Mathematical Programming 13 (1977) 121–139.
North-Holland Publishing Company

LINEAR MULTIPLE OBJECTIVE PROGRAMS WITH ZERO-ONE VARIABLES

Gabriel R. BITRAN*
University of São Paulo, SP, Brazil

Received 6 August 1976
Revised manuscript received 17 March 1977

Theoretical results are developed for zero-one linear multiple objective programs. Initially a simpler program, having as a feasible set the vertices of the unit hypercube, is studied. For the main problem an algorithm, computational experience, parametric analysis and indifference sets are presented. The mixed integer version of the main problem is briefly discussed.

Key words: Zero-one multiple objective programs, Multiple objective optimization, Vector optimization, Multi criteria optimization, Efficient points, Integer programming.

1. Introduction

Multiple objective programs with continuous variables have been exhaustively treated in the literature [3, 5, 7, 9]. However very little has been done for the zero-one case. Shapiro [8] mentions several applications and relates our central problem to recent results in integer programming duality theory [1]. The main objective of this paper is to provide theoretical results that hopefully will lead to a better understanding of this problem. As a by-product an algorithm to determine all efficient solutions is obtained and computational results are presented. The linear multiple objective program with zero-one variables is written as

(P) $\qquad \max\{Cx\colon x \in F\}$

where $F = \{x \in \mathbf{R}^n : Ax \leqq b,\ x_j = 0,\ 1,\ j \in J\}$, C is a $p \times n$ matrix, A is an $m \times n$ matrix, b is an $m \times 1$ vector and $J = \{1, 2, \ldots, n\}$.

A typical practical application that can be reduced to this model is the "Project Selection Problem". The columns a^j of A correspond to projects to be selected or rejected by p interested parties on the basis of the $p \times 1$ evaluation vectors c^j (the columns of C).

In this paper the partial ordering relation $x \geq y$ means $x_j \geqq y_j$ $j \in J$ with at least a strict inequality. The set of interest in a multiple objective program is the set of efficient solutions. Specifically $x^0 \in F$ is said to be efficient in (P) if there is no

* This research was partially supported by the "Fundação Carlos Alberto Vanzolini".

other $x \in F$ satisfying $Cx \geq Cx^0$. The set of efficient solutions to (P) is denoted by EF(P).

A central difference between convex and discrete multiple objective programs is that in the former case, if the Kuhn–Tucker constraint qualification holds, every efficient point in (P) maximizes a linear functional of the type λCx, for a $\lambda \in R^p$ $\lambda > 0$, on the feasible set [2, 5]. In the later case this may not happen as is shown in the example below:

$$\max\left\{\begin{pmatrix} 1 & -2 \\ -1 & 3 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}: x_1 = 0, 1 \text{ and } x_2 = 0, 1\right\}. \tag{1.1}$$

The point $(x_1, x_2) = (0, 0)$ is efficient in (1.1) but does not maximize a functional λCx, for any $\lambda \geq 0$, on F.

The plan of the paper is as follows. In Section 2 the auxiliary problem

(P′) $\max\{Cx\colon x_j = 0, 1, j \in J\}$

is studied in detail. This problem has shown to be very useful for our purpose, a reason being that every efficient point in (P′), that is feasible in (P), belongs to EF(P). The set $V = \{v \in \mathbf{R}^n : v_j = 0, 1 \text{ or } -1, j \in J\}$ and a point to set map defined on V are introduced. They play a central role in the theory developed here. Section 3 is devoted to the analysis of (P). The results obtained in Section 2 are extended to (P). An algorithm to generate EF(P) is described. Similarly to what is done in Kornbluth's paper [6] on the continuous multiple objective linear program, we characterize, for the zero-one case, the indifference set

$$S(x^*) = \left\{\lambda \geqq 0 \sum_{i=1}^{p} \lambda_i = 1\colon x^* \text{ solves } \max\{\lambda Cx\colon x \in F\}\right\}$$

in terms of a linear system and the set EF(P). An approximation to $S(x^*)$ independent of EF(P) is also obtained. Section 3 is concluded with results on parametric analysis. In Section 4 we comment on the mixed zero-one version of (P) and determine a local approximation to indifference sets. Some concluding remarks and areas for future research are briefly discussed in Section 5. For future reference we define the following sets and problems.

$J = \{1, 2, \ldots, n\}$.

$\text{DUC} = \{x \in \mathbf{R}^n : x_j = 0, 1, j \in J\}$.

$\text{UC} = [\text{DUC}] = \{x \in \mathbf{R}^n : 0 \leqq x_j \leqq 1, j \in J\}$ where [] indicates the convex hull.

(P): $\max\{Cx\colon x \in F\}$ where $F = \{x \in \mathbf{R}^n : Ax \leqq b,\ x \in \text{DUC}\}$, C is a $p \times n$ matrix, A is an $m \times n$ matrix and b is an $m \times 1$ vector.

(P′): $\max\{Cx\colon x \in \text{DUC}\}$.

(P″): $\max\{Cx\colon x \in \text{UC}\}$.

EF(P) $\equiv$ the set of efficient points in (P). ($x^0 \in F$ is efficient in (P) if there is no $x \in F$ such that $Cx \geq Cx^0$).

EF(P′) and EF(P″) are respectively the set of efficient points in (P′) and (P″). They are defined similarly to EF(P).

$\text{EF(P}')^c$ and EF(P)^c are the sets of non-efficient points in (P′) and (P) respectively.

The difference of two sets A and B is denoted by $A - B = \{x \in A : x \notin B\}$. Lower case letters are used to denote vectors. Superscripts differentiate vectors and subscripts indicate the components of a vector. The transpose sign is not indicated when the scalar product of two vectors is clear. Otherwise a capital T is used as a subscript.

2. The auxiliary problem (P′)

This section is devoted to the characterization of EF(P′) through the set $V = \{v^i \in \mathbf{R}^n,\ i \in K \mid Cv^i \geq 0$ and $v^i_j = 0,\ 1$ or $-1,\ i \in K,\ j \in J\}$ where $K = \{1, 2, \dots, k\}$. The vectors v^i $i \in K$ are contained in the reverse polar cone to the cone defined by the rows of C. They are the directions of preference, i.e., if $x^2 = x^1 + v^i$ for some $i \in K$ then $Cx^2 \geq Cx^1$. We say that x^2 dominates x^1 in the direction v^i. The set of points, in DUC, dominated in the direction v^i can be characterized as the image of the following point to set map

$$M : V \to \text{DUC} \quad \text{as} \quad M(v^i) = \{x^{ir}\}_r,$$

where

$$\begin{cases} x^{ir}_j = 0 & \text{if } v^i_j = 1, \\ x^{ir}_j = 1 & \text{if } v^i_j = -1, \\ x^{ir}_j = 0, 1 & \text{if } v^i_j = 0. \end{cases}$$

The cardinality of $M(v^i)$, for each $i \in K$, is greater than one if at least a component of v^i is zero.

At this point we are in position to introduce the first results.

Lemma 2.1. (a) *If $x \in M(v)$ for some $v \in V$, then $x + v \in \text{DUC}$ and $C(x + v) \geq Cx$, i.e., x is nonefficient in* (P′).

(b) *If $x \in \text{DUC}$ is nonefficient in* (P′) *there exists a $v \in V$ such that $x \in M(v)$.*

(c) *Let $x \in \text{DUC}$ and $x \notin M(v)$ for some $v \in V$. Then $x + v \notin \text{DUC}$.*

Proof. (a) The definitions of the map $M(.)$ and of the set V imply respectively $M(v) + \{v\} \subset \text{DUC}$ and $C(x + v) \geq Cx$.

(b) If $x \in \text{EF(P}')^c$ there is $x^1 \in \text{DUC}$ satisfying $C(x^1 - x) \geq 0$. By letting $v = x^1 - x$ it follows that for $j \in J$, $v_j = 1$, -1 or 0 implies that $x_j = 0$, 1, 0 or 1 respectively, i.e. $x \in M(v)$.

(c) $x \notin M(v)$ implies that for some $j \in J$ either $v_j = 1$ and $x_j = 1$ or $v_j = -1$ and $x_j = 0$. Therefore $x + v \notin \text{DUC}$.

Theorem 2.2. *$\bigcup_{i \in K} M(v^i)$ is the set of non-efficient points in* (P′), *i.e.*, $\bigcup_{i \in K} M(v^i) = \text{EF(P}')^c$.

Proof. Lemma 2.1(a) and 2.1(b) show respectively that $\bigcup_{i \in K} M(v^i) \subseteq \text{EF(P}')^c$ and $\text{EF(P}')^c \subseteq \bigcup_{i \in K} M(v^i)$.

Corollary 2.3. $V=\emptyset$ *implies* $\mathrm{EF}(\mathrm{P}')=\mathrm{DUC}$.

Each $M(v^i)$, $i\in K$ can be characterized as the set of optimal basic solutions to a linear programming problem:

Theorem 2.4. *For each* $i\in K$, $M(v^i)$ *is the set of extreme points that solve* $\min\{v^i x\colon x\in \mathrm{UC}\}$.

Proof. Note that for every $x\in M(v^i)$ and every $j\in J$ we have $v^i_j x_j=-1$ if $v^i_j=-1$ and $v^i_j x_j=0$ otherwise.

Let (P'') be the corresponding problem to (P') obtained when the integrality conditions are relaxed to $0\leqq x_j\leqq 1$, $j\in J$ and let CV be the cone generated by v^i, $i\in K$. Consider the preference cone $\mathrm{PC}=\{p\in \mathbf{R}^n : Cp\geq 0\}$. An alternative definition for efficiency is: given a set X, $x^0\in X$ is efficient if $X\cap(\{x^0\}+\mathrm{PC})=\{x^0\}$. Note that $\mathrm{CV}\subseteq \mathrm{PC}$. As DUC is the set of extreme points of UC, when the efficient points in (P') are contained in $\mathrm{EF}(\mathrm{P}'')$, we have $\mathrm{EF}(\mathrm{P}')$ directly generated by algorithms developed for continuous linear multiple objective programs [2, 3, 9]. Unfortunately this condition is not always true. To be more explicit if $x^0\in \mathrm{EF}(\mathrm{P}')$, i.e., $\mathrm{DUC}\cap(\{x^0\}+\mathrm{PC})=\{x^0\}$, it is possible that $\mathrm{UC}\cap(\{x^0\}+\mathrm{PC})\neq\{x^0\}$ or even $\mathrm{UC}\cap(\{x^0\}+\mathrm{CV})\neq\{x^0\}$ as shown below.

Example 2.5.

$$n=4,\quad C=\begin{bmatrix}-1 & -1 & 0 & 0\\ 1 & 1 & 0 & \\ 1 & 2 & 1 & -2\\ -1 & -2 & -1 & 0\\ 0 & 0 & 1 & 0\end{bmatrix},\quad \mathrm{DUC}=\{x\in R^4\colon x_j=0,1,j\in J\}$$

and $\mathrm{UC}=\{x\in R^4 : 0\leqq x_j\leqq 1, j\in J\}$. It follows that $V=\{v^1=(-1,1,1,1),\ v^2=(1,-1,1,0)\}$. By Theorem 2.2, $x^0=(0,0,0,0)$ is efficient in (P'). However $x^1=(0,0,1,\frac{1}{2})$ is contained in CV and satisfies $C(x^1-x^0)\geq 0$. Thus $x^0\notin \mathrm{EF}(\mathrm{P}'')$. This type of example cannot be obtained in a lower dimension space. When for every $x^0\in\mathrm{EF}(\mathrm{P}')$, $\mathrm{UC}\cap(\{x^0\}+\mathrm{CV})=\{x^0\}$ we say that (P') is regular.

The preference set that is relevant to zero-one multiple objective programs is CV. When (P') is regular we can reduce the obtainment of $\mathrm{EF}(\mathrm{P}')$ to the determination of efficient extreme points in

$$(\bar{\mathrm{P}}')\qquad \max\{\bar{C}x\colon x\in \mathrm{UC}\}$$

Where $\bar{C}$ is the objective matrix whose rows are a set of generators of the reverse polar cone to CV. Since in some cases the determination of $\bar{C}$ is simple, see example 2.6, it is important to determine sufficient conditions for (P') to be regular.

Example 2.6. Suppose

$$C=\begin{bmatrix}2 & 3 & -6\\ 3 & 2 & -4\\ 0 & 0 & 2\end{bmatrix},$$

then $V=\{(1,0,0),\ (0,1,0),\ (1,1,0)\}$ and

$$\bar{C}=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\\ 0 & 0 & -1\end{bmatrix}.$$

We say that a subset $\{v^i\}_{i=1}^q$ of V is a set of generators of V if any v^j, $j\in K$ can be written as $v^j=\sum_{i=1}^q \alpha_i v^i$ with $\alpha=(\alpha_1,\ldots,\alpha_q)\geq 0$. Note that $0\in\mathbf{R}^n$ is not in V. Let e^i be a column vector with $e_i^i=1$, $e_j^i=0$, $j\neq i$.

Lemma 2.7. *If* $\{v^i\}_{i=1}^q$, *with* $v^i=e^i$ *or* $-e^i$, *is a set of generators of* V, *then* $\mathrm{EF(P')}=\{x\in\mathrm{DUC}$ *such that* $x_i=\max(0,v_i^i)$, $i=1,2,\ldots,q\}$.

Proof. Assume $x\in\mathrm{EF(P')}$ is such that for a fixed i, $x_i\neq\max(0,v_i^i)$. Then if $v^i=e^i(-e^i)$ we have that $x+e^i(x-e^i)$ is an element of DUC contradicting the assumption.

Theorem 2.8. *A sufficient condition for* (P′) *to be regular is that* $\{v^i\}_{i=1}^q$, *with* $v^i=e^i$ *or* $-e^i$, *be a system of generators of* V. *Moreover the hyperplane* $\sum_{i=1}^q v_T^i x=\sum_{i=1}^q v_T^i x^0$ *separates* UC *and the set* $\{x^0\}+\mathrm{CV}$ *at* x^0 *for every* $x^0\in\mathrm{EF(P')}$.

Proof. Assume $x^0\in\mathrm{EF(P')}$. By Lemma 2.7 $x_i^0=\max(0,v_i^i)$, $i=1,2,\ldots,q$. Let $z=\sum_{i=1}^q v^i$. Any $v\in V$ can be written as: $v=\sum_{i=1}^q\alpha_i v^i$ with $(\alpha_1,\ldots,\alpha_q)\geq 0$ thus, $z_T(x^0+v)=z_Tx^0+z_Tv=z_Tx^0+\sum_{i=1}^q\alpha_i>z_Tx^0$, showing that $\{x^0\}+\mathrm{CV}$ is on one side of the hyperplane $z_Tx=z_Tx^0$. To see that UC is on the opposite side, note that the n adjacent vertices x^{0s} $(s=1,2,\ldots,n)$ to x^0 in DUC are obtained from x^0 by changing its n components one at a time from 0 to 1 or from 1 to 0. Therefore, since $x_i^0=\max(0,v_i^i)$ for $1\leqq i\leqq q$ and $z_s=0$ for $q+1\leqq s\leqq n$ we have

$$\begin{aligned}
&x^{0s}=x^0-v_s^s e^s \quad \text{for } 1\leqq s\leqq q,\\
&z_Tx^{0s}=z_Tx^0-z_T(v_s^s e^s)<z_Tx^0,\quad s=1,2,\ldots,q,\\
&z_Tx^{0s}=z_Tx^0,\quad s=q+1,\ldots,n,
\end{aligned}$$

proving that z defines a separating hyperplane. Furthermore x^0 solves

$$\max\{z_Tx:\ x\in\mathrm{UC}\}. \tag{2.1}$$

Corollary 2.9. *If* $\{v^i\}_{i=1}^q$, *with* $v^i=e^i$ *or* $-e^i$, *is a system of generators of* V, *then* EF(P′) *is the set of optimal extreme points of* (2.1).

The set V can be obtained through an implicit enumeration scheme. Since the number of potential elements of V is of the order of 3^n it is important to find strong rules that reduce significantly the enumeration. Lemmas 2.10 and 2.11 are a first step in this direction.

Let J_0 be a subset of J.

Lemma 2.10. *Assume v^1, $v^2 \in V$, $v_j^1 = v_j^2$, $j \notin J_0$ and $v_j^1 = 0$, $j \in J_0$. Then $M(v^2) \subset M(v^1)$.*

Proof. Let $y \in M(v^2)$. By Lemma 2.1(a) $y + v^2 \in \mathrm{DUC}$. Hence, $y_j = 0$ or 1 and $y_j + v_j^2 = 0$ or 1, $j \in J$. Clearly $y_j + v_j^1 = 0$ or 1, $j \notin J_0$. Moreover since $v_j^1 = 0$ we also have $y_j + v_j^1 = 0$ or 1, $j \in J_0$. Thus, $y + v^1 \in \mathrm{DUC}$. By Lemma 2.1(c) it follows that $y \in M(v^1)$.

This lemma suggests one start the enumeration with vectors having as many zero components as possible. In fact, if some $v^1 \in V$ has m zero components we discard at least $3^m - 1$ vectors of the type v^2 because if any of them is in V we will have $M(v^2) \subset M(v^1)$.

Lemma 2.11. *If C is non-negative, the only n-vectors necessary to construct* EF(P′) *are $v^i = e^i$, $i \in J$.*

The proof follows by Lemma 2.10 and the fact that any $v \in V$ must have at least one component, corresponding to a non-zero column of C, equal to one.

Next we present a result that will be useful in the algorithm to obtain EF(P).

Lemma 2.12. *Let v^1 and v^2 be as defined in Lemma 2.10. Then $M(v^2) + \{v^2\} \subset M(v^1) + \{v^1\}$.*

Proof. Assume $x \in M(v^2)$. Define $y_j = x_j + v_j^2$, $j \in J_0$ and $y = x_j$, $j \notin J_0$. By Lemma 2.1(a) for $j \in J_0$ we have $y_j = 0$ or 1. Clearly $y \in M(v^1)$ and $y + v^1 = x + v^2$.

In Section 3 an algorithm to obtain EF(P) is presented together with computational results. However, theoretical results, as the next three lemmas, may suggest alternative algorithms.

Lemma 2.13. *Let $x^1 \in M(v^1)$ and $x^2 \in M(v^2)$ be such that $x^2 = x^1 + v^1$. Then $v_j^1 v_j^2 \leqq 0$, $j \in J$.*

Proof. Since $x^2 = x^1 + v^1$ and $x^i + v^i \in \mathrm{DUC}$, $i = 1, 2$ we have $x^1 + v^1 + v^2 \in \mathrm{DUC}$. Then,

if $v_j^1 = 1 \;\Rightarrow x_j^1 = 0$ and $v_j^2 = 0$ or -1,
if $v_j^1 = -1 \Rightarrow x_j^1 = 1$ and $v_j^2 = 0$ or 1 and
if $v_j^1 = 0 \;\Rightarrow x_j^1 = 0$ (or 1) and $v_j^2 = 0$ or 1 (0 or -1).

Thus $v_j^1 v_j^2 \leqq 0$, $j \in J$.

Lemma 2.14. *Let $x^0 \in M(v^1)$ and $x^0 \in M(v^2)$. Then $v_j^1 v_j^2 \geqq 0$, $j \in J$.*

Proof. $x^0 \in M(v^1) \cap M(v^2)$ implies, together with the definition of the map M, that if for some $j \in J$, v_j^1 and v_j^2 are nonzero then $v_j^1 = v_j^2$.

Lemma 2.15. *Let $x^0 \in \mathrm{DUC}$, $x^0 - v^1 \in \mathrm{DUC}$ and $x^0 - v^2 \in \mathrm{DUC}$. Then $v_j^1 v_j^2 \geqq 0$, $j \in J$.*

Proof. For any $j \in J$ if $x_j^0 = 1(0)$ it follows that $v_j^1 = 0$ or 1 (0 or -1) and $v_j^2 = 0$ or 1 (0 or -1). Therefore $v_j^1 v_j^2 \geqq 0$, $j \in J$.

3. The analysis of (P)

In this section we apply the results, obtained up to this point to analyse (P). Besides the trivial observation as to the inclusion of the extra feasibility condition $Ax \leqq b$, it is important to note that although every $x \in \mathrm{EF(P')} \cap F$ is efficient in (P), a non-efficient point in (P′) is not necessarily non-efficient in (P). The following lemma relates (P) and (P′).

Lemma 3.1. (a) *If for some $u \in K$, $Av^u \leqq 0$ then $F \cap M(v^u) \subseteq \mathrm{EF(P)}^c$.*

(b) *Assume $x^0 \notin \mathrm{EF(P')}$, $x^0 \in \mathrm{EF(P)}$ and let $I(x^0) = \{i \in K : x^0 \in M(v^i)\}$. Then $x^0 + v^i \notin F$ for all $i \in I(x^0)$.*

Proof. (a) Let $x \in F \cap M(v^u)$. Then, $x + v^u \in \mathrm{DUC}$ and $A(x + v^u) = Ax + Av^u \leqq Ax \leqq b$, i.e., $x + v^u \in F$. Since $C(x + v^u) \geq Cx$ it follows that $x \in \mathrm{EF(P)}^c$.

(b) Let $x^0 \in \mathrm{EF(P)}$. Since $C(x^0 + v^i) \geq Cx^0$ it follows that $x^0 + v^i \notin F$ for all $i \in I(x^0)$

Let $s_u = \max_{x \in \mathrm{DUC}} (b_u - A_u x)$, $u = 1, 2, \ldots, m$ where b_u and $A_u = (A_{u1}, \ldots, A_{un})$ are the uth component and row of b and A respectively.

$s_u = b_u - \min_{x \in \mathrm{DUC}} A_u x = b_u - \Sigma_{A_{uj}<0} A_{uj}$, if all $A_{uj} \geqq 0$ define $s_u = b_u$.

Lemma 3.2. *If for every $i \in K$ there is $1 \leqq u(i) \leqq m$ such that $A_{u(i)} v^i > s_{u(i)}$, then every point of F is efficient in* (P).

Proof. Suppose $x^0 \in F$ and $x^0 \in \mathrm{EF(P)}^c$. There is $\bar{x} \in F$ such that $C(\bar{x} - x^0) \geq 0$. By letting $v^i = \bar{x} - x^0$ we have $v^i \in V$. Thus,

$$A_{u(i)}\bar{x} = A_{u(i)}x^0 + A_{u(i)}v^i > \min_{x \in \mathrm{DUC}} A_{u(i)}x + s_{u(i)} = b_{u(i)},$$

Contradiction!

Lemma 3.2 gives a sufficient condition for $\mathrm{F} = \mathrm{EF(P)}$. For the application mentioned in Section 1 it is useful to establish relations between a project, or equivalently the value of a binary variable, and the elements of EF(P).

Lemma 3.3. (a) *If for a fixed* j, $v_j^i = 1(-1)$ *for all* $i \in K$, *then any* $x \in F$ *with* $x_j = 1(0)$ *is efficient in* (P).

(b) *If some* $v \in V$ *is such that* $v = e^i(-e^i)$ *and* $Av \leqq 0$, *then for any* $x \in \mathrm{EF(P)}$ *we have* $x_i = 1(0)$.

Proof. (a) If $x \in F$ and $x_j = 1(0)$ it follows that for all $i \in K$, $v_j^i + x_j = 2(-1)$. Thus, $x + v^i \notin \mathrm{DUC}$, $i \in K$ and therefore $x \in \mathrm{EF(P)}$.

(b) If $x \in F$ and $x_i \neq 1(0)$ we have that $x + v \in F$ and consequently $x \notin \mathrm{EF(P)}$.

The results presented contain enough information to develop an algorithm to determine EF(P). Specifically Lemmas 2.10, 2.11 and Lemmas 3.1(a), 2.12 and 3.1(b) play a relevant role in steps 1 and 4 respectively.

Algorithm

The idea of the algorithm is to obtain EF(P) as the union of the points efficient both in (P′) and (P) with the nonefficient points in (P′) which are efficient in (P). This is accomplished in step 5. Recall that by Lemma 3.1(b) a nonefficient point x^0 in (P′) is efficient in (P) if it is not dominated in F in the directions v^i for all i such that $x^0 \in M(v^i)$. The algorithm first determines (step 1) the v^i's necessary to generate $\bigcup_{i=1}^k M(v^i)$. By using Lemmas 2.10 and 2.11 one often needs just a subset of V which is denoted by $\{v^i\}_{i=1}^r$. The nonefficient points in (P′) are directly obtained (step 2) as $\mathrm{EF(P')}^c = \bigcup_{i=1}^r M(v^i) = \bigcup_{i=1}^k M(v^i)$. The elements of DUC efficient both in (P′) and (P) are those in $\mathrm{EF(P')} \cap F$. These are determined in step 3 by first computing EF(P′) as $\mathrm{DUC} - \mathrm{EF(P')}^c$ and next intersecting this set with F. In step 4(a) the set $\Psi = \bigcup_{i \in I^1} M(v^i)$ is obtained. According to Lemma 3.1(a) it is the set of nonefficient points in (P′) which if feasible in (P) will also be nonefficient in (P). These points are excluded from $M(v^i)$, $i \in I^2$ in step 4(c). The elements of DUC which dominates the points in $M(v^i)$ in the direction v^i are determined in step 4(b). At this point it is important to recall that because of Lemmas 2.10 and 2.12 it is sufficient to consider the subset $\{v^i\}_{i=1}^r$ instead of V. In step 4(d) the points efficient both in (P) and (P′) are used to reduce the candidates as elements in EF(P). These candidates are reduced further in steps 4(e) and (f) by first checking for feasibility in (P) and next by verifying if they are dominated in the direction v^i by a point in F. Finally, the subtraction in step 4(g) is necessary because the sets used in the algorithm up to this point may not be disjoint.

The algorithm is as follows:

Let $I^1 = \{1 \leqq i \leqq r : Av^i \leqq 0\}$ and $I^2 = \{1, 2, \ldots, r\} - I^1$.

Step 1. Through an implicit enumerative scheme generate the subset $\bar{v} = \{v^i\}_{i=1}^r$ of V, using Lemmas 2.10 and 2.11, which is necessary to construct $\mathrm{EF(P')}^c = \bigcup_{i=1}^r (Mv^i) = \bigcup_{i=1}^k M(v^i)$.

Note that $r \leqq k$.

Step 2. Obtain $\mathrm{EF(P')}^c$.

Step 3. Obtain $\mathrm{EF(P')} \cap F = (\mathrm{DUC} - \mathrm{EF(P')}^c) \cap F$.

Step 4. Obtain

(a) $\Psi = \bigcup_{i\in I^1} M(v^i)$,
(b) $\Delta_i = M(v^i) + \{v^i\}$, $i \in I^2$,
(c) $\Omega_i = M(v^i) - \Psi$, $i \in I^2$,
(d) $\Lambda_i = \Omega_i - \{x \in \Omega_i : \exists y \in \mathrm{EF(P')} \cap F \text{ with } Cy \geq Cx\}$, $i \in I^2$,
(e) $\eta_i = \Lambda_i \cap F$, $i \in I^2$,
(f) $\theta_i = \{x \in \eta_i : \exists y \in \Delta_i \cap F \text{ with } Cy \geq Cx\}$.
$\theta_i^c = \eta_i - \theta_i$, $i \in I^2$,
(g) $\phi = \bigcup_{i\in I^2} \theta_i - (\bigcup_{i\in I^2} \theta_i^c)$.
Step 5. $\mathrm{EF(P)} = (\mathrm{EF(P')} \cap F) \cup \phi$. (3.1)

The proof of equality (3.1) is based on the lemmas mentioned above. Lemma 3.1(a) justifies the exclusion of the elements of ψ as possible candidates to EF(P). The procedure described in items (b) to (g) of step 4 to obtain the set ϕ is due to Lemmas 2.10, 2.12 and 3.1(b).

The following straightforward numerical example illustrates the algorithm.

Example 3.4. Consider the problem $\max\{[\begin{smallmatrix}1&0\\0&1\end{smallmatrix}]x: x \in F\}$ where $F = \{x \in R^2 : x_2 \leqq \frac{1}{2}, x_i = 0, 1, i = 1, 2\}$. We have

Step 1. $\{v^i\}_{i=1}^r = \{v^1 = (1, 0), v^2 = (0, 1)\}$ (note that the set V includes besides v^1 and v^2 the vector $v^3 = (1, 1)$). It is clear that $I^1 = \{1\}$ and $I^2 = \{2\}$.

Step 2. $M(v^1) = \{(0, 0), (0, 1)\}$, $M(v^2) = \{(0, 0), (1, 0)\}$ and $\mathrm{EF(P')}^c = M(v^1) \cup M(v^2)$.

Step 3. $\mathrm{EF(P')} = \{(1, 1)\}$, $\mathrm{EF(P')} \cap F = \emptyset$.

Step 4. (a) $\Psi = \{(0, 0), (0, 1)\}$
(b) $\Delta_2 = \{(0, 1), (1, 1)\}$
(c) $\Omega_2 = \{(1, 0)\}$
(d) $\Lambda_2 = \{(1, 0)\}$
(e) $\eta_2 = \{(1, 0)\}$
(f) $\theta_2 = \{(1, 0)\}$, $\theta_2^c = \emptyset$
(g) $\phi = \theta_2 = \{(1, 0)\}$.

Step 5. $\mathrm{EF(P)} = \{(1, 0)\}$.

Computational results

Tables 1 and 2 present the computational results obtained with 30 problems of the type

$$\max\{Cx: Ax \leqq b, x \in \mathrm{DUC}\}$$

The elements of matrices C and A were randomly generated in the interval [0, 99] and a density of negative elements equal to 20% was used in both matrices. The elements of b were randomly generated in the interval [0, 999]. The computer used is a Borroughs B6700. The program was written in Fortran GH. In Tables 1 and 2 the dimensions of C and A are indicated as well as cpu times in seconds, the number of elements of V necessary to obtain $\mathrm{EF(P')}^c$, the number of elements in I^1, the number of elements in EF(P'), EF(P) and F. Finally, for comparison purpose, we present in the column "Time By Definition"

Table 1

		Time to obtain EF(P′) in (sec)	Time to obtain EF(P) in (sec)	Total time (sec)	No. of elements in $\bar{V}$	No. of elements in I^1	No. of elements in EF(P′)	No. of elements in EF(P)	No. of pts in F	Time by definition
$A_{4\times 6}$	1.	1.191	1.233	2.424	5	1	2	2	54	1.029
	2.	1.090	1.343	2.433	4	0	4	3	16	0.500
	3.	1.713	1.799	3.512	8	0	11	11	64	3.911
$C_{2\times 6}$	4.	1.232	2.224	3.456	5	0	2	1	55	2.333
	5.	1.296	1.834	3.130	6	1	1	1	9	0.240
	6.	1.129	1.510	2.639	3	0	8	8	58	4.308
$A_{4\times 6}$	7.	1.132	1.160	2.292	2	0	16	16	63	6.228
	8.	1.253	0.915	2.168	3	1	14	12	54	3.595
	9.	1.307	1.510	2.817	4	0	11	10	57	4.167
$C_{3\times 6}$	10.	1.424	1.272	2.696	4	0	20	20	64	6.827
	11.	1.490	1.799	3.289	5	0	9	9	56	6.072
$A_{4\times 6}$	12.	1.223	2.243	3.466	5	0	2	2	42	3.027
	13.	1.572	2.161	3.733	6	0	7	7	47	4.297
$C_{4\times 6}$	14.	2.305	1.099	3.404	5	0	36	36	64	9.590
	15.	1.625	2.319	3.945	6	0	7	9	58	6.067

Table 2

		Time to obtain EF(P′) in (sec)	Time to obtain EF(P) in (sec)	Total time (sec)	No. of elements in $\bar{V}$	No. of elements in I^1	No. of elements in EF(P′)	No. of elements in EF(P)	No. of pts in F	Time by definition
	16.	34.497	25.242	59.739	9	2	8	8	492	96.956
$A_{4\times9}$	17.	35.051	29.243	64.294	7	0	4	5	367	65.498
	18.	33.335	33.406	66.740	7	0	6	6	512	175.963
$C_{2\times9}$	19.	33.932	35.174	69.106	8	0	3	2	435	104.874
	20.	34.064	22.673	56.737	7	0	6	3	245	41.685
	21.	36.588	28.449	65.036	9	1	15	13	340	75.316
$A_{4\times9}$	22.	90.500	30.013	120.512	19	7	46	41	428	147.007
	23.	45.332	22.300	67.361	10	0	25	19	279	105.150
$C_{3\times9}$	24.	33.565	23.991	57.556	6	0	29	29	512	228.494
	25.	32.377	18.615	50.992	7	0	4	9	20	2.245
	26.	53.103	26.052	79.154	11	1	24	12	336	90.071
$A_{4\times9}$	27.	35.450	33.698	69.148	7	0	6	16	356	169.708
	28.	44.067	29.649	73.716	7	0	29	29	512	321.374
$C_{4\times9}$	29.	35.072	35.850	70.921	7	0	7	7	508	194.041
	30.	36.419	28.971	65.389	6	0	12	15	399	164.898

the cpu times corresponding to the obtainment of EF(P), for each of the 30 problems, by the most trivial algorithm which consists in applying directly the definition of efficient point in (P) given in the introduction.

From Tables 1 and 2 we observe that:

(1) The number of elements of V necessary to generate EF(P') is surprisingly small indicating that Lemma 2.10 is powerful.

(2) There is no conclusive evidence that the partition of the set $\{1, 2, \ldots, r\}$ in I^1 and I^2, in the algorithm, improves its efficiency.

(3) The algorithm is more efficient than the definition in obtaining EF(P) when the number of elements in F is large when compared with 2^n (the number of elements in DUC). In particular, since the first column of Tables 1 and 2 correspond to steps 1, 2 and 3 of the algorithm, we see that when $F = \text{DUC}$ (problems 3, 10, 14, 18, 24 and 28) the time required by the algorithm to obtain $\text{EF(P)} = \text{EF(P')}$ is significantly smaller than the definition time.

(4) For a fixed number of variables the cpu time for the algorithm does not vary strongly when the number of rows in C are between 2 and 4.

	15 First problems		15 Last problems	
	Algorithm	Definition	Algorithm	Definition
Mean time	3.0269	4.1460	69.1114	132.2186
Standard deviation	0.5652	2.5603	15.8827	80.5142

(5) The fact that the numbers of elements in EF(P') and EF(P) are close, indicates that the constraints cut, most of the time, nonefficient points in DUC.

Indifference sets

It was mentioned in Section 1 that not every element of EF(P) maximizes a functional of the type λCx with $\lambda > 0$ on F. However for any $\lambda > 0$, every solution to

$$(\text{P}\lambda) \qquad \max\{\lambda Cx\colon x \in F\}$$

is efficient in (P). The linear functional λCx corresponds to the assignment of weights to the p criteria. As these weights are usually subjective it is useful to investigate the set of weights for which a given element of EF(P) solves (Pλ). Let

$$S = \{\lambda \in R^p : \sum_{i=1}^{p} \lambda_i = 1, \lambda_i \geqq 0, i = 1, 2, \ldots, p\}, \quad x^u \in \text{EF(P)},$$

$$S(x^u) = \{\lambda \in S\colon x^u \text{ solves } (\text{P}\lambda)\},$$

$$\overline{\text{EF}}(\text{P}) = \{x \in \text{EF(P)}\colon x \text{ solves } (\text{P}\lambda) \text{ for some } \lambda \in S\}.$$

To provide a characterization of $S(x^u)$, the indifference set for x^u, we need the following result:

Lemma 3.5. *Let* $\lambda \in S$ *and* $F(\lambda) = \{y \in F : \lambda Cy = \max\{\lambda Cx\colon x \in F\}\}$. *Then* $F(\lambda) \cap \overline{EF}(P) \neq \emptyset$.

Proof. If $\lambda > 0$ the result is trivial. Assume $\lambda_1 > 0, \dots, \lambda_d > 0$, $\lambda_{d+1} = \cdots = \lambda_p = 0$, $d < p$ and x^0 solves $\max\{\Sigma_{i=d+1}^p c_i x \colon x \in F(\lambda)\}$, where c_i is the ith row of C. We show that $x^0 \in \mathrm{EF(P)}$. Suppose $x^0 \notin \mathrm{EF(P)}$, then there is $\bar{x} \in F$ such that

$$C\bar{x} \geq Cx^0 \tag{3.2}$$

If $\bar{x} \notin F(\lambda)$ we have $\lambda C\bar{x} < \lambda Cx^0$ and thus $c_j\bar{x} < c_j x^0$ for some $j \in \{1, 2, \dots, d\}$ contradicting (3.2). If $\bar{x} \in F(\lambda)$, $\lambda C\bar{x} = \lambda Cx^0$ and $\Sigma_{i=d+1}^p c_i\bar{x} \leqq \Sigma_{i=d+1}^p c_i x^0$ implying $\lambda C\bar{x} + \Sigma_{i=d+1}^p c_i\bar{x} \leqq \lambda Cx^0 + \Sigma_{i=d+1}^p c_i x^0$ which contradicts (3.2). Hence $x^0 \in \mathrm{EF(P)}$. Moreover $x^0 \in \overline{\mathrm{EF}}(\mathrm{P})$.

Theorem 3.6. *$S(x^u)$ is the set of $\lambda \in S$ satisfying*

$$\lambda C(x^u - x^i) \geqq 0, \quad \textit{for all } x^i \in \overline{\mathrm{EF}}(\mathrm{P}), \quad \textit{with } \lambda \in S \tag{3.3}$$

Proof. If $\lambda^0 \in S(x^u)$ then $\lambda^0 C(x^u - x) \geqq 0$ for all $x \in F$ and λ^0 satisfies (3.3). Assume λ^0 is a solution to (3.3). Let $x \in F$ and $x \notin \overline{\mathrm{EF}}(\mathrm{P})$. By Lemma 3.5 there is $\bar{x} \in \overline{\mathrm{EF}}(\mathrm{P})$ that solves $(\mathrm{P}\lambda^0)$. Thus, $\lambda^0 C(\bar{x} - x) \geqq 0$. But, $\lambda^0 C(x^u - x) = \lambda^0 C(x^u - \bar{x}) + \lambda^0 C(\bar{x} - x) \geqq 0$.

When EF(P) is known and we are not able to distinguish the subset $\overline{\mathrm{EF}}(\mathrm{P})$, $S(x^u)$ can still be characterized by the system

$$\lambda C(x^u - x^i) \geqq 0 \quad \text{for all } x^i \in \mathrm{EF(P)}, \quad \text{with } \lambda \in S \tag{3.4}$$

It is straightforward to prove the equivalence of (3.3) and (3.4). However, if EF(P) is not known it is possible to approximate $S(x^u)$. The n extreme points adjacent to x^u in DUC can be written as $x^u + r^j$, $j \in J$ where $r_i^j = 0$, $i \neq j$ and $r_j^j = 1(-1)$ if $x_j^u = 0(1)$.

For any $x \in \mathrm{DUC}$, $x - x^u$ is a non-negative combination of the vectors r^j, $j \in J$, i.e.,

$$x = x^u + \sum_{j \in J} \gamma_j r^j \quad \text{with } \gamma_j \geqq 0 \tag{3.5}$$

Consider the system

$$\lambda Cr^j \leqq 0, \quad j \in J \quad \text{with } \lambda \in S \tag{3.6}$$

Lemma 3.7. *Let $\bar{S}(x^u)$ be the subset of S satisfying* (3.6). *Then $\bar{S}(x^u) \subseteq S(x^u)$, i.e., $\bar{S}(x^u)$ is an approximation to $S(x^u)$.*

Proof. Suppose $\lambda^0 \in \bar{S}(x^u)$. Since $F \subseteq \mathrm{DUC}$, (3.5) holds for any $x \in F$. Thus, $\lambda^0 C(x^u - x) = -\lambda^0 \Sigma_{j \in J} \gamma_j Cr^j \geqq 0$. Hence $\lambda^0 \in S(x^u)$.

This lemma is illustrated by the following example.

Example 3.8. Let (P) be

$$\max\left\{\begin{bmatrix} 2 & 0 & 1 \\ -2 & 0 & -1 \\ 1 & 1 & 1 \end{bmatrix} x:\quad -2x_1 - 2x_3 \leqq -1,\ x_j = 0, 1,\ j = 1, 2, 3\right\}.$$

It can be shown that $V = \{(0,1,0)\}$, $\mathrm{EF(P)} = \{x^1 = (0,1,1);\ x^2 = (1,1,1);\ x^3 = (1,1,0)\}$ and $\overline{\mathrm{EF}}\mathrm{(P)} = \{(1,1,1), (0,1,1)\}$. From (3.4) and (3.6) we have that $S(x^1)$ is the solution set of the system:

$$\begin{aligned} -2\lambda_1 + 2\lambda_2 - \lambda_3 &\geqq 0, \\ \lambda_1 + \lambda_2 + \lambda_3 &= 1, \\ \lambda_1,\quad \lambda_2,\quad \lambda_3 &\geqq 0. \end{aligned}$$

$\bar{S}(x^1)$ is the solution set of the system

$$\begin{aligned} +2\lambda_1 - 2\lambda_2 + \lambda_3 &\leqq 0, \\ -\lambda_1 + \lambda_2 - \lambda_3 &\leqq 0, \\ \lambda_1 + \lambda_2 + \lambda_3 &= 1, \\ \lambda_1,\quad \lambda_2,\quad \lambda_3 &\geqq 0. \end{aligned}$$

$S(x^1)$ and $\bar{S}(x^1)$ are represented in Figure 1 by areas ABD and BCD respectively.

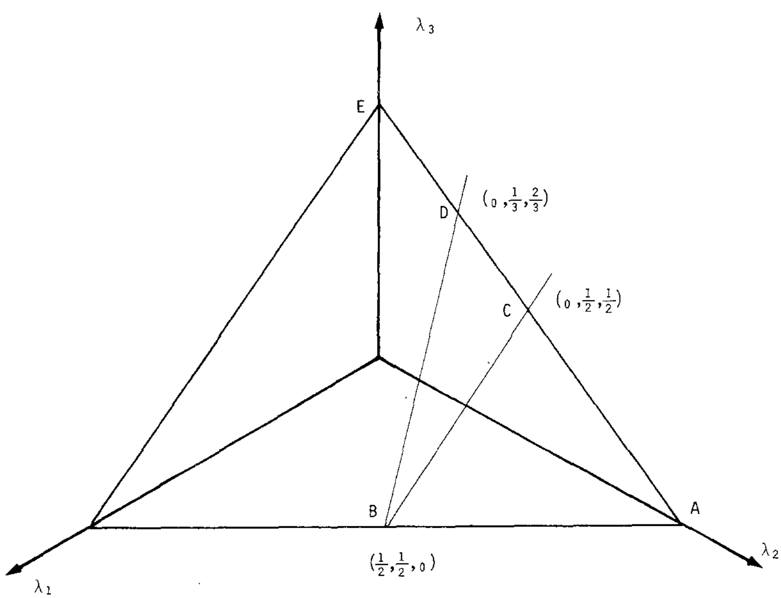

Fig. 1.

Lemma 3.9. *If* $x^u \notin \mathrm{EF(P'')}$ *then* $\{\lambda > 0 : \lambda \in \bar{S}(x^u)\} = \emptyset$.

Proof. $x^u \notin \mathrm{EF(P'')}$ implies that there is $x \in \mathrm{UC}$, $x = x^u + \Sigma_{j\in J}\, \alpha_j r^j$, $\alpha =$

$(\alpha_1, \ldots, \alpha_n) \geq 0$ and $C(x - x^u) = \sum_{j\in J} \alpha_j Cr^j \geq 0$. But for any $\lambda > 0$, $\sum_{j\in J} \alpha_j \lambda Cr^j > 0$. By (3.6) it follows that $\lambda \notin \bar{S}(x^u)$.

Parametric analysis

We conclude this section with some considerations on parametric analysis. A change in A or b may affect the condition for a point to be efficient or not. The analysis varies according to the information on V and whether or not the point is efficient in (P′). The variations in A and b are represented respectively by the $m \times n$ matrix ΔA and the $m \times 1$ vector Δb. The perturbed problem (P) is denoted by $(P\theta_1\theta_2)$: $\max\{Cx: x \in F(\theta_1, \theta_2)\}$ where

$$F(\theta_1, \theta_2) = \{x \in \mathbf{R}^n : (A + \theta_1 \Delta A)x \leqq b + \theta_2 \Delta b \text{ and } x_j = 0, 1, j \in J\}, \quad \theta_1, \theta_2 \in R.$$

Lemma 3.10. *Suppose $x^0 \in \mathrm{EF}(\mathrm{P}')$ and that A and b are perturbed by $\theta_1 \Delta A$ and $\theta_2 \Delta b$ respectively. Then, $x^0 \in \mathrm{EF}(\mathrm{P}\theta_1\theta_2)$ as long as $x^0 \in F(\theta_1, \theta_2)$.*

Proof. $x^0 \in \mathrm{EF}(\mathrm{P}')$ implies that for all $i \in K$, $x^0 + v^i \notin \mathrm{DUC}$. Since $F(\theta_1, \theta_2) \subseteq \mathrm{DUC}$ we also have $x^0 + v^i \notin F(\theta_1, \theta_2)$, $i \in K$. Therefore $x^0 \in \mathrm{EF}(\mathrm{P}\theta_1\theta_2)$ as long as $x^0 \in F(\theta_1, \theta_2)$.

Lemma 3.11. *Suppose $x^0 \notin \mathrm{EF}(\mathrm{P}')$ and that A and b are perturbed by $\theta_1 \Delta A$ and $\theta_2 \Delta b$ respectively. Moreover assume V is known. Then, $x^0 \in \mathrm{EF}(\mathrm{P}\theta_1\theta_2)$ as long as $x^0 \in F(\theta_1, \theta_2)$ and $(A + \theta_1 \Delta A)(x^0 + v^i) \not\leqq b + \theta_2 \Delta b$ for all $i \in I(x^0) = \{j \in K : x^0 \in M(v^j)\}$.*

Proof. Note that according to Lemma 2.1(b) $I(x^0) \neq \emptyset$. By Lemma 2.1(c) we know that $x^0 + v^i \notin F(\theta_1, \theta_2) \subseteq \mathrm{DUC}$ for $i \notin I(x^0)$. Thus $x^0 \in \mathrm{EF}(\mathrm{P}\theta_1\theta_2)$ as long as $x^0 \in F(\theta_1, \theta_2)$ and $x^0 + v^i \notin F(\theta_1, \theta_2)$, $i \in I(x^0)$.

4. The mixed linear multiple objective problem with zero-one variables

We denote the mixed zero-one version of (P) by (MP): $\max\{[C_1x + C_2y]: (x, y) \in \mathrm{MF}\}$ where $\mathrm{MF} = \{(x, y) \in \mathbf{R}^{n+n'} : A_1x + A_2y \leqq b, x_j = 0, 1, j \in J, y \geqq 0\}$, A_1, A_2, C_1, C_2 are $m \times n$, $m \times n'$, $p \times n$ and $p \times n'$ matrices respectively and b is an $m \times 1$ vector.

A pair (x^0, y^0) is said to be efficient in (MP) if there is no $(x, y) \in \mathrm{MF}$ such that $C_1(x - x^0) + C_2(y - y^0) \geq 0$. The set of efficient points in (MP) is denoted by EF(MP).

(MP) is considerably more difficult to solve than (P). Besides presenting the same inconvenience as (P) (see example 4.1), its efficient set cannot be always obtained as union of convex combinations of subsets of extreme efficient points in (MP) as shown in Example 4.2.

Example 4.1. Consider (MP) with

$$C_1 = \begin{bmatrix} 1 & 2 \\ 2 & 4 \\ 2 & -1 \\ -2 & 1 \end{bmatrix}, \qquad C_2 = \begin{bmatrix} 2 \\ -5 \\ 0 \\ 0 \end{bmatrix},$$

$\mathrm{MF} = \{-x_2 + y \leqq 0;\ x_1 = 0, 1;\ x_2 = 0, 1;\ y \geqq 0\}$.

It is not difficult to verify that $(x_1, x_2, y) = (0, 0, 0) \in \mathrm{EF(MP)}$ and does not maximize $\lambda C_1 x + \lambda C_2 y$ over MF for any $\lambda \geq 0$ such that $(\lambda C_1, \lambda C_2) \neq 0$.

Example 4.2. Consider (MP) with

$$C_1 = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}, \qquad C_2 = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix},$$

$\mathrm{MF} = \{x_1 = 0, 1;\ y \leqq 1;\ y \geqq 0\}$.
MF consists of the segments $[(0, 0), (0, 1)]$ and $[(1, 0), (1, 1)]$. All points of MF are efficient in (MP) with the exception of $(0, 0)$. Note that $(C_1, C_2)(1, 1)_{\mathrm{T}} \geq (C_1, C_2)(0, 0)_{\mathrm{T}}$.

Lemma 4.3. *A sufficient condition condition for* EF(MP) *to be empty is that the system* $C_2 r \geq 0$, $A_2 r \leqq 0$, $r \geqq 0$ *has solution.*

Proof. Assume $(x, y) \in \mathrm{MF}$ and the conditions of the lemma hold. Then, $A_1 x + A_2(y + r) = A_1 x + A_2 y + A_2 r \leqq A_1 x + A_2 y \leqq b$ and $y + r \geqq 0$, i.e., $(x, y + r) \in \mathrm{MF}$. Since $C_1 x + C_2(y + r) \geq C_1 x + C_2 y$ we conclude that $(x, y) \notin \mathrm{EF(MP)}$.

When $\mathrm{EF(MP)} \neq \emptyset$ efficient points in (MP) can be obtained by solving

$$(\mathrm{MP}\lambda) \qquad \max\{\lambda C_1 x + \lambda C_2 y : (x, y) \in \mathrm{MF}\} \quad \text{for any } \lambda > 0.$$

In fact every solution to (MPλ) with $\lambda > 0$ is efficient in (MP). In what follows we explore indifference sets and possible approximations. Let $(x^e, y^e) \in \mathrm{EF(MP)}$ and $S(x^e, y^e) = \{\lambda \in S : (x^e, y^e) \text{ solves } (\mathrm{MP}\lambda)\}$ denote the indifference set for (x^e, y^e). In characterizing $S(x^e, y^e)$ we refer to Benders' method [4, pp. 134–138].

At iteration k, the set of known extreme points and extreme rays of (4.1), below, are denoted by $T(k)$ and $Q(k)$ respectively. Assume that $\lambda^0 \in S(x^e, y^e)$. Thus

$$\{u \colon uA_2 \geqq \lambda^0 C_2,\ u \geqq 0\} \neq \emptyset \tag{4.1}$$

Steps 1, 2 and 3 of the method are as follows (for a detailed description see [4, pp. 138].

Step 1. Initialization.

Step 2. Solve the integer programming problem

$$\begin{aligned}\max\quad & u_0, & & \\ & u_0 \leqq \lambda^0 C_1 x + u^t(b - A_1 x) & & \text{every } u^t \in \mathrm{T}(k), \\ & 0 \leqq s^q(b - A_1 x) & & \text{every } s^q \in Q(k), \\ & x_j = 0, 1, & & j \in J.\end{aligned} \tag{4.2}$$

Call the optimal solution to (4.2) (u_0^k, x^k). Go to Step 3.

Step 3. Solve the LP

$$\begin{aligned}u_0^*(x^k) = \lambda^0 C_1 x^k + \min\quad & u(b - A_1 x^k), \\ \text{s.t.}\quad & uA_2 \geqq \lambda^0 C_2, \\ & u \geqq 0.\end{aligned} \tag{4.3}$$

The algorithm terminates if $u_0^*(x^k) = u_0^k$. Otherwise update $T(k)$ and $Q(k)$ and go to step 2.

Assume that (MPλ^0) was solved by Benders' method in k^0 iterations. Clearly $S(x^e, y^e)$ is the set of $\lambda \in S$ such that

$$\lambda C_1 x^e + \lambda C_2 y^e \geqq \lambda C_1 x + \lambda C_2 y \quad \text{for all } (x, y) \in \mathrm{MF}. \tag{4.4}$$

Unfortunately (4.4) is not operative. However, it is possible to obtain a local approximation to $S(x^e, y^e)$ around the point λ^0.

When solving (MPλ^0) by Benders' method each of the u^t in $T(k^0)$ solves (4.3) at some iteration and therefore has a corresponding optimal basis denoted by (B_2^t). Recall that the $u^t \in T(k^0)$ are extreme points of (4.1). By linear programming parametric analysis we have that each u^t in $T(k^0)$ will remain optimal in its corresponding problem (4.3) as long as $\lambda C_2(B_2^t)^{-1} \geqq 0$. Let u^e be the element of $T(k^0)$ which solves (4.3) with $x^k = x^e$ and consider the system

$$\lambda \in S, \tag{4.5}$$

$$\lambda C_2(B_2^t)^{-1} \geqq 0 \quad \text{for all t such that } u^t \in T(k^0), \tag{4.6}$$

$$\begin{aligned}\lambda C_1 x^e + \lambda C_2 y^e \geqq \max\ & u^0, \\ \text{s.t.}\quad & u^0 \leqq \lambda C_1 x + u^t(b - A_1 x) \quad \text{for all t such that } u^t \in T(k^0), \\ & 0 \leqq s^q(b - A_1 x) \quad \text{for all } q \text{ such that } s^q \in Q(k^0), \\ & x_j = 0, 1, \quad j \in J.\end{aligned} \tag{4.7}$$

Theorem 4.4. *The set of* λ*'s which satisfy* (4.5) *to* (4.7) *is contained in* $S(x^e, y^e)$. *We denote this set by* $S(\lambda^0, x^e, y^e)$.

Proof. Assume $\bar{\lambda}$ satisfies (4.5) to (4.7). Then, every $u^t \in T(k^0)$ is feasible in

$$\min\{u(b - A_1 x^e)\colon uA_2 \geqq \bar{\lambda} C_2,\ u \geqq 0\} \tag{4.8}$$

and in particular u^e is an optimal solution. By linear programming duality it follows that

$$u^t(b - A_1 x^e) \geqq \bar{\lambda} C_2 y^e u^t \in T(k^0) \quad \text{and} \quad u^e(b - A_1 x^e) = \bar{\lambda} C_2 y^e$$

(y^e is the optimal solution to the dual to (4.8)). Therefore

$$\bar{\lambda}C_1x^e + u^t(b - A_1x^e) \geqq \bar{\lambda}C_1x^e + \bar{\lambda}C_2y^e = \bar{\lambda}C_1x^e + u^e(b - A_1x^e),\ u^t \in T(k^0). \tag{4.9}$$

But, (4.7), (4.9) and the termination criterion of Benders' method imply that (x^e, y^e) solves (MP$\bar{\lambda}$) and thus, $\bar{\lambda} \in S(x^e, y^e)$.

Clearly $S(\lambda^0, x^e, y^e)$ is an approximation to $S(x^e, y^e)$ around λ^0. The reader should note that the matrices $(B_2^t)^{-1}$ are generated at Step 3 during the solution of (MPλ^0).

5. Conclusions

The main difficulty in obtaining EF(P) is that some, or all, of its points may not maximize a functional of the type λCx, with $\lambda \geq 0$, over $\bar{F} = \{x \in \mathbf{R}^n : Ax \leqq b$ and $x \in \mathrm{UC}\}$. In fact they may not maximize the functional even over F. Otherwise we could use any algorithm for continuous linear multiple objective programs. This inconvenience points out the necessity to develop special methods for the zero-one case. In this paper we presented results based essentially on the set V. We showed that this set contains enough information to generate EF(P′) and to identify the potential elements of DUC to be efficient in (P).

Although the number of possible candidates to V is of the order of 3^n, Lemma 2.10 provides a rule that reduces significantly the number of vectors to be enumerated. The concept of regular efficient point in (P′) was introduced. It is intimately related to the cone CV.

The reader should note that most propositions of this paper are valid even if the set F is not defined by a system of linear inequalities as long as the zero-one conditions are included.

The results presented are far from being complete and much research and practical experimentation remains to be done. It is important to investigate sufficient conditions for any $x \in$ EF(P) to be efficient in (Q): $\max\{Cx\colon x \in \bar{F}\}$ or in ($\bar{\text{P}}$): $\max\{\bar{C}x\colon x \in \bar{F}\}$ when $\bar{C}$ can be easily determined. Parametric analysis in (P) as well as its mixed integer version are still to be fully explored. Finally it is the author's belief that new algorithms, possibly more efficient than the one presented, can be developed from the theory presented. These algorithms will be certainly needed since for $n > 10$ the cardinality of some $M(v)$ can be very large.

Acknowledgment

The author is grateful to Professor Jeremy F. Shapiro, from MIT, for bringing to his attention the problem treated here and to the referee Professor J. Philip for his suggestions to improve the readability of the paper.

References

[1] D.E. Bell and J.F. Shapiro, "A finitely convergent duality theory for zero-one integer programming", MIT Operations Research Center Report OR 043-75 (May 1975).

[2] G.R. Bitran, "Admissible points and vector optimization: a unified approach", unpublished Ph.D. thesis, Operations Research Center, MIT (Cambridge, MA, 1975).

[3] J.P. Evans and R.E. Steuer, "A revised simplex method for linear multiple objective programs", *Mathematical Programming* 5 (1973) 54–72.

[4] R.S. Garfinkel and G.L. Nemhauser, *Integer programming* (Wiley, New York, 1972).

[5] A.M. Geoffrion, "Proper efficiency and the theory of vector maximization", *Journal of Mathematical Analysis and Applications* 22 (1968) 618–630.

[6] J.S.H. Kornbluth, "Duality, indifference and sensitivity analysis in multiple objective programming", *Operations Research Quarterly* 25 (1974) 599–614.

[7] J. Philip, "Algorithms for the vector maximization problem", *Mathematical Programming* 2 (1972) 207–229.

[8] J.F. Shapiro, "Multiple criteria public investment decision making by mixed integer programming", in: *Proceedings of the conference on multicriterion decision making* Jouy En Josas, May 1975, to appear.

[9] P.L. Yu and M. Zeleny, "The set of all nondominated solutions in linear cases and a multicriteria simplex method", *Journal of Mathematical Analysis and Applications* 49 (1975) 430–468.