Turkish Journal of Mathematics

http://journals.tubitak.gov.tr/math/

Research Article

Turk J Math
(2018) 42: 1031 – 1048

doi:10.3906/mat-1702-12

Generation of efficient and ϵ-efficient solutions in multiple objective linear programming

Zohra Sabrina DELHOUM[1], **Sonia RADJEF**[2,3,*], **Fatima BOUDAOUD**[1]
[1]Department of Mathematics, Faculty of Exact and Applied Sciences, Mathematics and Applications Laboratory, University of Oran 1, Ahmed Ben Bella, Oran, Algeria
[2]Department of Mathematics, Faculty of Mathematics and Computer Science, Oran University of Science and Technology - Mohamed Boudiaf (USTO-MB), El M'naouer, Oran, Algeria
[3]Lamos Research Unit, University of Bejaia, Bejaia, Algeria

Received: 04.02.2017 • **Accepted/Published Online:** 25.09.2017 • **Final Version:** 08.05.2018

Abstract: We develop an algorithm to solve a multiple objective linear programming problem with bounded variables. It is based on the scalarization theorem of optimal solutions of multiobjective linear programs and the single objective adaptive method. We suggest a process for the search for the first efficient solution without having to calculate a feasible solution, and we elaborate a method to generate efficient solutions, weakly efficient solutions, and ϵ-efficient solutions. Supporting theoretical results are established and the method is demonstrated on a numerical example.

Key words: Multiobjective linear program, bounded variables, ϵ-optimality criterion, adaptive method, efficiency, weak efficiency, ϵ-efficiency

1. Introduction

Multiobjective programming has applications in many academic areas such as in the fields of engineering, economics, mining, life sciences, and finance. It has been a topic of research since the 1960s. However, while this type of optimization has great interest, the simplex algorithm has long been the most exact method used to solve multiobjective linear programming problems [7]. More recently, algorithms to solve these types of programs in objective space have been developed [2]. Since the discovery of interior point algorithms to solve linear programs in polynomial time, efforts to apply such methods to deal with multiobjective programming are evident. Another focus of interest is based on scalarization. Duality theory for multiobjective linear programming has been also studied.

On the other hand, single linear programming with bounded variables models many problems in real situations and major effort has been made in solving such problems. It is also clear that the multiple objective paradigm appears naturally in this context. The study of the combination of these two areas of optimization is therefore relevant; it is the main objective of this paper.

In parallel, in the 1980s, Loridan [12] introduced the notion of ϵ-efficient solutions for multiobjective programs. White [16] then proposed several concepts of approximate solutions for these programs and drafted methods for their generation. In the last decades, ϵ-efficient solutions have been examined in the literature by

*Correspondence: soniaradjef@yahoo.fr
2000 *AMS Mathematics Subject Classification:* 90C05, 65K05, 90C29

many authors. This paper deals with the generation of ϵ-efficient solutions in the case of multiobjective linear programming with bounded variables.

Following the preceding work in [15], a new method is proposed. The authors develop a method to solve a multiple objective linear programming problem with upper and lower bounded decision variables. The method is an extension of the direct support method [14, 15], known in single objective linear programming. This method is considered as an intermediate method between the interior methods and the simplex one. Indeed, using the direct support method, the initial feasible solution can be an extreme point, an interior point, or any point on the edge. This method [11] is usually used to solve linear programs with bounded variables.

In [15], the authors suggested an efficiency test of a nonbasic variable and a new process to search the initial efficient extreme point. Using the direct support method principle, they prescribed an algorithm to generate all the efficient extreme points, the ϵ-efficient extreme points, and the ϵ-weakly efficient extreme points of the problem.

The authors, in [15], used the simplex metric when solving mono-objective programs.

In this work, we use the principle of the adaptive method [10], which is considered more general than the direct support method in single objective linear programming. Indeed, we suggest to use another metric called the adapted metric, where we consider all suboptimal indexes by which we build improvement of the objective function and the maximum step along this direction. This method avoids the preliminary transformation of the constraints and it enables us to treat the problem as it stands without making any changes. It manipulates the bounds as they are initially expressed. It is easy to use and it generates a large benefit in time and memory space. Comparisons with algorithms well known in the field of single objective programming [10, 11] have shown that this method is more efficient than the other approaches in this context.

We also exploit the suboptimality criterion of this adaptive method to find ϵ-efficient solutions of our multiple objective problem.

Another contribution of our work is that we propose a new process to search for the first efficient solution without calculating a feasible solution.

The article is structured as follows. First, we state our purpose and some definitions. The basic concepts of efficiency, of ϵ-efficiency, and their proprieties are established in Section 3 and Section 4. In Section 5 we give a process to find a first efficient solution. Section 6 deals with the development of a method to generate the efficient solutions. The detailed algorithm is given in Section 7 and a numerical example to demonstrate the applicability of the suggested method is given in Section 8. Finally, we give a conclusion in Section 9.

2. Basic terminology and problem formulation

We consider the following multiobjective linear programming problem with bounded variables:

$$\begin{cases} \max Z(x) = Cx, \\ x \in S. \end{cases} \tag{2.1}$$

The feasible set is defined as follows:

$$S = \{x \in \mathbb{R}^n : Ax = b,\, d_1 \leq x \leq d_2\}. \tag{2.2}$$

The indices of constraints and decision variables are respectively denoted by:
$I = \{1, 2, \cdots, m\}$, $J = \{1, 2, \cdots, n\}$, such that:
$J = J_B \cup J_N$ with $J_B \cap J_N = \phi$, $|J_B| = m$.

Thus, $x = x(J) = (x_j, j \in J)$, $d_1 = d_1(J) = (d_{1_j}, j \in J)$, and $d_2 = d_2(J) = (d_{2_j}, j \in J)$ are n-vectors; $b = (b_i, i \in I)$ is an m-vector; and $A = A(I, J)$ is an $m \times n$-matrix, such that $\text{rang}A = m < n$.

We define the criterion function as follows:

$$Z(x) = \begin{pmatrix} z_1(x) \\ z_2(x) \\ \vdots \\ z_p(x) \end{pmatrix} = \begin{pmatrix} c_1^T x \\ c_2^T x \\ \vdots \\ c_p^T x \end{pmatrix} = Cx, \tag{2.3}$$

where $C = C(K, J)$ with $K = \{1, 2, \cdots, p\}$ is a $p \times n$-matrix, whose rows are the n-vectors $c_k^T, k \in K$.

Then we can split the vectors and matrices as follows:

$$x = \begin{pmatrix} x_B \\ x_N \end{pmatrix}, \quad x_B = x(J_B) = (x_j,\ j \in J_B), \quad x_N = x(J_N) = (x_j,\ j \in J_N),$$

$$C = \begin{pmatrix} C_B \\ - \\ C_N \end{pmatrix}, C_B = C(K, J_B), \quad C_N = C(K, J_N),$$

$$A = A(I, J) = (a_{ij},\ 1 \leq i \leq m, \quad 1 \leq j \leq n), A = (A_B | A_N), \quad A_B = A(I, J_B), \quad A_N = A(I, J_N).$$

We give the following definitions:

- The set $J_B \subset J$, $|J_B| = m$, is said to be *support* if $\det A_B = \det A(I, J_B) \neq 0$.
- The couple $\{x, J_B\}$ formed by the feasible solution x and the support J_B is called *the support feasible solution*. It is said to be *nondegenerate* if

$$d_{1_j} < x_j < d_{2_j}, \quad \forall j \in J_B.$$

- The $p \times n$-matrix $E^T = C_B A_B^{-1} A - C$ is called *the reduced cost matrix*, where $E = (E_B^T, E_N^T)$, such that $E_B = E(K, J_B), E_N = E(K, J_N)$.
- The potential matrix is defined by: $U = C_B A_B^{-1}$.

Remark 2.1 *In this work, we suppose that S is a bounded set and the problem is nondegenerate; therefore, all the feasible solutions have at least m noncritical components, with $m = rang(A)$.*

Remark 2.2 *The support feasible solution is a more general concept than one of the basic feasible solutions. A support feasible solution can be an interior point, a boundary point, or an extreme point of S, while a basic feasible solution is always an extreme point.*

3. Efficient solutions and their properties

Definition 3.1 *A feasible decision $x^0 \in \mathbb{R}^n$ is said to be an efficient solution (or Pareto optimal solution) for the problem (2.1) if there is no other feasible solution $x \in S$ such that $Cx \geq Cx^0$ and $Cx \neq Cx^0$.*

Let S^E be the efficient solutions set of the problem (2.1).

Definition 3.2 *A feasible decision $x^0 \in \mathbb{R}^n$ is said to be a weakly efficient solution (or Slater optimal solution) for the problem (2.1) if there is no other feasible solution $x \in S$ such that $Cx > Cx^0$.*

Then we recall the following classical theorems:

Theorem 3.3 *A feasible decision x^0 is efficient if and only if:*

$$\exists \lambda \in \mathbb{R}^p, \lambda > 0 : \lambda^T C x^0 = \max_{x \in S} \lambda^T C x.$$

Theorem 3.4 *A feasible decision x^0 is weakly efficient if and only if:*

$$\exists \lambda \in \mathbb{R}^p, \lambda \geq 0 : \lambda^T C x^0 = \max_{x \in S} \lambda^T C x.$$

Theorem 3.5 *(Efficiency criterion) Let $\{x, J_B\}$ be a support feasible solution for the problem (2.1) and $k \in K$.*

If

$$\begin{cases} E_{kj} \geq 0, & \text{if } x_j = d_j^-, \quad j \in J_N, \\ E_{kj} \leq 0, & \text{if } x_j = d_j^+, \quad j \in J_N, \\ E_{kj} = 0, & \text{if } d_j^- < x_j < d_j^+, \quad j \in J_N, \end{cases} \tag{3.1}$$

then x is a weakly efficient solution for the problem (2.1).

If the support feasible solution is nondegenerate, then those relations are also necessary to have x weakly efficient.

For a discussion of some basic theoretical properties and other approaches to the problem see, for example, the references [1, 5, 8, 9, 15].

Multiple objective linear programming with bounded variables involves determining the whole set of the efficient and all weakly efficient solutions of the problem (2.1) for given C, A, b, d_1 and d_2.

Our method also determines ϵ-efficient solutions.

4. ϵ-Efficient solutions and their properties

Definition 4.1 *Let $\epsilon \in \mathbb{R}^p, \epsilon \geq 0$. A feasible decision $x^\epsilon \in S$ is said to be ϵ-efficient for the problem (2.1) if there exists an efficient feasible solution $x \in S$ such that $c_k^T x - c_k^T x^\epsilon \leq \epsilon_k, \forall k \in K$.*

Let S_ϵ^E be the ϵ-efficient solutions set of the problem (2.1).

The following proprieties are the direct results of the definition of ϵ-efficient solutions:

Proposition 4.2 *1. $S^E \subset S_\epsilon^E, \forall \epsilon > 0$ and $S^E = S_\epsilon^E$, for $\epsilon = 0$.*

2. *If* $\epsilon_1 > \epsilon_2 > 0$, *then* $S^E_{\epsilon_2} \subset S^E_{\epsilon_1}$.

Lemma 4.3 *[3] A feasible solution* $x^\epsilon \in S$ *is said to be* ϵ*-efficient for the problem (2.1) if and only if there exists an efficient solution* $x^0 \in S$ *such that for all vectors* $\lambda \in \mathbb{R}^p_+$, $\sum_{k=1}^p \lambda_k = 1$, *satisfying the condition* $\lambda^T Cx^0 = \max_{x\in S} \lambda^T Cx$, *the following inequality holds:*

$$\lambda^T(Cx^0 - Cx^\epsilon) \leq \epsilon.$$

Definition 4.4 *The value*

$$\beta_k(x, J_B) = \sum_{j\in J_N, E_{kj}>0} E_{kj}(x_j - d_j^-) + \sum_{j\in J_N, E_{kj}<0} E_{kj}(x_j - d_j^+)$$

is called the ϵ*-efficiency formula of the objective* k, $k \in K$.

Theorem 4.5 *(Characterization of an* ϵ*-efficient solution)*

Let $\{x, J_B\}$ *be a support feasible solution of the problem (2.1) and* ϵ *an arbitrary vector of* $\mathbb{R}^p_+$.

If there exists $k \in \{1, \dots, p\}$ *such as* $\beta_k(x, J_B) \leq \epsilon_k$, *then* x *is* ϵ_k*-weakly efficient for the problem (2.1).*

If $\beta(x, J_B) = (\beta_k(x, J_B), k \in \{1, \dots, p\}) \leq \epsilon$, *then* x *is* ϵ*-efficient.*

5. Finding an initial efficient point

5.1. Finding an initial efficient point using Isermann's procedure

Inspired by the Isermann's procedure to find an initial efficient solution, we give a procedure by taking into account the specificity of the constraints of the problem (2.1). The single linear programs introduced in the procedure will be resolved using the adaptive method [10, 13].

The procedure is given in the following steps:

Step (1): Get the optimal solution $(u^0, v^0, \gamma^0, \alpha^0, \beta^0)$ for the following linear program:

$$\begin{cases} \min(b - Ad_1)^T u + (d_2 - d_1)^T \gamma, \\ u^T A - v^T C + \gamma^T - \alpha^T = 0, \\ v - \beta = e, \\ \alpha, \gamma, \beta \geq 0, \end{cases} \tag{5.1}$$

where $e \in \mathbb{R}^p$ is a vector whose entries are each unity. Then go to step (2).

Otherwise, stop the process.

Step (2): Get an efficient solution for the problem (2.1) by finding the optimal solution for the following program:

$$\begin{cases} \max(v^0)^T Cx, \\ Ax = b, \\ d_1 \leq x \leq d_2, \end{cases} \tag{5.2}$$

using the adaptive method [10].

To establish the validity of this procedure, we prove the following results.

Theorem 5.1 *If the program (5.1) admits an optimal solution* $(u^0, v^0, \gamma^0, \alpha^0, \beta^0)$*, then the program (5.2) admits an optimal solution. Besides, this solution is efficient for the program (2.1).*

Proof Let $\lambda \in \mathbb{R}^p, \lambda > 0,$ and consider the following linear program:

$$\begin{cases} \max \lambda^T Cx, \\ Ax = b, \\ d_1 \leq x \leq d_2. \end{cases} \tag{5.3}$$

We put $y = x - d_1$ in the linear program (5.3). We have:

$$\begin{cases} \max \lambda^T Cy + \lambda^T Cd_1, \\ Ay = b - Ad_1, \\ y \leq d_2 - d_1, \\ y \geq 0. \end{cases} \tag{5.4}$$

Let $(u^0, v^0, \gamma^0, \alpha^0, \beta^0)$ be the optimal solution of (5.1). The dual of the program (5.2) is given by

$$\begin{cases} \min u^T(b - Ad_1) + \gamma^T(d_2 - d_1), \\ u^T A + \gamma^T \geq (v^0)^T C, \\ \gamma \geq 0. \end{cases} \tag{5.5}$$

As $(u^0, v^0, \gamma^0, \alpha^0, \beta^0)$ is an optimal solution of the program (5.1), then (u^0, γ^0) is a feasible solution for the program (5.5). Since the set S is not empty, the program (5.2) is feasible. From the duality theory, the program (5.4), with $\lambda = v^0$, admits an optimal solution. □

The following theorem allows us to find an efficient solution for the multiple objective program (2.1) by solving one linear program with bounded variables.

Theorem 5.2 *[4, 6] The following linear program:*

$$\begin{cases} \max e^T Cx, \\ x \in S, \end{cases} \tag{5.6}$$

has an optimal solution if and only if the multiple objective program (2.1) admits an efficient solution.

Theorem 5.3 *The program (5.1) has an optimal solution if and only if the multiple objective program (2.1) admits an efficient solution.*

Proof We give the dual program of (5.1) by

$$\begin{cases} \max e^T z, \\ Ay = b - Ad_1, \\ -Cy + z = 0, \\ y \leq d_2 - d_1, \\ y \geq 0, \\ z \geq 0. \end{cases} \tag{5.7}$$

If we put $y + d_1 = x$, then we get the following program:

$$\begin{cases} \max e^T z, \\ Ax = b, \\ z = Cx - Cd_1, \\ d_1 \leq x \leq d_2, \\ z \geq 0. \end{cases} \tag{5.8}$$

We have $z = Cx - Cd_1 \geq 0$, and the program (5.8) is equivalent to the following one:

$$\begin{cases} \max e^T Cx - e^T Cd_1, \\ Ax = b, \\ d_1 \leq x \leq d_2. \end{cases} \tag{5.9}$$

However, as $e^T Cd_1$ is a constant value, then the program (5.9) is equivalent to the program (5.6).

By applying Theorem 5.2 and according to the duality theory, the theorem is established. □

The following theorem describes a class of multiobjective linear problems for which Isermann's procedure is valid.

Theorem 5.4 *Isermann's method is valid if* $T = \{x \in S, Cx \geq 0\}$ *is nonempty.*

We propose to use the new method for generating an initial efficient point presented in the next section.

5.2. Finding an initial efficient point using the Benson–Radjef–Bibi procedure

The authors in [15] developed a procedure to find an initial efficient solution. However, in their method, they used the direct support method to solve the single linear programs introduced in the procedure. Here, we use the same procedure but we solve the single linear programs using the adaptive method [10, 13].

The procedure is given by the following steps:

Step (1): Find a feasible solution $x^0 \in S$.

Step (2): If $S \neq \emptyset$, find the optimal solution $(u^0, w^0, \gamma^0, \alpha^0)$ of the following linear program:

$$\begin{cases} \min u^T(-Cx^0 + Cd_1) + w^T(b - Ad_1) + \gamma^T(d_2 - d_1), \\ u^T C - w^T A - \gamma^T + \alpha^T = -e^T C, \\ u, \alpha, \gamma \geq 0, \end{cases} \tag{5.10}$$

and go to step (3).

Otherwise, stop the process, as the problem (2.1) is unfeasible.

Step (3): Get an efficient solution for the problem (2.1) by finding the optimal solution of the following program using the adaptive method [10]:

$$\begin{cases} \max(u^0+e)^T Cx, \\ Ax = b, \\ d_1 \leq x \leq d_2. \end{cases} \tag{5.11}$$

6. Generating efficient points

In this phase, we use the adaptive method principle. Starting from the first efficient solution, we calculate a neighbor solution, and we test whether it is efficient. If it is not, we return to another efficient solution and we reiterate the process. Then a test of efficiency of a nonbasic variable is necessary.

6.1. Efficiency test of a nonbasic variable

To test the efficiency of a solution x^* in the multiple objective linear program (2.1), we use the test developed by Radjef and Bibi [13, 15]. Here, the solution of the single objective programs is done using the adaptive method. We introduce the vector v of dimension p and we define the following linear program:

$$\begin{cases} \max g = e^T v, \\ Ax = b, \\ Cx - v = Cx^*, \\ d_1 \leq x \leq d_2, \\ v \geq 0. \end{cases} \tag{6.1}$$

Theorem 6.1 *If* $\max g = 0$*, then* x^* *is efficient. Otherwise,* x^* *is not efficient.*

6.2. Construction of the new efficient solution

To find the efficient solution, we introduce into the basis, one by one, the nonbasic variables of the initial efficient solution found in the first phase.

To calculate the new efficient solution $\overline{x} = x + \theta^0 l$, we choose a direction of improvement $l \in \mathbb{R}^n$ and a maximum step θ^0 along this direction such as $z_{k_0}(\overline{x}) \geq z_{k_0}(x)$.

Let k_0 be the criterion satisfying the following relation:

$$\sum_{j\in J_N} |E_{k_0 j}| = \max_{k=\overline{1,p}} \left(\sum_{j\in J_N} |E_{kj}| \right).$$

In J_N, we set $\theta = 1$, and we put

$$l_j = \begin{cases} d_{1_j} - x_j, & \text{if } E_{k_0 j} > 0, \\ d_{2_j} - x_j, & \text{if } E_{k_0 j} < 0, \\ 0 & \text{if } E_{k_0 j} = 0, \quad j \in J_N. \end{cases} \tag{6.2}$$

In J_B, we put $l(J_B) = -A_B^{-1} A_N l(J_N)$ to get $A\overline{x} = b$, so that $\overline{x}$ satisfies $d_1 \leq \overline{x} \leq d_2$. On the other hand, the maximum step θ^0 along the direction l must verify

$$d_{1_j} - x_j \leq \theta^0 l_j \leq d_{2_j} - x_j,\ j \in J_B.$$

Therefore, we have

$$\theta_j = \begin{cases} \frac{d_{2_j} - x_j}{l_j}, & \text{if } \ l_j > 0, \\ \frac{d_{1_j} - x_j}{l_j}, & \text{if } \ l_j < 0, \\ \infty, & \text{if } \ l_j = 0, \quad j \in J_B, \end{cases}$$

with

$$\theta_{j_0} = \min_{j \in J_B} \theta_j,$$

where j_0 is the candidate index to come out of the basis.

The maximal step is

$$\theta^0 = \min(1, \theta_{j_0}).$$

The new feasible solution is $\bar{x} = x + \theta^0 l$.

6.2.1. Calculation of $\beta(\bar{x}, J_B)$

For $k \in \{1, \cdots, p\}$, we have:

$$\begin{aligned} \beta_k(\bar{x}, J_B) &= \sum_{E_{kj} > 0, j \in J_N} E_{kj}(\bar{x}_j - d_{1_j}) + \sum_{E_{kj} < 0, j \in J_N} E_{kj}(\bar{x}_j - d_{2_j}) \\ &= \beta_k(x, J_B) + \theta^0 \left(\sum_{E_{kj} > 0, j \in J_N} E_{kj} l_j + \sum_{E_{kj} < 0, j \in J_N} E_{kj} l_j \right). \end{aligned}$$

Then

$$\beta_{k_0}(\bar{x}, J_B) = \beta_{k_0}(x, J_B) + \theta^0 \left(\sum_{E_{k_0 j} > 0, j \in J_N} E_{k_0 j} l_j + \sum_{E_{k_0 j} < 0, j \in J_N} E_{k_0 j} l_j \right).$$

We replace l_j given by the relations (6.2). Thus, we have

$$\beta_{k_0}(\bar{x}, J_B) = (1 - \theta^0) \beta_{k_0}(x, J_B).$$

From this expression, we deduce that:

- If $\theta^0 = 1$ then $\overline{x}$ is optimal for the objective k_0; we also say that $\overline{x}$ is Slater efficient. We consider all the nonbasic variables.
- If $\beta_{k_0}(\bar{x}, \bar{J}_B) < \epsilon_{k_0}$, we have $\bar{x}$ an ϵ-optimal for the objective k_0; we also say that $\bar{x}$ is ϵ-weakly efficient.
- If $\beta_{k_0}(\bar{x}, \bar{J}_B) > \epsilon_{k_0}$, we start a new iteration with the new support solution $\{\bar{x}, \bar{J}_B\}$, and we change J_B as follows:

6.2.2. Change of support

Changing support consists of the change of E to $\overline{E}$, and U to $\overline{U}$, so that:

$$\beta_{k_0}(\bar{x}, J_B) \leq \beta_{k_0}(\bar{x}, \bar{J}_B).$$

To this end, we put:

$$\bar{E}_{k_0j} = E_{k_0j} + \sigma_0 t_j,$$

$$\bar{U}_{k_0j} = U_{k_0j} + \sigma_0 t_j,$$

where t is the reduction direction of the dual function and σ_0 is the maximal step along this direction.

Calculation of t and σ_0 :

The step θ^0 is given by:

$$\theta^0 = \min(1, \theta_{j_0}) = \theta_{j_0},\ j_0 \in J_B.$$

We seek an index $j_1 \in J_N$, which will enter into the basis instead of j_0.

To this end, we put:

$$t_j = \begin{cases} -sign(l_{j_0}), & \text{if } \ j = j_0, \\ 0, & \text{if } \ j \in J_B \backslash j_0, \end{cases}$$

$$t(J_N) = t(J_B) A_B^{-1} A_N,$$

and we calculate

$$\sigma_0 = \sigma_{j_1} = \min_{j \in J_N}(\sigma_j),$$

with

$$\sigma_j = \begin{cases} \frac{-E_{k_0j}}{t_j}, & \text{if} \qquad E_{k_0j} t_j < 0, \\ 0, & \text{if} \qquad (E_{k_0j} = 0 \text{ and } \ x_j \neq d_{1_j} \text{ for } \ t_j > 0) \ \text{ or} \\ & \qquad\qquad (E_{k_0j} = 0 \text{ and } \ x_j \neq d_{2_j} \text{ for } \ t_j < 0),\ j \in J_N, \\ \infty, & \text{else .} \end{cases}$$

We have $\overline{E}_{k_0j_1} = 0$.

The new support is $\overline{J}_B = (J_B \setminus j_0) \cup j_1$.

We can notice that:

$$\begin{aligned} \beta_{k_0}(\bar{x}, \bar{J}_B) &= \sum_{\bar{E}_{k_0j} > 0, j \in \overline{J}_N} \bar{E}_{k_0j}(\bar{x}_j - d_{1_j}) + \sum_{\bar{E}_{k_0j} < 0, j \in \overline{J}_N} \bar{E}_{k_0j}(\bar{x}_j - d_{2_j}) \\ &= (1 - \theta^0)\beta_{k_0}(x, J_B) + \sigma_0 \left(\sum_{E_{k_0j} > 0, j \in \overline{J}_N} t_j(\bar{x}_j - d_{1_j}) + \sum_{E_{k_0j} < 0, j \in \overline{J}_N} t_j(\bar{x}_j - d_{2_j}) \right). \end{aligned}$$

From this expression, we deduce that:

- If $\beta_{k_0}(\bar{x}, \bar{J}_B) > \epsilon_{k_0}$, we call the test program to verify the efficiency of this solution; if it is efficient, we will begin a new iteration with the new support feasible solution $\{\bar{x}, \bar{J}_B\}$.

- If $\beta_{k_0}(\bar{x}, \bar{J}_B) < \epsilon_{k_0}$, we have $\bar{x}$ an ϵ-optimal solution for the objective k_0; we also say that $\bar{x}$ is ϵ-weakly efficient.
- If $\beta_{k_0}(\bar{x}, \bar{J}_B) = 0$, then we find the Slater efficient solution. Then we consider another nonbasic variable to start a new iteration.

7. Algorithm of the method

The steps of the method to search for the efficient solutions are given in the following algorithm:

I. Find the first efficient solution by using the following procedure:

- Find an optimal solution $(u^0, w^0, \gamma^0, \alpha^0)$ of the program (5.10).
- Obtain an optimal solution solution of the program (5.11).

  Let x^1 be the obtained solution, go to II.

II. Set $s = 1$:

(1) Let $\{x^s, J_B\}$ a support feasible solution and $\epsilon \geq 0$.

- Calculate $U = C_B A_B^{-1}$.
- Calculate $E = (E_B, E_N) = UA - C$.

(2) Choose the criterion k_0.

- Calculate $\beta_{k_0}(x^s, J_B)$:
  - If $\beta_{k_0}(x^s, J_B) > \epsilon_{k_0}$, go to (3).
  - If $\beta_{k_0}(x^s, J_B) < \epsilon_{k_0}$, then x^s is ϵ_{k_0}-weakly efficient, go to (6).
  - If $\beta_{k_0}(x^s, J_B) = 0$, then x^s is weakly efficient, go to (5).

(3) Calculate the new feasible solution by using the following procedure:

- Calculate the vector l.
- Determine the index j_0 and the maximal step θ^0.
- Set $s = s + 1$.
- Calculate $x^s = x^{s-1} + \theta^0 l$, s^{th} feasible solution.
- Calculate $\beta_{k_0}(x^s, J_B) = (1 - \theta^0)\beta_{k_0}(x^{s-1}, J_B)$:
  - If $\beta_{k_0}(x^s, J_B) > \epsilon_{k_0}$, go to (4).
  - If $\beta_{k_0}(x^s, J_B) < \epsilon_{k_0}$, then x^s is ϵ_{k_0}-weakly efficient, go to (6).
  - If $\theta^0 = 1$, then x^s is weakly efficient solution, go to (5).

(4) Change the support:

- Calculate the vector t.
- Calculate σ_0, and determine index j_1.
- Determine the new support $\overline{J}_B = (J_B \setminus j_0) \cup j_1$.
- Calculate $\beta(x^s, \bar{J}_B)$.
  - If $\beta_{k_0}(x^s, \overline{J}_B) > \epsilon_{k_0}$, go to (7).

  - If $\beta_{k_0}(x^s, \overline{J}_B) < \epsilon_{k_0}$, then x^s is ϵ_{k_0}-weakly efficient, go to (6).
  - If $\beta_{k_0}(x^s, \overline{J}_B) = 0$, then x^s is weakly efficient solution, go to (5).

(5) Introduction of the jth corresponding column leads to an unprocessed basis?

- If so, go to (6).
- Else, stop, all the solutions are found.

(6) Can we improve another objective?

- If so, go to (1).
- Else, stop, all the solutions are found.

(7) Consider the program (6.1) with $x^* = x^s$:

- If $\max g = 0$, the solution x^s is efficient.
- Else, go to (6).

8. Numerical example

Consider the following linear programming problem with bounded variables:

$$\begin{cases} \max z_1(x) = x_1 + 2x_2, \\ \max z_2(x) = x_1 - 2x_3, \\ \max z_2(x) = -x_1 + x_3, \\ x_1 + x_2 + x_4 = 1, \\ x_2 + x_5 = 2, \\ x_1 - x_2 + x_3 + x_6 = 4, \\ -1 \leq x_1 \leq 1, \\ -2 \leq x_2 \leq 2, \\ -3 \leq x_3 \leq 3, \\ -4 \leq x_4 \leq 4, \\ -5 \leq x_5 \leq 5, \\ -6 \leq x_6 \leq 6. \end{cases} \tag{8.1}$$

We have $x^0 = \begin{pmatrix} 0 & 0 & 0 & 1 & 2 & 4 \end{pmatrix}$ as a feasible solution.

I. Calculation of the first Pareto efficient solution:

First, we solve the program (5.10); by identification we have:

$$\begin{cases} \min(-u_1 + 5u_2 - 2u_3 + 8w_1 + 9w_2 + 12w_3 + 2\gamma_1 + 4\gamma_2 + 6\gamma_3 + 8\gamma_4 + 10\gamma_5 + 12\gamma_6), \\ u_1 + u_2 - u_3 - w_1 - w_3 - \gamma_1 + \alpha_1 = -1, \\ 2u_1 - w_1 - w_2 + w_3 - \gamma_2 + \alpha_2 = -2, \\ -2u_2 + u_3 - w_3 - \gamma_3 + \alpha_3 = 1, \\ -w_1 - \gamma_4 + \alpha_4 = 0, \\ -w_2 - \gamma_5 + \alpha_5 = 0, \\ -w_3 - \gamma_6 + \alpha_6 = 0, \\ u \geq 0, \alpha \geq 0, \gamma \geq 0. \end{cases}$$

To solve this program, we put:

$w = w^+ - w^-$ with $w^+ = \max(0, w)$ and $w^- = \max(0, -w)$,

and so we get the following program:

$$\begin{cases} \min(-u_1 + 5u_2 - 2u_3 + 8w_1^+ + 9w_2^+ + 12w_3^+ - 8w_1^- - 9w_2^- - 12w_3^- + 2\gamma_1 + 4\gamma_2 + 6\gamma_3 + 8\gamma_4 + 10\gamma_5 + 12\gamma_6), \\ u_1 + u_2 - u_3 - w_1^+ - w_3^+ + w_1^- + w_3^- - \gamma_1 + \alpha_1 = -1, \\ 2u_1 - w_1^+ - w_2^+ + w_3^+ + w_1^- + w_2^- - w_3^- - \gamma_2 + \alpha_2 = -2, \\ -2u_2 + u_3 - w_3^+ + w_3^- - \gamma_3 + \alpha_3 = 1, \\ -w_1^+ + w_1^- - \gamma_4 + \alpha_4 = 0, \\ -w_2^+ + w_2^- - \gamma_5 + \alpha_5 = 0, \\ -w_3^+ + w_3^- - \gamma_6 + \alpha_6 = 0, \\ u \geq 0, w^+ \geq 0, w^- \geq 0, \alpha \geq 0, \gamma \geq 0. \end{cases}$$

The optimal solution to this program is:

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

To obtain the first Pareto efficient solution, we solve the following program:

$$\begin{cases} \max(u^0 + e)^T Cx, \\ Ax = b, \\ d_1 \leq x \leq d_2. \end{cases}$$

Identification yields:

$$\begin{cases} \max(2x_2), \\ x_1 + x_2 + x_4 = 1, \\ x_2 + x_5 = 2, \\ x_1 - x_2 + x_3 + x_6 = 4, \\ -1 \leq x_1 \leq 1, \\ -2 \leq x_2 \leq 2, \\ -3 \leq x_3 \leq 3, \\ -4 \leq x_4 \leq 4, \\ -5 \leq x_5 \leq 5, \\ -6 \leq x_6 \leq 6. \end{cases}$$

The optimal solution to this program by the adaptive method is: $\begin{pmatrix} 1 & 2 & -1 & -2 & 0 & 6 \end{pmatrix}$. Then our first Pareto efficient solution is: $\begin{pmatrix} 1 & 2 & -1 & -2 & 0 & 6 \end{pmatrix}$.

We set $J_B = \{3, 4, 5\}$ and $J_N = \{1, 2, 6\}$, and go to II.

II. Search all Pareto efficient solutions associated to our first Pareto efficient solution:

First iteration: Let $\{x^1, J_B\}$ be a support feasible solution and $\epsilon = \begin{pmatrix} 0.01 \\ 0.01 \\ 0.01 \end{pmatrix}$.

- Calculate the reduced cost matrix E:

  $$E = C_B A_B^{-1} A - C.$$

  We have:

  $$E_N = \begin{pmatrix} -1 & -2 & 0 \\ -3 & 2 & -2 \\ 2 & -1 & 1 \end{pmatrix} \text{ and } E_B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

  Thus,

  $$E = \begin{pmatrix} -1 & -2 & 0 & 0 & 0 & 0 \\ -3 & 2 & 0 & 0 & 0 & -2 \\ 2 & -1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

- Determine the criterion k_0 such as:

  $$\sum_{j \in J_N} |E_{k_0 j}| = \max(\sum_{j \in J_N} |E_{1j}|, \sum_{j \in J_N} |E_{2j}|, \sum_{j \in J_N} |E_{3j}|) = \sum_{j \in J_N} |E_{2j}| = 7,$$

  so $k_0 = 2$.

- We have $\beta_2(x^1, J_B) = 8 > \epsilon_2$, and then x^1 is not optimal for the objective $k_0 = 2$.

- Calculate the direction of improvement l:

$$\begin{cases} l_N = \begin{pmatrix} l_1 & l_2 & l_6 \end{pmatrix} = \begin{pmatrix} 0 & -4 & 0 \end{pmatrix}, \\ l_B = \begin{pmatrix} l_3 & l_4 & l_5 \end{pmatrix} = \begin{pmatrix} -4 & -4 & 4 \end{pmatrix}. \end{cases}$$

- Calculate the maximum step:

$$\theta_{j_0} = \min_{j \in J_B} \theta_j = \min(\theta_3, \theta_4, \theta_5) = \theta_3 = 0.5,$$

so $j_0 = 3$ and $\theta^0 = \min(1, \theta_{j_0}) = 0.5$.

- Calculate x^2:

$$x^2 = x^1 + \theta^0 l = \begin{pmatrix} 1 & 0 & -3 & 0 & 2 & 6 \end{pmatrix}$$

- Calculate $\beta_2(x^2, J_B) = 4 > \epsilon_2$, so $\{x^2, J_B\}$ is not optimal for the objective $k_0 = 2$.

  Changing the support:

- $t = \begin{pmatrix} 1 & -1 & 1 & 0 & 0 & 1 \end{pmatrix}$,

- $\sigma = \begin{pmatrix} 3 & 2 & 0 & 0 & 0 & 2 \end{pmatrix}$,

- $\sigma_0 = \sigma_{j_1} = \sigma_2 = 2$, so $j_1 = 2$.

  Our new support is: $J_B = (J_B \setminus j_0) \cup j_1 = \{2, 4, 5\}$.

  $\beta_2(x^2, J_B) = 0$, and then $\{x^2, J_B\}$ is optimal for the objective $k_0 = 2$, so x^2 is a Slater efficient solution for the problem (8.1).

- Consider the program (6.1). By identification, we have the following program:

$$\begin{cases} \max g = v_1 + v_2 + v_3, \\ x_1 + 2x_2 - v_1 = 1, \\ x_1 - 2x_3 - v_2 = 7, \\ -x_1 + x_3 - v_3 = -4, \\ x_1 + x_2 + x_4 = 1, \\ x_2 + x_5 = 2, \\ x_1 - x_2 + x_3 + x_6 = 4, \\ -1 \le x_1 \le 1, \\ -2 \le x_2 \le 2, \\ -3 \le x_3 \le 3, \\ -4 \le x_4 \le 4, \\ -5 \le x_5 \le 5, \\ -6 \le x_6 \le 6, \\ v \ge 0. \end{cases}$$

We found $\max g = 0$. Hence, the solution x^2 is Pareto efficient for our program (8.1).

Second iteration: Let $\{x^2, J_B\}$ be a support feasible solution. We have:

$$E_N = \begin{pmatrix} -3 & -2 & -2 \\ -1 & 2 & 0 \\ 1 & -1 & 0 \end{pmatrix} \text{ and } E_B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Thus,

$$E = \begin{pmatrix} -3 & 0 & -2 & 0 & 0 & -2 \\ -1 & 0 & 2 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \end{pmatrix}.$$

- Determine the criterion k_0 such as:

$$\sum_{j\in J_N} |E_{k_0 j}| = \max(\sum_{j\in J_N} |E_{1j}|, \sum_{j\in J_N} |E_{2j}|, \sum_{j\in J_N} |E_{3j}|) = \sum_{j\in J_N} |E_{1j}| = 7,$$

  so $k_0 = 1$.

- We have $\beta_1(x^2, J_B) = 12 > \epsilon_1$, and then x^2 is not optimal for the objective $k_0 = 1$.

- Calculate the direction of improvement l:

$$\begin{cases} l_N = \begin{pmatrix} l_1 & l_3 & l_6 \end{pmatrix} = \begin{pmatrix} 0 & 6 & 0 \end{pmatrix}, \\ l_B = \begin{pmatrix} l_2 & l_4 & l_5 \end{pmatrix} = \begin{pmatrix} 6 & -6 & -6 \end{pmatrix}. \end{cases}$$

- Calculate the maximum step:

$$\theta_{j_0} = \min_{j\in J_B} \theta_j = \min(\theta_2, \theta_4, \theta_5) = \theta_2 = \frac{1}{3},$$

  so $j_0 = 2$ and $\theta^0 = \min(1, \theta_{j_0}) = \frac{1}{3}$.

- Calculate x^3:

$$x^3 = x^2 + \theta^0 l = \begin{pmatrix} 1 & 2 & -1 & -2 & 0 & 6 \end{pmatrix}.$$

  We can notice that $x^3 = x^1$, so we can deduce that x^1 is optimal for the objective $k_0 = 1$. Moreover, x^3 is Pareto efficient for our program (8.1).

  Let $J_B = \{1, 2, 3\}$ be an unprocessed basis.

  Third iteration: Let $\{x^1, J_B\}$ be a support feasible solution.

  Calculate the reduced cost matrix E:

$$E_N = \begin{pmatrix} 1 & 1 & 0 \\ 3 & -5 & -2 \\ -2 & 3 & 1 \end{pmatrix} \text{ and } E_B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Thus,

$$E = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 & -5 & -2 \\ 0 & 0 & 0 & -2 & 3 & 1 \end{pmatrix}.$$

- Determine the criterion k_0 such as:

$$\sum_{j \in J_N} |E_{k_0 j}| = \max(\sum_{j \in J_N} |E_{1j}|, \sum_{j \in J_N} |E_{2j}|, \sum_{j \in J_N} |E_{3j}|) = \sum_{j \in J_N} |E_{2j}| = 10,$$

  so $k_0 = 2$. Thus, we cannot improve another objective and so we stop the algorithm.

- The Pareto efficient solutions found are:

$$x^1 = \begin{pmatrix} 1 & 0 & -3 & 0 & 2 & 6 \end{pmatrix} \text{ and } x^2 = \begin{pmatrix} 1 & 2 & -1 & -2 & 0 & 6 \end{pmatrix}.$$

9. Conclusion

This paper was devoted to developing a new method to solve a multiobjective linear program with bounded variables using the adaptive method, based on the use of an adapted metric. The constructed method generates a large benefit in time and memory space. This is supported by the fact that we use the adaptive method to solve our single linear programs, and it was showed that this method is very efficient, especially in the case of degenerate problems. It is worth noticing that the use of the simplex method is inappropriate for this kind of problem.

We first introduced a new procedure to find a first efficient solution. Subsequently, we developed a detailed procedure to calculate the efficient solutions and the weakly efficient solutions. We have also characterized the ϵ-efficient solutions for a multicriteria linear programming problem with bounded variables. We exploited the suboptimal criterion of the adaptive method in single objective programming to get the ϵ-efficient solutions and the ϵ-weakly efficient solutions to the problem. Moreover, we provided a global algorithm to describe major steps in implementing the method. Finally, we presented a numerical example showing the applicability of the method.

References

[1] Benayoun R, de Montgolfier J, Tergeny J, Larichev O. Linear programming with multiple objective functions: step method (STEM). Math Program 1971; 1: 366-375.

[2] Benson HP. An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem. J Glob Optim 1998; 13: 1-24.

[3] Dimkov MP. Research on the multicriteria linear programming problems. In: Gabasov R, Kirillova FM, editors. Optimal Control Problems. Moscrow, USSR: Nauka i Technika, 1981, pp. 25-42.

[4] Ecker JG, Hegner NS. On computing an initial efficient extreme point. J Oper Res Soc 1978; 29: 1005-1007.

[5] Ecker JG, Kouada IA. Finding efficient points for linear multiple objective programs. Math Program 1975; 8: 375-377.

[6] Ecker JG, Kouada IA. Finding all efficient extreme points for multiple objective linear programs. Math Program 1978; 14: 249-261.

[7] Ehrgott M, Wiecek MM. Multiobjective programming. In: Figueira J, Greco S, Ehrgott M, editors. Multiple Criteria Decision Analysis: State of the Art Surveys. New York, NY, USA: Springer, 2005, pp. 708.

[8] Evans JP, Steuer RE. A revised simplex method for linear multiple objective programs. Math Program 1973; 5: 54-72.

[9] Evans JP, Steuer RE. Generating efficient extreme points in linear multiple objective programming: two algorithms and computing experience. In: Cochrane JL, Zeleny M, editors. Multiple Criteria Decision Making. Columbia, SC, USA: University of South Carolina Press, 1973, pp. 349-365.

[10] Gabasov R. Adaptive Method of Linear Programming. Preprint. Karlsruhe, Germany: University of Karlsruhe, 1993.

[11] Gabasov R, Kirillova FM, Tyatyushkin AI. Constructive methods of optimization. Minsk, Belarus: P.I.-University Press, 1984.

[12] Loridan P. ϵ-Solutions in vector minimization problems. J Optimiz Theory App 1984; 43: 265-276.

[13] Radjef S. Application de la méthode adaptée aux problèmes multicritère. PhD, A Mira University, Bejaia, Algeria, 2011 (in French).

[14] Radjef S, Bibi MO. An effective generalization of the direct support method. Math Probl Eng 2011; 2011: 374390.

[15] Radjef S, Bibi MO. A new algorithm for linear multiobjective programming problems with bounded variables. Arab J Math 2014; 3: 79-92.

[16] White DJ. Epsilon efficiency. J Optimiz Theory App 1986; 49: 319-337.