Journal of Global Optimization **18:** 321–336, 2000.

Conical Algorithm in Global Optimization for Optimizing over Efficient Sets

NGUYEN V. THOAI
Department of Mathematics, University of Trier, 54286 Trier, Germany (e-mail: thoai@uni-trier.de)

Abstract. The problem of optimizing some contiuous function over the efficient set of a multiple objective programming problem can be formulated as a nonconvex global optimization problem with special structure. Based on the conical branch and bound algorithm in global optimization, we establish an algorithm for optimizing over efficient sets and discuss about the implementation of this algorithm for some interesting special cases including the case of biobjective programming problems.

Key words: Multiple objective programming, Biobjective programming, Optimizing over efficient sets, Global Optimization, Conical algorithm

1. Introduction

In general, the multiple objective programming problem can be formulated as follows:

$$\max\ c_i(x)\ (i = 1, \cdots, k),\ \text{s.t. } x \in X, \tag{MOP}$$

where X is a closed subset of $\mathbb{R}^n$ and $c_i(x)$ $(i = 1, \cdots, k)$ are continuous functions defined on X.

Let $c(x)$ be the vector function having components $c_i(x)$ $(i = 1, \cdots, k)$. A point $x \in X$ is called an *efficient* (or *nondominated* or *Pareto-optimal) solution* of Problem (MOP), if there is no point $y \in X$ such that $c(y) \geqslant c(x)$ and $c(y) \neq c(x)$.

The concept of efficient solutions plays a central role in multiple objective optimization, see, e.g., [23, 27]. One of the important and interesting approaches in multiple objective optimization is the problem of optimizing some function over the set of efficient solutions. More precisely, denoting by E_X the set of all efficient solutions of Problem (MOP), and letting f be a real-valued function defined on X, we consider the optimization Problem

$$\min \{f(x) : x \in E_X\}, \tag{P}$$

Optimizing over the efficient set is a very hard task. The main difficulty is that the efficient set, in general, is nonconvex, even in the case where the functions $c_i(x)$ $(i = 1, \cdots, k)$ are linear and X is a polyhefral set. The problem of optimizing over efficient sets has been first considered by Philip in [20]. Subsequently, because of its interesting mathematical aspects as well as its wide range of applications, this problem has attracted the attention of several authors, (cf. e.g., [1–10, 13, 14, 18, 19, 22, 25] and references given therein).

The purpose of this paper is to handle Problem (P) using numerical techniques in global optimization. One of the most promising approaches in global optimization is the branch and bound scheme. A realization of this general scheme called conical algorithm has been developed for solving concave minimizing problems and some related nonconvex problems (cf. [11, 12, 15–17, 26]). In [12], the conical algorithm has been implemented within a decomposition scheme for solving global optimization problems having some special structure. Based on this decomposition idea, we propose in the present paper a conical algorithm for Problem (P), which can be implemented for many interesting cases, in particular for the case of biobjective optimization problems, i.e., the case where in Problem (MOP) one has $k = 2$.

In the next section, we formulate Problem (P) as a global optimization problem with special structure, for which a conical algorithm is established in Section 3. Section 4 contains a very simple implementation of the conical algorithm for the case of biobjective optimization problems. Some preliminary computational results are reported in the last section.

2. Formulation as a global optimization problem with special structure

A weak form of efficient solutions is the concept of *weakly–efficient solutions*. A point $x \in X$ is called a *weakly–efficient solution* of Problem (MOP), if there is no point $y \in X$ such that $c(y) > c(x)$.

In order to construct a global optimization problem with special structure, we denote by W_X the set of all weakly–efficient solutions of Problem (MOP) and consider the following programming problem.

$$\min \{f(x) : x \in W_X\}. \tag{P'}$$

Problem (P′) is a relaxed form of Problem (P) and was considered e.g. in [2]. In [14], a conial algorithm has been presented for the case where $c_i(x)$, $i = 1, \dots, k$, are linear, X is a polyhedral set and $f(x)$ is a convex function. The algorithm given in [14] is in fact a special implementation of the general algorithm to be established in the present article. Some of following results can be found in [14]. For the completeness of presentation, however, they are sometimes recalled.

From Problem (MOP), we define a set in $\mathbb{R}^k$, the space of objective functions which is sometimes called the outcome space,

$$Z = \{z \in \mathbb{R}^k : z_i = c_i(x) \ (i = 1, \cdots, k), \ x \in X\}. \tag{1}$$

Further, for each $z \in \mathbb{R}^k$, define two subsets $L(z)$ and $L^+(z)$ of $\mathbb{R}^k$ by

$$L(z) = \{v \in \mathbb{R}^k : z - v \leqslant 0, \ v \in Z\}, \tag{2}$$

$$L^+(z) = \{v \in \mathbb{R}^k : z - v < 0,\ v \in Z\}. \tag{3}$$

The relationship between efficient (weakly–efficient) solutions of Problem (MOP) and the sets defined in (2)–(3) can be formulated as follows.

PROPOSITION 1. *If x^* is an efficient (weakly–efficient) solution of Problem (MOP) and $z^* = c(x^*)$, then $|L(z^*)| = 1$, i.e., $L(z^*) = \{z^*\}$, ($|L^+(z^*)| = 0$, i.e., $L^+(z^*) = \emptyset$). Conversely, if $z^* \in Z$ satisfies $|L(z^*)| = 1$ ($|L^+(z^*)| = 0$), then each point $x^* \in X$ satisfying $z^* = c(x^*)$ is an efficient (weakly–efficient) solution of Problem (MOP).*

Proof. Actually, this proposition contains equivalent definitions of efficient and weakly–efficient solutions. We show here the equivalency concerning efficient solutions. For weakly–efficient solutions, the proof is straightforward.

Let $x^* \in X$ be an efficient solution, and let $z^* \in \mathbb{R}^k$ defined by $z^* = c(x^*)$. Then, by definition, for each $x \in X$, $c(x) \geqslant c(x^*)$ implies $c(x) = c(x^*)$, i.e., the system

$$c(x) - v = 0,\ v \geqslant c(x^*),\ x \in X \tag{4}$$

has an unique solution (x^*, v^*) with $v^* = c(x^*) = z^*$, i.e., $|L(z^*)| = 1$.

Next, let $z^* \in \mathbb{R}^k$ such that $|L(z^*)| = 1$ and let $x^* \in X$ satisfying $z^* = c(x^*)$. Then, it follows that System (4) has an unique solution, i.e., for each $x \in X$, $v \geqslant c(x^*)$ implies $v = c(x) = c(x^*)$. Thus, by definition, x^* is an efficient solution of Problem (MOP). □

For the construction of a global optimization problem with special structure, we use the following notations.

$$E_Z = \{z \in Z : |L(z)| = 1\} \tag{5}$$

$$W_Z = \{z \in Z : |L^+(z)| = 0\} \tag{6}$$

$$G = \{z \in \mathbb{R}^k : z \leqslant v \text{ for some } v \in Z\}. \tag{7}$$

PROPOSITION 2. *(i) If $Z \neq \emptyset$, then int$G \neq \emptyset$.*
(ii) If $E_Z \neq \emptyset$, then $E_Z \subseteq W_Z = Z \cap \partial G$.

(For a set S, we denote by int$S$ and ∂S respectively the interior and the boundary of S).

Proof. (i) We can write

$$G = \{z \in \mathbb{R}^k : z = v - d,\ v \in Z,\ d \in \mathbb{R}^k_+\} = Z - \mathbb{R}^k_+.$$

Therefore, it follows that $intG \supset int(z - \mathbb{R}^k_+) \neq \emptyset$, whenever there exists a point $z \in Z$.

(ii) Let $z \in E_Z$, i.e., $|L(z)| = 1$. Then $Z \cap \{v : z < v\} = \emptyset$, i.e., $z \in W_Z$. Suppose there is a point $u \in W_Z$ such that $u \in intG$. Then, from the definition of G, there is a point $v \in Z$ such that $u \in int(v - \mathbb{R}^k_+)$, i.e., $u < v$, which is a contradiction to $u \in W_Z$. Thus, we have $W_Z \subseteq Z \cap \partial G$. On the other hand, let $z \in Z \cap \partial G$. Suppose there is a point $v \in Z$ such that $z < v$. Then $z \in int(v - \mathbb{R}^k_+)$, i.e., $z \in int G$, which is a contradiction to $z \in \partial G$. Thus, $Z \cap \{v : z < v\} = \emptyset$, i.e., $z \in W_Z$. This implies that $Z \cap \partial G \subseteq W_Z$ and hence, $W_Z = Z \cap \partial G$. □

In view of Proposition 2, we consider the optimization problem

$$\min\{f(x) : x \in X,\ z = c(x) \in Z \cap \partial G\}, \tag{Q}$$

which is equivalent to Problems (P), (P′) in the following sense.

PROPOSITION 3. *If x^* is an optimal solution of (P'), then (x^*, z^*) with $z^* = c(x^*)$ is an optimal solution of (Q). Conversely, if (x^*, z^*) is an optimal solution of (Q), then x^* is an optimal solution of (P'). If, in addition, $z^* \in E_Z$, then x^* is optimal to (P) as well.*

Proof. From Propositions 1 and 2, it follows that Problem (P′) is equivalent to the problem

$$\min\{f(x) : c(x) - z = 0,\ x \in X,\ |L^+(z)| = 0\} =$$
$$\min\{f(x) : x \in X,\ z \in Z \cap \partial G\}$$

in the sense stated above. If, in addition, $z^* \in E_Z$, where (x^*, z^*) is an optimal solution of Problem (Q), then, since $E_Z \subseteq W_Z$, it follows that x^* is also an optimal solution of (P). □

In general, Problem (Q) is a highly nonlinear optimization problem, even for the case that f is linear and X, Z are polyhedral sets. However, employing the special structure of the constraint $z \in Z \cap \partial G$, we can establish in the next section a conical algorithm for handling Problem (Q) and discuss some interesting implementable cases. Of course, our first aim is to obtain an optimal solution of Problem (P). Therefore, although Problem (Q) is under consideration, our algorithm will be designed in a way towards the first aim. Convergence properties of the algorithm are discussed in detail in Propotions 6 and 7. Preliminary computational experiments show, however, that for the linear case where X is a polyhedron, and the functions $c_i(x)$, $f(x)$ are all linear, the algorithm always yields an optimal solution of Problem (P).

3. Conical algorithm

To our purpose, we assume in what follows that the set W_Z defined in (6) is compact, so that Problem (Q) has an optimal solution whenever the function $f(x)$ is continuous. The set Z defined in (1) is called to be $\mathbb{R}^k_+$*–convex* if the set $G = Z - \mathbb{R}^k_+$ is convex. For the establishment of our algorithm we also assume that the set Z is $\mathbb{R}^k_+$–convex. A sufficient condition for this assumption is, e.g., the following.

PROPOSITION 4. *If in Problem (MOP), X is a convex subset of $\mathbb{R}^n$ and the functions $c_i(x)$ $(i = 1, \dots, k)$ are concave on X, then the set Z defined by (1) is $\mathbb{R}^k_+$–convex.*

Proof. By definition, we have

$$\begin{aligned} G = Z - \mathbb{R}^k_+ &= \{z \in \mathbb{R}^k : z \leqslant u, u = c(x), x \in X\} \\ &= \{z \in \mathbb{R}^k : z - c(x) \leqslant 0, x \in X\}. \end{aligned}$$

Thus, the set G is nothing but the projection of the set

$$\{(x, z) \in \mathbb{R}^n \times \mathbb{R}^k : z - c(x) \leqslant 0, x \in X\}$$

on $\mathbb{R}^k$. From the assumtion, this is a convex subset of $\mathbb{R}^n \times \mathbb{R}^k$, therefore, it follows (cf. e.g., [21]) that its projection G is a convex subset of $\mathbb{R}^k$. □

As mentioned in the introduction, the conical algorithm belongs to the branch and bound scheme, in which two basic operations are needed: branching and bounding. We begin the establishment of the conical algorithm with these basic operations.

3.1. CONICAL PARTITION IN $\mathbb{R}^k$

Let v^0 be a point in $\mathbb{R}^k$ such that $v^0 \in int G$ and the set $K^0 = \{z \in \mathbb{R}^k : v^0 \leqslant z\}$ contains the set W_Z. Such a point v^0 can be found, e.g., when $\min\{c_i(x) : x \in X\}$ exists for each $i = 1, \dots, k$. In this case we can choose $v^0 \in int G$ satisfying

$$v^0_i \leqslant \min\{c_i(x) : x \in X\} \ (i = 1, \dots, k). \tag{8}$$

Let

$$\sigma^i = v^0 + e^i \quad (i = 1, \dots, k), \tag{9}$$

where e^i $(i = 1, \dots, k)$ are the unit vectors of $\mathbb{R}^k$ and let

$$S^0 = [\sigma^1, \dots, \sigma^k] \tag{10}$$

be the $(k-1)$–simplex with vertices $\sigma^1, \dots, \sigma^k$. Then we obtain the convex polyhedral cone $K^0 = K(S^0)$ of dimension k, which has k edges emanating from v^0, passing through the vertices of the simplex S^0. Throughout this article, by a 'cone' or 'conical partition set' we always mean a convex polyhedral cone of dimension k, having k edges emanating from v^0, passing through k vertices of a $(k-1)$–simplex $S \subset S^0$, respectively.

Let K be a cone contained in K^0. A collection $\{K_1, \dots, K_r\}$ of cones is called a *conical partition* of K if

$$\bigcup_{j=1}^{r} K_j = K \quad \text{and} \quad int\ K_j \cap int\ K_i = \emptyset \quad \text{for} \quad j \neq i.$$

Each K_j is called a conical partition set of K.

In the context of our conical algorithm, at the beginning, a cone K^0 is constructed as above. Thereafter, at each iteration, a cone is divided into finitely many subcones using certain standard partition rules. For more details on various conical partition rules we refer, e.g., to [11, 15–17] and [26]. For convergence proofs of conical algorithms, the most useful characterization of a conical partition process is the concept of *exhaustiveness*. A nested subsequence $\{K_q\}$, $K_q \supset K_{q+1}$ $\forall q$, is called exhaustive if the intersection $\bigcap_{q=1}^{\infty} K_q$ is a ray (a halfline emanating from the point v^0). A conical partition process is called exhaustive if every nested subsequence of cones generated throughout the algorithm is exhaustive. A typical example for exhaustive partition processes is the well-known *conical bisection*, see, e.g., [26].

3.2. POLYHEDRAL PARTITION IN $\mathbb{R}^{n+k}$

According to the conical partition in $\mathbb{R}^k$ discussed above, we obtain a kind of polyhedral partitions in $\mathbb{R}^{n+k}$ which we call K-partition. A collection $\{F_1, \cdots, F_r\}$ of subsets of $\mathbb{R}^{n+k}$ is called a *K-partition* of $\mathbb{R}^{n+k}$, if

$$F_j = \mathbb{R}^n \times K_j \;\; (j = 1, \ldots, r), \tag{11}$$

and $\{K_1, \ldots, K_r\}$ forms a conical partition of $\mathbb{R}^k$. The sets F_j are called *K-partition sets*. A K-partition of an element of a K-partition is defined similarly by using a conical partition of the corresponding cone K, i.e., we say that the collection $\{F_1, \ldots, F_r\}$ forms a K-partition of $F = \mathbb{R}^n \times K$, where K is a cone, if $F_j = \mathbb{R}^n \times K_j$, $(j = 1, \ldots, r)$ and $\{K_1, \ldots, K_r\}$ forms a conical partition of K.

3.3. LOWER BOUNDS

In this subsection we consider the following task. Let $K = K(S)$ be a cone contained in $K^0 = K(S^0)$ and let $F = \mathbb{R}^n \times K$. We intend to compute a lower bound of the function $f(x)$ over a part of the feasible set of Problem (Q), which is contained in F. More precisely, we intend to compute a lower bound $\mu(K)$ of the optimal value of the problem

$$\min\{f(x) : x \in X,\; z = c(x) \in Z \cap \partial G,\; (x, z) \in F\}. \tag{12}$$

Let $S = [s^1, \ldots, s^k]$ and for each $i = 1, \ldots, k$, let v^i denote the intersection point of the ith edge of K with the set ∂G. Note that the ith edge of K emanates from v^0 passing s^i, and that $v^0 \in intG$. Further, let V be the $(k \times k)$ matrix with the columns $(v^1 - v^0), \ldots, (v^k - v^0)$. Our method for computing a lower bound $\mu(K)$ is based on the following result.

PROPOSITION 5. *A lower bound $\mu(K)$ of the optimal value of Problem (12) is obtained by solving the following program (in variables $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}^k$):*

$$\begin{aligned} \mu(K) \;=\; & \min\; f(x) \\ \text{s.t.}\quad & c(x) - V\lambda = v^0 \\ & x \in X \\ & \sum_{i=1}^{k} \lambda_i \geqslant 1,\;\; \lambda \geqslant 0. \end{aligned} \tag{13}$$

(As usual, we set $\mu(K) = +\infty$ if Problem (13) is infeasible).

Proof. Each point $z \in K$ is uniquely represented by

$$z = v^0 + V\lambda,\;\; \lambda \geqslant 0.$$

Let

$$E^+ = \left\{ z \in \mathbb{R}^k : z = v^0 + V\lambda,\;\; \sum_{i=1}^{k} \lambda_i \geqslant 1 \right\}$$

be one of the two halfspaces generated by the hyperplane containing the k linearly independent points $v^1, \ldots, v^k$. Then, from the convexity of the set G it follows that the set

$$\{(x, z) \in \mathbb{R}^n \times \mathbb{R}^k : x \in X,\;\; z = c(x),\;\; z \in Z \cap \partial G,\;\; (x, z) \in F\}$$

is contained in the set

$$\begin{aligned} &\{(x, z) \in \mathbb{R}^n \times \mathbb{R}^k : x \in X,\;\; z = c(x),\;\; z \in E^+,\;\; (x, z) \in F\} = \\ &\{(x, z) \in \mathbb{R}^n \times \mathbb{R}^k : x \in X,\;\; v^0 + V\lambda = c(x),\;\; \sum_{i=1}^{k} \lambda_i \geqslant 1,\;\; \lambda \geqslant 0\}. \end{aligned}$$

Therefore, we have

$$\begin{aligned} \mu(K) &= \min\{f(x) : x \in X,\;\; v^0 + V\lambda = c(x),\;\; \sum_{i=1}^{k} \lambda_i \geqslant 1,\;\; \lambda \geqslant 0\} \\ &\leqslant \min\{f(x) : x \in X,\;\; z = c(x),\;\; z \in Z \cap \partial G,\;\; (x, z) \in F\}. \qquad \square \end{aligned}$$

In the following remark, we show a simple way to determine intersection points v^i $(i = 1, \ldots, k)$ and discuss some interesting special cases for computing the lower bound $\mu(K)$.

REMARK 1. (i) For each $i = 1, \ldots, k$, the intersection point v^i of the ith edge of K with the set ∂G is computed by

$$v^i = v^0 + \alpha_i (s^i - v^0), \tag{14}$$

where, by the definition of G, the number α_i is computed by

$$\begin{aligned}\alpha_i &= \max\{\alpha : (v^0 + \alpha(s^i - v^0)) \in G\}\\ &= \max\{\alpha : (v^0 + \alpha(s^i - v^0)) \leqslant v,\ v = c(x),\ x \in X,\ \alpha \geqslant 0\}\\ &= \max\{\alpha : -c(x) + (s^i - v^0)\alpha \leqslant -v^0,\ x \in X,\ \alpha \geqslant 0\}.\end{aligned} \tag{15}$$

(ii) For the case that $f(x)$ is a linear function and (MOP) is a linear multiple objective programming problem, i.e., $c_i(x)$ $(i = 1, \ldots, k)$ are linear functions and X is a polyhedral set, Problems (13) and (15) are ordinary linear programs.

(iii) If $f(x)$ is a convex function and (MOP) is a convex multiple objective programming problem, i.e., $c_i(x)$ $(i = 1, \ldots, k)$ are linear functions and X is a convex set defined by

$$X = \{x \in \mathbb{R} : g_j(x) \leqslant 0\ (j = 1, \ldots, m)\},$$

with g_j $(j = 1, \ldots, m)$ being convex functions, then Problems (13) and (15) are ordinary convex programming problems. It is worth noting that each (MOP) with concave functions $c_i(x)$ $(i = 1, \ldots, k)$ and convex set X can be transformed into an equivalent convex multiple objective programming problem considered in this case. For computing a lower bound $\mu(K)$, the convex functions $f(x)$ and $g_j(x)$ $(j = 1, \ldots, m)$ can be iteratively approximated by convex piecewise linear functions, so that (13) can be formulated equivalently as linear programs (see, e.g., [14]).

(iv) If $f(x)$ is a concave function and (MOP) is a convex multiple objective programming problem in the sense of (iii), then (13) is a concave minimization problem with a special structure which can be solved by several decomposition techniques in global optimization (cf., e.g., [11, 17, 24]).

(v) If f is a composite function given by

$$f(x) = \varphi(c_1(x), \ldots, c_k(x)) = \varphi(z_1, \ldots, z_k)$$

where $\varphi(z)$ is a concave (or more generally, quasiconcave) function defined on a suitable subset of $\mathbb{R}^k$ and (MOP) is a convex multiple objective programming problem (in the sense of (iii)), then, similarly to the case (iv), (13) can be solved e.g. by the decomposition techniques discussed in [11, 17, 24, 25].

REMARK 2. If K and K' are conical partition sets satisfying $K \supset K'$, then the lower bounds given in Proposition 5 have the useful monotonicity property that $\mu(K) \leqslant \mu(K')$.

3.4. UPPER BOUNDS

A point (x, z) is feasible to Problem (Q), if $c(x) = z$ and $z \in W_Z = Z \cap \partial G$. We call a point (x, z) *efficiently-feasible solution* of Problem (Q), if $c(x) = z$ and $z \in E_Z$.

At the begining of the algorithm, if an efficient solution $x^0 \in E_X$ is available, then $(x^0, c(x^0))$ is an efficiently-feasible solution of Problem (Q). In this case we define a set $Q_0 = \{(x^0, z^0)\}$ with $z^0 = c(x^0)$. Notice that several methods for computing efficient solutions of Problem (MOP) can be found, e.g., in [23, 27]. A first upper bound for f over the set of efficiently-feasible solutions of Problem (Q) is then given by $\beta_0 = f(x^0)$. If we want to save computing an efficient solution of (MOP), then we simply set $Q_0 = \emptyset$ and $\beta_0 = +\infty$.

Throughout the algorithm, more and more efficiently-feasible solutions of Problem (Q) can be detected and thereby upper bounds can be iteratively improved.

For each conical partion set K, following points can be checked to enlarge the collection of efficiently-feasible solutions of Problem (Q).

First, for each $i = 1, \ldots, k$, let (x^i, α_i) be an optimal solution of Problem (15) and let $z^i = c(x^i)$. Then from Proposition 1 and part (ii) of Proposition 2, it follows that the point (x^i, z^i) is a feasible solution and the efficiently-feasibility of this point should be checked.

Next, let $(x(K), \lambda(K))$ be an optimal solution of Problem (13) (whenever this problem is solvable). Then the point

$$(x(K), z(K)) = (x(K), V\lambda(K) + v^0) = (x(K), c(x(K))) \tag{16}$$

should be taken for checking efficiently-feasibility.

For a given point $(\overline{x}, \overline{z})$, the examination of efficiently-feasibility is performed by Proposition 1, namely, $(\overline{x}, \overline{z})$ is an efficiently-feasible solution if and only if $|L(\overline{z})| = 1$, i.e., the system

$$c(x) - v \geqslant 0,\ v \geqslant c(\overline{x}),\ x \in X \tag{17}$$

has an unique solution $(\overline{x}, \overline{z})$. This is the case if

$$\max\left\{\sum_{i=1}^{k} v_i : c(x) - v \geqslant 0,\ v \geqslant c(\overline{x}),\ x \in X\right\} = \sum_{i=1}^{k} c_i(\overline{x}).$$

For each conical partion set K, we denote by $Q(K)$ the set of all efficiently-feasible solutions found as above. The number

$$\min\{f(x) : (x, z) \in Q(K)\}$$

yields an upper bound for the optimal value of Problem (Q).

3.5. THE ALGORITHM

Using notations and basic operations discussed in the previous subsections, we can formulate the conical branch and bound algorithm for handling Problem (Q) as follows.

Conical Algorithm:
Initialization:

Construct a cone $K^0 = K(S^0)$ (Subsection 3.1); Compute lower bound $\mu(K^0)$ (Subsections 3.2–3.3); Construct Q_0, $Q(K^0)$ (Subsection 3.4); Set $Q_0 \leftarrow Q_0 \cup Q(K^0)$; If $Q_0 \neq \emptyset$, then compute upper bound $\beta_0 = \min\{f(x) : (x, z) \in Q_0\}$ and choose $(\xi^0, \zeta^0) \in Q_0$ such that $f(\xi^0) = \beta_0$. If $Q_0 = \emptyset$, then set $\beta_0 = +\infty$; Store $(x(K^0), z(K^0))$ ($(x(K^0), z(K^0)$ is determined in (16) according to K^0); Set $\mathcal{K}_0 = \{K^0\}$, $\mu_0 = \mu(K^0)$, $q = 0$.

Iteration q:

If $\beta_q = \mu_q < +\infty$, then stop: (ξ^q, ζ^q) is an optimal solution of Problem (Q) and ξ^q is an optimal solution of problem (P) (Proposition 6 below). Otherwise, perform a conical partition of K^q obtaining $K_1^q, \cdots, K_r^q$;
For $i = 1, \ldots, r$ do
Compute $\mu(K_i^q)$; Set $Q_q \leftarrow Q_q \cup Q(K_i^q)$.
end for
Set $Q_{q+1} = Q_q$; If $Q_{q+1} \neq \emptyset$, then compute

$$\beta_{q+1} = \min\{f(x) : (x, z) \in Q_{q+1}\}$$

and choose $(\xi^{q+1}, \zeta^{q+1}) \in Q_{q+1}$ such that $f(\xi^{q+1}) = \beta_{q+1}$, otherwise, set $\beta_{q+1} = +\infty$; Set

$$\mathcal{K}_{q+1} = \mathcal{K}_q \setminus \{K^q\} \cup \{K_i^q : \mu(K_i^q) < \beta_{q+1},\ i = 1, \cdots, r\};$$

If $\mathcal{K}_{q+1} \neq \emptyset$, then set $\mu_{q+1} = \min\{\mu(K) : K \in \mathcal{K}_{q+1}\}$ and choose $K^{q+1} \in \mathcal{K}_{q+1}$ such that $\mu_{q+1} = \mu(K^{q+1})$, otherwise, set $\mu_{q+1} = \beta_{q+1}$; Go to iteration $q + 1$.

Convergence properties of the conical algorithm are discussed in following results.

PROPOSITION 6. *If the algorithm terminates at iteration q (by criterion $\beta_q = \mu_q < +\infty$), then the point ξ^q is an optimal solution of Problem (P).*

Proof. If the algorithm terminates at iteration q, then $\beta_q = \mu_q$ and we obtain the point (ξ^q, ζ^q) with $f(\xi^q) = \beta_q = \mu_q$. Since μ_q is a lower bound of the optimal value of Problem (Q), the point (ξ^q, ζ^q) is an optimal solution of the problem (Q). Therefore, it follows from Proposition 3 that x^q is an optimal solution of Problem (P), since (ξ^q, ζ^q) is an efficiently–feasible solution, i.e., $\zeta^q \in E_Z$. □

If the algorithm is not finite, it generates an infinite nested subsequence $\{K^\nu\}$ of cones satisfying $K^\nu \supset K^{\nu+1}$ for all ν.

For each q such that the corresponding Problem (13) is solvable, let $(x^q, z^q) = (x(K^q), z(K^q))$, where $(x(K^q), z(K^q))$ is determined in (16). The convergence of the algorithm can be stated as follows.

PROPOSITION 7. *Assume that throughout the algorithm, the conical partition process is exhaustive in the sense that each nested subsequence $\{K^\nu\}$ shrinks to a* *ray (cf. Subsection 3.1). Then every cluster point* (x^*, z^*) *of* $\{(x^q, z^q)\}$ *is an optimal solution of Problem (Q) and hence* x^* *is an optimal solution of Problem (P′). If, in addition,* (x^*, z^*) *is efficiently-feasible, then* x^* *is also an optimal solution of Problem (P).*

Proof. Let (x^*, z^*) be a cluster point of $\{(x^q, z^q)\}$ and let $\{(x^\nu, z^\nu)\}$ be a subsequence converging to (x^*, z^*). By passing to a subsequence if necessary, we obtain the corresponding nested sequence $\{K^\nu\}$.

Since $\{K^\nu\}$ shrinks to a ray, K^*, it follows that $\{z^*\} = K^* \cap \partial G \cap Z = K^* \cap W_Z$ (Proposition 2). This implies that (x^*, z^*) is a feasible solution of Problem (Q) and hence it is an optimal solution, because $f(x^*) = \lim_{\nu\to\infty} f(x^\nu) = \lim_{\nu\to\infty} \mu(K^\nu)$ is a lower bound for the optimal value of this Problem. The remaining statement of this proposition follows from the definition of efficiently-feasibility (Subsection 3.4) and Proposition 3. □

4. The case of biobjective programming problem

If $k = 2$ in Problem (MOP), then (MOP) is called a biobjective programming problem. This interesting special case was considered by many authors (cf. e.g., [5, 7, 9, 10, 22]). In this section we show that for the case of biobjective programming problem, the conical algorithm in the previous section can be implemented in a very simple way. Moreover, it is shown that the convergence of the resulting algorithm for this case can be obtained without using the assumption on exhaustiveness of the conical partition process.

4.1. CONICAL PARTITION IN $\mathbb{R}^2$

Using the very simple structure of the set E_Z in $\mathbb{R}^2$, one can construct a point v^0 such that the first cone $K^0 = \{z \in \mathbb{R}^2 : v^0 \leqslant z\}$ has the property

$$E_Z = K^0 \cap \partial G. \tag{18}$$

Let $u \in \mathbb{R}^2$ be computed by

$$u_i = \max\{c_i(x) : x \in X\} \quad (i = 1, 2). \tag{19}$$

Then we obtain the following simple assertion.

PROPOSITION 8. *If the set* Z *is* $\mathbb{R}^2_+$*–convex, and for each* $i = 1, 2$*, the program* $\max\{c_i(x) : x \in X\}$ *has an unique optimal solution, then* $E_Z = W_Z$.

Proof. From Proposition 2, we know that $E_Z \subseteq W_Z = Z \cap \partial G$. Suppose there is $z^* \in W_Z$ such that $z^* \notin E_Z$. Then $|L(z^*)| > 1$, i.e., there is $v \in Z$ such that $v \geqslant z^*$ and $v \neq z^*$. Since $z^* \in \partial G$, it is not possible that $z^* < v$, (because $z^* < v$ implies $z^* \in int(v - \mathbb{R}^2_+) \subset intG$). Thus, there is an index i such that $z^*_i = v_i$.

If $z_i^* = v_i < \max\{c_i(x) : x \in X\}$, then, since $z^* \in \partial G$, it follows that $v \in \partial G$. Thus, ∂G contains a line segment parallel to the jth axis ($j \neq i$), which contradicts the convexity of G. If $z_i^* = v_i = \max\{c_i(x) : x \in X\}$, then, since $v \neq z^*$, there are two different points, x^1 and x^2 in X such that $z^* = c(x^1)$ and $v = c(x^2)$, and $c_i(x^1) = c_i(x^2) = \max\{c_i(x) : x \in X\}$, which is also a contradiction to the assumption of the proposition. □

So, if the assumptions of Proposition 8 are fulfilled, then the point v^0 is obtained by setting

$$v_1^0 = c_1(x^2) \quad \text{and} \quad v_2^0 = c_2(x^1),$$

where x^i is the unique optimal solution of (19) according to $c_i(x)$ ($i = 1, 2$). Notice that in this case, the set E_Z ($= W_Z$) is exactly the part of ∂G connecting two points $(c_1(x^1), c_2(x^1))$, $(c_1(x^2), c_2(x^2))$. Thus, by construction, $E_Z = \{z \in \mathbb{R}^2 : v^0 \leqslant z\} \cap \partial G = K^0 \cap \partial G$.

If the set Z is $\mathbb{R}^2_+$–convex and Problem (19) has more than one optimal solution for an index i, then for $j \neq i$ set

$$v_j^0 = \max\{c_j(x) : x \in X,\ c_i(x) = u_i\}.$$

By the same argument as above, we also have $E_Z = K^0 \cap \partial G$.

It is worth noting that, unfortunately, the above construction of v^0 can not be applied for $k \geqslant 3$.

Throughout the conical algorithm, let K be a conical partition set in $\mathbb{R}^2$ (cf. Subsection 3.1) and let the (2×2) matrix $V = (v^1, v^2)$ be defined as in Subsection 3.3.

Further, let $(x(K), z(K)) = (x(K), V\lambda(K) + v^0)$ be computed by (16).

Consider the following programming problem

$$\begin{aligned}
&\max\ e\lambda = \sum_{i=1}^{k} \lambda_i \\
&c(x) - V\lambda = v^0 \\
&x \in X \\
&\sum_{i=1}^{k} \lambda_i \geqslant 1,\ \lambda \geqslant 0,
\end{aligned} \tag{20}$$

where $e = (1, 1) \in \mathbb{R}^2$.

Let $t(K)$ be the optimal value of Problem (20). If $t(K) = 1$, then it follows that the set

$$\left\{ z \in \mathbb{R}^k : z = c(x) = V\lambda + v^0,\ x \in X,\ \sum_{i=1}^{k} \lambda_i \geqslant 1,\ \lambda \geqslant 0 \right\}$$

is contained in the set

$$\left\{ z \in \mathbb{R}^k : z = V\lambda + v^0,\ \sum_{i=1}^{k} \lambda_i = 1,\ \lambda \geqslant 0 \right\},$$

which defines a supporting hyperplane of G. This implies that for each feasible solution (x, λ) of Problem (20), (i.e., of Problem (13)), the point $z = V\lambda + v^0$ is contained in the set $K \cap \partial G$, which is a subset of $E_Z = K^0 \cap \partial G$. Thus, the point $(x(K), z(K)) = (x(K), V\lambda(K) + v^0)$ is an efficiently-feasible solution of Problem (Q), so that it is chosen to update the upper bound β. This implies that $\mu(K) = f(x(K)) \geqslant \beta$ and therefore, the cone K is removed from further consideration.

For the case $t(K) > 1$, let $(\overline{x}(K), \overline{\lambda}(K))$ be an optimal solution of Problem (20) and let

$$\overline{z}(K) = v^0 + V\overline{\lambda}(K). \tag{21}$$

Then the cone K is divided into two subcones K_1, K_2 having respectively corresponding matrices $V_1 = (\overline{z}(K), v^2)$, $V_2 = (v^1, \overline{z}(K))$.

Obviously we have

$$(\overline{x}(K), \overline{z}(K)) \in K \cap \partial G, \tag{22}$$

and therefore, $(\overline{x}(K), \overline{z}(K))$ is an efficiently-feasible solution and is chosen to update the upper bound.

4.2. CONVERGENCE OF THE ALGORITHM

By using the conical partition in Subsection 4.1, we obtain for the conical algorithm following convergence properties, which are completely released from the assumption on exhaustiveness of the conical partition process.

PROPOSITION 9. *(i) If the functions $c_1(x), c_2(x)$ are linear and X is a polyhedral set, then the algorithm always terminates after finitely many iterations yielding an optimal solution of Problem (P).*

(ii) In the general case, if the algorithm is not finite and for each cone K^q, let $(\overline{x}^q, \overline{z}^q) = (\overline{x}(K^q), \overline{z}(K^q))$, where $(\overline{x}(K^q), \overline{z}(K^q))$ is computed by (22) according to K^q. Then every cluster point $(\overline{x}, \overline{z})$ of the infinite sequence $\{(\overline{x}^q, \overline{z}^q)\}$ is an optimal solution of Problem (Q) and $\overline{x}$ is an optimal solution of Problem (P).

Proof. (i) If the functions $c_1(x), c_2(x)$ are linear and X is a polyhedral set, then $Z = \{z \in \mathbb{R}^2 : z_i = c_i(x)(i = 1, 2), x \in X\}$ is a polyhedral set as well. Moreover, the set $E_Z = K^0 \cap \partial G$ is a connected path of (finitely many) edges of Z between two points (v_1^0, u_2), (u_1, v_2^0). By construction, each cone K^q has the corresponding matrix $V^q = (v^{q1}, v^{q2})$, where v^{q1}, v^{q2} are break points of the path E_Z. The point $\overline{z}(K^q)$ computed by (21) according to K^q is also a break point of E_Z, which is used for the partition of K^q. Since $t(K^q) > 1$, it follows that $\overline{z}(K^q) \neq v^{qi}$ $(i =$

$1, 2)$. Thus, we have $\bar{z}(K^q) \neq \bar{z}(K^j)$ for $j < q$, which implies that the algorithm terminates after finitely many iterations yielding an optimal solution of (Q), since the number of break points of E_Z is finite. From (18), it follows that each optimal solution of (Q) yields an optimal solution of (P).

(ii) Let $(\bar{x}, \bar{z})$ be a cluster point of $\{(\bar{x}^q, \bar{z}^q)\}$ and let $\{(\bar{x}^\nu, \bar{z}^\nu)\}$ be a subsequence converging to $(\bar{x}, \bar{z})$ such that for each ν, the corresponding cone K^ν is generated from $K^{\nu-1}$ by a conical partition.

Since for each ν, the point $\bar{z}^\nu$ is one of two columns of matrix $V^{\nu+1}$ and $\bar{z}^\nu \to \bar{z}$, it follows that $t(K^\nu) \to 1$, (recall that $t(K^\nu)$ is the optimal value of Problem (20) according to K^ν). This implies that $\beta_\nu - \mu_\nu = \beta_\nu - \mu(K^\nu) \leqslant f(\bar{x}^\nu) - \mu_\nu \to 0$ for $\nu \to +\infty$, i.e., $(\bar{x}, \bar{z})$ is an optimal solution of Problem (Q) and therefore, in view of (18), $\bar{x}$ is an optimal solution of Problem (P). □

5. Preliminary computational experiments

To test the conical algorithm, we consider a linear multiple objective programming problem of the form

$$\max\ c_i(x) = c^i x = z_i \ (i = 1, \dots, k), \quad \text{s.t. } Ax \leqslant b,\ x \geqslant 0,$$

where $c^i \in \mathbb{R}^n$ $(i = 1, \dots, k)$, A is a real $m \times n$ matrix and $b \in \mathbb{R}^m$. We assume that the feasible set $\{x \in \mathbb{R}^n : Ax \leqslant b, x \geqslant 0\}$ is bounded. The function $f(x)$ is also assumed to be linear, i.e., $f(x) = fx$ where $f \in \mathbb{R}^n$. Denote by C the matrix with the rows c^i $(i = 1, \dots, k)$.

For the test, we modify the conical algorithm as follows. The algorithm terminates at iteration q if $\beta_q - \mu_q \leqslant \varepsilon|\beta_q|$ for $\beta_q \neq 0$ and $\beta_q - \mu_q \leqslant \varepsilon$ for $\beta_q = 0$, where $\varepsilon > 0$ is a given tolerance. The point (ξ^q, ζ^q) is called an *ε–optimal solution* of Problem (Q) and ξ^q is an *ε–optimal solution* of Problem (P).

For each triple (m, n, k) $(6 \leqslant m \leqslant 90,\ 20 \leqslant n \leqslant 200$ and $2 \leqslant k \leqslant 4)$ the algorithm was run on 20 randomly generated test problems. Each test problem is generated in the following way. The matrices C, A and the vectors b, f are generated by using a pseudo-random number generator. We notice that for solving linear subproblems and systems of linear inequalities we used an own code based on the well known simplex method. Test problems were run on a Sun SPARC station 10 Modell 20 workstation. Numerical results are summarized in Table 1.

Finally, it is worth noting that in all cases considered within the test, the conical algorithm terminates after finitely many iterations yielding a 10^{-2}–optimal solution of Problem (P).

Acknowledgement

The author would like to thank two anonymous reviewers for constructive comments and suggestions which helped to improve the first version of this article.

Table 1. Computational Results

m	n	k	ITER	CMAX	TIME
20	50	2	24	5	0.08
50	100	2	37	7	2.92
70	100	2	23	3	3.70
90	100	2	34	6	5.83
20	200	2	38	6	9.67
50	200	2	50	9	10.30
15	20	3	383	77	5.12
20	20	3	341	65	6.42
22	20	3	82	13	1.43
50	50	3	339	67	7.12
50	60	3	426	73	11.66
50	120	3	277	40	23.23
10	30	4	1457	219	24.32
20	30	4	7481	3130	31.41
20	50	4	5246	844	22.89
20	80	4	0	12	6.33
20	100	4	8	6	5.11
30	100	4	442	64	11.20

ITER: Average number of iterations.
CMAX: Maximal number of cones stored at an iteration.
TIME: Average CPU-Time in seconds.

References

1. Benson, H.P. (1984), Optimization over the Efficient Set, *Journal of Mathematical Analysis and Applications* 98: 562–580.
2. Benson, H.P. (1986), An Algorithm for Optimizing over the Weakly-Efficient Set, *European Journal of Operational Research* 25: 192–199.
3. Benson, H.P. (1991), An All–Linear Programming Relaxation Algorithm for Optimizing over the Efficient Set, *Journal of Global Optimization* 1: 83–104.
4. Benson, H.P. (1995), A geometrical analysis of the outcome set in multiple objective convex programs with linear criterion functions, *Journal of Global Optimization* 6: 231–251.
5. Benson, H.P. and Lee, D. (1996), Outcome–based Algorithm for Optimizing over the Efficient Set of a Bicriteria Linear Programming Problem, *Journal of Optimization Theory and Applications* 88: 77–105.
6. Bolintineanu, S. (1993), Minimization of a quasiconcave function over an efficient set, *Mathematical Programming* 61: 89–110.
7. Dan, N.D. and Muu, L.D. (1996), A parametric Simplex Method for Optimizing over the Efficient Set of a Bicriteria Linear Programming Problem, *Acta Mathematica Vietnamica* 21: 59–67.

8. Dauer, J.P. and Fosnaugh, T.A. (1995), Optimization over the Efficient Set, *Journal of Global Optimization* 7: 261–277.
9. Fülöp, J. and Muu, L.D., Branch and Bound Variant of an Outcome-Based Algorithm for Optimizing over the Efficient Set of a Bicriteria Linear Programming Problem. To appear in *Journal of Optimization Theory and Applications*.
10. Geoffrion, A.M. (1967), Solving Bicriterion Mathematical Programs, *Operations Research*, 15: 39–54.
11. Horst, R., Pardalos, P.M. and Thoai, N.V. (1995), *Introduction to Global Optimization*, Kluwer, Dordrecht, The Netherlands.
12. R. Horst and N.V. Thoai (1994), Constraint Decomposition Algorithms in Global Optimization, *Journal of Global Optimization* 5: 333–348.
13. Horst, R. and Thoai, N.V. (1997), Utility Function Programs and Optimization over the Efficient Set in Multiple-Objective Decision Making, *Journal of Optimization Theory and Applications* 92: 609–635.
14. Horst, R. and Thoai, N.V. (1999), Maximizing a Concave Function over the Efficient or Weakly-Efficient Set, *European Journal of Operational Research*, 117: 239–252.
15. Horst, R. and Thoai, N.V. (1999), DC Programming: Overview, *Journal of Optimization Theory and Applications*, 103: 1–43.
16. Horst, R., Thoai, N.V. and Benson, H.P. (1991), Concave minimization via conical partitions and polyhedral outer approximation, *Mathematical Programming* 50: 259–274.
17. Horst, R. and Tuy, H. (1993), *Global Optimization: Deterministic Approaches*, 2nd edition, Springer–Verlag, Berlin, New York.
18. Le-Thi H.A., Pham D.T. and Muu L.D. (1996), Numerical Solution for Optimization over the Efficient Set by D.C. Optimization Algorithms, *Operations Research Letters* 19: 117–128.
19. Muu, L.D. and Luc, L.T. (1996), On Equivalence between Convex Maximization and Optimization over the Efficient Set, *Vietnam Journal of Mathematics* 24: 439–444.
20. Philip, J. (1972), Algorithms for the Vector Maximization Problem, *Mathematical Programming* 2: 207–229.
21. Rockafellar, R.T. (1970), *'Convex Analysis'*, Princeton University Press, Princeton, New Jersey.
22. Solanski, R.S. and Cohon, J.L. (1989), Approximating the Noninferior Set in Linear Biobjective Programs Using Multiparametric Decomposition, *European Journal of Operational Research*, 41: 355–366.
23. Steuer, R.E. (1985), *Multiple Criteria Optimization: Theory, Computation and Application*, Wiley, New York, 1985.
24. Thoai, N.V. (1991), Global Optimization Approach for Solving the Convex Multiplicative Programming Problem, *Journal of Global Optimization*, 1: 341–357.
25. Thoai, N.V. (2000), A class of optimization problems over the efficient set of a multiple criteria nonlinear programming problem, *European Journal of Operational Research* 122: 58–68.
26. Thoai, N.V. and Tuy, H. (1980), Convergent Algorithms for Minimizing a Concave function, *Mathematics of Operations Research* 5: 556–566.
27. Yu, P.L. (1985), *Multiple Criteria Decision Making: Concepts, Techniques and Extensions*, Plenum, New York.