

An Objective Penalty Functions Algorithm for Multiobjective Optimization Problem

Zhiqing Meng, Rui Shen, Min Jiang

College of Business and Administration, Zhejiang University of Technology, Hangzhou, China

E-mail: {mengzhiqing, shenrui, jiangmin}@zjut.edu.cn

Received October 24, 2011; revised November 20, 2011; accepted December 7, 2011

Abstract

By using the penalty function method with objective parameters, the paper presents an interactive algorithm to solve the inequality constrained multi-objective programming (MP). The MP is transformed into a single objective optimal problem (SOOP) with inequality constraints; and it is proved that, under some conditions, an optimal solution to SOOP is a Pareto efficient solution to MP. Then, an interactive algorithm of MP is designed accordingly. Numerical examples show that the algorithm can find a satisfactory solution to MP with objective weight value adjusted by decision maker.

Keywords: Multiobjective Optimization Problem, Objective Penalty Function, Pareto Efficient Solution, Interactive Algorithm

1. Introduction

The interactive algorithm is very efficient in solving multi-objective optimization problems of many fields, while the penalty function is a very important method in solving optimization problems with constraints. Hence, based on objective penalty function, we propose an interactive algorithm which provides a versatile tool in finding solutions to multi-objective optimization problems with constraints. In solving multi-objective optimization problems, the interactive algorithm provides a way to adjust objective weight value between the decision maker and computer, so that solution space is readily understood, which also makes it easier in use and more convenient in operation.

In this paper the following inequality constrained multi-objective programming is considered:

$$\begin{aligned} \min f(x) &= (f_1(x), f_2(x), \dots, f_q(x)) \\ \text{s.t. } x &\in X, g_i(x) \leq 0, i = 1, 2, \dots, m, \end{aligned} \quad (\text{MP})$$

where $f_j: R^n \rightarrow R^1$, $g_i: R^n \rightarrow R^1$ for $j \in J = \{1, 2, \dots, q\}$, $i \in I = \{1, 2, \dots, m\}$ and X is a subset of R^n .

It is good to find out a satisfactory solution to (MP) such that all objective values are optimal, but this is obviously difficult in general. Hence, lots of efforts have been devoted to this area to find an efficient method, and

up until now many algorithms are presented [1-11].

In 1971, Benayoun *et al.* firstly presented an interactive algorithm STEM for linear multiobjective programming [1]. Its idea is the first in finding out a solution of an ideal value to every objective, obtaining better solution by improving unsatisfactory objectives value, and keeping within concession value of satisfactory objectives. With man-machine conversation, interactive algorithms provide a method to solve MP. There are many interactive approaches, as it is not possible for a decision maker to know all the objective values of MP. Then through interactive algorithms, he may gradually learn objective value changes and thus in the interactive procedure may determine his preferences to objectives. As to the dissatisfactory objectives, he may get satisfactory solution by modifying some parameters he gives, e.g. the ideal objective values and the weights of objectives.

For example, Geoffrion, Dyer and Feinberg (1972) gave an interactive approach to multi-criterion optimization, where they defined a non-explicitly criterion functions to show the DM's overall preference [2]. Zionts and Wallenius (1976) also presented a man-machine interactive mathematical programming method, where the overall utility function is assumed to be implicitly a linear function and more generally a concave function of the objective functions [3]. Furthermore, Rosinger (1981) studied the algorithm which is a modification of the steepest ascent method, giving at each iteration a signifi-

cant freedom and ease for the decision-maker’s self-expression, and requiring a minimal information on his local estimate of the steepest-ascent direction [4]. Zionts and Wallenius (1983) developed a method for interactive multiple objective linear programming by assuming an unknown pseudo concave utility function satisfying certain general properties [5]. Sadagopan and Ravinderan (1986) developed several interactive procedures for solving multiple criteria nonlinear programming problems based on the generalized reduced gradient method for solving single objective nonlinear programming problems [6]. Siegfried (1990) presented an interactive algorithm for nonlinear vector optimization problems, after solving only two optimization problems [7]. Kassem (1995) dealt with an interactive stability of multiobjective nonlinear programming problems with fuzzy parameters in the constraints [8]. Aghezzaf and Ouaderhman (2001) proposed an interactive interior point method for finding the best compromise solution to a multiple objective linear programming problem [9]. Abo-Sinna and Abou-El-Enien (2006) extended the technique of order preference by similarity ideal solution (TOPSIS) for solving large scale multiple objective programming problems involving fuzzy parameters [10]. Luque, Ruiz and Steuer pointed out that many interactive algorithms have two main features: 1) they help a decision maker (DM) learn about a problem while solving it, and 2) they put to work iteratively any new insights gained during the solution process to help the DM navigate to a final solution [11].

It is difficult to define an appropriate utility function for MP in the interactive algorithms. By using objective penalty functions as utility functions for MP, the paper obtains a satisfactory solution, when the decision maker is allowed, in the interactive algorithm, to choose another weight of objectives for some dissatisfactory objectives time and again. So our interactive algorithm has two advantages: 1) it is able to find out an efficient solution to each new MP with better convergence, 2) it can control the change of objectives such that a more satisfactory solution is obtained. What needs to focus on for the decision maker is the objective changes. Numerical examples show that the proposed interactive algorithm has faster convergence effect in Section 3.

The remainder of this paper is organized as follows. In Section 2, we provide results of the penalty problem of MP with penalty parameters. In Section 3, we present an interactive algorithm to solve the MP. Numerical examples show that the proposed algorithm has good convergence and can control objective changes by changing objective weights.

2. An Objective Penalty Function

In this section, an objective penalty function of (MP) is introduced.

For (MP), let $g_i^+(x) = \max\{g_i(x), 0\}$ for $i \in I$ and define a function $Q: R^1 \rightarrow R^1$ as follows:

$Q(t) = \max\{t, 0\}^2$. Then Q is increasing and differentiable at any $t \in R^1$.

A feasible set of (MP) is denoted as $X_0 = \{x \in X \mid g_i(x) \leq 0, i \in I\}$. $\bar{x} \in X_0$ is called a Pareto efficient solution if there is no $x \in X_0$ such that $f(x) \leq f(\bar{x})$ and $f(x) \neq f(\bar{x})$.

Let an objective weight value $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_q)^T > 0$ be a given positive vector, and $M \in R^1$ be objective parameters of $(f_1(x), f_2(x), \dots, f_q(x))$. Then define

$$f_0(x) = \sum_{j \in J} \lambda_j Q(f_j(x) - M).$$

Theorem 2.1 If \bar{x} is an optimal solution to the following problem:

$$\min f_0(x) \quad \text{s.t. } x \in X_0, \quad (P_\lambda)$$

and for all $j \in J$, $M < f_j(\bar{x})$, then \bar{x} is a Pareto efficient solution to (MP).

Proof. Suppose that \bar{x} is not a Pareto efficient solution to (MP), then there is an $x' \in X_0$ such that $f(x') \leq f(\bar{x})$ and $f(x') \neq f(\bar{x})$. It follows from the assumption that

$$f(x') - M \leq f(\bar{x}) - M, \quad \forall j \in J,$$

and there is at least such a $j \in J$ that $f(x') - M < f(\bar{x}) - M$. Hence, we have $f_0(x') < f_0(\bar{x})$, which results in a contradiction.

From Theorem 2.1, we learn that a Pareto efficient solution to (MP) can be found out by solving the single objective problem (P_λ) . Furthermore, the problem (P_λ) can be transformed into an unconstrained optimization by using nonlinear penalty function, which is defined as:

$$\begin{aligned} F(x, \lambda, M) &= f_0(x) + M^2 \sum_{i \in I} g_i^+(x) \\ &= \sum_{j \in J} \lambda_j Q(f_j(x) - M) + M^2 \sum_{i \in I} g_i^+(x). \end{aligned}$$

where $M < 0$. Consider the following nonlinear penalty optimization problem:

$$\min F(x, \lambda, M) \quad \text{s.t. } x \in X_0. \quad P_\lambda(M)$$

Theorem 2.2 Suppose that M is constant, x_M^* is an optimal solution to $P_\lambda(M)$, and for all $j \in J$, $M < f_j(x_M^*)$. If x_M^* is a feasible solution to (MP), then x_M^* is a Pareto efficient solution to (MP).

Proof. Let x^* be an optimal solution to $P_\lambda(M)$. From the given conditions, we have

$$f_0(x_M^*) = F(x_M^*, \lambda, M) \leq F(x^*, \lambda, M) = f_0(x^*).$$

Since x_M^* is a feasible solution to (MP) and x^* is an optimal solution to $P_\lambda(M)$, $f_0(x_M^*) \geq f_0(x^*)$. So, x_M^* is an optimal solution to $P_\lambda(M)$. From Theorem 2.1, x_M^* is a Pareto efficient solution to (MP).

Theorem 2.1 and Theorem 2.2 give a good way to solve (MP). The objective parameter M required in Theorem 2.2 may exist, as shown in the following example.

Example 2.1 Consider the MP problem:

$$\begin{aligned} \min f(x_1, x_2) &= (x_1^2, x_2^2) \\ \text{s.t. } -x_1 &\leq 0, -x_2 \leq 0. \end{aligned} \tag{P2.1}$$

It is clear that $(x_1^*, x_2^*) = (0, 0)$ is a Pareto efficient solution to (P2.1) and the objective value is $(0, 0)$. Let's take $M < 0$, $\lambda_1 > 0$ and $\lambda_2 > 0$. Define the penalty function:

$$\begin{aligned} F(x, \lambda, M) &= \lambda_1 \max\{x_1^2 - M, 0\}^2 + \lambda_2 \max\{x_2^2 - M, 0\} \\ &\quad + \max\{0, -x_1\} + \max\{0, -x_2\}. \end{aligned}$$

It is clear that $(x_1, x_2) = (0, 0)$ is an optimal solution to $P_\lambda(M)$ [with $M < 0$].

It is proved in [13] that the stability of constrained penalty function can ensure exactness. So in this paper we define the stability of objective penalty function to ensure equivalence between $P_\lambda(M)$ and (P_λ) .

Let a perturbed problem of (P_λ) defined as

$$\begin{aligned} \min f_0(x) \\ \text{s.t. } x \in X, g_i(x) \leq s_i, i = 1, 2, \dots, m, \end{aligned} \tag{P_\lambda(s)}$$

where $s = (s_1, s_2, \dots, s_M)$.

Definition 2.1 For $s \in R^n$, let x^* be an optimal solution to (P_λ) and x_s^* be an optimal solution to $P_\lambda(s)$. If

$$f_0(x^*) - f_0(x_s^*) \leq M^2 |s|,$$

where $|s| = \sum_{i=1}^m |s_i|$, $s = (s_1, s_2, \dots, s_M)$, then problem (P_λ)

is called stable for M .

Theorem 2.3 Let x^* be an optimal solution to (P_λ) . Then, problem (P_λ) is stable for M if and only if x^* is an optimal solution to $P_\lambda(M)$.

Proof. First, if problem (P_λ) is stable for M , it is hereby proved that x^* is an optimal solution to $P_\lambda(M)$. Assume that x^* is not an optimal solution to $P_\lambda(M)$, then, there is some x' such that

$$F(x', \lambda, M) < F(x^*, \lambda, M) = f_0(x^*).$$

That is

$$f_0(x') \leq f_0(x') + M \sum_{i \in I} g_i^+(x') < f_0(x^*).$$

If x' is a feasible solution to (P_λ) , we will get a contradiction. Since x^* be an optimal solution to (P_λ) , we have

$$f_0(x^*) \leq f_0(x')$$

such that $f_0(x') < f_0(x^*) \leq f_0(x')$, then x^* is not an optimal solution to (P_λ) . Hence, we have that x' is not any feasible solution to (P_λ) and $\sum_{i \in I} g_i^+ > 0$. Let

$s'_i = \max\{g_i(x'), 0\}$ for $i = 1, 2, \dots, m$. Let x_s^* be an optimal solution to $P_\lambda(s')$. So, it is clear to have $f_0(x_s^*) \leq f_0(x')$. We have

$$\begin{aligned} f_0(x_s^*) + M^2 \sum_{i \in I} s'_i &\leq f_0(x') + M^2 \sum_{i \in I} g_i^+(x') \\ &= F(x', \lambda, M) < f_0(x^*) \end{aligned}$$

and

$$f(x^*) - f_0(x_s^*) > M^2 |s'|.$$

Hence, the problem (P_λ) is not stable for M .

Next, let's prove that the problem (P_λ) is stable for M , under the condition that if x^* is an optimal solution to $P_\lambda(M)$. Let x_s^* be an optimal solution to $(P_\lambda(s))$. Since x^* is an optimal solution to $P_\lambda(M)$, we have

$$F(x^*, \lambda, M) \leq f_0(x_s^*) + M^2 \sum_{i \in I} g_i^+(x_s^*).$$

That is

$$f_0(x^*) \leq f_0(x_s^*) + M^2 \sum_{i \in I} |s_i|.$$

We have that problem (P_λ) is stable for M .

Example 2.2 Consider the problem (P2.1) and its (P2.1)(s):

$$\begin{aligned} \min f(x_1, x_2) &= (x_1^2, x_2^2) \\ \text{s.t. } -x_1 &\leq s_1, -x_2 \leq s_2. \end{aligned} \tag{P2.1)(s)}$$

$x_s^* = \max\{0, -s_1\}$, $\max\{0, -s_2\}$ is a Pareto efficient solution to (P2.1)(s). Let's take $M < 0$. Define the penalty function:

$$\begin{aligned} F(x, \lambda, M) &= \lambda_1 \max\{x_1^2 - M, 0\}^2 + \lambda_2 \max\{x_2^2 - M, 0\}^2 \\ &\quad + \max\{0, -x_1\} + \max\{0, -x_2\}. \end{aligned}$$

We have that stable condition holds as follows:

$$\begin{aligned} f_0(x^*) - f_0(x_s^*) &= (\lambda_1 + \lambda_2) \lambda_1 \left(\max\{0, -s_1\}^2 - M \right)^2 \\ &\quad - \lambda_2 \left(\max\{0, -s_2\} - M \right)^2 \\ &\leq M^2 \left(\max\{0, s_1\} + \max\{0, s_2\} \right). \end{aligned}$$

Based on Theorem 2.2 and Theorem 2.3, we develop an algorithm to compute (MP). It solves the problem $P_\lambda(M)$ sequentially and we name it Objective Penalty

Function Algorithm of Multiobjective Optimization Problem (OPFAMOP for short).

OPFAMOP Algorithm:

Step 1: Choose $\lambda > 0, x^1, M_1 < 0, N > 1$ and $k = 1$.

Step 2: Take the violation x^k as the starting point for solving the problem: $\min_{x \in X} F(x, \lambda, M_k)$. Let x^{k+1} be an optimal solution.

Step 3: If x^{k+1} is a feasible solution to (MP) and $M_k < f_j(x^{k+1})$ for all $j \in J$, stop and x^{k+1} is a Pareto efficient solution to (MP). Otherwise, let $M_{k+1} = NM_k, k = k + 1$ and go to Step 2.

The convergence of the OPFAMOP algorithm is proved in the following theorem. Let

$$S(L, f_0) = \{x^k \mid Lf_0(x^k), k = 1, 2\},$$

which is called an L-level set. We say that $S(L, f_0)$ is bounded if, for any given $L > 0, S(L, f_0)$ is bounded.

Theorem 2.4 Suppose that $f_j(j \in J)$ and $g_i(i \in I)$ are continuous on R^n , and the L-level set $S(L, f_0)$ is bounded. Let $\{x^k\}$ be the sequence generated by the OPFAMOP algorithm.

1) If $\{x^k\} (k = 1, 2, \dots, \bar{k})$ is a finite sequence (i.e., the OPFAMOP algorithm stops at the \bar{k} -th iteration), then $x^{\bar{k}}$ is a Pareto efficient to (MP).

2) If $\{x^k\}$ is an infinite sequence and there is some $k' > 1$ such that $f_j(x^{k+1}) > M_k (\forall j \in J)$ for all $k > k'$, then $\{x^k\}$ is bounded and any limit point of it is a Pareto efficient to (MP). Otherwise, for some $j \in J, f_j(x^k) \rightarrow -\infty$ as $k \rightarrow +\infty$.

Proof. 1) If the OPFAMOP Algorithm terminates at the \bar{k} th iteration and the second situation of Step 3 occurs, by Theorem 2.1 and Theorem 2.2, $x^{\bar{k}}$ is a Pareto efficient to (MP).

2) Suppose that $\{x^k\}$ is an infinite sequence and there is some $k' > 1$ such that $f_j(x^{k+1}) > M_k (\forall j \in J)$ for all $k > k'$. Let x' be a feasible solution to (MP).

We first show that the sequence $\{x^k\}$ is bounded. Since x^k is an optimal solution to $\min_{x \in X} F(x, \lambda, M_k)$,

$$f_0(x^{k+1}) \leq F(x^{k+1}, \lambda, M_k) \leq f_0(x'), \quad k = 1, 2, \dots$$

Hence, the L-level set $S(f_0(x'), f_0)$ is bounded, then the sequence $\{x^k\}$ is bounded. Without loss of generality, we assume $x^k \rightarrow \bar{x}$. And, for any $x \in X_0$, we have

$$0 < f_0(x^{k+1}) + M_k \sum_{i \in I} g_i^+(x^{k+1}) \leq f_0(x), \quad \forall k > k'.$$

That is

$$\sum_{i \in I} g_i^+(x^{k+1}) \leq \frac{1}{M_k^2} \sum_{j \in J} (f_j(x) - f_j(x^{k+1})) (f_j(x^{k+1}) + f_j(x) - 2M_k), \forall k > k'.$$

It is clear that $M_k \rightarrow -\infty$ as $k \rightarrow +\infty$. By letting $k \rightarrow +\infty$ in the above equation, we obtain

$\sum_{i \in I} g_i^+(\bar{x}) = 0$. Hence, \bar{x} is a feasible solution of (MP) and $f_0(\bar{x}) \leq f_0(x)$. Therefore, \bar{x} is Pareto efficient to (MP).

3. An Interactive Algorithm

In this section, we propose an interactive algorithm by the objective penalty function. There are many approaches of the MP problem to be transformed into a single objective optimal problem, such as a non-explicitly criterion function [2-4,9], nonlinear utility functions [5,6], weighting Tchebycheff function [8,9] and TOPSIS method [10]. So, our proposed approach is novel.

According to the OPFAMOP Algorithm, we can select $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_M)^T$ to get an approximate Pareto efficient solution to (MP).

Definition 3.1 A vector $x_\epsilon \in X$ is ϵ -feasible if

$$g_i(x_\epsilon) \leq \epsilon, \quad \forall i \in I.$$

Now, we present the following interactive algorithm IOPFAMOP based on the OPFAMOP.

IOPFAMOP:

Step 1: Choose $\lambda^1, x^1, \epsilon > 0, N > 1, K > 1$ and $s = 1$.

Step 2: By using the OPFAMOP Algorithm, compute problem $P_\lambda^s(M)$.

Step 2.1: Let $M_1, k = 1$.

Step 2.2: Solve the problem: $\min_{x \in X} F(x, \lambda^s, M_k)$, to get an ϵ -feasible solution x^k .

Step 2.3: If $k < K$, modify the penalty objective values $M_{k+1} = NM_k$.

Otherwise, $k = K$, let $x^s = x^k$ and go to Step 3.

Step 2.4: Let $k = k + 1$ and go to Step 2.1.

Step 3: The decision maker analyzes the objective $(f_1(x^s), f_2(x^s), \dots, f_q(x^s))$: if the solution x^s is satisfactory, then stop; otherwise, the decision maker will modify the weight values of objective, and go to Step 4.

Step 4: Deal with all the unsatisfactory objectives $f_j(x^s)$ as per the following procedure repeatedly: if the decision maker wants to increase j th objective value, then a $\Delta_j^s > 0$ should be given, then let $\lambda_{j+1}^s := \lambda_j^s + \Delta_j^s$, if the decision maker wants to decrease j th objective value, then a $\Delta_j^s > 0$ should be given, then let $\lambda_{j+1}^s := \lambda_j^s - \Delta_j^s$. Finally, let $s := s + 1$ and go to Step 2.

Remark: By Theorem 2.4, we may get an approximate Pareto efficient solution to (MP) in Step 2. Hence, using the above interactive algorithm, we may, after many interactive steps, obtain satisfactory objective values by modifying weight λ_i , as shown in Examples 3.1 to 3.3.

It is well known that each interactive sub-problem

need solve constrained optimization problem in the existed approach [1-11], but each interactive sub-problem in our approach only need solve unconstrained optimization problem.

We apply the above interactive algorithm IOPFAMOP to two examples programmed by Matlab6.5. The aim of numerical examples is to check the convergence of the interactive algorithm IOPFAMOP and control the changes of objective.

Example 3.1 Consider the following linear programming problem:

$$\begin{aligned} \min f(x_1, x_2) &= (-2x_1 - x_2, -x_1 - 4x_2) \\ \text{s.t. } 2x_1 + 3x_2 &\leq 6 \\ -x_1 &\leq 0, -x_2 \leq 0 \end{aligned} \tag{P4.1}$$

We wish to find out a solution such that every objective function value is close to each other.

Let penalty function

$$\begin{aligned} F(x_1, x_2; \lambda, M) &= \lambda_1 Q(-2x_1 - x_2 - M) \\ &+ \lambda_2 Q(-x_1 - 4x_2 - M) + M^2 \max\{2x_1 + 3x_2 - 6, 0\} \\ &+ M^2 \max\{-x_1, 0\} + M^2 \max\{-x_2, 0\}. \end{aligned}$$

Let $M_1 = -10, N = 4, K = 3$, error of an approximate solution (x_1, x_2) :

$$e(x^s) = \sum_{i \in I} g_j^+(x_s^*),$$

then different approximate solutions (x_1, x_2) are obtained by selecting different (λ_1, λ_2) (as shown in **Table 3.1**).

Remarks for Table 3.1

Step 1: The decision maker (DM) first takes a weight value $(\lambda_1^1, \lambda_2^1) = (0.5, 0.5)^T$.

By the interactive algorithm, the DM obtains the objective function value $f(x_1^1, x_2^1) = (-4.068164, -5.414795)$ at approximate solution $(x_1^1, x_2^1) = (1.551123, 0.965918)$. Because the second objective value f_2 is less than the first objective value f_1 , the DM will improve weight value λ_1 in the Step 2.

Step 2: Then, the DM takes a second weight value $(\lambda_1^2, \lambda_2^2) = (0.6, 0.5)^T$. By the interactive algorithm, the DM obtains the objective function value $f(x_1^2, x_2^2) = (-4.570135, -4.787331)$ at approximate solution $(x_1^2, x_2^2) = (1.927601, 0.714933)$. In order to decrease the first objective value f_1 , the DM still need to improve

Table 3.1. Numerical results for (P4.1).

s	$(\lambda_1^s, \lambda_2^s)$	$e(x^s)$	(x_1^s, x_2^s)	(f_1^s, f_2^s)
1	(0.50, 0.50)	0.00	(1.551123, 0.965918)	(-4.068164, -5.414795)
2	(0.60, 0.50)	0.00	(1.927601, 0.714933)	(-4.570135, -4.787331)
3	(0.70, 0.50)	0.00	(2.201802, 0.532132)	(-4.935736, -4.330330)
4	(0.63, 0.50)	0.00	(2.019328, 0.653781)	(-4.692437, -4.634454)

weight value λ_1 in the next step.

Step 3: Then, the DM takes a third weight value $(\lambda_1^3, \lambda_2^3) = (0.7, 0.5)^T$. By the interactive algorithm, the DM obtains the objective function value $f(x_1^3, x_2^3) = (-4.935736, -4.330330)$ at approximate solution $(x_1^3, x_2^3) = (2.201802, 0.532132)$. This time, the DM need to decrease weight value λ_1 in the next step.

Step 4: Then, the DM take a forth weight value $(\lambda_1^4, \lambda_2^4) = (0.63, 0.5)^T$. By the interactive algorithm, the DM obtains the objective function value $f(x_1^4, x_2^4) = (-4.692437, -4.634454)$ at approximate solution $(x_1^4, x_2^4) = (2.019328, 0.653781)$. Then, the DM is satisfied with the approximate solution, and wishes to stop.

Example 3.2 Consider the problem:

$$\begin{aligned} \min f(x_1, x_2) &= (x_1 - 2x_2, -2x_1 + x_2, -x_1 - x_2), \\ \text{s.t. } x_2 &\leq 2x_1^4 - 8x_1^3 + 8x_1^2 + 2, \\ x_2 &\leq 4x_1^4 - 32x_1^3 + 88x_1^2 - 96x_1 + 36, \\ 0 &\leq x_1 \leq 3, \\ 0 &\leq x_2 \leq 4. \end{aligned} \tag{P4.2}$$

Let penalty function

$$\begin{aligned} F(x_1, x_2; \lambda, M) &= \lambda_1 \max\{x_1 - 2x_2 - M, 0\}^2 \\ &+ \lambda_2 \max\{-2x_1 + 2x_2 - M, 0\}^2 \\ &+ \lambda_3 \max\{-x_1 - x_2 - M, 0\}^2 \\ &+ M^2 \max\{x_2 - 2x_1^4 + 8x_1^3 - 8x_1^2 - 2, 0\} \\ &+ M^2 \max\{x_2 - 4x_1^4 + 32x_1^3 - 88x_1^2 + 96x_1 - 36, 0\} \\ &+ M^2 \max\{-x_1, 0\} + M^2 \max\{-x_2, 0\} \\ &+ M^2 \max\{x_1 - 3, 0\} + M^2 \max\{x_2 - 4, 0\} \end{aligned}$$

Let $M_1 = -1, N = 2, K = 3$, error of an approximate solution (x_1, x_2) :

$$e(x^s) = \sum_{i \in I} g_j^+(x_s^*),$$

then we get numerical results for $s = 6$ in **Table 3.2**.

In **Table 3.2**, from $s = 1$ to $s = 3$, the second objective value f_2^s improves from -2.109783 to -2.555347 . Now, the DM wishes to find a solution such that three objectives are as small as possible with the second objective less than -2.4 , and the first objective less than -2.5 . Then, when $s = 4, 5, 6$, the first objective value f_1^s improves from -2.111084 to -2.549624 . So, the summing of objective values from f_1^s to f_3^s improves from -9.585379 to -9.920799 . And finally, the DM gets a satisfactory solution $(x_1^6, x_2^6) = (2.457059, 2.503341)$.

Example 3.3 Consider the problem: (Illustrative example in [11])

Table 3.2. Numerical results for (P3.2).

s	$(\lambda_1^s, \lambda_2^s, \lambda_3^s)$	$e(x^s)$	(x_1^s, x_2^s)	(f_1^s, f_2^s, f_3^s)
1	(0.50, 0.50, 0.50)	0.00	(2.417748, 2.725713)	(-3.033678, -2.109783, -5.143461)
2	(0.50, 0.60, 0.50)	0.00	(2.443024, 2.583925)	(-2.724826, -2.302124, -5.026949)
3	(0.50, 0.70, 0.50)	0.00	(2.475504, 2.395661)	(-2.315818, -2.555347, -4.871165)
4	(0.55, 0.70, 0.50)	0.00	(2.491432, 2.301258)	(-2.111084, -2.681605, -4.792690)
5	(0.60, 0.70, 0.50)	0.00	(2.475939, 2.393096)	(-2.310252, -2.558783, -4.869035)
6	(0.65, 0.70, 0.50)	0.00	(2.457059, 2.503341)	(-2.549624, -2.410776, -4.960400)

$$\min f(x) = \left((x_1 - 2)^2 + 2(x_2 - 3)^2, x_3^2 + 2x_4^2, 2(x_5 - 5)^2 + (x_6 - 1)^2, 3x_7 + x_8^2 \right),$$

s.t. $x_2 + x_3 + x_4 - x_5 = 2,$

$$x_1^2 + x_2^2 \leq 5,$$

$$x_1 - x_3 \leq 0,$$

$$x_2 - x_4 \leq 0,$$

$$x_1 + x_2 + x_5 + x_7 - 2x_8 = 0,$$

$$x_7^2 + x_8^2 \leq 8,$$

$$x_1, x_2, x_5, x_6 \leq 0,$$

$$x_3 \geq 1, x_8 \geq 0.5.$$

(P4.3)

Let penalty function

$$\begin{aligned} F(x; \lambda, M) = & \lambda_1 \max \left\{ (x_1 - 2)^2 + 2(x_2 - 3)^2 - M, 0 \right\}^2 \\ & + \lambda_2 \max \left\{ x_3^2 + 2x_4^2 - M, 0 \right\}^2 \\ & + \lambda_3 \max \left\{ 2(x_5 - 5)^2 + (x_6 - 1)^2 - M, 0 \right\}^2 \\ & + \lambda_4 \max \left\{ 3x_7 + x_8^2 - M, 0 \right\}^2 \\ & + M^2 \max \{ x_2 + x_3 + x_4 - x_5 - 2, 0 \} \\ & + M^2 \max \{ -x_2 - x_3 - x_4 + x_5 + 2, 0 \} \\ & + M^2 \max \{ x_1^2 + x_2^2 - 5, 0 \} + M^2 \max \{ x_1 - x_3, 0 \} \\ & + M^2 \max \{ x_2 - x_4, 0 \} \\ & + M^2 \max \{ x_1 + x_2 + x_5 + x_7 - 2x_8, 0 \} \\ & + M^2 \max \{ -x_1 - x_2 - x_5 - x_7 + 2x_8, 0 \} \end{aligned}$$

$$\begin{aligned} & + M^2 \max \{ x_7^2 + x_8^2 - 8, 0 \} + M^2 \max \{ x_1, 0 \} \\ & + M^2 \max \{ x_2, 0 \} + M^2 \max \{ x_5, 0 \} + M^2 \max \{ x_6, 0 \} \\ & + M^2 \max \{ 1 - x_3, 0 \} + M^2 \max \{ 0.5 - x_8, 0 \}. \end{aligned}$$

Let $M_1 = -1, N = 4, K = 5,$ error of an approximate solution $x:$

$$e(x^s) = \sum_{i \in I} g_i^+(x_i^s),$$

then we get numerical results for $s = 6$ in **Table 3.3.**

In **Table 3.3,** when $s = 1, 2,$ the second objective value f_2^s and the fourth objective value f_4^s improves from $f_2^s = 9.808986$ and $f_4^s = 8.310570$ to $f_2^s = 7.674058$ and $f_4^s = 4.762737.$ From $s = 3$ to $s = 4,$ the first objective value f_1^s and the second objective value f_2^s improves from $f_1^s = 7.480056$ and $f_2^s = 6.965056$ to $f_1^s = 6.844538$ and $f_2^s = 6.563474.$ The DM wishes to keep the four objective values $(f_2^s, f_1^s, f_3^s, f_4^s)$ less than $(7, 7, 13, 5).$

For $s = 6,$ the DM obtains the four objective values $(f_2^s, f_1^s, f_3^s, f_4^s) = (6.292457, 6.723388, 12.731173, 4.799065),$ with the summing of the four objective values being 30.546081. In the iteration 3 of Mariano [11], they obtained four groups

$$\begin{aligned} (f_2^s, f_1^s, f_3^s, f_4^s) & := \{(7.06, 7.04, 12.75, 4.16); \\ & (6.74, 7.26, 12.13, 4.43); \\ & (6.72, 6.91, 13.12, 4.27); \\ & (6.61, 7.03, 12.76, 4.39)\} \end{aligned}$$

with the sums of the four objective values being 31.01; 30.56; 31.02; 30.79 respectively, and the DM chooses $(7.06, 7.04, 12.75, 4.16).$

Table 3.3. Numerical results for (P3.3).

s	$(\lambda_1^s, \lambda_2^s, \lambda_3^s, \lambda_4^s)$	$e(x^s)$	$(f_1^s, f_2^s, f_3^s, f_4^s)$	$f_1^s + f_2^s + f_3^s + f_4^s$
1	(0.50, 0.50, 0.50, 0.50)	0.00	(4.110485, 9.808986, 5.093159, 8.310570)	27.323199
2	(0.50, 1.00, 0.50, 1.00)	0.00	(7.215821, 7.674058, 9.779637, 4.762737)	29.432253
3	(0.50, 1.50, 0.50, 1.00)	0.00	(7.480056, 6.965056, 11.862982, 4.244595)	30.552688
4	(0.60, 1.50, 0.50, 1.00)	0.00	(6.844538, 6.563474, 13.309395, 4.335055)	31.052463
5	(0.60, 1.50, 0.55, 1.00)	0.00	(6.774419, 7.179232, 11.202925, 4.755234)	29.911810
6	(0.60, 1.60, 0.55, 1.00)	0.00	(6.292457, 6.723388, 12.731173, 4.799065)	30.546081

4. Conclusions

In this paper, using the nonlinear penalty function method with objective parameters, we present an interactive algorithm to solve the multi-objective programming with inequality constraints. With this algorithm, we can readily find out a satisfactory solution. When objective parameter M is increased, we may obtain a stable solution, but unsatisfactory. Then, by adopting different weights in the algorithm, we can go on interacting with computer and get many approximate different solutions, among which we can choose a satisfactory one. By the objective penalty function, new algorithms for multiobjective programming and bilevel multiobjective programming deserve further study.

5. Acknowledgements

This research is supported by the National Natural Science Foundation of China under grant 10971193 and the Natural Science Foundation of Zhejiang Province with grant Y6090063. We thank the referees for their helpful comments on a previous version of the paper.

6. References

- [1] R. Benayoun, J. de Montgoler, J. Tergny and O. Larichev, "Linear Programming with Multiple Objective Functions: Stem Method (STEM)," *Mathematical Programming*, Vol. 1, No. 3, 1971, pp. 355-375. [doi:10.1007/BF01584098](https://doi.org/10.1007/BF01584098)
- [2] A. M. Geoffrion, J. S. Dyer and A. Feinberg, "An Interactive Approach for Multi-Criterion Optimization, with an Application to the Operation of an Academic Department," *Management Science*, Vol. 19, No. 4, 1972, pp. 457-368. [doi:10.1287/mnsc.19.4.357](https://doi.org/10.1287/mnsc.19.4.357)
- [3] S. Zionts and J. Wallenius, "An Interactive Programming Method for Solving the Multiple Criteria Problem," *Management Science*, Vol. 22, No. 6, 1976, pp. 652-663. [doi:10.1287/mnsc.22.6.652](https://doi.org/10.1287/mnsc.22.6.652)
- [4] E. E. Rosinger, "Interactive Algorithm for Multiobjective Optimization," *Journal of Optimization Theory and Applications*, Vol. 35, No. 3, 1981, pp. 339-365. [doi:10.1007/BF00934907](https://doi.org/10.1007/BF00934907)
- [5] S. Zionts and J. Wallenius, "An Interactive Multiple for a Class of Underlying Nonlinear Utility Functions," *Management Science*, Vol. 29, No. 5, 1983, pp. 519-529. [doi:10.1287/mnsc.29.5.519](https://doi.org/10.1287/mnsc.29.5.519)
- [6] S. Sadagopan and A. Ravinderan, "An Interactive Algorithm for Multiple Criteria Nonlinear Programming Problems," *European Journal of Operational Research*, Vol. 25, No. 2, 1986, pp. 247-257. [doi:10.1016/0377-2217\(86\)90089-5](https://doi.org/10.1016/0377-2217(86)90089-5)
- [7] S. Helbig, "An Interactive Algorithm for Nonlinear Vector Optimization," *Applied Mathematics and Optimization*, Vol. 22, No. 1, 1990, pp. 147-151. [doi:10.1007/BF01447324](https://doi.org/10.1007/BF01447324)
- [8] M. Abd El-Hady Kassem, "Interactive Stability of Multiobjective Nonlinear Programming Problems with Fuzzy Parameters in the Constraints," *Fuzzy Sets and Systems*, Vol. 73, No. 2, 1995, pp. 235-243. [doi:10.1016/0165-0114\(94\)00317-Z](https://doi.org/10.1016/0165-0114(94)00317-Z)
- [9] B. Aghezzaf and T. Ouaderhman, "An Interactive Interior Point Algorithm for Multiobjective Linear Programming Problems," *Operations Research Letters*, Vol. 29, No. 4, 2001, pp. 163-170. [doi:10.1016/S0167-6377\(01\)00089-X](https://doi.org/10.1016/S0167-6377(01)00089-X)
- [10] M. A. Abo-Sinna and T. H. M. Abou-El-Enien, "An Interactive Algorithm for Large Scale Multiple Objective Programming Problems with Fuzzy Parameters through TOPSIS Approach," *Applied Mathematics and Computation*, Vol. 177, 2006, pp. 515-527. [doi:10.1016/j.amc.2005.11.030](https://doi.org/10.1016/j.amc.2005.11.030)
- [11] M. Luque, F. Ruiz and R. E. Steuer, "Modified Interactive Chebyshev Algorithm (MICA) for Convex Multiobjective Programming," *European Journal of Operational Research*, Vol. 204, No. 3, 2010, pp. 557-564. [doi:10.1016/j.ejor.2009.11.011](https://doi.org/10.1016/j.ejor.2009.11.011)
- [12] Z. Q. Meng, Q. Y. Hu and C. Y. Dang, "A Penalty Function Algorithm with Objective Parameters for Nonlinear Mathematical Programming," *Journal of Industrial and Management Optimization*, Vol. 5, No. 3, 2009, pp. 585-601. [doi:10.3934/jimo.2009.5.585](https://doi.org/10.3934/jimo.2009.5.585)
- [13] F. H. Clarke, "Optimization and Nonsmooth Analysis," John-Wiley & Sons, New York, 1983.