Preprint submitted to *Optimization Methods & Software*

Finding the Edgeworth-Pareto hull and its application to optimization over the efficient set of multiple objective discrete linear programs

P.T. Hoai[a], L.D. Muu[b*] and T.N. Thang[a]

[a]*School of Applied Mathematics and Informatics, Hanoi University of Science and Technology, Hanoi, Vietnam;* [b]*Institute of Mathematics, VAST, Hanoi, Vietnam*

We propose a new algorithm for finding the Edgeworth-Pareto hull for multiple objective linear mathematical programming problems, where the constraint set is given by a finite number of points. We apply the proposed algorithm to minimizing an increasingly quasiconcave function over the efficient set by using the Edgeworth-Pareto hull in outcome space. When the cardinality of the discrete constrained set is large, one can replace this set by the vertices of its convex hull. Application to minimization problems of an increasingly quasi-concave function over the efficient set by using the Edgeworth-Pareto hull is discussed. The obtained computational results on a lot number of randomly generated problems show the efficiency of the proposed algorithm for some cases.

Keywords: multiple objective discrete optimization; Edgeworth-Pareto hull; optimization over efficient set; outcome space algorithm

AMS Subject Classification: 65K10; 90C29

1. Introduction

The multiple objective (vector) mathematical programming problem involves simultaneous optimization (minimization or maximization) of $p \geq 2$ criterion functions over a nonempty set. For this problem the concept of an efficient Pareto solution plays an important role. The efficient set, even for the case when the constrained set is polyhedral convex and the criterion functions are affine, may not be convex. Thus finding the entire or a significant part of this set, in general, is a difficult global optimization problem, especially for large size problems. Fortunately, in many practical problems, the number of the criterion functions is much smaller than the dimension of the decision variable. For this case, some properties for the efficient set can be derived and understood more easily by focusing on the efficient outcome set rather than on the efficient set itself. The outcome set approach has been used in studies concerning the existence of the efficient solution. It also has been used for key properties of the efficient set such as the connectedness, stability, contractibility and optimization over the efficient set, see e.g., [3, 4, 9–11]. Numerous instances of the beneficial use of outcome set can be cited, for example, [3, 4, 7, 9, 21].

On the other hand, the concept of the Edgeworth-Pareto efficiency has been used for studies multiple objective programming problems thanks to the fact that this set has particular properties, see e.g. [23], such as it is a full-dimensional convex set, all

*Corresponding author. Email: ldmuu@math.ac.vn

of its vertices are efficient. These facts can be employed in some applications such as in optimization over the efficient set. There exist several approaches for approximating the Edgeworth-Pareto hull such as the Haussdorff approximation [13, 24], the iterative polyhedral approximation based on a synthesis of the ideas of the branch-and-bound method [23].

The problem of optimizing a real function over the efficient set has been introduced by Philip [22] for the linear case. After the appearance of the works by Benson [5, 6] this problem has attracted much attention of the researchers, see e.g. [1, 8, 12, 16, 17, 19, 25, 26] and the references therein. However to our best knowledge there is no method for this problem in the case when the constraint set is a finite set of points. Such a finite set is often given by methods of statistics for practical problems.

The purpose of this paper is to propose a new algorithm for determining the vertex-set in outcome space of the Edgeworth-Pareto hull for multiple objective linear problems, where the constraint set is given as a finite number of points, and to apply it to minimizing an increasingly quasiconcave function over the efficient set of such a discrete multiple objective program by using the Edgeworth-Pareto hull in outcome space rather than the efficient set itsself. The main points of the problem to be solved and the proposed algorithm are the followings:

• The constrained set X of the multiple objective linear problem under consideration can be a discrete set given by a finite number of points in the decision space;

• The proposed algorithm can use algorithms for finding the vertices of the convex hull in the outcome space for the discrete constraint set. The number of these vertices is often significantly smaller than the number of points of X. This fact allows that the method can handle multiple objective discrete programs whose constrained set may be very large;

• The algorithm is a finite recursive procedure in outcome space that can compute all vertices of the Edgeworth-Pareto hull;

• The optimization problem over the efficient set of a multiple objective linear program whose constraint set is given by a finite number of points is solved by using the proposed algorithm for determining the vertex-set of the Edgeworth-Pareto hull in the outcome space. The latter set is convex, and, in general, much smaller than the efficient set in outcome space (see Fig. 1 below).

The paper is organized as follows. The next section contains some properties concerning the Egdeworth-Pareto hull and related optimization problem over this set. The third section is devoted to description of the proposed algorithm for finding the Egdeworth-Pareto hull and its application to optimization over the efficient set. We close the paper with some computational results and experiments for the proposed algorithms.

2. Preliminaries on the Edgeworth-Pareto hull

First we recall that the convex hull conv$D$ of a set D is the intersection of all convex sets containing D. It is well known from the convex analysis that every vertices of conv$D$ belongs to D, and that conv$D$ is a bounded polyhedral convex set (polytope) if D is a finite set.

In what follows, as usual, for two vectors $a, b \in \mathbb{R}^p$, the notation $a \geq b$ means $a - b \in \mathbb{R}^p_+ = \{y = (y_1, ..., y_p)^T | y_1 \geq 0, \ldots, y_p \geq 0\}$. By $V(P)$ we denote the vertex-set of a polyhedral convex set P.

We consider the multiple linear objective optimization problem that is given as

$$\text{Vmin}\{f(x) = Cx : x \in X\}, \tag{VP}$$

where $X \subset \mathbb{R}^n$ is a set of finite points in $\mathbb{R}^n$, C is a $(p \times n)$-real matrix. Recall that a point $x^* \in X$ is said to be an *efficient solution* for Problem (VP) if $x \in X$, $Cx \leq Cx^*$ then $Cx^* = Cx$. Let X_E denote the set of all efficient solutions for Problem (VP).

The set $Y = C(X) \subset \mathbb{R}^p$, is called the *outcome set* (or image) of X. Since X is finite, Y is finite too. Denote the convex hull of Y by conv$Y$. Recall [15, 23] that the set conv$Y + \mathbb{R}^p_+$ is called the *Edgeworth-Pareto hull* of Y, denoted by $Y^\diamond$, which means

$$Y^\diamond = \{y \in \mathbb{R}^p | y = z + d, z \in \text{conv}Y, d \in \mathbb{R}^p_+\}.$$

From the definition, it is easy to check that

$$Y \subseteq Y^\diamond, V(Y^\diamond) \subseteq Y_E,$$

where Y_E stands for the efficient set of Y. It is easy to see that

$$Y_E = \{y \in Y : \exists x \in X_E, y = Cx\}.$$

Consider the following optimization problem over the efficient set

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

where $\varphi : \mathbb{R}^p \to \mathbb{R}$, $f : \mathbb{R}^n \to \mathbb{R}^p$. The outcome-space reformulation of Problem (P) is given by

$$\min\{\varphi(y) : y \in Y_E\}. \tag{OP}$$

The next result is a direct consequence of Proposition 2.1 in [7].

Proposition 2.1 *A point x^* is a global optimal solution of Problem (P) if and only if $y^* \in Y_E$, with $f(x^*) = y^*$, is a global optimal solution of Problem (OP).*

Now let us consider the optimization problem over the Edgeworth-Pareto hull related to (OP) which is defined as

$$\min\{\varphi(y) : y \in Y^\diamond\} \qquad (OP^\diamond).$$

The following proposition gives a relationship between Problems (OP) and $(OP^\diamond)$.

Proposition 2.2 *Suppose that $\varphi(y)$ is lower semi-continuous and quasiconcave on $Y^\diamond$, then*

i) If problem $(OP^\diamond)$ is solvable then one of the optimal solutions of $(OP^\diamond)$ attains at a vertex of $Y^\diamond$;

ii) If $y^* \in V(Y^\diamond)$ is an optimal solution of $(OP^\diamond)$, then it is also an optimal solution of (OP).

Proof. The first part of the assertion is obvious because the function φ is quasiconcave and $Y^\diamond$ is convex. Suppose that $y^* \in V(Y^\diamond)$ is an optimal solution of $(OP^\diamond)$, which

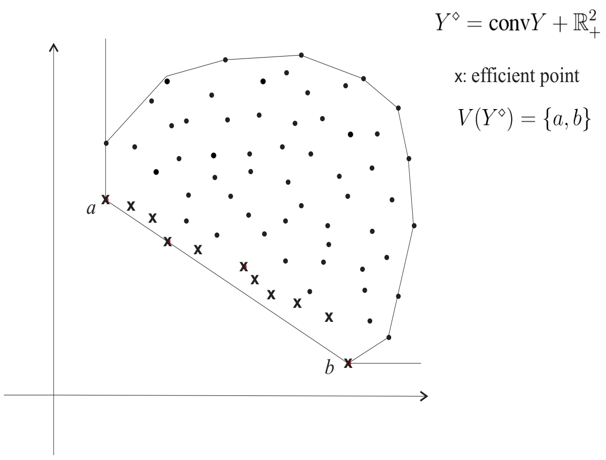

Figure 1. The Edgeworth Pareto hull

means

$$\varphi(y^*) \leq \varphi(y) \text{ for all } y \in Y^\diamond.$$

Since $Y_E \subseteq Y \subseteq Y^\diamond$, it implies that

$$\varphi(y^*) \leq \varphi(y) \text{ for all } y \in Y_E.$$

Since $y^* \in V(Y^\diamond)$, by $V(Y^\diamond) \subseteq Y_E$, we have $y^* \in Y_E$. Thus y^* is an optimal solution of problem (OP). ■

Remark 1 If $\varphi(y)$ is an increasing function in the sense that $y \geq y'$, implies $\varphi(y) \geq \varphi(y')$, and $\varphi(y)$ is lower semi-continuous and quasiconcave then problem $(OP^\diamond)$ is solvable. Thus, by Proposition 2.2, every optimal solution of problem

$$\min\{\varphi(y) \mid y \in V(Y^\diamond)\}$$

is also an optimal solution of (OP).

3. Description of the algorithms

Propositions 2.1 and 2.2 suggest that for solving Problem (OP) one can evaluate the objective function φ at vertices of the Edgeworth-Pareto hull in the outcome space. In the case the discrete set X is large and the number of the criteria is much smaller than the dimension of decision variable (often in practice), one suggests the use of an existing algorithm for computing all vertices of the convex hull of the image set Y of X. Fig. 1 is an example in $\mathbb{R}^2$ showing that the number of vertices of conv$Y$ is much smaller than that of Y.

Now we are going to present an algorithm for finding all vertices of the Edgeworth-Pareto hull. Suppose that $V(Y^\diamond) = \{v^1, v^2, ..., v^l\}$. From the definition of $Y^\diamond$, it is obviously that for every $y \in Y^\diamond$ there exists a vector $d \in \mathbb{R}^p_+$ such that $y = z + d$, where z is a convex combination of $\{v^1, v^2, ..., v^l\}$, that for each $k \in \{1, 2, ..., l\}$, $v^k \notin \text{conv}(\{v^1, v^2, ..., v^l\} \setminus \{v^k\}) + \mathbb{R}^p_+$, that dim$Y^\diamond = p$, and that $V(Y^\diamond) \subset Y$.

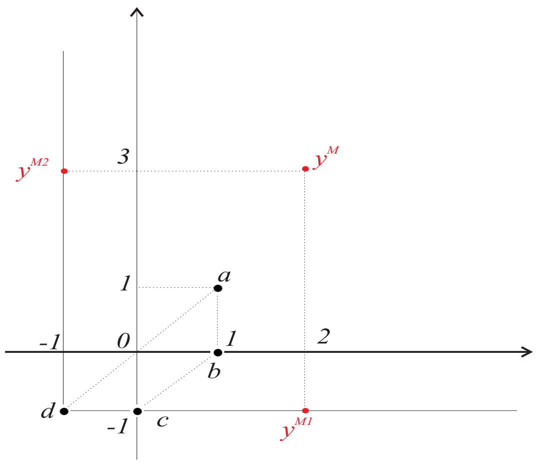

Figure 2. An illustration for Algorithm 1

The algorithm below for finding $V(Y^\diamond)$ can be regarded as an induction procedure for the dimension of the set $Y^\diamond$. Suppose that for each coordinate j we have computed all vertices of the Edgeworth-Pareto hull of Y_j where Y_j is obtained from Y by deleting j-th coordinate. Then the algorithm for computing $V(Y^\diamond)$ from $V(Y_j^\diamond)$, for every $j = 1, ..., p$ can be described in detail as follows.

Algorithm 1

- **Step 1:** Find $y^M \in \mathbb{R}^p$ such that $y_i^M > \max\{y_i | y \in Y\}$.
- **Step 2:** For each $j = 1, ..., p$
  - Set $Y_j := \{p_j(y) | y \in Y\}$, where $p_j(y)$ is obtained from $y^T := (y_1, ..., y_{j-1}, y_j, y_{j+1}, ... y_p)$ by deleting j-th coordinate.
  - Compute $V(Y_j^\diamond)$ in the space $\mathbb{R}^{p-1}$ and set

$$V_j := \{v \in \mathbb{R}^p | p_j(v) \in V(Y_j^\diamond), v_j = y_j^M\}.$$

- **Step 3:** Let $Y' := Y \cup (\bigcup_{j=1}^{p} V_j) \cup \{y^M\}$ and define $V(Y^\diamond)$ by taking

$$V(Y^\diamond) = V(\text{conv}Y') \cap Y.$$

Remark 2 It is easily seen that if $Y_j^\diamond \subset \mathbb{R}$ then

$$V(Y_1^\diamond) = \min\{y_2 | y = (y_1, y_2) \in Y\}$$

and

$$V(Y_2^\diamond) = \min\{y_1 | y = (y_1, y_2) \in Y\}.$$

To illustrate the algorithm, let us consider the following simple example for $p = 2$ (Fig. 2).

Suppose that

$$Y := \{a, b, c, d\} \ \text{ with } \ a := (1, 1), b := (1, 0), c := (0, -1), d := (-1, -1).$$

Then, by definition

$$Y_1 = \{1, 0, -1, -1\},\ Y_2 = \{1, 1, 0, -1\}.$$

Thus

$$Y_1^\diamond = \{-1, 0, 1\} + \mathbb{R}_+ = [-1, +\infty),\ Y_2^\diamond = \{-1, 0, 1\} + \mathbb{R}_+ = [-1, +\infty).$$

Hence

$$V(Y_1^\diamond) = \{-1\},\ V(Y_2^\diamond) = \{-1\}.$$

Take

$$y_1^M := 2 > \max\{y_1 : (y_1, y_2) \in Y\} = 1,\ y_2^M := 3 > \max\{y_2 : (y_1, y_2) \in Y\} = 1.$$

Then, by definition

$$V_1 = \{v = (v_1, v_2) \in \mathbb{R}^2, v_2 \in V(Y_1^\diamond), v_1 = y_1^M = 2\} = \{(2, -1)\} = \{y^{M1}\}$$

$$V_2 = \{v = (v_1, v_2) \in \mathbb{R}^2, v_1 \in V(Y_2^\diamond), v_2 = y_2^M = 3\} = \{(-1, 3)\} = \{y^{M2}\}.$$

Thus

$$Y' = Y \cup \{V_1, V_2\} \cup \{y^M\} = Y \cup \{(2, -1), (-1, 3)\} \cup \{(2, 3)\}.$$

Since $V(\text{conv}Y') \cap Y = \{(-1, -1)\}$, we have $V(Y^\diamond) = \{(-1, -1)\}$.

Now we show the validity and finite convergence of the algorithm.

THEOREM 3.1 *Algorithm 1 is well defined and terminates after a finite number of steps yielding all vertices of the Edgeworth-Pareto hull $Y^\diamond$.*

Proof. We divide the proof into several steps:

(i) Firstly, we show that there exists a set $S = \{v^1, v^2, ..., v^l\} \subset Y$ such that $V(\text{conv}Y') = (\bigcup_{j=1}^{p} V_j) \cup \{y^M\} \cup S$.

Indeed, from $y \leq y^M$ for all $y \in \text{conv}Y'$, it follows that y^M is a vertex of $\text{conv}Y'$.

For the case $j = 1$, let $v \in V_1$ and suppose that $v \notin V(\text{conv}Y')$. Since $v_1 = y_1^M \geq y_1$ for all $y = (y_1, ..., y_p) \in \text{conv}Y'$, we see that v is a convex combination of points in Y' in the facet $y_1 = y_1^M$, which means that

$$v = \sum_{v^\mu \in V_1} \lambda_\mu v^\mu + \lambda_M y^M \tag{1}$$

$$= \sum_{v^\mu \in V_1} \lambda_\mu v^\mu + \lambda_M q + \lambda_M (y^M - q), \tag{2}$$

where $\sum_{v^\mu \in V_1} \lambda_\mu + \lambda_M = 1, \lambda_\mu, \lambda_M \in [0, 1]$, $q \in Y$.

From (2) we can write

$$p_1(v) = \sum_{v^\mu \in V_1} \lambda_\mu p_1(v^\mu) + \lambda_M p_1(q) + \alpha,$$

where $p_1(v^\mu), p_1(q) \in Y_1 = p_1(Y)$, $\alpha = \lambda_M p_1((y^M - q)) \in p_1(\mathbb{R}^p_+) = \mathbb{R}^{p-1}_+$. But $p_1(v) \in V(Y_1^\diamond)$ we see that $\lambda_\mu = 1$ with $p_1(v_\mu) = p_1(v)$, $\lambda_\mu = 0$ with $p_1(v_\mu) \neq p_1(v)$ and $\lambda_M = 0$. Combining with (1) we can conclude that v is a vertex of convY'. The same argument can be used for $j = 2, ..., p$.

(ii) Next we show that for each $j \in \{1, ..., p\}$ and $v \in V_j$, there exists $\bar{v} \in S$ such that $p_j(\bar{v}) = p_j(v)$. We suppose that $j = 1$, for other j the argument is the same. Fix $v \in V_1$ and let $\bar{v} = \text{argmin}\{y_1 | y \in Y, p_1(y) = p_1(v)\}$. We prove that $\bar{v} \in S$. Indeed, since $p_1(\bar{v}) \in V(Y_1^\diamond)$, the point $\bar{v}$ cannot be a convex combination of points in Y' whose projections on $p_1(\mathbb{R}^p_+) := \{(y_2, ..., y_p), y_j \geq 0 \;\forall j\}$ are different from $p_1(\bar{v})$. On the other hand, since

$$\bar{v}_1 = \min\{y_1 | y \in Y, p_1(y) = p_1(v)\} = \min\{y_1 | y \in Y', p_1(y) = p_1(v)\},$$

we have $\bar{v} \notin \text{conv}(Y' \setminus \{\bar{v}\}) + \mathbb{R}^p_+$ as the desired conclusion.

(iii) Now we show $Y^\diamond = S^\diamond$.

Obviously, $S^\diamond \subset Y^\diamond$, so it is sufficient to show that $Y^\diamond \subset S^\diamond$. Indeed, take an $y \in Y^\diamond$ then there exist $z \in \text{conv}Y, d \in \mathbb{R}^p_+$ such that $y = z + d$. Since $z \in \text{conv}Y \subset \text{conv}Y'$, we can express z as a convex combination of $V(\text{conv}Y')$ that means

$$z = \sum_{k=1}^{l} \lambda_k v^k + \sum_{j=1}^{p} \sum_{v^j_\mu \in V_j} \lambda^j_\mu v^j_\mu + \lambda_M y^M, \tag{3}$$

where $\lambda_k, \lambda^j_\mu, \lambda_M \in [0, 1]$ and $\sum\limits_{k=1}^{l} \lambda_k + \sum\limits_{j=1}^{p} \sum\limits_{v^j_\mu \in V_j} \lambda^j_\mu + \lambda_M = 1$.

By(ii), for each $v^j_\mu \in V_j$ there is $\bar{v}^j_\mu \in S$ such that $v^j_\mu = \bar{v}^j_\mu + \alpha^j_\mu e^j$ with $\alpha^j_\mu > 0$. Thus, from (3) it follows that

$$z = \sum_{k=1}^{l} \lambda_k v^k + \sum_{j=1}^{p} \sum_{v^j_\mu \in V_j} \lambda^j_\mu \bar{v}^j_\mu + \lambda_M v^1 + \sum_{j=1}^{p} \sum_{v^j_\mu \in V_j} \alpha^j_\mu e^j + \lambda_M (y^M - v^1), \tag{4}$$

$$= \sum_{k=1}^{l} \lambda_k v^k + \sum_{j=1}^{p} \sum_{v^j_\mu \in V_j} \lambda^j_\mu \bar{v}^j_\mu + \lambda_M v^1 + \bar{d}, \tag{5}$$

where $\bar{d} = \sum\limits_{j=1}^{p} \sum\limits_{v^j_\mu \in V_j} \alpha^j_\mu e^j + \lambda_M (y^M - v^1) \in \mathbb{R}^p_+$, which implies that $y \in S^\diamond$ and $Y^\diamond \subset R^\diamond$.

(iv) Finally, we claim that $V(Y^\diamond) = S = V(\text{conv}Y') \cap Y$.

For this purpose, first we show that $v^k \notin \text{conv}(\{v^1, v^2, ..., v^l\} \setminus \{v^k\}) + \mathbb{R}^p_+$, for each $k \in \{1, 2, ..., l\}$.

Indeed, suppose in contrary that $v^k = u + \beta$, where u is a convex combination of $\{v^1, v^2, ..., v^l\} \setminus \{v^k\}$ and $\beta \in \mathbb{R}^p_+$. Note that for each $j \in \{1, ..., p\}$, $v^k + (y^M_j - v^k_j)e^j \in \text{conv}(Y')$ and $y^M_j > v^k_j$ therefore there is an $t_j > 0$ such that $v^k + te^j \in \text{conv}Y'$

for $0 \leq t \leq t_j$. Since convY' is convex, conv$(\bigcup_{j=1}^{p}\{v^k + te^j | 0 \leq t \leq t_j\}) \subset$ convY'. Let $v := v^k + \lambda\beta$, then there exists $\lambda > 0$ satisfying

$$v \in \text{conv}(\bigcup_{j=1}^{p}\{v^k + te^j | 0 \leq t \leq t_j\})$$

which means $v \in$ convY', and therefore $v^k = \frac{\lambda}{\lambda+1}u + \frac{1}{\lambda+1}v$ which contradicts the fact $v^k \in V(\text{conv}Y')$.

From (iii) and (iv) we can conclude that $V(Y^\diamond) = S = V(\text{conv}Y') \cap Y$.

■

Remark 3 **1.** Since conv(Y) = conv$(V(\text{conv}Y))$ in Algorithm 1 one can replace Y with $V(\text{conv}Y)$. In many cases, the set $V(\text{conv}Y)$ is significantly smaller than Y. In the case when the number of the criteria is relatively small, there exist existing efficient algorithms such as Quickhull ones [2] for computing $V(\text{conv}Y)$.

2. When $p = 2$, Algorithm 1 for computing $V(Y^\diamond)$ can be modified as follows.

- Step 1:
  $a_1 = \min\{y_1 | y \in Y\}$, $a_2 = \min\{y_2 | y \in Y, y_1 = a_1\}$, $a = (a_1, a_2)$.
  $b_2 = \min\{y_2 | y \in Y\}$, $b_1 = \min\{y_1 | y \in Y, y_2 = b_2\}$, $b = (b_1, b_2)$.
- Step 2: If $a = b$ then $V(Y^\diamond) = \{a\}$.
  Otherwise, let T be the set of all points of Y that lie below the line connecting a and b, that means

$$T = \{y = (y_1, y_2) \in Y \mid (y_1 - a_1)/(b_1 - a_1) - (y_2 - a_2)/(b_2 - a_2) \leq 0\}$$

  Then $V(Y^\diamond) = V(\text{conv}(T \cup \{a, b\}))$.

Having the vertices of the Edgeworth-Pareto hull we can solve Problem $(OP^\diamond)$ thereby, by Propositions 2.1, 2.2, to obtain a global solution of (OP) and (P) in the case of Problem $(OP^\diamond)$ has a solution. The algorithm is the following.

Algorithm 2

- **Step 1.** Use an algorithm, for example in [2], to compute all vertices $V(\text{conv}Y)$ of the convex hull of the outcome set $Y = f(X) = C.X$.
- **Step 2.** Apply Algorithm 1 with $Y := V(\text{conv}Y)$ to compute the vertex-set $V(Y^\diamond)$ of the Edgeworth-Pareto hull $Y^\diamond$.
- **Step 3.** Evaluating the objective function φ at every vertex of $V(Y^\diamond)$ to obtain a vertex that is a global optimal solution to $(OP^\diamond)$.

Remark 4 The following simple example shows that Problem $(OP^\diamond)$ has nonsolution while Problem (OP) solvable. In this case Algorithm 2 cannot be used.

Example. Consider the outcome set $Y = \{a, b, c, d\}$, where $a = (1, 3), b = (2, 3), c = (2.5, 2.5), d = (3, 1)$ as in Fig.3. It is easily seen that $Y_E = \{a, c, d\}$ and $V(Y^\diamond) = \{a, d\}$. The function $\varphi(y) = -y_1^2 - y_2^2$ is continuous and quasiconcave on $Y^\diamond$. The solution of problem (OP) is c and $c \notin V(Y^\diamond)$.

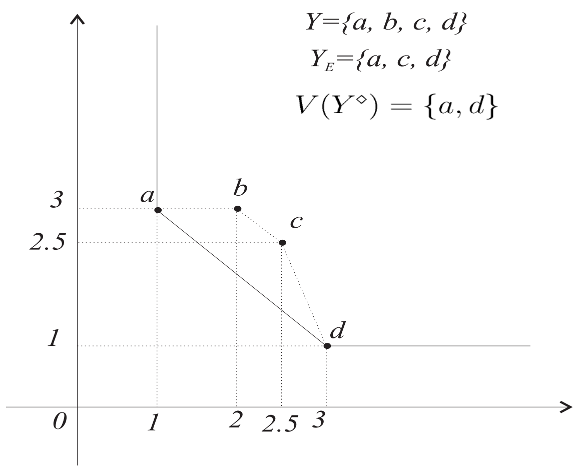

Figure 3. Example

4. Numerical experiments and results

We implemented the proposed algorithms by coding in Matlab R2012a, on an PC with RAM 8G, Intel core i7 2.26 GHz and Win7-64bit operating system.

To compute the convex hull of Y we use the Matlab command "convhulln" which was designed based on Quickhull algorithm [2].

The data is given as follows.

- $f(x) = Cx$, where the matrix $C \in \{0,1\}^{p\times n}$ is generated randomly.
- X including m vectors in R^n is regarded as a matrix of order $m \times n$. The entries of X is generated randomly in the interval $[0, 1000]$.
- The funtion φ is given by

$$\varphi_1(y) = \min\{B^1(y)^T z \mid z \in \{0,1\}^n, a^T z \le b\}$$

or

$$\varphi_2(y) = \min\{B^2(y)^T z \mid z \in \{0,1\}^n, a^T z \le b\}$$

where $a \in R^m$ is an integer vector randomly generated in $[0, 1000]$. The vector $b = \frac{1}{2}\sum\limits_{i=1}^{n} a_i$ is chosen as the capacity of Knapsack problem [18] and $B^k(y) : R^p \to R^n, k = 1, 2$ are given by

$$B_i^1(y) = \Pi_{j=1}^n y_j^{\alpha_j^i}, \alpha_j^i \ge 0, y_j > 0, i = 1, ..., n; j = 1, ..., p$$

and

$$B_i^2(y) = \sum_{j=1}^{n} \alpha_j^i \log(y_j + 1), \alpha_j^i \ge 0, y_j > 0, i = 1, ..., n; j = 1, ..., p.$$

Table 1. Computational results for Algorithm 1.

p	n	$m = 10^5$			$m = 10^6$		
		$\#V(\text{conv}Y)$	$\#V(Y^\diamond)$	CPU(s)	$\#V(\text{conv}Y)$	$\#V(Y^\diamond)$	CPU(s)
	20	17.4	3.6	0.0952	20.2	4.3	0.7972
2	50	15.3	3.6	0.1092	17.6	3.9	0.9313
	100	15.7	3.8	0.1544	17.7	3.8	1.3775
	20	80.8	9.1	0.117	111.8	9.9	0.9859
3	50	77.7	9.4	0.1778	94	8.9	1.0889
	100	72.2	7.2	0.2059	87.9	9.5	1.3806
	20	301	18.6	0.1763	450.9	18.4	1.2511
4	50	256.3	12.5	0.1981	307.2	15.3	1.4087
	100	231.8	13.3	0.2262	307.2	15.3	1.4087
	20	853.3	23.2	0.5772	1477.6	38	2.5943
5	50	658.7	21.8	0.4586	1060	25.8	2.3228
	100	621.8	19.8	0.4352	967.2	24.7	2.5225

It is clear that φ_1 and φ_2 are increasing and quasiconcave with respect to y. So by Remark 1 Problem $(OP^\diamond)$ is solvable and we can apply Algorithm 2 for solving Problem (OP).

To evaluate φ at each y we solve an integer program by using CVX 2.1 containing Gurobi Solver 6.0 (academic version at http://cvxr.com/cvx/ and http://www.gurobi.com/).

We tested Algorithms 1 and 2 for problems with $m = 10^6$, $m = 10^5$ and $p = 2, 3, 4, 5$. The coefficients $\{\alpha_j^i\}_{j=1,\dots,p}^{i=1,\dots,n}$ are randomly generated in $[0, 1]$, $[0, 0.5]$ and in $[0, 0.25]$ for $n = 20$, $n = 50$ and $n = 100$ respectively.

For each problem size we run 10 instances randomly generated and take the average over them. In tables below by $\#S$ we denote the average of the cardinality of a set S for 10 tested instances, where S can be $V(\text{conv}Y), V(Y^\diamond)$ or Y_E. We compared Algorithm 2 with the enumeration algorithm in [14] (denoted by Algorithm 3) that generates every point in Y_E and evaluates the objective function at each generated point to obtain the solution set. From the obtained computational results we can conclude the followings:

• Algorithm 1 for computing the Edgeworth-Pareto hull of a multiple objective discrete linear program is efficient when the cardinality of the constrained set can be large (10^6), while the number of the criteria is relatively small;

• Algorithm 2, which uses the Edgeworth-Pareto hull, for optimizing an increasingly quasiconcave lower semi-continuous function over the efficient set of a multiple objective discrete linear program is much more efficient than the enumeration algorithm in [14], whenever the constrained set is sufficiently large and the evaluation of the objective function requires much computational times.

Table 2. Comparison for running time between Algorithm 2 and Algorithm 3.

						φ_1-CPU(s)		φ_2-CPU(s)	
m	p	n	$\#V(\text{conv}Y)$	$\#V(Y^\diamond)$	$\#Y_E$	Algo. 2	Algo. 3	Algo. 2	Algo. 3
10^5	2	20	17.4	3.6	8.1	**0.4961**	1.5647	**0.4118**	1.4945
		50	15.3	3.6	6.2	**0.5148**	1.4305	**0.4212**	1.3884
		100	15.7	3.8	6.7	**0.5476**	1.2355	**0.468**	1.2168
	3	20	80.8	9.1	25.8	**0.883**	3.4382	**0.8268**	3.4164
		50	77.7	9.4	24.4	**1.0436**	3.705	**0.9204**	3.6379
		100	72.2	7.2	15.8	**0.9079**	3.3524	**0.8315**	3.2885
	4	20	301	18.6	55.4	**1.7628**	8.2446	**1.702**	8.1838
		50	256.3	12.5	40.4	**1.3073**	8.1417	**1.1778**	7.9202
		100	231.8	13.3	40.7	**1.3666**	6.1527	**1.2558**	5.9187
	5	20	853.3	23.2	73.3	**2.6767**	17.6843	**2.4476**	17.0915
		50	658.7	21.8	74.7	**2.5085**	16.0198	**2.3478**	15.8404
		100	621.8	19.8	71.7	**2.028**	10.1697	**1.989**	10.0543
10^6	2	20	20.2	4.3	8.4	**1.1716**	6.4912	**1.1794**	6.4319
		50	17.6	3.9	6.9	**1.2527**	7.5161	**1.2698**	7.5161
		100	17.7	3.8	7.4	**1.7004**	5.2385	**1.7098**	5.1886
	3	20	111.8	9.9	23.9	**1.8548**	14.1774	**1.8174**	14.11654
		50	94	8.9	26.4	**1.9812**	17.6796	**1.8751**	17.5361
		100	87.9	9.5	27.9	**2.3057**	19.8917	**2.1887**	19.848
	4	20	450.9	18.4	61.2	**2.8532**	57.0823	**2.6676**	56.8982
		50	356.5	15.7	61	**2.4617**	41.3278	**2.4461**	41.0922
		100	307.2	15.3	64.3	**2.6411**	42.4744	**2.6005**	42.2607
	5	20	1477.6	38	153.5	**5.694**	119.2986	**5.577**	119.26276
		50	1060	25.8	96.3	**4.6691**	90.2169	**4.4164**	90.1218
		100	967.2	24.7	104.9	**4.6597**	79.9053	**4.4725**	79.7103

5. Conclusion

We have proposed an induction algorithm in outcome space for finding the Edgeworth-Pareto hull of multiple objective discrete linear programming problems, where the constrained set is given by a finite number of points in the decision space. In the case the cardinality of the constrained set is very large, in order to reduce computational times, we have used an existing algorithm such as Quickhull one to replace the constrained set by the vertices of its convex hull. Application to minimization problems of an increasingly quasiconcave function over the efficient set by using the Edgeworth-Pareto hull has been discussed. Some numerical results showing the efficiency of the proposed algorithms have been reported.

References

[1] L.T.H. An, L.D. Muu, P.D. Tao, *Numerical solution for optimization over the efficient set by d.c. optimization algorithms*, Oper. Res. Lett. 19 (1996), pp. 117-128.

[2] C.B. Barber, D.P. Dobkin and H.T. Huhdanpaa, *The Quickhull Algorithm for Convex Hulls*, ACM Transactions on Mathematical Software 22(4) (1996), pp. 469-483.

[3] H.P. Benson, *Existence of efficient solutions for vector maximization problem*, J. Optim. Theory Appl. 26 (1978), pp. 569-580.

[4] H.P. Benson, *Efficiency and proper efficiency in vector maximization with respect to cones*, J. Optim. Theory Appl.39 (1984), pp. 125-132.

[5] H.P. Benson, *Optimization over the efficient set*, J. Math. Anal. Appl. 98 (1984), pp. 562-580.

[6] H.P. Benson, *An all-linear programming relaxation algorithm for optimizing over the efficient set*, J. Glob. Optim. 1 (1991), pp. 84-104 .

[7] H.P. Benson, *A geometrical analysis of the efficient outcome set in multiple objective convex programs with linear criterion functions*, J. Glob. Optim. 5 (1995), pp. 231-251.
[8] S. Bilintineanu, *Minimization of a quasi-concave function over an efficient set*, Math. Prog. 61 (1993), pp. 89-110.
[9] G.R. Bitran and T.L. Magnanti, *The structure of admissible point with respect to cone dominance*, J. Optim. Theory Appl. 29 (1979), pp. 573-614.
[10] J.P. Dauer, *Analysis of objective space in multiple objective linear programming*, J. Math. Anal. Appl. 126 (1987), pp. 579-583.
[11] J.P. Dauer and J.H. Liu, *Solving multiple objective linear programs in objective space*, European J. Oper. Res. 46 (1990), pp. 350-357.
[12] J.P. Dauer and T. Fosnaugh, *Optimization over the efficient set*, J. Glob. Optim. 7 (1995) , pp. 261-277.
[13] R.V. Efremov, *Convergence of Haussdorf approximation method for the Edgeworth-Pareto hull of a compact set*, Comput. Math. and Math. Phys. 25 (2015), pp. 1171-1178.
[14] M. Ehrgott. *Multicriteria Optimization*, Springer, Berlin, 2005.
[15] M. Ehrgott, X. Gandibleux and A. Przybylski, *Exact Methods for Multi-Objective Combinatorial Optimisation*, Inter. Series in Oper. Res. & Manag. Sci. 233 (2016), 817-850.
[16] J. Fülöp, *A cutting plane algorithm for linear optimization over the efficient set*, in: Generalized Convexity, Lecture notes in Economics and Mathematical System, vol. 405 (1994), pp. 374-385, Springer-Verlag, Berlin.
[17] R. Horst, N.V. Thoai, Y. Yamamoto and D. Zenke, *On optimization over the efficien set of linear multicriteria programming*, J. Optim. Theory Appl. 134 (2007), pp. 433-443.
[18] G. Kirlik, S. Sayın. *A new algorithm for generating all nondominated points of multiobjective discrete optimization problems*, Eur. J. Oper. Res. 232 (2014), pp. 479-488.
[19] N.T.B. Kim, L.D. Muu, *On the projection of the efficient set and potential application*, Optim. 51 (2002), pp. 401-421.
[20] D. T. Luc, *Theory of Vector Optimization*, Springer-Verlag, Berlin, Germany, 1989.
[21] P.H Naccache, *Stability in Multicriteria Optimization*, J. Math. Anal. Appl. 68 (1979), pp. 441-453.
[22] J. Philip, *Algorithms for vector maximization problem*, Math. Prog. 2 (1972), pp. 85-106.
[23] A.I. Pospelov, *Approximating the Convex Edgeworth Pareto Hull in Integer Multi objective Problems with Monotone Criteria*, Comput. Math. & Math. Physics 49 (10) (2009), pp. 1686-1699.
[24] A.I. Pospelov, *Haussdorf methods for approximating the convex Edgeworth-Pareto hull in integer problems with monotone objectives*, Comput. Math. & Math. Phys. 56 (2016), pp. 1388-1401.
[25] H.Q. Tuyen and L.D. Muu, *Biconvex programming approach to optimization over the weakly efficient set of a multiple objective affine fractional programming problem*, Oper. Res. Lett. 28 (2001), pp. 81-92.
[26] P.T. Thach, H. Konno and D. Yokoda, *Dual approach to optimization on the set of Pareto-optimal solutions*, J. Optim. Theory Appl. 88 (1996), pp. 869-707.
[27] P.L. Yu, *Multiple-Criteria Decision Making*, Plenum Press, New York and London, 1985.