Journal of Computational and Applied Mathematics 261 (2014) 158–171

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A numerical method for constructing the Pareto front of multi-objective optimization problems

E. Khorram [a,*], K. Khaledian [a], M. Khaledyan [b]

[a] Faculty of Mathematics and Computer Sciences, Amirkabir University of Technology, 424, Hafez Avenue, 15914 Tehran, Iran

[b] Faculty of Electrical and Computer Engineering, K.N.Toosi University of Technology, Seyedkhandan Bridge, beside the Ministry of Communication, Tehran, Iran

ARTICLE INFO

Article history:
Received 8 August 2011
Received in revised form 26 January 2013

Keywords:
Multi-objective optimization
Pareto front
Pascoletti and Serafini scalarization
Nonlinear optimization
Nonconvex optimization

ABSTRACT

In this paper, a new numerical method is presented for constructing an approximation of the Pareto front of multi-objective optimization problems. This method is based on the well-known scalarization approach by Pascoletti and Serafini. The proposed method is applied to four test problems that illustrate specific difficulties encountered in multi-objective optimization problems, such as nonconvex, disjoint and local Pareto fronts. The effectiveness of the proposed method is demonstrated by comparing it with the NSGA-II algorithm and the Normal Constraint method.

1. Introduction

Multi-objective optimization is the process of simultaneously optimizing two or more objectives subject to certain constraints.

We consider a multi-objective optimization problem (MOP) as follows:

$$\textbf{\textit{MOP}}: \min_{x\in X} f(x) = (f_1(x), f_2(x), \dots, f_p(x)), \tag{1}$$

where $X \subset \mathbb{R}^n$ is a nonempty feasible set, and f is a vector-valued function composed of p $(p \geq 2)$ real-valued functions. The image of X under f is denoted by $Y := f(X) \subseteq \mathbb{R}^p$ and referred to as the image space.

For $y, \hat{y} \in \mathbb{R}^p$,

- $y < \hat{y}$ means $y_k < \hat{y}_k$ for all $k = 1, \dots, p$,
- $y \leqq \hat{y}$ means $y_k \leq \hat{y}_k$ for all $k = 1, \dots, p$,
- $y \leq \hat{y}$ means $y \leqq \hat{y}$ but $y \neq \hat{y}$.

In this paper, we use the above componentwise orders to order the objective space and define the cone $\mathbb{R}^p_{\geqq} = \{x \in \mathbb{R}^p | x \geqq 0\}$.

Definition 1.1. A feasible solution $\hat{x} \in X$ is called an efficient solution of MOP (1) if there is no $x \in X$ such that $f(x) \leq f(\hat{x})$. If $\hat{x} \in X$ is efficient then $f(\hat{x})$ is called a nondominated point.

Definition 1.2. A feasible solution $\hat{x} \in X$ is called a weakly efficient solution of MOP (1) if there is no $x \in X$ such that $f(x) < f(\hat{x})$. If $\hat{x} \in X$ is weakly efficient then $f(\hat{x})$ is called a weakly nondominated point.

* Corresponding author. Tel.: +98 021 6406322; fax: +98 021 649793.
E-mail addresses: eskhor@aut.ac.ir (E. Khorram), khaledian.k@aut.ac.ir (K. Khaledian), mkh.1986@gmail.com (M. Khaledyan).

0377-0427/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cam.2013.11.007

The set of all efficient and weakly efficient solutions of MOP (1) are denoted by X_E and X_{WE}, respectively, and their images are called Pareto front and weak Pareto front, denoted by Y_N and Y_{WN}, respectively.

Definition 1.3. The point $f^* = (f_1^*, \dots, f_p^*)^T$, where $f_i^* = \min_{x \in X} f_i(x)$ for $i = 1, \dots, p$, is called the Ideal point of MOP (1).

A common approach for determining the solutions of a MOP is reformulating it to a parameters dependent scalar optimization problem. By solving the scalar problem for a variety of parameters, several solutions of the MOP are generated. For a survey on scalarizing (and non-scalarizing) techniques, the reader is referred to [1]. There has been a great deal of effort by researchers in developing methods to generate an approximation of the Pareto front, see e.g. [2–11]. In most of these works, various types of scalarization are considered. In this paper, we concentrate on the Pascoletti and Serafini scalarization [12], and present a numerical method based on it for constructing an approximation of the Pareto front of general MOPs (nonlinear and nonconvex). An advantage of the Pascoletti and Serafini scalarization is that it is very general in the sense that many other scalarization approaches such as the weighted Tchebycheff method and the ε-constraint method can be seen as a special case of it; see [6]. Further, it is defined for general partial orderings rather than the natural ordering [12]. Our proposed method shares conceptual similarity with the approach proposed in [6], in its use of the Pascoletti and Serafini scalarization.

The Pascoletti and Serafini scalarization has two parameters named a and r, selected from $\mathbb{R}^p$. The method proposed in [6] attempts to limit the choices of a from $\mathbb{R}^p$, and still obtain all the efficient solutions. This method might be hard to verify in practice, since to implement the approach in [6], one needs to obtain a restricted hyperplane in the objective space where the parameter a changes on it. Obtaining this restricted hyperplane is simple for two objective problems, but in three and more than three objective problems is hard in practice. In comparison with the method in [6], our method can locate the Pareto front quite easily in most cases. In Section 2.2 the method in [6] is briefly discussed (for more information see [6, pp. 31–47]).

We proceed as follows. In Section 2, the Pascoletti and Serafini scalarization is briefly reviewed and some of its properties are given. In Section 3, the new parameter restriction for the Pascoletti and Serafini scalarization is proposed, and based on it the Pareto front is approximated. In Section 4, the proposed method is applied to four test problems and the results are compared with the results from NSGA-II and NC methods. Finally, conclusions are presented in Section 5.

2. Pascoletti and Serafini scalarization

In this section, a brief review of the Pascoletti and Serafini scalarization [12] is given. Pascoletti and Serafini propose the following scalar optimization problem with parameters $a, r \in \mathbb{R}^p$, in order to characterize weakly efficient and efficient solutions of MOP (1) w.r.t. the ordering cone K:

$$\begin{aligned}
&\min t\\
&\text{s.t.}\\
&a + tr - f(x) \in K, \quad (SP(a,r))\\
&x \in X,\\
&t \in \mathbb{R}.
\end{aligned}$$

In this paper we assume $K = \mathbb{R}^p_{\geqq}$.

In order to solve the $SP(a, r)$ problem, the ordering cone $-\mathbb{R}^p_{\geqq}$ is moved in direction r (or $-r$) on the line $a + tr$ starting in the point a until the set $(a + tr - \mathbb{R}^p_{\geqq}) \bigcap f(X)$ is reduced to the empty set. The smallest value $\bar{t}$ for which $(a + \bar{t}r - \mathbb{R}^p_{\geqq}) \bigcap f(X) \neq \emptyset$ is the minimal value of $SP(a, r)$ [6]. This is illustrated in Fig. 1 for a two-objective problem.

2.1. Properties of the Pascoletti and Serafini scalarization

The Pascoletti and Serafini scalarization is very general, in the sense that many other scalarization approaches such as the ε-constraint method and the weighted Tchebycheff method can be seen as a special case of it; see [6]. For example, the formulation of the weighted Tchebycheff method is as follows:

$$\min_{x \in X} \max_{i=1,\dots,p} w_i(f_i(x) - a_i),$$

where $w_i > 0$ for $i = 1, \dots, p$ and $a \in \mathbb{R}^p$. If we set $K = \mathbb{R}^p_{\geqq}$ and $r_i = \frac{1}{w_i}$, then the weighted Tchebycheff method and the Pascoletti and Serafini scalarization coincide.

In the following, Section b of Theorem 2.1 in [6] is given.

Theorem 2.1. *Let $\hat{x}$ be an efficient solution of MOP (1), then $(0, \hat{x})$ is an optimal solution of $SP(a, r)$ for the parameters $a = f(\hat{x})$ and arbitrary $r \in \mathbb{R}^p_{\geqq} \setminus \{0\}$.*

Remark. Note that, as mentioned in [6], it is possible that for some parameters $a, r \in \mathbb{R}^p$ the problem $SP(a, r)$ is unbounded from below.

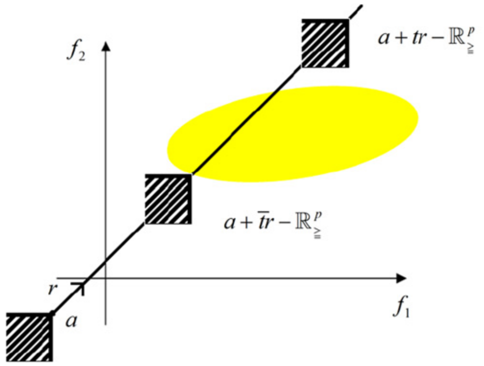

Fig. 1. Finding minimal value of $SP(a, r)$ problem.

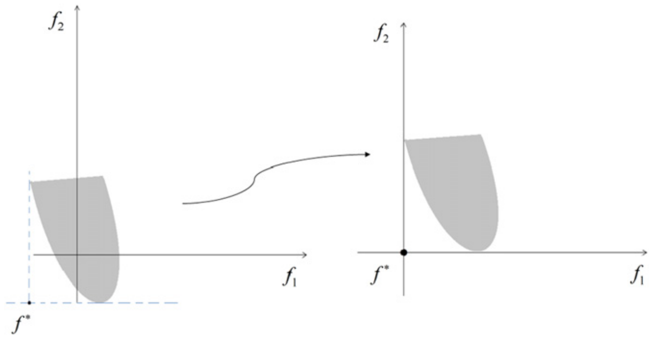

Fig. 2. The Ideal point is shifted to the origin.

2.2. Restriction of the parameter a

As a result of Theorem 2.1, we can find all efficient solutions of MOP (1) for a constant parameter $r \in \mathbb{R}^p_{\geq} \setminus \{0\}$ by varying the parameter $a \in \mathbb{R}^p$ only. In [6], a method is used to limit the choices of a from $\mathbb{R}^p$, and still obtain all the efficient solutions for MOPs that have the Ideal point. In the following, this method is briefly reviewed.

Restriction of the parameter a in the two-objective case. For the two-objective case, consider $r \in \mathbb{R}^2 \setminus \{0\}$ and the hyperplane $H = \{y \in \mathbb{R}^2 | b_1 y_1 + b_2 y_2 = \beta\}$ with $b \in \mathbb{R}^2 \setminus \{0\}$ and $\beta \in \mathbb{R}$. First, the points $\bar{x}_1 = \text{argmin}_{x\in X} f_1(x)$ and $\bar{x}_2 = \text{argmin}_{x\in X} f_2(x)$ are determined, then the points $f(\bar{x}_1)$ and $f(\bar{x}_2)$ are projected in direction r onto the line H. The projection points $\bar{a}_1 \in H = \{y \in \mathbb{R}^2 | b_1 y_1 + b_2 y_2 = \beta\}$ and $\bar{a}_2 \in H$ are given by

$$\bar{a}_i := f(\bar{x}_i) - r\frac{b^T f(\bar{x}_i) - \beta}{b^T r}, \quad i = 1, 2.$$

Theorem 2.17 in [6] shows that it is sufficient to consider parameters $a \in H^a$ with the set H^a given by $H^a = \{y \in H | y = \lambda \bar{a}_1 + (1-\lambda)\bar{a}_2, \lambda \in [0, 1]\}$ to approximate the whole Pareto front.

Restriction of the parameter a in the general case. For three or more objective cases the above method is not correct. In [6] this is shown by Example 2.19. So [6] proposed a weaker restriction of the set H for the parameter a, by projecting the image set $f(X)$ in direction r onto the set H. Thus the set $\tilde{H} := \{y \in H | y + tr = f(x), t \in \mathbb{R}, x \in X\} \subset H$ is determined. The set $\tilde{H} \subset H$ has in general an irregular boundary and is therefore not suitable for a systematic procedure. Hence in [6] the set $\tilde{H}$ is embedded in a $(p-1)$-dimensional cuboid $H^0 \subset \mathbb{R}^p$ which is chosen as minimal as possible. The method of obtaining H^0 is given in [6], which is difficult in practice.

3. A new parameter restriction for the Pascoletti and Serafini scalarization

In this section, a new parameter restriction for the Pascoletti and Serafini scalarization is proposed. To this end, we assume that MOP (1) has the Ideal point and we proceed as follows.

First, in order to determine the Ideal point, the problems $\min_{x\in X} f_i(x)$ for $i = 1, \ldots, p$ are solved. Let $f_i^*, \ i = 1, \ldots, p$ be the respective optimal value of $\min_{x\in X} f_i(x)$. The Ideal point is denoted by $f^* = (f_1^*, \ldots, f_p^*)^T$.

Then, the objective functions are redefined as $f(x) \longleftarrow f(x) - f^*$, i.e., the Ideal point is shifted to the origin. This is shown for a two-objective problem in Fig. 2. In the rest of this section it is assumed that the Ideal point is equal to zero and the

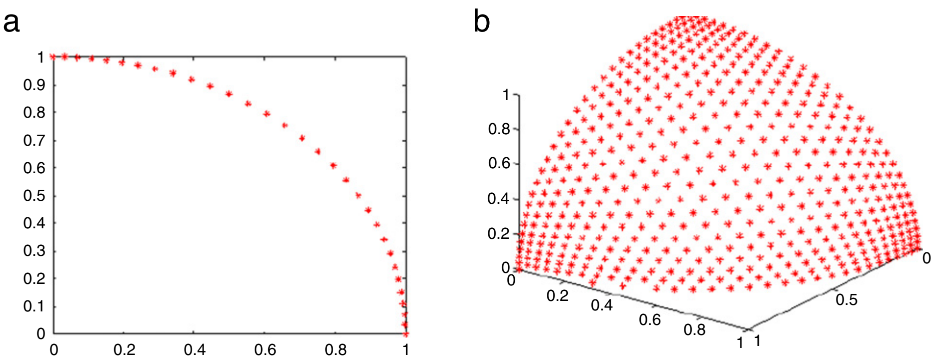

Fig. 3. The set R with $\delta = \frac{1}{30}$ for two-objective (a) and three-objective (b) problems.

objective functions are nonnegative. Then the set $R = \{\frac{\beta}{\|\beta\|_2} | \beta \in \mathbb{R}^p, \sum_{i=1}^p \beta_i = 1, \beta_i \geq 0\}$ is defined ($\|\cdot\|_2$ is the Euclidean norm).

Now, we assume that the parameter a is equal to zero, and the parameter r is chosen from the set R, i.e., each point on the set R can be considered as a direction r. So the direction r is defined as $r = \frac{\beta}{\|\beta\|_2}$ for some $\beta \in \mathbb{R}^p$ for which $\sum_{i=1}^p \beta_i = 1$ and $\beta_i \geq 0$. A strategy to generate an even spread of combinations vectors β has been proposed in [3]. In this strategy, the values of each component of β are simply $0, \delta, 2\delta, \ldots, 1$, which $\delta \leq 1$ is a fixed stepsize, such that δ^{-1} is a nonnegative integer. This even spread of β generates even spaced points on the set R. In two-objective problems the set R is the first quarter of the unit circle $\sum_{i=1}^2 y_i^2 = 1$ and in three-objective problems it is the first octant of the unit sphere $\sum_{i=1}^3 y_i^2 = 1$. The set R with $\delta = \frac{1}{30}$ is shown for two-objective and three-objective cases in Fig. 3(a) and (b), respectively.

In the following theorem, it is shown that by considering $a = 0$ and varying $r \in R$, all efficient solutions of MOP (1) can be found.

Theorem 3.1. *Suppose that $\bar{x} \in X$ is an efficient solution of MOP (1). Let $a = 0$ and $R = \{\frac{\beta}{\|\beta\|_2} | \beta \in \mathbb{R}^p, \sum_{i=1}^p \beta_i = 1, \beta_i \geq 0\}$, then there are a parameter $r \in R$ and some $\bar{t} \in \mathbb{R}$ such that $(\bar{t}, \bar{x})$ is an optimal solution of $SP(a, r)$. This holds for instance for*

$$\begin{cases} (\bar{t}, a, r) = \left(\|f(\bar{x})\|_2, 0, \dfrac{f(\bar{x})}{\|f(\bar{x})\|_2} \right) & \text{if } f(\bar{x}) \neq 0 \\ (\bar{t}, a, r) = (0, 0, r \in R) & \text{if } f(\bar{x}) = 0. \end{cases}$$

Proof. In the case $f(\bar{x}) \neq 0$, it is obvious that $\frac{f(\bar{x})}{\|f(\bar{x})\|_2} \in R$, because by choosing $\beta_i = \frac{f_i(\bar{x})}{\sum_{i=1}^p f_i(\bar{x})}$ for each $i = 1, \ldots, p$, we have $\sum_{i=1}^p \beta_i = 1$ and $\beta_i \geq 0$ for $i = 1, \ldots, p$. (Note that it is assumed that all the objective functions are nonnegative.) Considering $r = \frac{f(\bar{x})}{\|f(\bar{x})\|_2} = \frac{\beta}{\|\beta\|_2} \in R$, $\bar{t} = \|f(\bar{x})\|_2$ and $a = 0$, we have

$$a + \bar{t}r - f(\bar{x}) = 0 + \|f(\bar{x})\|_2 \frac{f(\bar{x})}{\|f(\bar{x})\|_2} - f(\bar{x}) = 0 \in \mathbb{R}^p_{\geqq}.$$

So, the point $(\bar{t}, \bar{x})$ is feasible for the $SP(a, r)$ problem. Suppose to the contrary that $(\bar{t}, \bar{x})$ is not an optimal solution of the $SP(a, r)$ problem. Then there exists a $\acute{t} \in \mathbb{R}$ with $\acute{t} < \bar{t}$ and a point $\acute{x} \in X$ such that $(\acute{t}, \acute{x})$ is an optimal solution of $SP(a, r)$. Considering $a = 0$, $r = \frac{f(\bar{x})}{\|f(\bar{x})\|_2} = \frac{\beta}{\|\beta\|_2} \in R$ and $\acute{t} < \bar{t} = \|f(\bar{x})\|_2$, we have

$$\left(a + \acute{t} \frac{f(\bar{x})}{\|f(\bar{x})\|_2} - f(\acute{x}) \right) = \left(0 + \frac{\acute{t}}{\bar{t}} f(\bar{x}) - f(\acute{x}) \right). \tag{2}$$

From the feasibility of $(\acute{t}, \acute{x})$ for $SP(a, r)$, we have

$$\left(a + \acute{t} \frac{f(\bar{x})}{\|f(\bar{x})\|_2} - f(\acute{x}) \right) \geq 0. \tag{3}$$

Since $\bar{t} = \|f(\bar{x})\|_2 > 0$ and $\acute{t} < \bar{t}$, we have $\frac{\acute{t}}{\bar{t}} < 1$. So, from relation (2), we get

$$\left(0 + \frac{\acute{t}}{\bar{t}} f(\bar{x}) - f(\acute{x}) \right) < f(\bar{x}) - f(\acute{x}). \tag{4}$$

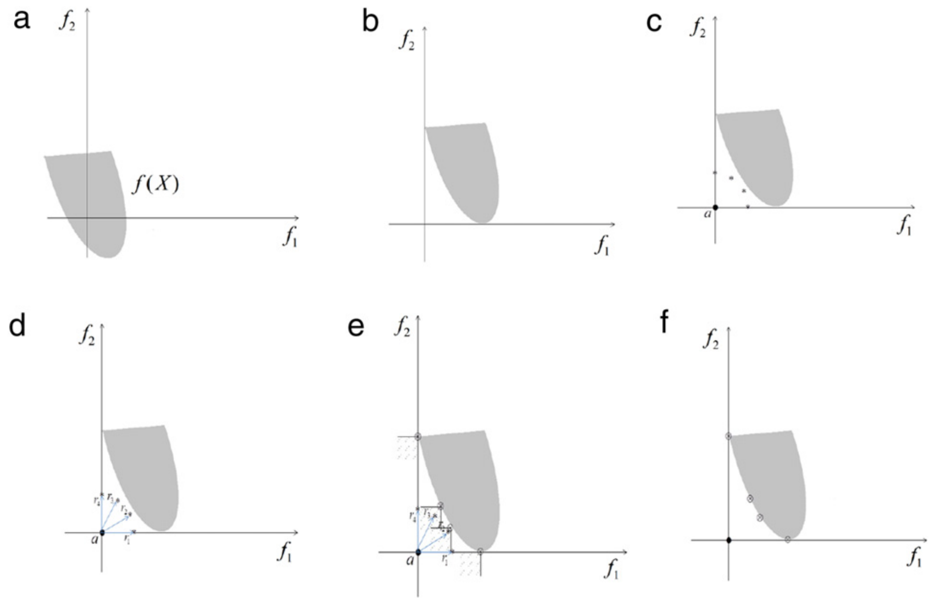

Fig. 4. Four iterations of Algorithm 1 for a two-objective problem. (a) Image space. (b) The Ideal point is shifted to the origin. (c) The set R (symbols $*$) and $a = 0$. (d) The directions $r \in R$. (e) Solving $SP(a = 0, r)$ for r_1, r_2, r_3 and r_4. (f) Symbols $\otimes$ are the obtained nondominated points.

From relations (3) and (4)

$$f(\bar{x}) - f(\acute{x}) > 0.$$

Hence, $f(\bar{x}) > f(\acute{x})$, and this is in contradiction to $\bar{x}$ being an efficient solution of MOP (1).

The proof of the case $f(\bar{x}) = 0$ is trivial. □

Now, based on the proposed parameter restriction, a numerical method for approximating the Pareto front is given. This method is described with an algorithm, namely Algorithm 1. Algorithm 1 is an iterative algorithm that constructs sets of points which approximates the Pareto front Y_N. At each iteration, the set of nondominated points (with respect to all points generated so far) is denoted by Y_L, and Y_L gives an approximation of the Pareto front Y_N.

At the initialization step, the Ideal point is found and shifted to the origin, also, Y_L is set empty.

Each iteration of Algorithm 1 consists of three steps.

The first step consists of determining the parameters a and r for the Pascoletti and Serafini scalarization. In this step we set $a = 0$ and $r \in R$.

The second step of the iteration consists of forming the $SP(a, r)$ problem using the direction $r \in R$ and $a = 0$, then solving it.

Finally, at the end of each iteration the set of nondominated points Y_L is updated, new nondominated points are added and dominated ones are removed.

In Fig. 4, four iterations ($\delta = \frac{1}{3}$) of Algorithm 1 are shown for a two-objective optimization problem.

The general scheme of Algorithm 1 is as follows.

Algorithm 1

Input: the multi-objective optimization problem.
Output: The set Y_L which is an approximation of the Pareto front.

Initialization:
– Let $f_i^* = \min_{x \in X} f_i(x)$ for $i = 1, \dots, p$ and $f^* = (f_1^*, \dots, f_p^*)^T$ being the Ideal point.
– Define $f(x) \longleftarrow f(x) - f^*$.
– Let $Y_L = \emptyset$.
– Initialize the value of δ to generate an even spread of combinations vectors β [3].
– Define the set $R = \{\frac{\beta}{\|\beta\|_2} | \beta \in \mathbb{R}^p, \sum_{i=1}^{p} \beta_i = 1, \beta_i \geq 0\}$ and let $a = 0$.

Main iteration repeat:
– Determine direction $r \in R$.
– Forming the $SP(a, r)$ problem using $r \in R$ and $a = 0$, then solving it.
– Update Y_L: Add to Y_L all nondominated points found in the current iteration, remove dominated ones from Y_L.

4. Numerical results

In this section, the performance of Algorithm 1 is evaluated by four test problems from [13,14]. These problems test the ability of an algorithm to overcome different landscape complexities and introduce specific difficulties in both converging to the Pareto front and maintaining a well spread distribution of the efficient solutions. The results from Algorithm 1 are compared with the results from the NSGA-II algorithm [4] and the Normal Constraint (NC) method [15], for all test problems. In Sections 4.4 and 4.5 a brief review of the NSGA-II algorithm and NC method are given, respectively. Also to measure the quality of the distribution of approximation points, measures of coverage [15] and spacing [16] are used. These measures are briefly discussed in Section 4.1. Results obtained by Algorithm 1, NSGA-II and NC method are compared in terms of the dominance of the obtained solutions (Definition 1.1), the number of function evaluations, CPU time required for solving each test problem and measures of coverage and spacing. Also, visualization plots are used to illustrate the distribution of solutions on the Pareto front. Moreover, to solve the scalar problems $SP(a, r)$ in Algorithm 1, we use the SQP algorithm in Matlab fmincon solver.

4.1. Measures of coverage and spacing

Measure of coverage examines whether all regions of the Pareto front are represented. In this paper the measure of coverage called "extension" [16] is used. Suppose $f^* = (f_1^*, \ldots, f_p^*)^T$ to be the Ideal point. Given a discrete representation of Y_N, we calculate the distance between each element of the Ideal point and Y_N as follows:

$$d(f_i^*, Y_N) = \min\{d(f_i^*, y) | y \in Y_N\}.$$

Finally, the extension is calculated as follows:

$$EX(Y_N) = \frac{\sqrt{\sum_{i=1}^{p} (d(f_i^*, Y_N))^2}}{p}.$$

For this measure, a small value is preferred to a larger value because the larger value could indicate that the representation is mostly in the center of the true Pareto front, with the surroundings being neglected.

Measure of spacing determines the distance between points in the representation. A representation with uniform spacing is desirable. Having equidistant points does not guarantee good coverage, so measure of spacing should be used in conjunction with a coverage measure. In this paper the measure of spacing called "evenness" [15] is used. For each point, y^i, in the discrete representation, two (hyper)spheres are constructed: the smallest and the largest spheres that can be formed between y^i and any other point in the set such that no other points are within the spheres. The diameters of the two spheres are denoted by d_l^i and d_u^i, respectively. The evenness, ξ, of a representation is then calculated with the following formula:

$$\xi = \frac{\sigma_d}{\hat{d}},$$

where $\hat{d}$ and σ_d are the mean and standard deviation of the set of minimum and maximum diameters for each point in the representation, respectively. A discrete representation with all points spaced equidistantly will have $\xi = 0$ because the d_l^i and d_u^i will all be equal (i.e., $\sigma_d = 0$).

4.2. Two-objective problems

Here, the test problems ZDT3 and ZDT2 are considered [13].

Modified ZDT3 test problem.

The modified ZDT3 test problem can be stated as follows:

$$\min f_1(x) = x_1,$$

$$\min f_2(x) = g(x)\left(1 - \left(\frac{f_1(x)}{g(x)}\right)^{0.5} - \frac{f_1(x)}{g(x)}\sin(10\Pi x_1)\right),$$

$$\text{where } g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i^2,$$

$$x_1 \in [0, 1] \quad \text{and} \quad x_i \in [-1, 1]\ \forall i = 2, 3, \ldots, n.$$

We use $n = 30$. This problem has a discontinuous Pareto front. The Pareto front corresponds to $0 \le x_1^* \le 1$ and $x_i^* = 0$ for $i = 2, \ldots, n$. The problem is solved by Algorithm 1 and the NC method with $\delta = \frac{1}{150}$ and NSGA-II with population of size 150 and generation of size 200. The convergence to the Pareto front and the distribution of solutions obtained by Algorithm 1

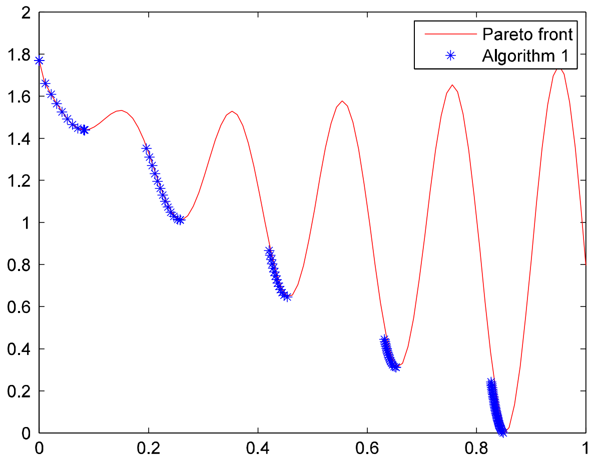

Fig. 5. Algorithm 1 with 292 function evaluations.

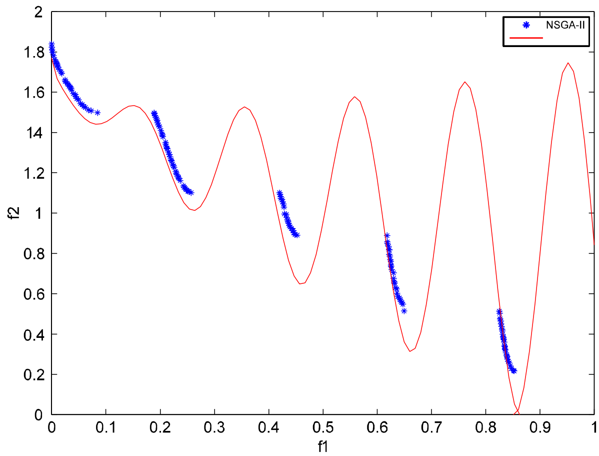

Fig. 6. NSGA-II with 30 000 function evaluations.

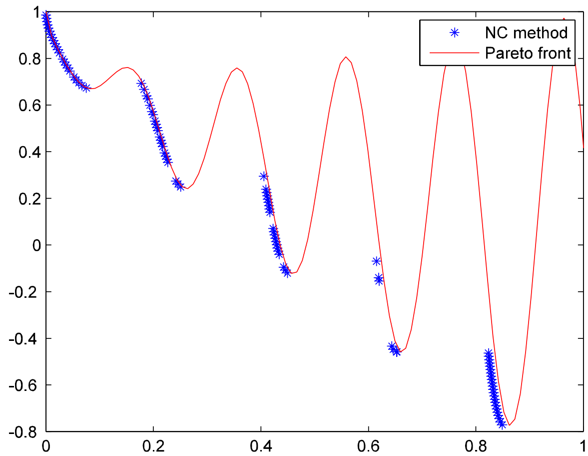

Fig. 7. NC method with 286 function evaluations.

after 292 function evaluations, NSGA-II after 30 000 function evaluations and NC method after 286 function evaluations for this problem are depicted in Figs. 5–7, respectively.

The results of solving ZDT3 by Algorithm 1, NSGA-II and NC method are presented in Table 1. Solutions obtained by Algorithm 1 dominate 87 and 33 solutions obtained by NSGA-II and NC methods, respectively, while none of the solutions

Table 1
Number of function evaluations, CPU time for solving ZDT3, measures of coverage and spacing and the dominance between solutions obtained by Algorithm 1, NSGA-II and NC method for ZDT3.

Method	*ZDT3 Problem*			
	Number of function evaluations	CPU time (s)	EX	ξ
Algorithm 1	292	17.1179	5.0000e−004	0.0913
NSGA-II	30 000	107.6434	0.3795	0.0583
NC method	286	17.334485	0.001	0.1096

Solutions obtained by Algorithm 1 dominate **87** solutions obtained by NSGA-II.
None of the solutions obtained by Algorithm 1 are dominated by NSGA-II.
Solutions obtained by NC method dominate **77** solutions obtained by NSGA-II.
Solutions obtained by NSGA-II dominate **30** solutions obtained by NC method.
Solutions obtained by Algorithm 1 dominate **33** solutions obtained by NC method.
None of the solutions obtained by Algorithm 1 are dominated by NC method.

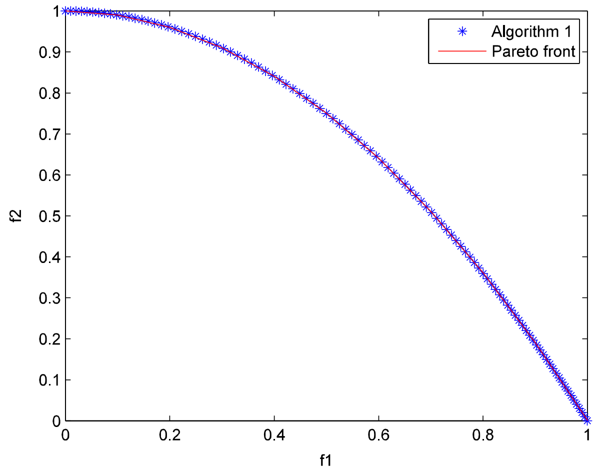

Fig. 8. Algorithm 1 with 417 function evaluations.

obtained by Algorithm 1 are dominated by NSGA-II and NC methods. From comparison of E and ξ for the three methods, it is concluded that the quality of the distribution of approximation points obtained by Algorithm 1 is better than that for the NSGA-II and NC methods. Comparison of Figs. 5–7 shows that the distribution of solutions obtained by Algorithm 1 is better than the NSGA-II and NC method.

Modified ZDT2 test problem.

The modified ZDT2 test problem [13] can be stated as follows:

$$\min f_1(x) = x_1,$$
$$\min f_2(x) = g(x)\left(1 - \left(\frac{x_1}{g(x)}\right)^2\right),$$
$$\text{where } g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i^2,$$
$$x_1 \in [0, 1] \quad \text{and} \quad x_i \in [-1, 1]\ \forall i = 2, 3, \ldots, n.$$

We use $n = 30$. This problem resorts to a nonconvex Pareto front. The efficient solutions correspond to $0 \le x_1^* \le 1$ and $x_i^* = 0$ for $i = 2, \ldots, n$. The ZDT2 is difficult due to the presence of 21^9 local Pareto fronts, which an algorithm must overcome before reaching the global Pareto front. The problem is solved by Algorithm 1 and the NC method with $\delta = \frac{1}{100}$ and NSGA-II with population of size 100 and generation of size 500. The convergence to the Pareto front and the distribution of solutions obtained by Algorithm 1 after 417 function evaluations, NSGA-II after 50 000 function evaluations and the NC method after 391 function evaluations for this problem are depicted in Figs. 8–10, respectively.

The results of solving the ZDT2 problem by Algorithm 1, NSGA-II and NC methods are presented in Table 2.

The solutions obtained by Algorithm 1 dominate 98 solutions obtained by NSGA-II, while none of the solutions obtained by Algorithm 1 are dominated by NSGA-II. None of the solutions obtained by Algorithm 1 are dominated by the NC method and none of the solutions obtained by the NC method are dominated by Algorithm 1. Also, from comparison of E and ξ for the three methods, it is concluded that the quality of the distribution of approximation points obtained by Algorithm 1 and the

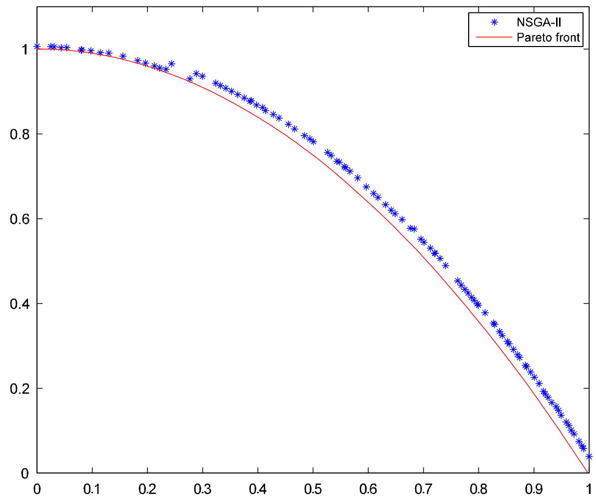

Fig. 9. NSGA-II with 50 000 function evaluations.

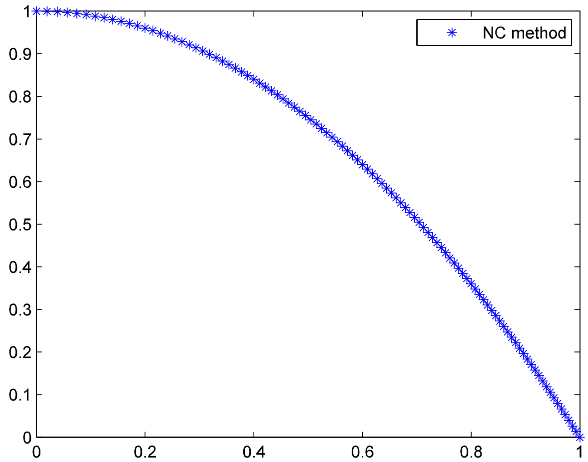

Fig. 10. NC method with 391 function evaluations.

Table 2
Number of function evaluations, CPU time for solving ZDT2, measures of coverage and spacing and the dominance between solutions obtained by Algorithm 1, NSGA-II and NC method for ZDT2.

Method	*ZDT2 problem*			
	Number of function evaluations	CPU time (s)	EX	ξ
Algorithm 1	417	7.6521	0	0.0013
NSGA-II	50,000	123.9843	0.0197	0.0088
NC method	351	8.355745	0	0.0011

Solutions obtained by Algorithm 1 dominate **98** solutions obtained by NSGA-II.
Solutions obtained by NC method dominate **99** solutions obtained by NSGA-II.
None of the solutions obtained by Algorithm 1 are dominated by NSGA-II.
None of the solutions obtained by NC method are dominated by NSGA-II.
None of the solutions obtained by Algorithm 1 are dominated by NC method.
None of the solutions obtained by NC method are dominated by Algorithm 1.

NC method is better than that by NSGA-II and this quality is almost identical for both Algorithm 1 and NSGA-II. Comparison of Figs. 8–10 shows that the distribution of solutions obtained by Algorithm 1 and the NC method is better than that for the NSGA-II.

4.3. Three-objective test problems

Here, two three-objective problems are considered [14].

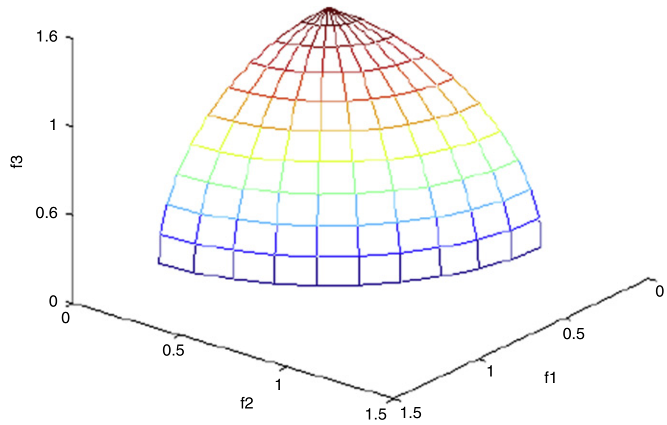

Fig. 11. The Pareto front of DTLZ2.

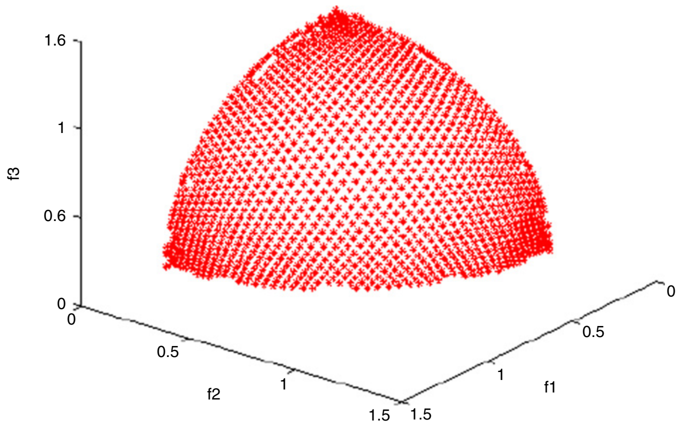

Fig. 12. Algorithm 1 with 3705 function evaluations.

DTLZ2 test problem.

The DTLZ2 test problem is a 12-variable problem:

$$\min f_1(x) = \cos\left(x_1 \frac{\Pi}{2}\right) \cos\left(x_2 \frac{\Pi}{2}\right) (1 + g(x)),$$

$$\min f_2(x) = \cos\left(x_1 \frac{\Pi}{2}\right) \sin\left(x_2 \frac{\Pi}{2}\right) (1 + g(x)),$$

$$\min f_3(x) = \sin\left(x_1 \frac{\Pi}{2}\right) (1 + g(x)),$$

$$\text{where } g(x) = 100\left(10 + \sum_{i=3}^{12}\left((x_i - 0.5)^2 - \cos\left(20\Pi(x_i - 0.5)\right)\right)\right),$$

$$0 \leq x_i \leq 1, \quad i = 1, \ldots, p.$$

The problem is an octant of a unit sphere centered at the origin of the objective space. The set of efficient solutions is

$$X_E = \{x \in \mathbb{R}^{12} | 0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1, x_i = 0.5 \text{ for all } i = 3, \ldots, 12\}.$$

Here, the Pareto front is the first octant of the unit sphere: $\sum_{i=1}^{3} y_i^2 = 1$, depicted in Fig. 11. Due to the multimodal function g, the search space contains $(11^{10} - 1)$ local Pareto fronts. The problem is solved by Algorithm 1 and the NC method with $\delta = \frac{1}{50}$ and NSGA-II with population of size 1000 and generation of size 100. The convergence to the Pareto front and the distribution of solutions obtained by Algorithm 1 after 3705 function evaluations, NSGA-II after 100 000 function evaluations and the NC method after 3815 function evaluations for this problem is depicted in Figs. 12–14, respectively.

The results of solving DTLZ2 problem by Algorithm 1, NSGA-II and NC methods are presented in Table 3.

Solutions obtained by Algorithm 1 dominate 972 solutions obtained by NSGA-II, while none of the solutions obtained by Algorithm 1 are dominated by NSGA-II. None of the solutions obtained by Algorithm 1 are dominated by the NC method and

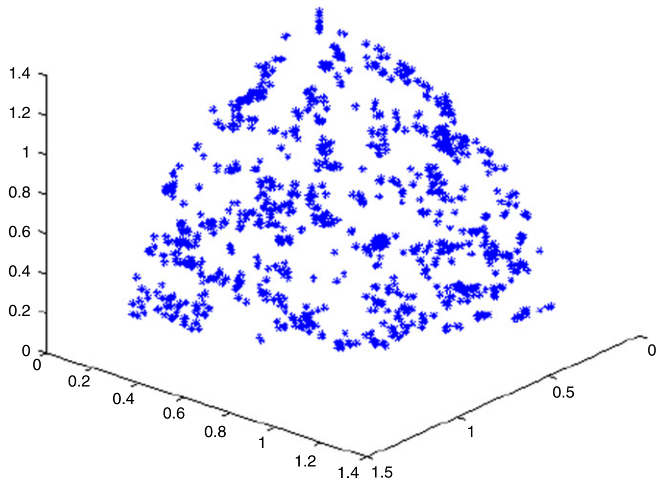

Fig. 13. NSGA-II with 100 000 function evaluations.

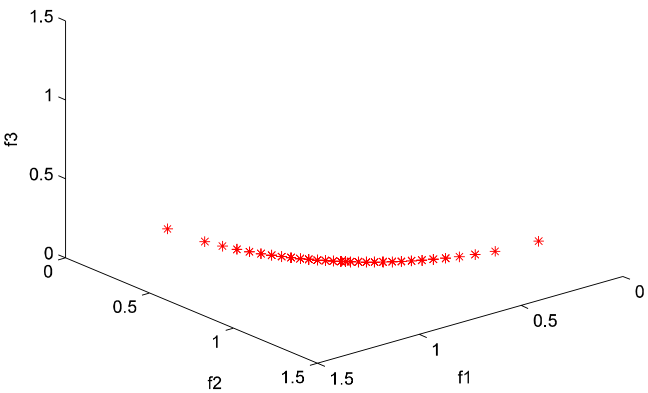

Fig. 14. NC method with 3815 function evaluations.

Table 3
Number of function evaluations, CPU time for solving DTLZ2, measures of coverage and spacing and the dominance between solutions obtained by Algorithm 1, NSGA-II and NC method for DTLZ2.

Method	*DTLZ2 problem*			
	Number of function evaluations	CPU time (s)	EX	ξ
Algorithm 1	3 705	89.08	0	0.2010
NSGA-II	100,000	825.7158	0.0111	0.3221
NC method	3 815	86.985878	0.3333	0.2632

Solutions obtained by Algorithm 1 dominate **972** solutions obtained by NSGA-II.
None of the solutions obtained by Algorithm 1 are dominated by NSGA-II.
None of the solutions obtained by Algorithm 1 are dominated by NC method.
None of the solutions obtained by NC method are dominated by NSGA-II.
None of the solutions obtained by NSGA-II are dominated by NC method.
None of the solutions obtained by NC method are dominated by Algorithm 1.

none of the solutions obtained by the NC method are dominated by Algorithm 1. From comparison of E and ξ for the three methods, it is concluded that the quality of the distribution of approximation points obtained by Algorithm 1 is better than that for the NSGA-II and NC methods. Comparison of Figs. 12–14 shows that Algorithm 1 assures a very good convergence and distribution for DTLZ2, while the performance of the NSGA-II algorithm and NC method are poor. Also the performance of the NSGA-II algorithm is much better than that of the NC method.

The comet problem.

The comet problem is a three-variable problem:

$$\min f_1(x) = \big(1 + x_3\big)(x_1^3 x_2^2 - 10x_1 - 4x_2),$$

$$\min f_2(x) = \big(1 + x_3\big)(x_1^3 x_2^2 - 10x_1 + 4x_2),$$

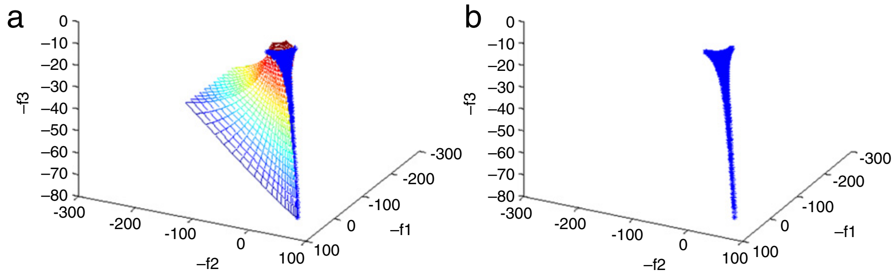

Fig. 15. (a) Image space and (b) Pareto front of the Comet problem.

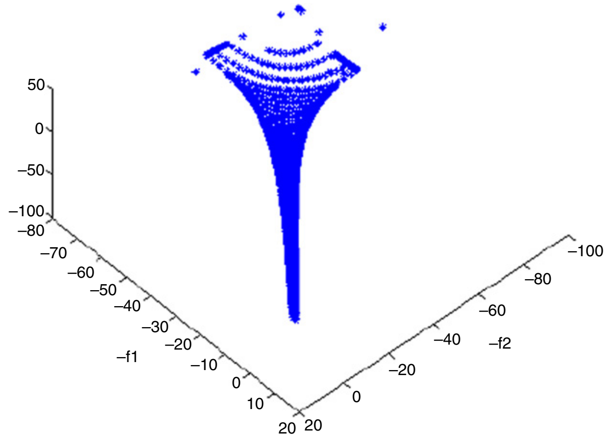

Fig. 16. Algorithm 1 with 13 495 function evaluations.

$$\min f_3(x) = 3(1+x_3)x_1^2,$$
s.t.
$$1 \le x_1 \le 3.5,$$
$$-2 \le x_2 \le 2,$$
$$0 \le x_3 \le 1.$$

The comet problem has a comet-like Pareto front. Starting from a widely spread region, the Pareto front continuously reduces to a thinner region. Finding a wide variety of solutions in both broad and thin portions of the Pareto region simultaneously is a challenging task. Here, we have chosen $g(x_3) = x_3$ and $0 \le x_3 \le 1$. The Pareto front corresponds to $x_3^* = 1$ and $-2 \le (x_1^*)^3 x_2^* \le 2$ with $1 \le x_1^* \le 3.5$. The image space of this problem is shown in Fig. 15(a). The Pareto front of this problem is shown in Fig. 15(b). To achieve a better representation we have drawn the negative of the objective function values.

The problem is solved by Algorithm 1 and the NC method with $\delta = \frac{1}{100}$ and NSGA-II with population of size 5000 and generation of size 100. The convergence to the Pareto front and the distribution of solutions obtained by Algorithm 1 after 13 495 function evaluations, NSGA-II after 100 000 function evaluations and the NC method after 12 204 function evaluations for this problem are depicted in Figs. 16–18, respectively.

The results of solving the comet problem by Algorithm 1, NSGA-II and NC methods are presented in Table 4.

Solutions obtained by Algorithm 1 dominate 79 solutions obtained by NSGA II, and solutions obtained by NSGA II dominate 3 solutions obtained by Algorithm 1. Solutions obtained by Algorithm 1 dominate 27 solutions obtained by NC method and none of the solutions obtained by Algorithm 1 are dominated by the NC method. From comparison of EX and ξ for the three methods, it is concluded that the quality of the distribution of approximation points obtained by Algorithm 1 is better than that for the NSGA-II and NC methods. Comparison of Figs. 16–18 shows that Algorithm 1 assures a very good convergence and distribution for the comet problem, while the performance of the NSGA-II algorithm and NC method are poor. Also the performance of the NSGA-II algorithm is much better than the NC method.

4.4. NSGA-II algorithm

NSGA-II [4] is a very popular multi-objective evolutionary algorithm (MOEA) based on Pareto domination. NSGA-II maintains a population P_t of size N at generation t and generates P_{t+1} from P_t in the following way.

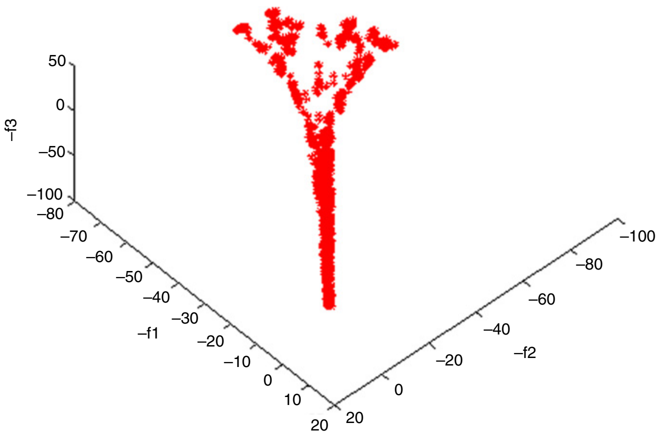

Fig. 17. NSGA-II with 100 000 function evaluations.

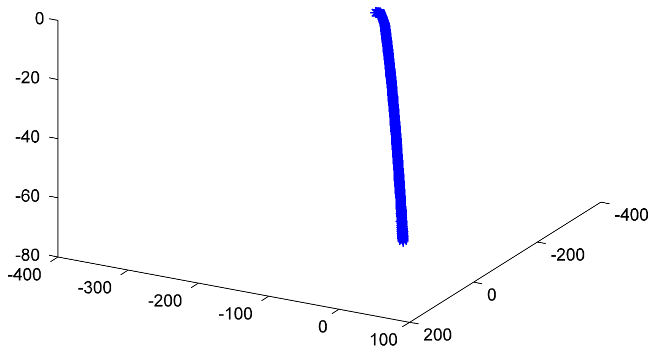

Fig. 18. NC method with 12 204 function evaluations.

Table 4
Number of function evaluations, CPU time for solving comet problem, measures of coverage and spacing and the dominance between solutions obtained by Algorithm 1, NSGA-II and NC methods for the comet problem.

Method	*Comet problem*			
	Number of function evaluations	CPU time (s)	EX	ξ
Algorithm 1	13 495	337.0103	0	0.5021
NSGA-II	100 000	811.2634	0.5148	0.3801
NC method	12 204	257.692384	9.3723	0

Solutions obtained by Algorithm 1 dominate **79** solutions obtained by NSGA-II.
Solutions obtained by NSGA-II dominate **3** solutions obtained by Algorithm 1.
Solutions obtained by Algorithm 1 dominate **27** solutions obtained by NC method.
Solutions obtained by NSGA-II dominate **21** solutions obtained by NC method.
Solutions obtained by NC method dominate **5** solutions obtained by NSGA-II.
None of the solutions obtained by Algorithm 1 are dominated by NC method.

Step 1: Use selection, crossover and mutation to create an offspring population Q_t from P_t.
Step 2: Choose N best solutions from $Q_t \bigcup P_t$ to form P_{t+1}.
The characteristic feature of NSGA-II is that it uses a fast nondominated sorting and crowding distance estimation procedure for comparing qualities of different solutions in Step 2 and selection in Step 1. The computational complexity of each generation in NSGA-II is $o(mN^2)$, where m is the number of the objectives and N is its population size. In this paper we use a polynomial mutation operator with $\eta_c = 10$ and $\eta_m = 10$ respectively. Details of implementation of the NSGA-II algorithm may be obtained from [4].

4.5. Normal Constraint (NC) method

The Normal Constraint method [13] claims to guarantee obtaining even representation of the complete Pareto front for a generic MOP problem by performing a series of optimizations. Each optimization in the series is performed subject to a reduced feasible design space. This method proceeds as follows. First, anchor points are determined. The ith anchor point is written as $f^{i*} = (f_1(x_i^*), \ldots, f_p(x_i^*))^T$, where $x_i^* = \operatorname{argmin}_{x \in X} f_i(x)$. The anchor points form the vertices of what is called the utopia hyperplane (in the objective space). Then the following problem for a set of evenly distributed points p_k on the utopia hyperplane is solved (this set is obtained according to [3], as described in Section 3):

$$\begin{aligned}
&\min f_j(x)\\
&\text{s.t.}\\
&x \in X,\\
&v_i(f - p_k) \leq 0, \quad \forall i \in \{1, \ldots, p\} \setminus \{j\},\\
&v_i(f^{j*} - f^{i*}) \leq 0, \quad \forall i \in \{1, \ldots, p\} \setminus \{j\},
\end{aligned}$$

where $j \in \{1, \ldots, p\}$.

However, this method has the potential to find dominated solutions, and thus a Pareto filter is necessary to remove these dominated solutions.

5. Conclusions

In this paper, with introduction of a new parameter restriction for the Pascoletti and Serafini scalarization, a numerical method for constructing an approximation of the Pareto front was proposed. The method generates a good distribution of the entire Pareto front for both convex and nonconvex Pareto fronts, and is formulated in a way which may be suitable for practical implementation. The proposed method was applied to four standard test problems and performed very well in them for obtaining the Pareto front. In all cases, the solutions obtained by the proposed method were better than the solutions obtained by the NC method and the NSGA-II algorithm.

References

[1] M. Ehrgott, M. Wiecek, Multiobjective programming, in: J. Figueira, S. Greco, M. Ehrgott (Eds.), Multicriteria Decision Analysis: State of the Art Surveys, Springer, New York, 2005, pp. 667–722.
[2] C. Audet, G. Savard, W. Zghal, Multiobjective optimization through a series of single-objective formulations, SIAM J. Optim. 19 (2008) 188–210.
[3] I. Das, J.E. Dennis, Normal-boundary intersection: a new method for generating the Pareto surface in nonlinear multicriteria optimization problems, SIAM J. Optim. 8 (1998) 631–657.
[4] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197.
[5] J. Dutta, C. Yalçin Kaya, A new scalarization and numerical method for constructing weak Pareto front of multi-objective optimization problems, J. Optim. (2011).
[6] G. Eichfelder, Adaptive Scalarization Methods in Multiobjective Optimization, Springer, Berlin, Heidelberg, 2008.
[7] G. Eichfelder, Scalarizations for adaptively solving multi-objective optimization problems, Comput. Optim. Appl. 44 (2009) 249–273.
[8] G. Eichfelder, An adaptive scalarization method in multi-objective optimization, SIAM J. Optim. 19 (2009) 1694–1718.
[9] A. Engau, M.M. Wiecek, Generating ε-efficient solutions in multiobjective programming, European J. Oper. Res. 177 (2007) 1566–1579.
[10] D. Gourion, D.T. Luc, Generating the weakly efficient set of nonconvex multiobjective problems, J. Global Optim. 41 (2008) 517–538.
[11] D.T. Luc, T.Q. Phong, M. Volle, Scalarizing functions for generating the weakly efficient solution set in convex multiobjective problems, SIAM J. Optim. 15 (2005) 987–1001.
[12] A. Pascoletti, P. Serafini, Scalarizing vector optimization problems, J. Optim. Theory Appl. 42 (1984) 499–524.
[13] E. Zitzler, K. Deb, L. Thiele, Comparison of multiobjective evolutionary algorithms: empirical results, Evol. Comput. J. 8 (2000) 125–148.
[14] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multiobjective optimization, in: A. Abraham, L. Jain, R. Goldberg (Eds.), Evolutionary Multiobjective Optimization, Springer-Verlag, London, 2005, pp. 105–145.
[15] A. Messac, C.A. Mattson, Normal constraint method with guarantee of even representation of complete Pareto frontier, AIAA J. 42 (2004) 2101–2111.
[16] H. Meng, X. Zhang, S. Liu, New quality measures for multiobjective programming, Lecture Notes in Comput. Sci. 3611 (2005) 1044–1048.