the above IVP gives a single point where the first-order weak Pareto-optimality conditions are fulfilled. After such a solution is obtained, a set of non-dominated solutions is obtained by perturbing it using a Brownian motion concept. The following stochastic differential equation is employed for this purpose:

$$d\mathbf{X}_t = -q(\mathbf{X}_t)d(t) + \varepsilon dB_t, \quad \mathbf{X}_0 = \mathbf{x}_0, \tag{1}$$

where $\varepsilon > 0$ and B_t is a n-dimensional Brownian motion having the following properties:

1. The expected value is zero,
2. The increments B_0, $(B_{t_1} - B_{t_0})$, $(B_{t_2} - B_{t_1})$ for every $t_0(= 0) < t_1 < t_2 < \ldots$ are stochastically independent, and
3. For every $s < t$, the increment $(B_s - B_t)$ is normally distributed with mean equal to zero and a variance equal to $(s - t)I_n$, where I_n is a n-dimensional identity matrix.

Figure 1 shows (in the objective space) how from a given point A the next iterate a is obtained. Thus, starting from an initial solution, a number of solutions converging to the efficient frontier

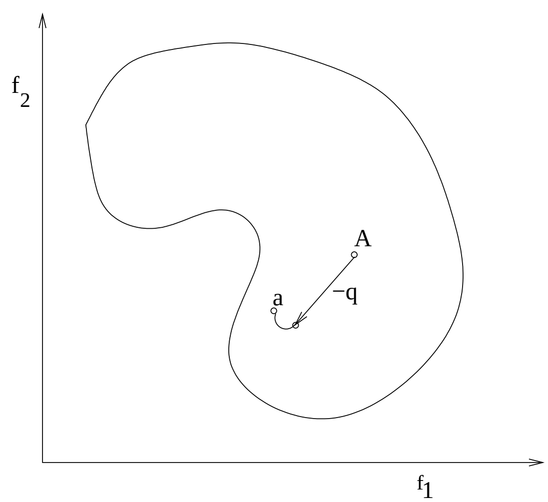

Figure 1: Schematic showing one iteration of SSM method.

are expected to be generated by this procedure. The $-q(\mathbf{X}_t)d(t)$ term in Equation 1 is the deterministic descent part, while the Brownian motion is the local random search term. In all simulations here, to solve the above equation numerically, we employ the Euler's method. The approach needs two parameters to be set properly: (i) the parameter ε which controls the amount of local search and (ii) the step size σ used in the Euler's approach which controls the accuracy of the integration procedure. At the end of a pre-specified number of iterations, a non-domination check of the obtained solutions is performed and the resulting solutions are declared as the obtained Pareto-optimal solutions. If the combination of Brownian motion and deterministic descent search is unable to converge to the global Pareto-optimal front, the algorithm may have difficulty in later finding many points on the global Pareto-optimal front. For more information on this algorithm, interested readers may refer to the original study [33].

3.2 Timmel's Population Based Method (TPM)

As early as in 1980, Timmel [35] proposed a population-based stochastic approach for finding multiple Pareto-optimal solutions of a differentiable multi-objective optimization problem. In this method, first, a feasible solution set (we call it a population) is randomly created. The non-dominated solutions ($\mathbf{X}_0 = \{\mathbf{x}_1^0, \mathbf{x}_2^0, \ldots, \mathbf{x}_s^0\}$) are identified and they serve as the first approximation to the Pareto-optimal set. Thereafter, from each solution $\mathbf{x}_k^0$, a child solution is created in the following manner:

$$\mathbf{x}_k^1 = -\sum_{i=1}^{M} t_1 u_i \nabla f_i(\mathbf{x}_k^0),$$

where u_i is a uniformly distributed random number (between 0 and 1) and t_1 is step-length in the first generation. It is a simple exercise to show that the above formulation ensures that not all functions can be worsened simultaneously. Thus, the child solution is either non-dominated to the parent solution $\mathbf{x}_k^0$, or it dominates the parent. However, the variation of the step-length over iterations must be made carefully to ensure convergence to the efficient frontier. The original study suggested the following sequence for updating the step-length t_j in the j-th generation:

$$\lim_{j\to+\infty} t_j = 0, \quad \sum_{j=1}^{\infty} t_j = \infty, \quad \sum_{j=1}^{\infty} t_j^2 < \infty.$$

Figure 2 shows the creation of a child a from parent A (in the objective space) during the j-th generation. It can be seen that not all children dominated their parents. Thus while in SSM the descent direction ensures descent of all objective functions it is not the case with the direction used in TPM. Thus a child may be nondominated to its parent and hence it will be included in the next generation unless some other solution dominates it. After the child population

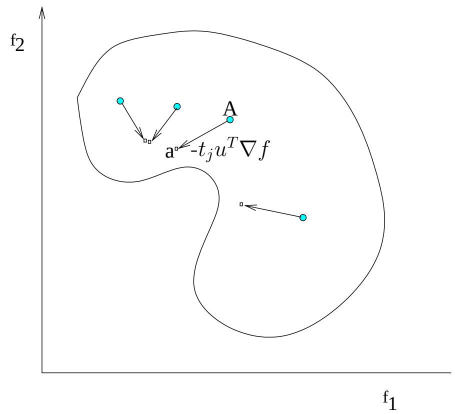

Figure 2: Schematic showing creation of a child a from parent A in the TPM method.

is created, it is combined with the parent population and only the non-dominated solutions are retained. Since a non-domination check is performed weak Pareto-optimal solutions are not there in the final optimized set. This set then becomes the second approximation to the Pareto-optimal set. This procedure is continued for a pre-specified number of iterations. Note that the population size can vary with iterations. In fact, in most problems, an increase in the population size is expected.

The step-length variation mentioned above ensures the following aspects:

1. The step size should slowly decrease to zero as solutions closer to the Pareto-optimal set are found and

2. The decrease of the step size must not be slow enough so that the algorithm gets caught in sub-optimal points.

Thus, it is clear that the update of the step length is a crucial part of the working of the algorithm and a tuning of the update strategy may have to be done for every problem. Here, we use the following strategy for varying t_j with generation j: $t_j = c/j$ (where c is a positive constant), which satisfies all the above-mentioned conditions. For the interested readers, we refer to the original study [35, 36] for further details. It is interesting to note that this algorithm uses an elitist strategy, in which best of parent and offspring populations is retained.

3.3 Normal Boundary Intersection Method (NBI)

The NBI method was developed by Das et. al. [6] for finding a uniformly spread Pareto-optimal solutions for a general nonlinear multi-objective optimization problem. The weighted-sum scalarization approach has a fundamental drawback of not being able to find a uniform spread of Pareto-optimal solutions, even if a uniform spread of weight vectors are used. The NBI approach uses a scalarization scheme with a property that a uniform spread in parameters will give rise to a near uniform spread in points on the efficient frontier. Also, the method is independent of the relative scales of different objective functions. The scalarization scheme is briefly described below.

Let us consider the following multi-objective problem (MP):

$$\begin{aligned} &\min_{\mathbf{x}\in S} \quad F(\mathbf{x}), \\ &\text{where} \quad S = \{\mathbf{x} \mid h(\mathbf{x}) = 0; g(\mathbf{x}) \leq 0, a \leq \mathbf{x} \leq b\}. \end{aligned} \tag{2}$$

Let $F^* = (f_1^*, f_2^*, \ldots, f_M^*)^T$ be the ideal point of the multi-objective optimization problem with M objective functions and n variables. Let the individual minimum of the functions be attained at $\mathbf{x}_i^*$ for each $i = 1, 2, \ldots, M$. The convex hull of the individual minima is then obtained. The simplex obtained by the convex hull of the individual minimum can be expressed as $\Phi\beta$, where $\Phi = (F(\mathbf{x}_1^*), F(\mathbf{x}_2^*), \ldots F(\mathbf{x}_M^*))$ is a $M \times M$ matrix and $\beta = \{(b_1, b_2, \ldots, b_M)^T \mid \sum_{i=1}^{M} b_i = 1\}$. The original study suggested a systematic method of setting β vectors in order to find a uniformly distributed set of efficient points. The NBI scalarization scheme takes a point on the simplex and then searches for the maximum distance along the normal pointing toward the origin. This obtained point may or may not be a Pareto-optimal point. Figure 3 shows the obtained solutions (i.e. a, b, c, d, e, f, g, h) corresponding to equidistant points (i.e. A, B, C, D, E, F, G, H) on the convex hull (i.e. line AH). It can be seen that points d, e and f are not Pareto-optimal but are still found using the NBI method. In non-convex situations, even the Pareto-optimal points which cannot be obtained by the usual weighted-sum schemes, are possible to be obtained by this method. For example, in Figure 3 solution x which lies in the non-convex region can be found by the NBI method while it cannot be obtained by the

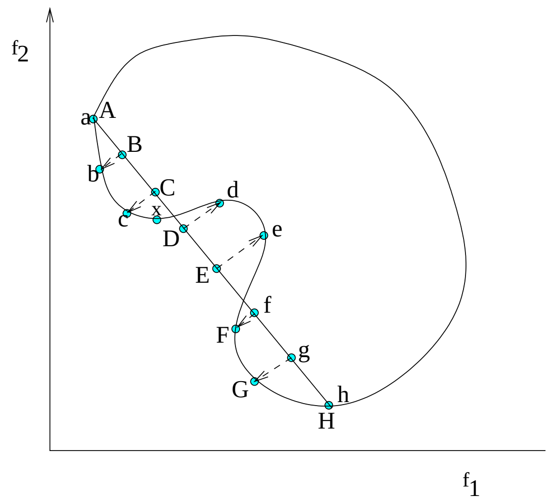

Figure 3: Schematic showing obtained solutions in NBI method.

weighted-sum schemes. The NBI subproblem (NBI$_\beta$) for a given vector β is as follows:

$$\begin{aligned} \max_{(\mathbf{x},t)} \quad & t, \\ \text{subject to} \quad & \Phi\beta + t\hat{n} = F(\mathbf{x}), \\ & \mathbf{x} \in S, \end{aligned} \tag{3}$$

where $\hat{n}$ is the normal direction at the point $\Phi\beta$ pointing towards the origin. The solution of the above problem gives the maximum t and also the corresponding Pareto-optimal solution, $\mathbf{x}$. The method works even when the normal direction is not an exact one, but a quasi-normal direction. The following quasi-normal direction vector is suggested in Das et al. [6]: $\hat{n} = -\Phi e$, where $e = (1, 1, \ldots, 1)^T$ is a $M \times 1$ vector. The above quasi-normal direction has the property that NBI$_\beta$ is independent of the relative scales of the objective functions.

In addition to sometimes obtaining non Pareto-optimal solutions in the NBI method, another limitation of the NBI procedure is that for dimensions more than two (in the objective space) the extreme Pareto-optimal solutions are not obtainable in all the cases. The limitation is due to the restriction of $0 \leq \beta_i \leq 1$ variable. This can be easily seen by considering a problem having a spherical efficient front satisfying $f_1^2 + f_2^2 + f_3^2 = 1$ in the range $f_1, f_2 \in [0, 1]$. As seen from Figure 4 there exist unexplored regions outside the simplex obtained by the convex hull of individual function minima.

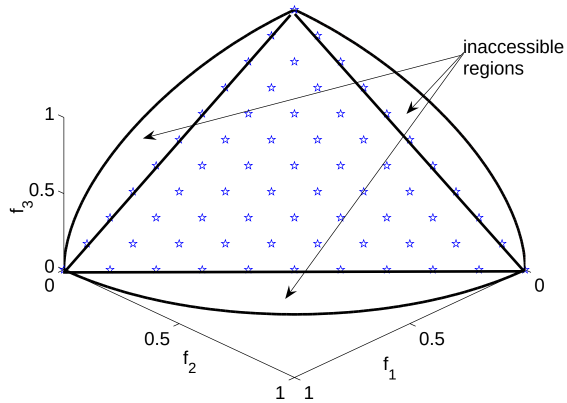

Figure 4: Plot showing difficulties in obtaining Pareto-optimal solutions outside the simplex by using NBI method.

3.4 Normal Constraint Method (NC)

Let us again consider the MP. The points $\mu^{i*} = F(x_i^*)$ for all $i = 1, 2, \ldots, M$ are called the *anchor* points. Let the M-dimensional hyper-plane passing through the anchor points be denoted by P^u and let μ^n denote the nadir point. Using the ideal and nadir points, let $\overline{f}_i$ denote the normalized i^{th} objective function f_i. The NC method uses the normalized function values to cope with disparate function scales. This is in contrast to the use of quasi-normal direction vector in NBI.

Next, let us denote by $\overline{N}_k$ (for all $k = 1, 2, \ldots, M-1$) the vectors from one fixed anchor point (assumed here as M^{th} only i.e. μ^{M*}) to all other anchor points, i.e. $\overline{N}_k = \overline{\mu}^{M*} - \overline{\mu}^{k*}$. Similar to NBI by systematically varying (in fixed increments) β $(= \{(b_1, b_2, \ldots, b_M)^T \mid \sum_{i=1}^{M} b_i = 1\})$ vector set of evenly distributed points on the ideal hyper-plane can be obtained. Let us denote by Z_β as the point on the ideal hyper-plane corresponding to vector β. The NC subproblem (NC$_\beta$) for a given vector β is as follows:

$$
\begin{array}{lll}
\min_{(\mathbf{x})} & \overline{f}_M(\mathbf{x}), & \\
\text{subject to} & \overline{N}_k^T(\overline{F} - Z_\beta) \leq 0, & \text{for } 1 \leq k \leq M\text{-}1 \\
& \mathbf{x} \in S &
\end{array}
\qquad (4)
$$

The normal constraint method uses inequality constraint reduction of the feasible space. Figure 5 shows the obtained solution using the NC method. The hatched part is the feasible region corresponding to point B in the ideal plane (i.e. AC). It can be seen that points b is

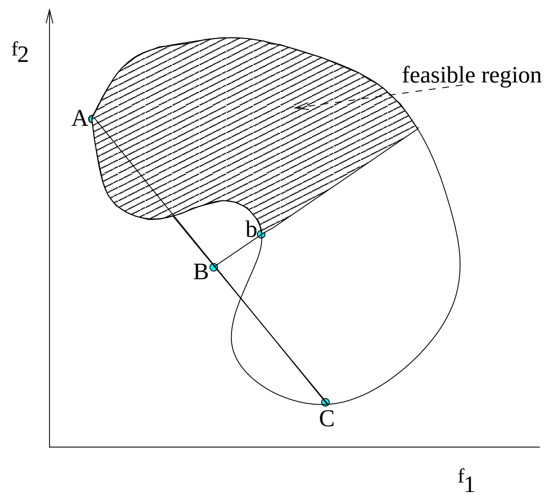

Figure 5: Schematic showing obtained solution in NC method.

not Pareto optimal but it is still found using the NC method. Since evenly distributed points on the ideal plane are used, the final points on the efficient front are likely to be more evenly distributed than the usual ϵ-constraint method.

In order to to alleviate the deleterious characteristic of NBI method of not obtaining extreme Pareto-optimal points in higher dimensional objective functions, Messac et. al. [31] developed a computationally efficient approach of finding the lower and upper limits of each variable β_i such that no region of the efficient frontier remains unexplored. To achieve this, first, a hypercube is created that encloses the entire feasible space in the objective space. Then the size of the ideal plane section is enlarged so that the projection of the hypercube can be made on the enlarged ideal plane. Finally, unnecessary regions (where no projection of the objective space lies on the enlarged ideal plane) are removed. It is to be noted that such a generalization is also possible with the NBI approach.

4 Comparison with an Evolutionary Method

In this section, we compare the above four classical generating methods with the elitist non-dominated sorting GA or NSGA-II [9] on a number of two and three-objective test problems. The test problems are chosen in such a way so as to systematically investigate various aspects of an algorithm. In the test problems, the exact knowledge of the Pareto-optimal front is available. For classical methods, a limited parametric study (for example parameters σ and ε in SSM and parameter c in TPM) is performed for each test problem and results from the best parameter setting are presented. For the NSGA-II, we use a standard real-parameter SBX and polynomial mutation operator with $\eta_c = 10$ and $\eta_m = 10$, respectively [8]. For all problems solved using NSGA-II, we use a population of size 100.

4.1 Two-Objective Test Problems

First, we consider two-objective ZDT test problems [8, 11]. The test problems are slightly modified so that they become unconstrained multi-objective optimization problems, as the SSM method is only able to tackle unconstrained problems in its present form. We are currently investigating a constrained version of SSM algorithm.

4.1.1 Modified ZDT1 Test Problem

The modified ZDT1 test problem can be stated as follows:

$$\begin{aligned} \text{Minimize} \quad & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} \quad & f_2(\mathbf{x}) = g(\mathbf{x}) \left(1 - \sqrt{\tfrac{x_1}{g(\mathbf{x})}}\right), \\ \text{where} \quad & g(x) = 1 + \tfrac{9}{n-1} \textstyle\sum_{i=2}^{n} x_i^2, \end{aligned} \tag{5}$$

where the box constraints are $x_1 \in [0, 1]$, and $x_i \in [-1, 1]$ for $i = 2, 3, \ldots, n$. Here, we choose $n = 30$. This modified ZDT1 problem has a convex Pareto-optimal front for which solutions correspond to $0 \leq x_1^* \leq 1$ and $x_i^* = 0$ for $i = 2, 3, \ldots, n$. Figure 6 shows the entire feasible space (in the objective space) for this problem as well as points in the objective space corresponding to 1,000 uniformly randomly generated points (within the box constraint) in the decision space. The upper boundary of the feasible region corresponds to $g(\mathbf{x}) = 10$ while the lower boundary corresponds to $g(\mathbf{x}) = 1$. This problem offers a difficulty in handling a large number of variables.

The Euler's method with a step size of $\sigma = 0.8$ along with $\epsilon = 0.05$ is used in SSM. An initial starting point is randomly created using the box constraints. It is to be noted that the SSM method requires gradient information. To make a fair comparison with an EMO methodology, gradients are calculated numerically here (using the central difference technique)

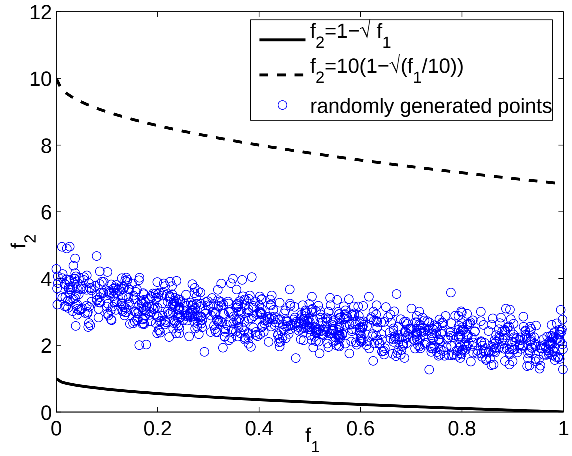

Figure 6: Feasible region of the modified ZDT1 problem.

and the overall function evaluations is recorded. Figure 7 shows the obtained distribution of efficient solutions after 20,000 (left plot) and 100,000 (right plot) function evaluations. Due to the use of a descent direction, the SSM method quickly converges near to the efficient frontier in this problem. However, the spread of solutions along the efficient frontier is very slow. Notice

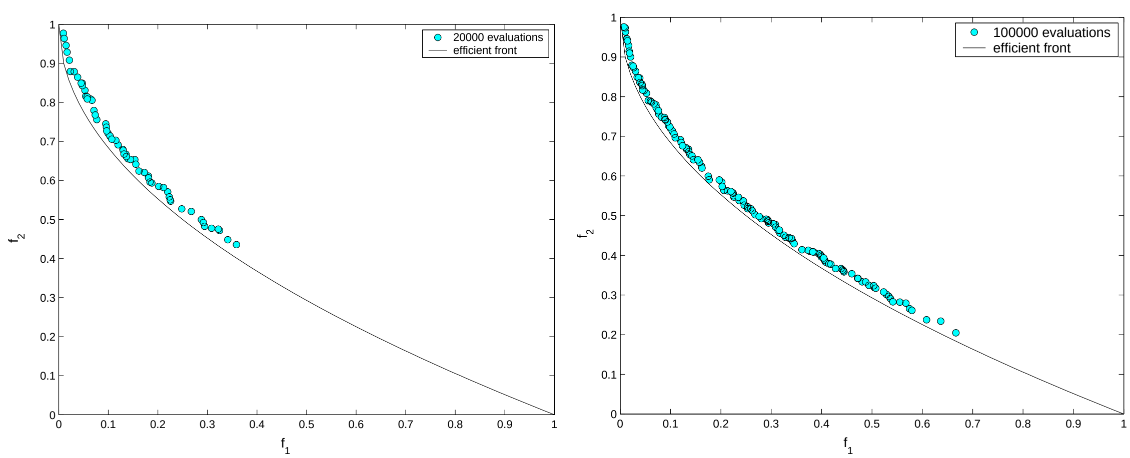

Figure 7: Performance of SSM method on ZDT1 (left and right).

that from 20,000 function evaluations till 100,000 function evaluations, the procedure finds a spread from $x_1 = 0.4$ to $x_1 = 0.7$. The use of a Brownian motion for obtaining a spread seems to be too generic to get a faster spread along the entire efficient frontier. After even 100,000 function evaluations, the solutions are not quite on to the efficient frontier. The numerical gradient evaluation is costly, requiring $2n$ function evaluations for each gradient. With a large number of variables, such methods may become computationally expensive. However, the

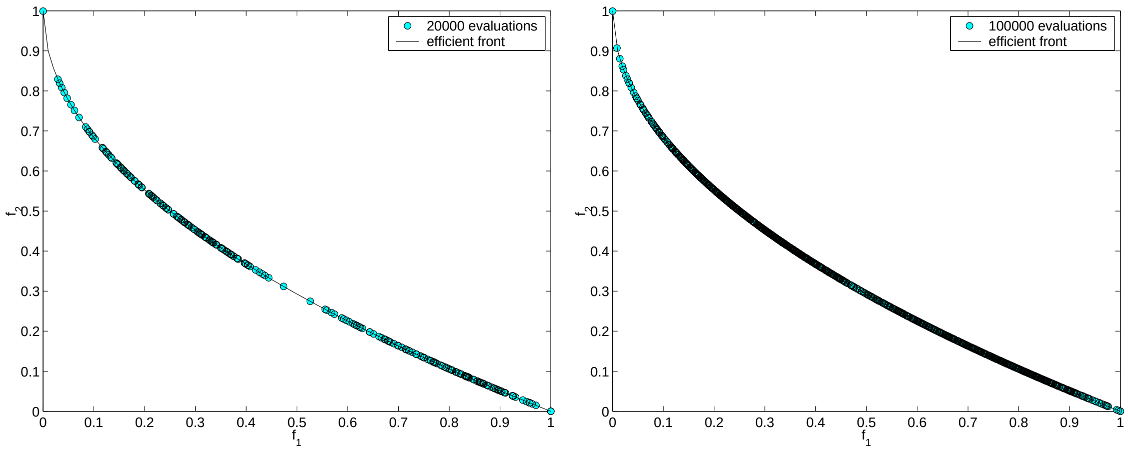

Figure 8: Performance of TPM method on ZDT1 (left and right).

simulation results show that this test problem does not offer too much of a difficulty to the SSM method in quickly converging near to the efficient frontier.

Next, we apply the TPM method. We begin the search with a single solution ($s = 1$), randomly created satisfying the box constraints. Figure 8 shows the obtained front after 20,000 (left plot) and 100,000 function evaluations. It is clear that the TPM method performs extremely well on ZDT1 both in terms of convergence to the Pareto-optimal front and maintenance of diversity.

The NBI method needs the computation of the ideal point and the anchor points. This requirement causes an added difficulty for the NBI method. The minimum solution of $f_1(\mathbf{x})$ is, in general, a weak Pareto-optimum solution, hence using a hyper-plane constructed using extreme weak Pareto-optimal solutions causes the NBI method to also converge to intermediate weak Pareto-optimal solutions. For example, the individual minimum for each of the two objective functions optimized using the sequential quadratic programming (SQP) method are as follows: $(0, 3.4278)$ and $(1, 0)$ for f_1 and f_2, respectively, whereas the corresponding Pareto-optimal solutions are $(0, 1)$ and $(1, 0)$, respectively. It is clear that some subproblems originating from some points on the hyper-plane constructed using the weak Pareto-optimal solutions may not result in finding true Pareto-optimal solutions. Thus solving these subproblems is a waste.

Here, the NBI subproblems are also solved using the SQP method. When finding multiple Pareto-optimal solutions for a fixed number of overall evaluations by using a single-objective optimizer sequentially and independently, one difficulty arises in distributing the total function calls to independent runs. Here, each SQP simulation is continued till convergence with a specified error value (i.e. termination tolerance on the function value is set as $\epsilon = 0.0001$ in the SQP code of Matlab software). Thereafter, the next SQP is started using the next β vector. The process is continued till the total number of function evaluations reaches 20,000 or 100,000, as the case may be. Based on the average number of evaluations needed in each SQP simulation, the number of β vectors are adjusted. A similar method for allocation of function evaluations is also done for other test problems. The obtained frontier is shown in Figure 9. As can be seen the weak Pareto-optimal solutions are also obtained. In solving the NBI sub-problems using SQP, two different methods of setting the starting point are used. In the first approach, as shown in Figure 9 (left plot), the NBI solution obtained using the previous solution of sub-problem is used as the starting point of the current sub-problem, while in the second approach (shown in Figure 9 (right plot)) a random initial solution is taken as the starting point of the current sub-problem. As many as 131 Pareto-optimal solutions are obtained using the first approach, while only 46 solutions are obtained using the second approach. Although intuitive, the first approach is observed to work much better than the second approach in this problem. In all other problems, we simply use the first approach, as also suggested in [6], unless otherwise stated. ¿From Figure 9, it can be also be seen that the NBI method is capable of finding a good spread of Pareto-optimal solutions even with 20,000 function evaluations in the ZDT1 problem. Since, a systematic set of initial points (β vectors) are considered in this approach, a good spread is obtained. If more densed β vectors are used, a more dense set of Pareto-optimal solutions could have been found.

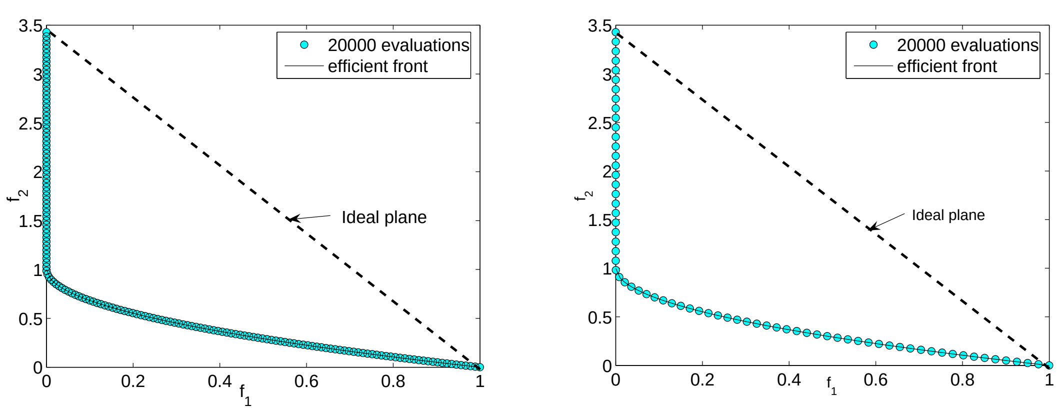

Figure 9: Performance of NBI method on ZDT1 (left and right).

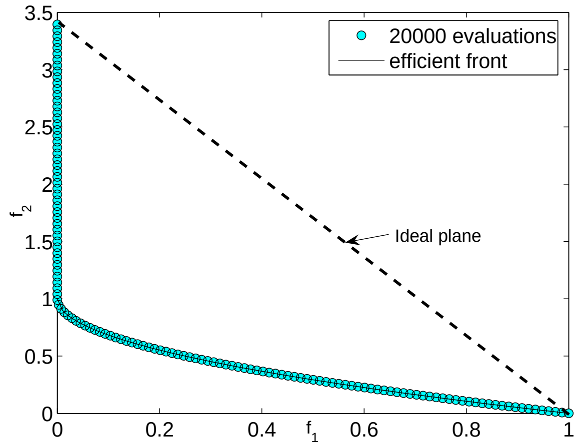

Figure 10: Performance of NC method on ZDT1.

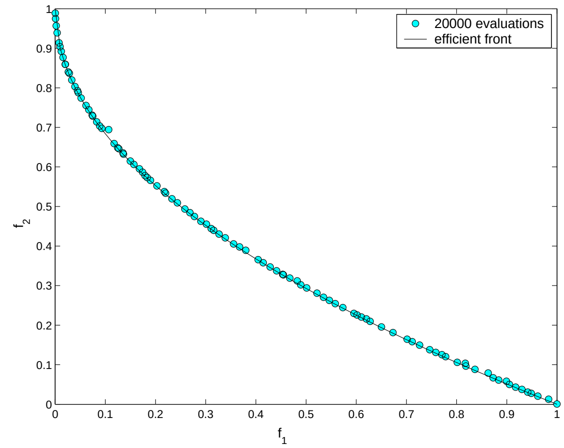

Figure 11: Performance of NSGA-II method on ZDT1.

Next, we apply the NC method using the same boundary points. The NC method uses an additional inequality constraint to restrict the feasible objective space and hence it allows a more flexible search than NBI, which requires handling of a nonlinear equality constraint. Figure 10 shows the obtained front after 20,000 function evaluations. It is clear that the NC method performs extremely well on ZDT1, both in terms of convergence and maintenance of diversity. However likewise NBI, a large number of NC subproblems are a waste since they find weak Pareto-optimal solutions.

Finally, we apply NSGA-II for a total of 20,000 function evaluations. Figure 11 shows that a good distribution is achieved. Based on all these simulations, it can be concluded that the ZDT1 problem is best solved by using a systematic procedure such as the NC or NBI method, whereas the population based approaches, that is, TPM and NSGA-II also perform well on this problem.

4.1.2 Modified ZDT2 Test Problem

The modified ZDT2 test problem can be stated as follows:

$$\begin{aligned} \text{Minimize} \quad & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} \quad & f_2(\mathbf{x}) = g(\mathbf{x}) \left(1 - \left(\frac{x_1}{g(\mathbf{x})}\right)^2\right), \\ \text{where} \quad & g(x) = 1 + \frac{9}{n-1} \sum_{i=2}^{n} x_i^2, \end{aligned} \qquad (6)$$

where the box constraints are $x_1 \in [0, 1]$ and $x_i \in [-1, 1]$ for $i = 2, 3, \ldots, n$. Here again we use $n = 30$. This problem resorts to a non-convex efficient frontier. The Pareto-optimal solutions correspond to $0 \leq x_1^* \leq 1$ and $x_i^* = 0$ for $i = 2, 3, \ldots, n$. This problem provides two difficulties to an optimization algorithm: (i) a large number of variables and (ii) a non-convex efficient frontier.

The Euler's method with a step size of $\sigma = 0.1$ along with $\epsilon = 0.01$ is used in the SSM algorithm. Figure 12 (left and right) shows the obtained distribution of solutions after 20,000 and 100,000 function evaluations, respectively. Although the convergence near the efficient front is quite similar to that in ZDT1, the distribution is poor.

In the TPM method, we use a population of size 100 randomly created satisfying the box constraints. Solutions after 20,000 function evaluations are shown in Figure 13. A good convergence and diversity of solutions is observed again.

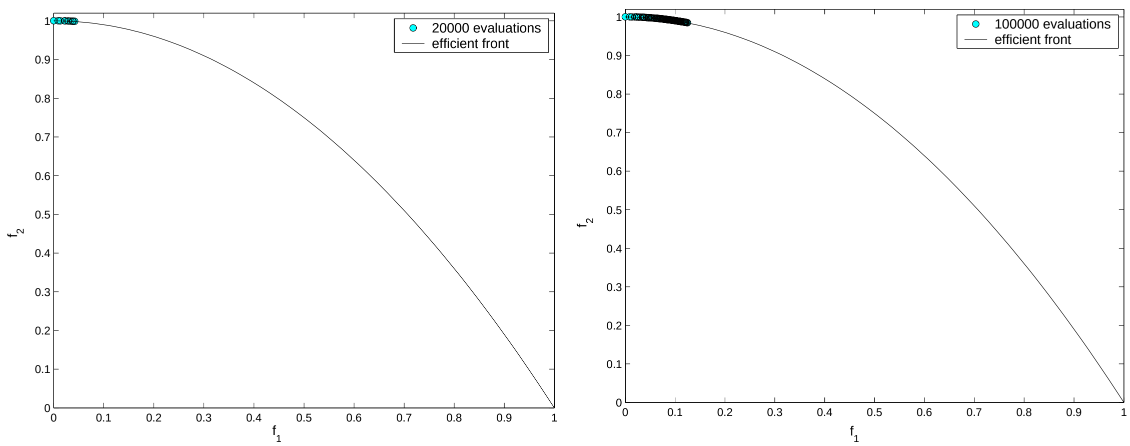

Figure 12: Performance of SSM method on ZDT2 (left and right).

Figure 14 shows the solutions obtained using the NBI method. Here, the individual minimum of each objective function are as follows: $(0, 3.4296)$ and $(1, 0)$ for f_1 and f_2, respectively. A good set of solutions even with 20,000 function evaluations is apparent from the figure. Since the search is performed along a direction towards the Pareto-optimal front, the NBI approach does not get affected by the convexity of the efficient frontier.

Figures 15 and 16 shows the NC and NSGA-II solutions for 20,000 function evaluations, respectively. It is also clear that the non-convexity of the efficient frontier does not provide any difficulties to NC, NBI, TPM and NSGA-II approaches, while the performance of the SSM approach is poor.

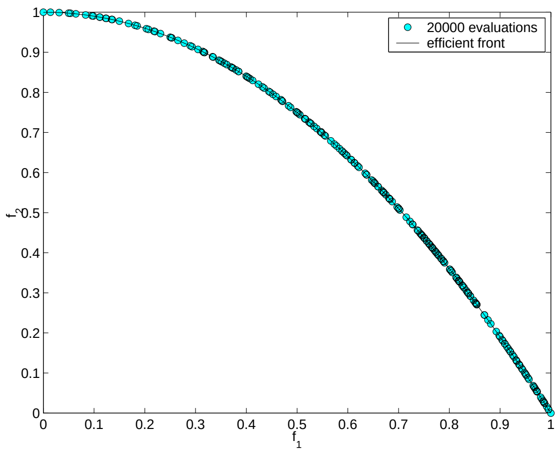

Figure 13: Performance of TPM method on ZDT2.

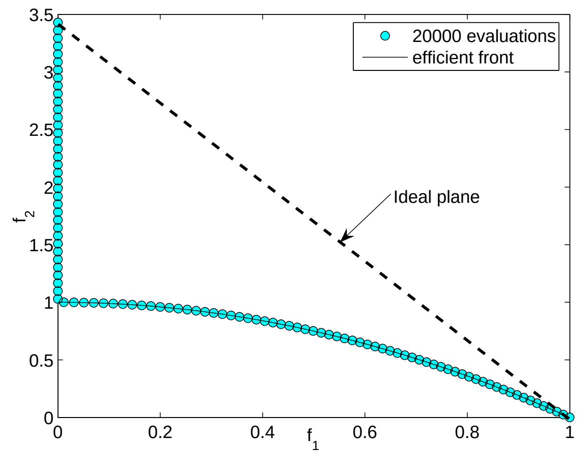

Figure 14: Performance of NBI method on ZDT2.

4.1.3 Modified ZDT3 Test Problem:

The modified ZDT3 test problem can be stated as follows:

$$\begin{aligned} \text{Minimize} \quad & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} \quad & f_2(\mathbf{x}) = g(x)\left(1 - \sqrt{\tfrac{x_1}{g(\mathbf{X})}} - \tfrac{x_1}{g(\mathbf{X})}\sin(10\pi x_1)\right), \\ \text{where} \quad & g(x) = 1 + \tfrac{9}{n-1}\textstyle\sum_{i=2}^{n} x_i^2, \end{aligned} \tag{7}$$

where the box constraints are $x_1 \in [0,1]$, and $x_i \in [-1,1]$ for $i = 2,3,\ldots,n$. We use $n = 30$. This problem has a convex discontinuous efficient frontier. The Pareto-optimal solutions correspond to $0 \leq x_1^* \leq 1$ and $x_i^* = 0$ for $i = 2,3,\ldots,n$. The Euler's method with a step size $\sigma = 0.5$ along with an $\epsilon = 0.01$ is used in SSM. Figure 17 shows the obtained distribution after 20,000 (left plot) and 100,000 (right plot) functions evaluations. Only a portion of the efficient frontier is discovered by this method. This example also indicated that SSM performs poorly in the case of multi-objective problems having disconnected efficient frontiers. The TPM method is applied with an initial population of size 5,000, randomly created satisfying the box constraints. Figure 18 (left and right) shows the obtained solutions after 20,000 and 100,000 evaluations, respectively. Two figures show that all disconnected efficient fronts are discovered by this method.

Figure 19 shows the distribution of the NBI method after 20,000 evaluations. Here, the

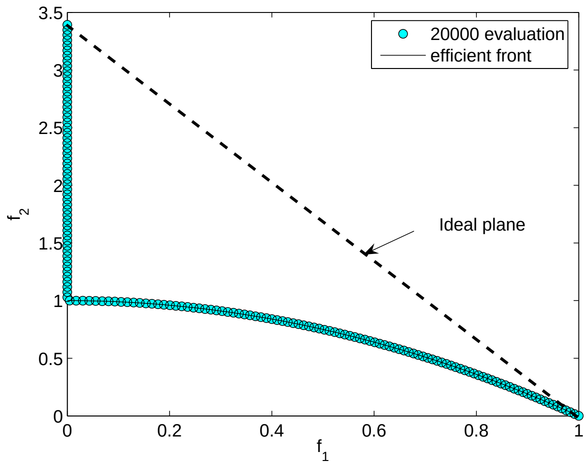

Figure 15: Performance of NC method on ZDT2.

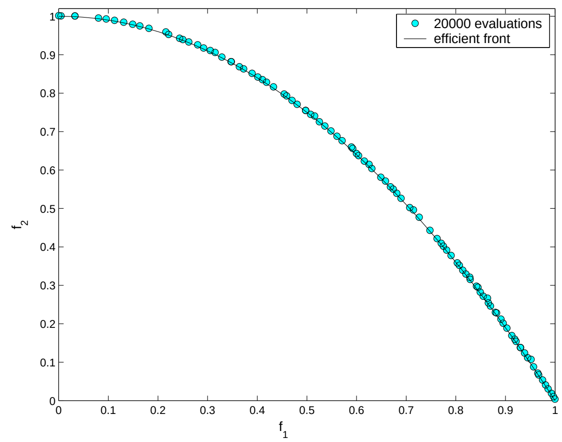

Figure 16: Performance of NSGA-II method on ZDT2.

individual minimum of each objective function are as follows: $(0, 3.9979)$ and $(0.9999, 0)$ for f_1 and f_2, respectively. It is apparent that although disconnected efficient fronts are discovered by this approach it takes a large number of function evaluations. A dense set of points can be obtained using this approach however the number of function evaluations required will be large. Also similar to that in ZDT1 and ZDT2 problems the method finds a large number of weak Pareto-optimal solutions (such as point D). As can be seen from Figure 19 only few subproblems find a Pareto-optimal solution (such as point A). Other subproblems find local Pareto-optimal solutions (like point B and C). The important point to observe here is that subproblems will find points B and point C (i.e. the local Pareto-optimal solutions) even when global solvers are used to solve these subproblems. This is an inherent drawback of searching only along an inclined line. Although the NBI method performed very well on ZDT1 and ZDT2 problems having a continuous efficient front, the disconnectedness of the efficient frontier seems to have provided difficulty to this approach. This method also ends up finding a dominated portion of the true efficient frontier. This is not surprising because the idea of non-domination is never applied in any step of the NBI approach.

Figure 20 shows the performance of the NC method on ZDT3. Since the NC method uses an inequality constraint formulation, the feasible space is more compared to corresponding sub-problem of NBI, thus there are more chances to be trapped in local optimal in the case of NC. For example in Figure 20, the subproblem corresponding to point Z_β obtains point A as

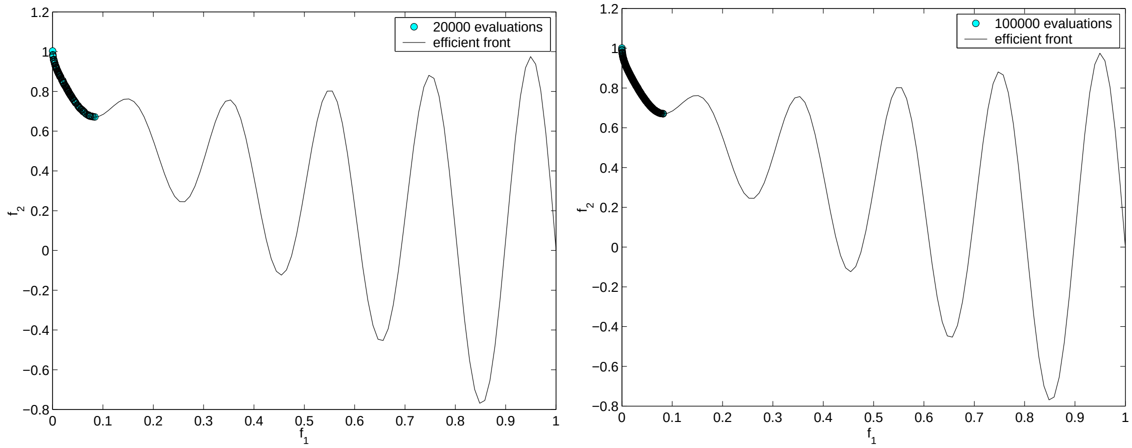

Figure 17: Performance of SSM method on ZDT3 (left and right).

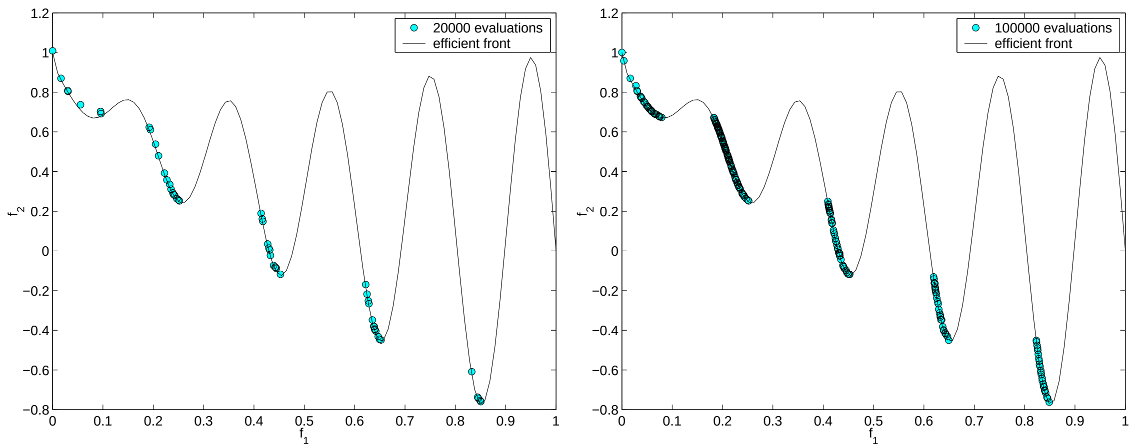

Figure 18: Performance of TPM method on ZDT3 (left and right).

solution while the corresponding solution would have been point B.

A comparison with the NSGA-II results (Figure 21) indicates the NSGA-II with only 20,000 evaluations is able to find all disconnected efficient fronts.

4.1.4 ZDT4 Test Problem:

Next, we use the 10-variable ZDT4 test problem [8]. This problem has a total of 100 distinct local efficient fronts in the objective space. The global Pareto-optimal solutions correspond to $0 \leq x_1^* \leq 1$ and $x_i^* = 0$ for $i = 2, 3, \ldots, n$. The algorithms face a difficulty in overcoming a large number of local fronts and converging to the global front.

The Euler's method with a step size of $\sigma = 0.1$ along with $\epsilon = 0.001$ is used in SSM. Only a few weak Pareto-optimal solutions ($f_1 = 0$ and $f_2 = 70$ to 70.4) are found after 20,000 evaluations. Since the SSM method requires functions to be twice continuously-differentiable

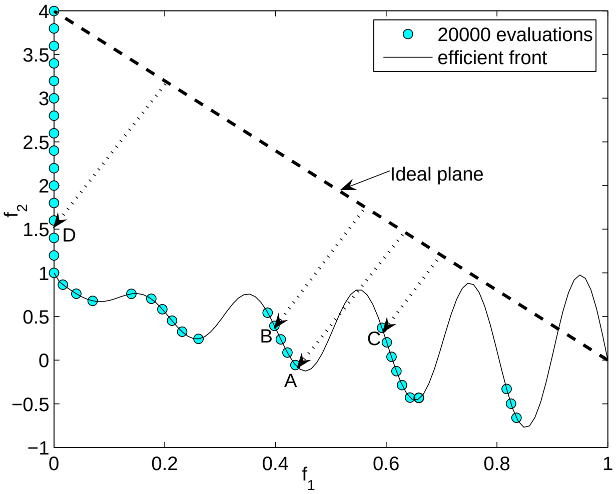

Figure 19: Performance of NBI method on ZDT3.

and since ZDT4 is not twice differentiable precisely at $x_1 = 0$, the gradient computation is erroneous at $x_1 = 0$, resulting in a failure of the method.

The TPM method is applied with 2,000 initial solutions randomly created satisfying the box constraints. Figure 22 shows that a set of dominated local-Pareto-optimal solutions is discovered after 20,000 (right plot) and 100,000 (left plot) evaluations. It can be seen that the population is stuck on three different local-Pareto-optimal fronts. The simple algorithm used in the TPM method can get stuck to a local-optimal solution and the ZDT4 problem with many local efficient frontiers provides enough difficulty to this approach for finding the true global efficient frontier. Also it can be easily seen that once the algorithm gets stuck to some local Pareto-optimal solution it is difficult to get out of it. Thus, a step size adaptation scheme may be important for the success of this algorithm.

The multi-modality of the search space also causes the NBI method to not find the global efficient frontier. Here, the individual minimum of each objective function are as follows: $(0, 81.1707)$ and $(1, 0)$ for f_1 and f_2, respectively. The SQP method is inadequate to find the global optimal solutions. Thus we do not show any result of NBI. Figure 23 shows the performance of the NC method after 100,000 evaluations. It also is not able to reach the global Pareto-optimal front. However, Figure 24 shows that NSGA-II with only 20,000 evaluations is able to converge to the global efficient frontier.

The problem ZDT4 provides difficulty in terms of multi-modality of the search space. It

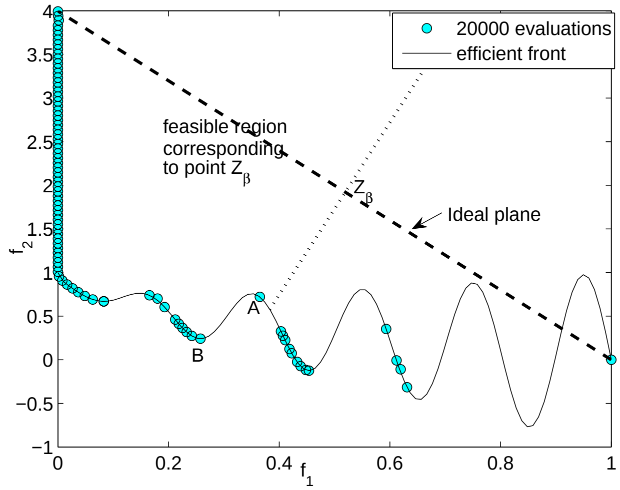

Figure 20: Performance of NC method on ZDT3.

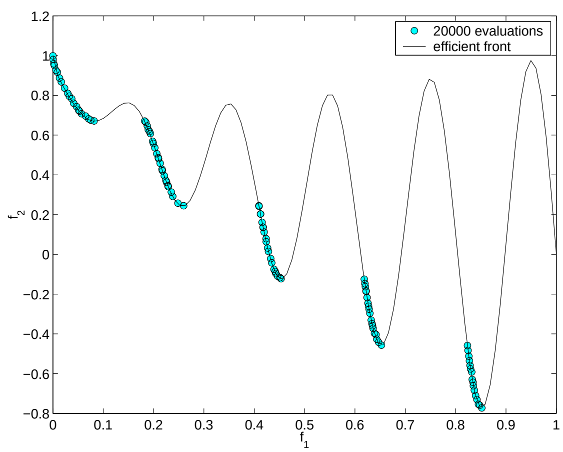

Figure 21: Performance of NSGA-II method on ZDT3.

is evident from the simulation results that the classical generating methods face enormous problems in overcoming the multi-modalities, whereas in this type of problems evolutionary multi-objective (EMO) methods are particularly found to be useful. This problem makes a clear distinction between population-based and point-by-point-based generating methods of solving multi-objective optimization problems. To converge to the efficient frontier in this problem, an optimization algorithm must have to overcome a number of local efficient frontiers. Since in finding every Pareto-optimal solution, such hurdles have to be overcome, it is too demanding for a classical point-by-point approach to expect to do the task well independently every time. On the other hand, a population-based evolutionary approach recombines the decision variable vectors of good solutions to create new and hopefully better solutions through its recombination

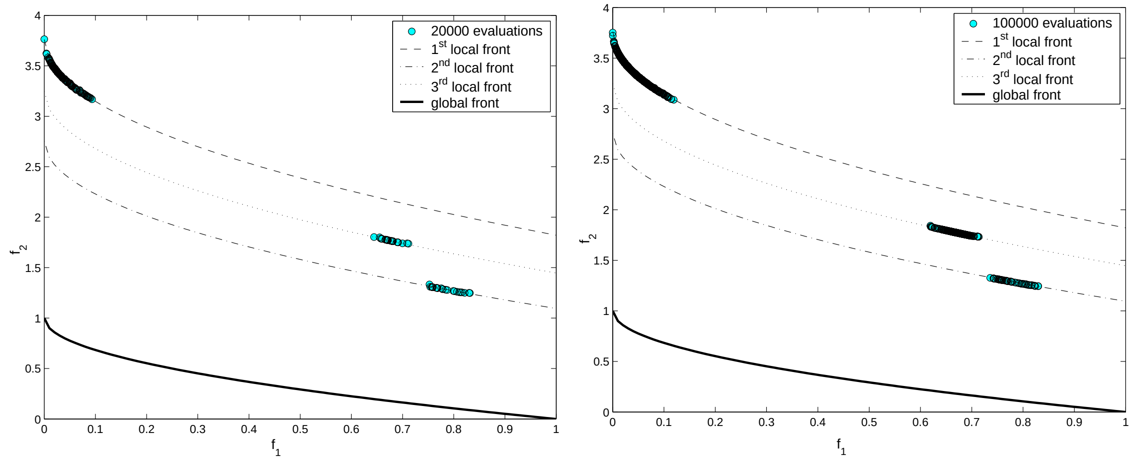

Figure 22: Performance of TPM method on ZDT4 (left and right).

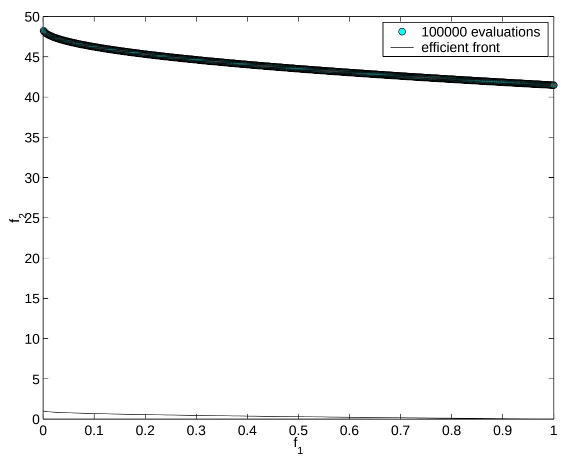

Figure 23: Performance of NC method on ZDT4.

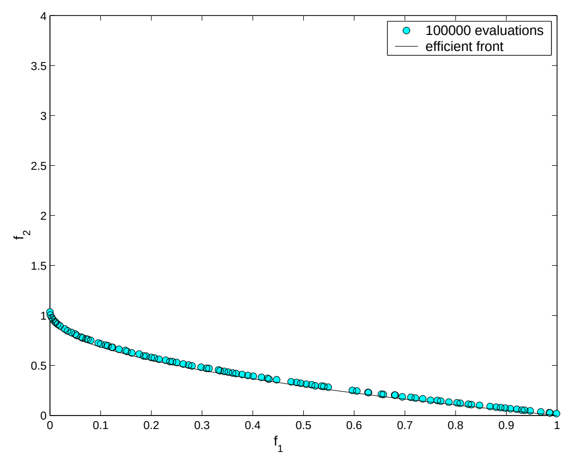

Figure 24: Performance of NSGA-II method on ZDT4.

operator. Thus, if one population member somehow reaches close to the global efficient frontier, it can *pull* the rest of population members close to the global frontier by means of population interactions, thereby causing a faster and more reliable search. Although, the trick of using the previous optimal solution to start a new subproblem can be used as an alternative, thereby ensuring starting with a close-by solution for subsequent subproblems, finding the first Pareto-optimal solution using a classical local search method also always remains as a hurdle. Thus, we can conclude that an iterative single-objective approach to generate a well-distributed and well-converged Pareto-optimal set may not always be a wise choice. If generation of the entire Pareto-optimal front is desired, it may be better to use a population-based approach which uses an *implicitly parallel* search through its population members, such as the algorithms motivated by evolutionary principles [8].

4.1.5 Modified ZDT6 Test Problem:

The $n = 10$ variable modified ZDT6 test problem is as follows:

$$
\begin{aligned}
\text{Minimize} \quad & f_1(\mathbf{x}) = 1 - \exp(-4x_1)\sin^6(4\pi x_1), \\
\text{Minimize} \quad & f_2(\mathbf{x}) = g(\mathbf{x})\left(1 - \left(\frac{f_1(\mathbf{x})}{g(\mathbf{x})}\right)^2\right), \\
\text{where} \quad & g(x) = 1 + 9\left(\sum_{i=2}^{n} x_i^2/(n-1)\right)^{0.25},
\end{aligned}
\tag{8}
$$

where the box constraints are $x_1 \in [0, 1]$ and $x_i \in [-1, 1]$ for $i = 2, 3, \ldots, n$. This problem has a non-convex and non-uniformly spaced Pareto-optimal solutions. The Pareto-optimal solutions correspond to $0 \leq x_1^* \leq 1$ and $x_i^* = 0$ for $i = 2, 3, \ldots, n$. The Euler's method with a step size of $\sigma = 0.15$ along with $\epsilon = 0.001$ is used in SSM. Figure 25 shows the distribution of obtained solution after 100,000 function evaluations. The algorithm is not able to find a well-converged set of solutions. Although there is no local efficient frontier at the location where the algorithm can get stuck, parameters play an important role in the success of SSM and in this problem it is seen that for small values of parameters there is an ascent in functions, instead of a descent in them. A theoretical analysis suggests that at each point $\mathbf{x}$ the direction $(-q(x))$ is a descent direction for all functions, however with a finite step size this result does not hold.

The TPM method with 1,000 initial random solutions produces a set of solutions closer to the efficient frontier, but there are only a few solutions found even after 100,000 function evaluations (Figure 26). For this problem, there is a slow improvement in each iteration and by the time the solution reaches near the efficient frontier, the step size t_i becomes very small and it would take a long time before the solutions fall on the efficient frontier.

The NBI method (Figure 27) performs poorly on this problem. Here, the individual minimum of each objective function are as follows: $(0.3883, 8.8097)$ and $(1, 0)$ for f_1 and f_2, respectively. Since the density of solutions along the frontier is non-uniform, the SQP method along with the NBI strategy is unable to find a good distribution. Similar is the case with NC method and is hence not shown here. On the other hand, with parameters $\eta_c = 10$, $\eta_m = 10$ after 100,000 evaluations NSGA-II is not able to convergence entirely to the efficient frontier (Figure 28, shown with squares). However with parameters $\eta_c = 1$, $\eta_m = 1$ NSGA-II intro-

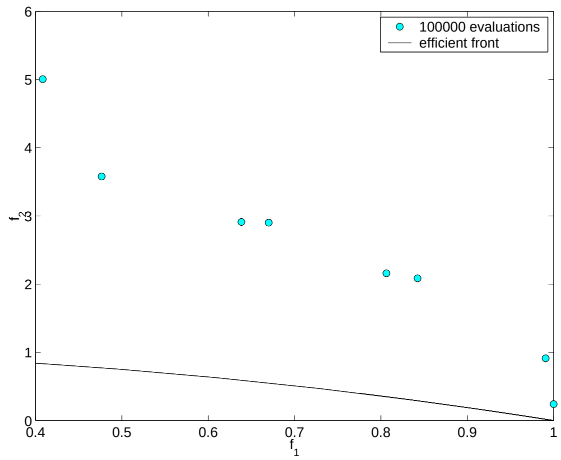

Figure 25: Performance of SSM method on ZDT6.

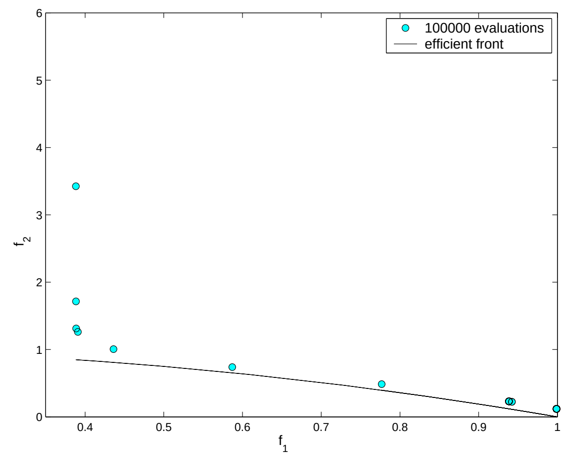

Figure 26: Performance of TPM method on ZDT6.

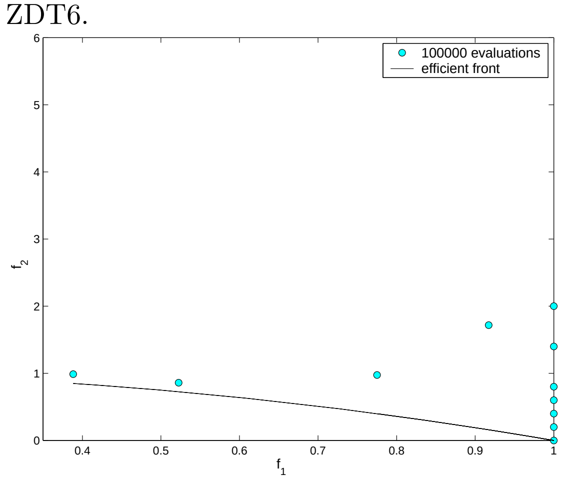

Figure 27: Performance of NBI method on ZDT6.

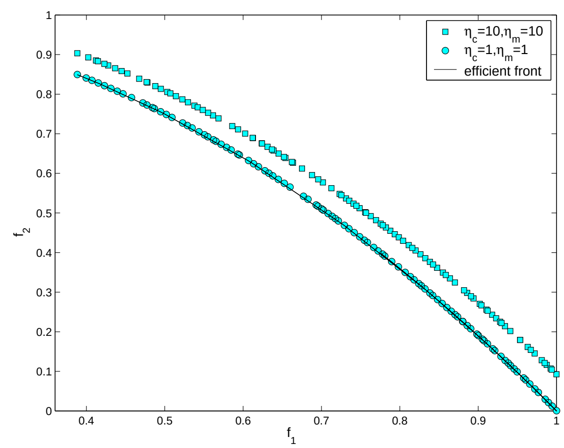

Figure 28: Performance of NSGA-II method on ZDT6.

duces more search, thereby causing NSGA-II to convergence and maintain a good distribution with 100,000 function evaluations (Figure 28, shown with circles).

Based on these simulations, we infer that a non-uniform density of solutions in the objective space (which occurs in many real-world problems [8]) provides enough difficulty to the classical generating methods. This is another class of optimization problems in which an EMO methodology performs comparatively better than the classical methods.

4.2 Three-Objective Test problems

Now, we consider a couple of three-objective test problems developed elsewhere [11] to study the effect of number of objectives with all five algorithms.

4.2.1 DTLZ2 Test Problem:

First, we consider the 12-variable DTLZ2 test problem [11] having a spherical efficient front satisfying $f_1^2 + f_2^2 + f_3^2 = 1$ in the range $f_1, f_2 \in [0, 1]$. The Euler's method with a step size of $\sigma = 0.1$ and $\epsilon = 0.01$ is used in SSM. Figure 29 shows all obtained solutions after 100,000 evaluations. It is clear that the SSM approach is able to get the solutions on the frontier, but the distribution of solutions (obtained mainly by the Brownian approach) is not adequate. It will take enormous number of evaluations for the algorithm to find a distribution across the complete efficient frontier. However as apparent from the figure, the descent property of the search direction requires only a few iterations to reach the efficient frontier.

The Timmel's method (TPM) is applied next with 1,000 initial random solutions. After 20,000 evaluations, the approach is able to find a reasonable coverage of the entire efficient frontier (Figure 30). It is interesting that the boundary solutions are adequately discovered by this approach.

The performance of NBI approach is similar. Here, the individual minimum of each objective function are as follows: $(0, 0.7449, 1.2736)$, $(0.9618, 0, 0.5179)$ and $(1.1516, 0.3886, 0)$ for f_1, f_2 and f_3, respectively. Initial solutions for the SQP sub-problems seem to be an important parameter in the success of these methods. As an example, Figure 31 shows the performance of NBI approach, after 100,000 evaluations, with Pareto-optimal points as the starting points. It finds a few well-distributed solutions. It is to be noted that in the case of DLTZ2, an entire Pareto-optimal curve (on say, f_2-f_3 plane) corresponds to the individual minima of an objective function (say, f_1). Thus it is unlikely, (as shown in Figure 31) the simplex obtained by convex combinations of individual function minima will cover the entire efficient front[1]. It is in this case that size of ideal plane section needs to be enlarged by the approach presented in [31]. Even with such an approach, although the diversity is excellent, the requirement of a large number of evaluations for a high-dimensional objective space is a drawback of this algorithm. Figure 32 shows the performance of the NC approach on this problem. It is seen that the SQP strategy along with the NC method is not able to find most of Pareto-optimal solutions.

The spread of solutions using NSGA-II (with 20,000 evaluations) is shown in Figure 33. Although the distribution is not as regular as in the NBI approach, the obtained solutions spread

[1]This problem is similar to finding the nadir point from individual minima, as discussed well in the classical multi-objective optimization literature [26].

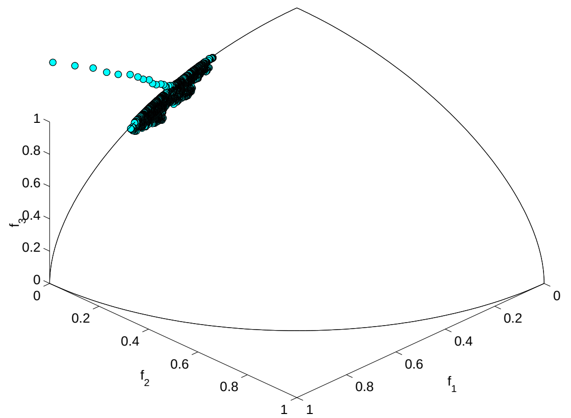

Figure 29: Performance of SSM method on DTLZ2 using 100,000 evaluations.

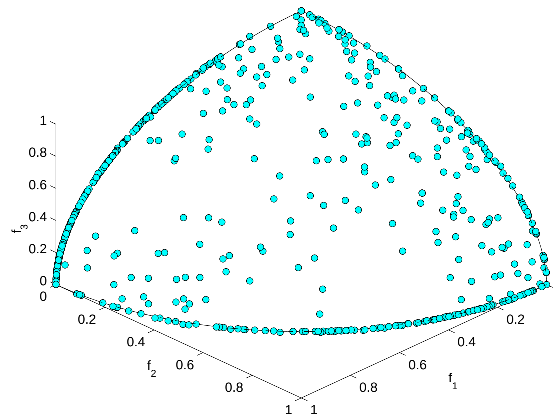

Figure 30: Performance of TPM method on DTLZ2 using 20,000 evaluations.

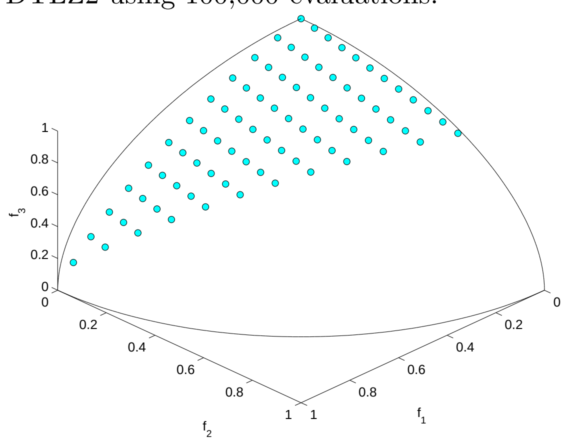

Figure 31: Performance of NBI method on DTLZ2 using 100,000 evaluations.

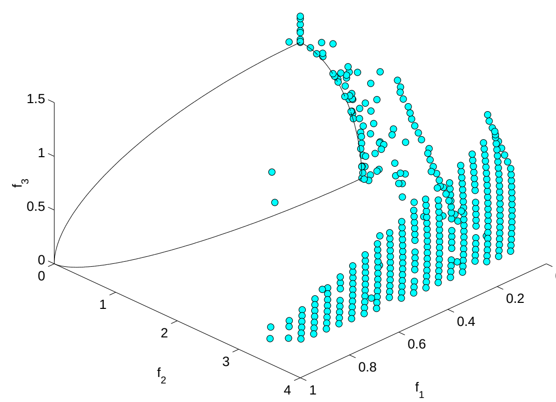

Figure 32: Performance of NC method on DTLZ2 using 100,000 evaluations.

across the entire front. As pointed elsewhere, a better niching operator than the crowding-distance operator, such as a clustered NSGA-II or a ϵ-MOEA [10] or another EMO such as SPEA2 [38] can employ a better distribution of solutions in problems having more than two objectives. The application of clustered NSGA-II on DTLZ2 is shown in Figure 34.

If these algorithms are applied on DTLZ3 which has a number of local efficient frontiers as in ZDT4, the classical algorithms will have similar difficulties in converging to the true efficient frontier. Thus, we do not show the results on DTLZ3.

4.2.2 DTLZ5 Test Problem:

The DTLZ5 is a 12-variable problem having a Pareto-optimal curve: $f_3^2 = 1 - f_1^2 - f_2^2$ with $f_1 = f_2 \in [0, \sqrt{0.5}]$. This problem, although a three-objective one, has a one-dimensional

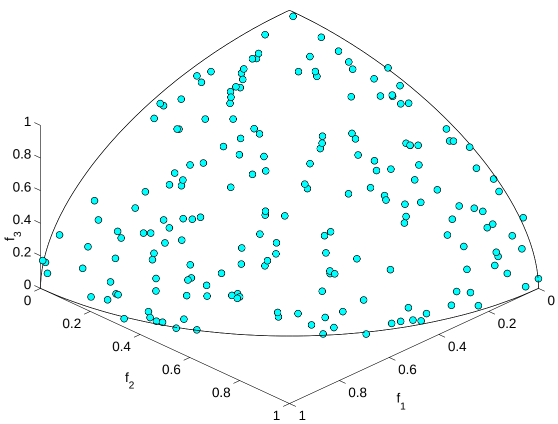

Figure 33: Performance of NSGA-II method on DTLZ2 using 20,000 evaluations.

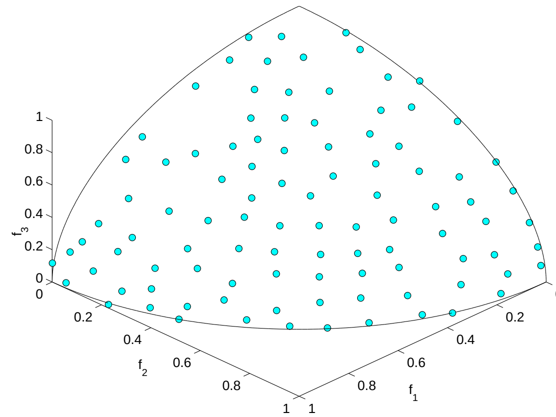

Figure 34: Performance of clustered NSGA-II method on DTLZ2 using 20,000 evaluations.

efficient frontier. The SSM, using Euler's method with a step size of $\sigma = 0.5$ and $\epsilon = 0.01$, finds the partial front after 100,000 evaluations, as shown in Figure 35. The TPM approach (with 500 initial random solutions) finds the complete front, as shown in Figure 36. The NC method is inadequate to find the global optimal solutions. On the other hand, the NBI approach finds a different one-dimensional curve as the efficient frontier (Figure 37). The single-objective minimum solutions obtained in this problem are as follows: $(0, 0, 1.0256)$, $(0, 0, 1.0864)$ and $(0.8962, 1.2517, 0)$ for f_1, f_2 and f_3, respectively. NBI moves along quasi-normal direction from the ideal plane, the intersections of these lines will not necessarily give Pareto-optimal solutions as shown in this example. We explain this matter in the following paragraph.

Consider Figure 39 showing a typical three-objective optimization problem (with an objective space SRQPTS) having a Pareto-optimal *curve*, as shown with a thicker line TP. In this problem, a single-objective minimization of f_1 and f_2 will result in any solution on the line TS. Say, we obtain solution A and B, respectively. Without a loss of generality, we assume these two solutions to be identical here. However, minimization of f_3 will result in any solution in the region PQR. Say, we obtain solution C, as marked in the figure. Then, the ideal plane degenerates to the line AC (or BC). An NBI approach from solutions from this line will result in the solutions shown with filled circles, which are not members of the true Pareto-optimal front (curve TP). Only in the event of special cases (such as the solution C', lying on an extended line OP, obtained as the minimum solution of f_3), true Pareto-optimal solutions will be obtained by the NBI approach. Similarly, if the single-objective minimization of f_1 and f_2 results

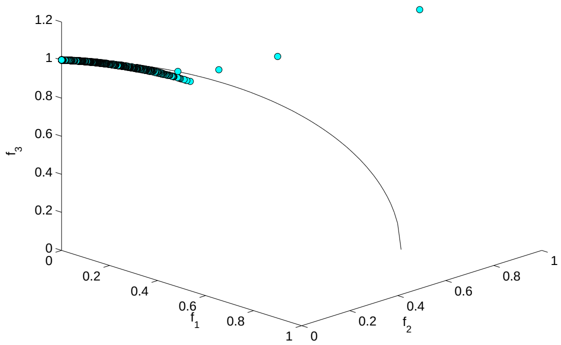

Figure 35: Performance of SSM method on DTLZ5.

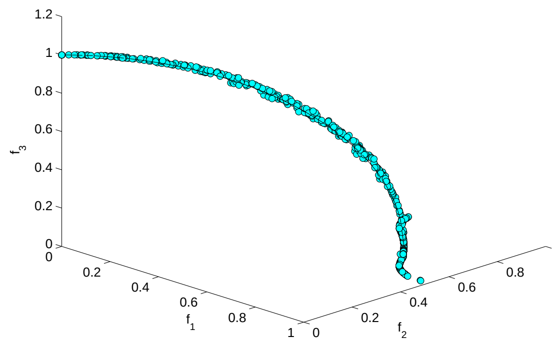

Figure 36: Performance of TPM method on DTLZ5.

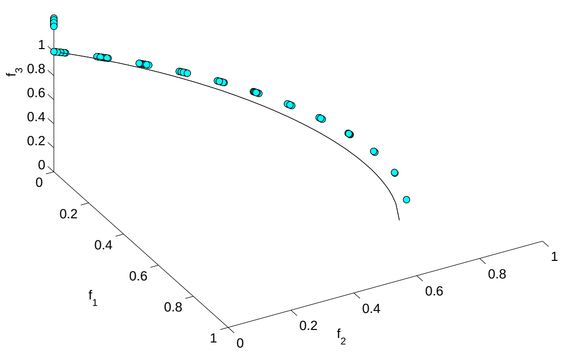

Figure 37: Performance of NBI method on DTLZ5.

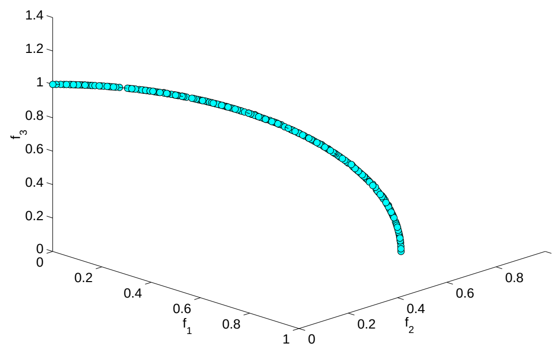

Figure 38: Performance of NSGA-II method on DTLZ5.

in slightly different solutions on the line TS (as in the present case with the SQP approach) the ideal plane is two-dimensional. However its projection on the efficient front would only be a small patch (similar to the case discussed in Figure 4) or the normal direction to the ideal plane will not contain the true Pareto-optimal front entirely (as in this case). Unfortunately, this difficulty is an inherent one for any such higher-dimensional problems, as was discussed in Figure 4 and as apparent here in higher-dimensional problems having a lower-dimensional Pareto-optimal front.

On the other hand, like TPM, NSGA-II (with 20,000 function evaluations) does not have any of these difficulties in finding a good distribution on the true Pareto-optimal frontier, as demonstrated in Figure 38. With NSGA-II, the convergence is better and free of any spurious solutions in comparison to the TPM approach.

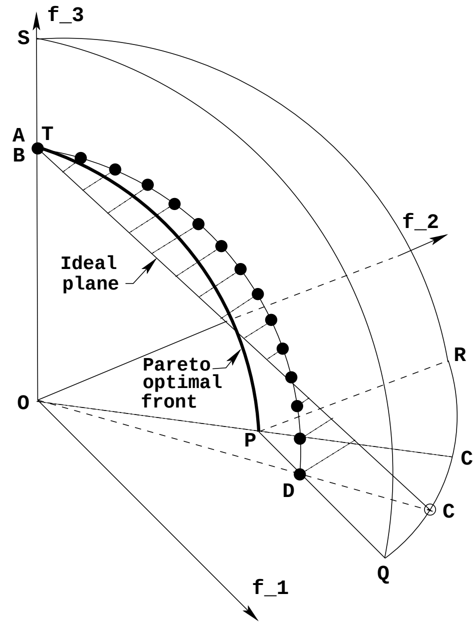

Figure 39: An illustration of a difficulty faced by NBI approach in problems like DTLZ5.

5 Conclusions

This study has brought into light four different classical generating methods which could be used to find a set of Pareto-optimal solutions in a single simulation run, similar to the principle of solving multi-objective optimization problems using evolutionary methods. The comparison of these methods with an evolutionary approach – the NSGA-II — on a number of test problems have adequately demonstrated that these methods perform very well when the problem size and search space complexity are small. Among the four methods, the SSM approach seems to be slow in finding a well-distributed solution set and with a limited number of evaluations the method ends up finding only a part of the entire efficient frontier. However, due to the use of a direction of descent on all objective functions simultaneously, it usually reaches a local efficient front quickly. It is of the view of the authors that such kind of descent information can also be tried to hybridize EMO algorithms for a better local search.

The TPM approach is similar to an elite-preserving population-based EMO approach with an exception that with iterations the population size can increase indefinitely, thereby making the latter iterations slow. The approach also requires fixing a step-size update scheme, which requires fine-tuning for every problem and the real effect has been found to show up while handling multi-modal and more complex optimization problems.

The NBI and NC approaches are systematic mathematical programming approaches in which a number of searches are performed from a uniformly-distributed set of points in the objective space. Although in easier problems, these methods have shown their strengths, particularly when a subproblem is started using the Pareto-optimal solution obtained after solving the previous subproblem, they have clearly showed their weaknesses in solving more complex problems, involving disconnectedness, multi-modality, multi-objectivity, and non-uniform density of solutions in the efficient frontier. It has also been discussed and demonstrated that evolutionary population-based methods (such as NSGA-II and others) are good candidates for finding a set of well-distributed and well-converged efficient solutions due to their implicitly parallel search abilities. On this account, classical iterative single-objective generating approaches are too demanding to solve complex multi-objective optimization problems. Moreover, the optimal solutions obtained using the NBI and NC approaches are not necessarily Pareto-optimal.

On a number of two and three-objective test problems, it has been observed that the TPM, NC and NBI are better approaches than the SSM approach. For higher dimensional problems TPM and NSGA-II have performed the best. However, for problems having multi-modal efficient fronts or non-uniform density of points in the objective space, all four classical methods have not performed well. They get stuck either to a local efficient frontier or to suboptimal solutions. On the other hand, on all problems considered here, the NSGA-II approach with almost an identical parameter setting, has performed well in achieving both convergence and diversity of solutions.

On another note, this study has shown the importance of the ZDT and DTLZ test problems proposed elsewhere [8, 11] in evaluating multi-objective optimization algorithms. Although simpler problems are comparatively easier to solve by almost all methodologies, algorithms were put to real test when ZDT3, ZDT4, ZDT6 and three dimensional DTLZ problems were tried.

With a clear demonstration of the strengths and weaknesses of classical and evolutionary multi-objective optimization algorithms in this study, it is time to recombine the positive aspects of two approaches and create new and more powerful hybrid optimization methods. Different avenues are possible. One way forward would be to replace the SQP or the single-objective optimization approach embedded to these classical algorithms with an evolutionary algorithm, thereby providing the global approach in the search. Another approach would be to use some of the classical directional search principles as additional operators in an EMO methodology for better convergence properties. Some such extensions would be an immediate focus for useful research and application in the area of efficient multi-objective optimization.

References

[1] R. M. Beeson and W. S. Meisel. The optimization of complex systems with respect to multiple criteria. In *IEEE Systems, Man and Cybernetics Group Annual Symposium*, pages 144–149, 1971.

[2] H. P. Benson. An outer approximation algorithm for generating all efficient exteme points in the outcome set of a multiple objective linear programming problem. *Journal of Global Optimization*, 13:1–24, 1998.

[3] H.P. Benson and S. Sayin. Towards Finding Global Representation of the Efficient Set in Multiple Objective Mathematical Programming. *Naval Research Logistics*, 44:47–67, 1997.

[4] C. A. C. Coello, D. A. VanVeldhuizen, and G. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems.* Boston, MA: Kluwer Academic Publishers, 2002.

[5] I. Das and J.E. Dennis. A closer look at drawbacks of minimizing weighted sum of objecties for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14(1):63–69, 1997.

[6] I. Das and J.E. Dennis. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal of Optimization*, 8(3):631–657, 1998.

[7] I. Das and J.E. Dennis. An improved technique for choosing parameters for Pareto surface generation using normal-boundary intersection. In *Proceedings of the Third World Congress on Structutal and Multidisciplinary Optimization*, 1999.

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

[9] 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.

[10] K. Deb, M. Mohan, and S. Mishra. Towards a quick computation of well-spread pareto-optimal solutions. In *Proceedings of the Second Evolutionary Multi-Criterion Optimization (EMO-03) Conference (LNCS 2632)*, pages 222–236, 2003.

[11] 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.

[12] L.C.W. Dixon and G. P. Szegö. The global optimization problem: An introduction. In *Towards Global Optimization, Vol 2*, pages 1–15. North-Holland, Amsterdam, 1978.

[13] M. Ehrgott. *Multicriteria Optimization*. Berlin: Springer, 2000.

[14] G. Fadel and Y. Li. Approximating the Pareto curve to help solve bi-objective design problems. *Structural and Multidisciplinary Optimization*, 23:280–296, 2002.

[15] J. Fliege and A. Heseler. Constructing approximations to the efficient set of convex quadratic multiobjective programs. Technical report, Ergebnisberichte in Angewandte Mathematik, Fachbereich Mathematik, Universität Dortmund, 2002.

[16] 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.

[17] C. Hillermeier. Generalized homotopy approach to multiobjective optimization. *Journal of Optimization Theory and Applications*, 110(3):557–583, 2001.

[18] C. Hillermeier. *Nonlinear Multiobjective Optimization*. Birkhauser Verlag, 2001.

[19] C. L. Hwang and K. Yoon. Multiple objective decision making, methods and applications: A state-of-the-art survey. In M. Beckmann and H. P. Kunzi, editors, *Lecture Notes in Economics and Mathematical Systems 169*. Springer-Verlag, Berlin, 1979.

[20] H. Ishibuchi and T. Murata. A multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and reviews*, 28(3):392–403, 1998.

[21] E. K. Karasakal and M. Köksalan. Generating a Representative Subset of the Efficient Frontier in Multiple Criteria Decision Making. Technical Report Working Paper 01-20, Faculty of Administration, University of Ottawa, Ottawa, Ontario, Canada, 2001.

[22] I. Y. Kim and O.de Weck. Multiobjective optimization via the adaptive weighted sum method. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2004.

[23] I. Y. Kim and O.de Weck. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and Multidisciplinary Optimization*, 29(2):149–158, 2005.

[24] J. Kim and S. K. Kim. A CHIM-based interactive Tchebycheff procedure for multiple objective decision making. *Computers and Operations Research*, In Press.

[25] K. Klamroth, J. Tind, and M. M. Wiecek. Unbiased approximation in multicriteria optimization. *Mathematical Methods of Operations Research*, 56:413–437, 2002.

[26] P. Korhonen, S. Salo, and R. Steuer. A heuristic for estimating nadir criterion values in multiple objective linear programming. *Operations Research*, 45(5):751–757, 1997.

[27] Y.I. Lim, P. Floquet, and X. Joulia. Efficient Implementation of the Normal Boundary Intersection(NBI) Method on Multiobjective Optimization Problems. *Industrial and Engineering Chemistry Research*, 40:648–655, 2001.

[28] T. Marler and J. Arora. Review of multi-objective optimization concepts and algorithms for engineering. Technical Report ODL-01.03, University of Iowa, Optimal Design Laboratory, 2003.

[29] A. Messac. Physical Programming: Effective optimization for computational design. *AIAA Journal*, 34(1):149–158, 1996.

[30] A. Messac and C. A. Mattson. Generating well-distributed sets of Pareto points for engineering design using Physical Programming. *Optimization and Engineering*, 3(4):431–450, 2002.

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

[32] K. Miettinen. *Nonlinear Multiobjective Optimization.* Kluwer, Boston, 1999.

[33] S. Schäffler, R. Schultz, and K. Weinzierl. Stochastic method for the solution of unconstrained vector optimization problems. *Journal of Optimization Theory and Applications*, 114(1):209–222, 2002.

[34] P. K. Shukla, K. Deb, and S. Tiwari. Comparing Classical Generating Methods with an Evolutionary Multi-objective Optimization Method. In C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler, editors, *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005*, pages 311–325, Guanajuato, México, March 2005. Springer. Lecture Notes in Computer Science Vol. 3410.

[35] G. Timmel. Ein stochastisches Suchverrahren zur Bestimmung der optimalen Kompromilsungen bei statischen polzkriteriellen Optimierungsaufgaben. *Wiss. Z. TH Ilmenau*, 26(5):159–174, 1980.

[36] G. Timmel. Modifikation eines statistischen Suchverfahrens der Vektoroptimierung. *Wiss. Z. TH Ilmenau*, 28(6):139–148, 1982.

[37] A. A. Törn. A sampling-search-clustering approach for exploration of the feasible/efficient solutions of MCDM problems. *Computers and Operations Research*, 7:67–69, 1980.

[38] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, editors, *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pages 95–100, Athens, Greece, 2001. International Center for Numerical Methods in Engineering (Cmine).

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