Journal of Applied Sciences Research, 3(8): 662-667, 2007

Using Feasible Direction to Find All Efficient Extreme Points for Multiple Objective Linear Programs (MOLP)

S.F. Tantawy and R.H. Sallam

Mathematics Department Faculty of Science, Helwan University (11795) Cairo, Egypt.

Abstract: Most of the current methods for solving multiple objective linear programs (MOLP) depend on the simplex tableau in multi objective form to find the set of efficient solutions of this problem. In this paper we present a feasible direction method to find all efficient extreme points for (MOLP). Our method based on the conjugate gradient projection method starting with an initial efficient extreme point we generate a sequence of feasible directions towards all efficient adjacent extremes of the above problem. A simple example is given to clarify this method.

Key words: Multiple objective linear program- efficient point-conjugate projection.

1-INTRODUCTION

The multiple objectives linear programming (MOLP) problems arises when several linear objective functions has to be maximized (or minimized) on a convex polytope. Different approaches have been suggested for solving this problem, among which are the ones suggested by Evans and Steven[3], Tamula and Mura[11]. Gal[4], lsermunm[9], Ecker and Kauda[1] and Ecker, Hegren and Kauda[2]. Most of these methods depend on the canonical simplex tableau in multiple objective forms to find the efficient set of (MOLP). In this paper we suggest a new method to find all efficient extreme points based on using a feasible direction of movement from an efficient extreme point to its adjacent one. In section 2 some notation and theory for (MOLP) are presented, our main result is given in section 3 with an illustrative example and finally a conclusion about the problem under consideration is made in section 4.

2-Notations and theory: Multiple objective linear programming (MOLP) problems arises when several linear objective functions has to be maximized (or minimized) on a convex polytope.

$X = \{x \in R^n / Ax \le b\}$

Where A is an (m+n) x n matrix and b Rm+n

We point out that the non negativity condition is added to the set of constraints. If C is a k x n matrix, then (MOLP) can be formulated as:

$$\begin{aligned} &\text{Maximize} && y = C\,x \\ &\text{Subject to:} && x \in X \end{aligned} \qquad (2.1)$$

Solving (MOLP) problem is to find the set of efficient solution E where

$$E = \{x \in X \text{ there is no } \bar{x} \in X \text{ such that } Cx \le C\bar{x}\} \qquad (2.2)$$

Consider the set $Y = \{y \in R^k \mid y = Cx,\ x \in X\}$ (2.3)

Then we say that y is non dominated point if $y \in Y$ and there is no $\bar{y}$ such that $y \le \bar{y}$. It is well known theorem for (MOLP)[4,13] that a feasible point x is efficient for multiple objective linear program if and only if there is a k vector $\lambda > 0$ (weights) such that x is an optimal solution for the linear program

$$\begin{aligned} &\text{Maximize } \lambda^T Cx \\ &\text{Subject to; } \in x\ x \end{aligned} \qquad (2.4)$$

The idea of using weights seems to be an attractive method to solve this kind of problems. It involves averaging the objectives into a composite objective and then maximizing the result but the difficulty of this method is in specifying these weights for solving the above linear program (2.4)

Consider the dual linear program of (2.4) in the form

$$\begin{aligned} &\text{Minimize} && w = u^T b \\ &\text{Subject to} && u^T A = \lambda^T C \\ & && u \ge 0 \end{aligned} \qquad (2.5)$$

On multiply the set of constraints of this dual problem by a matrix $T = (T_1 \mid T_2)$, where $T_1 = (C\,C^T)^{-1} C$ and the column of the matrix T_2
Constitute the bases of $N(C) = \{x;\ Cx = 0\}$.

We have $u^T A T_1 = \lambda^T$, $u^T A T_2 = 0$ and $u \ge 0$. (2.6)

If an $L \times (m+n)$ matrix P of non-negative entries is defined such that $PAT_2 = 0$, then a sub matrix $\bar{P}$ of the given matrix P satisfying

Corresponding Author: S.F. Tantawy, Mathematics Department Faculty of Science, Helwan University (11795) Cairo, Egypt.

$$\bar{P}\ AT_1 = \lambda^T > 0 \qquad (2.7)$$

will be important for specifying the positive weights needed for solving the linear programming (2.4), also our proposed method for solving (MOLP) will depend mainly on these previously specified weights given and can be summarized in two phases as follows:

Phase 1 is applied first to find an initial efficient extreme point x by solving the linear program.

Maximize $F(x) = e^T Cx$
Subject to: $Ax \leq b$

Where $e \in R^k$ with all entries equals 1. If $e^T C = c^T$, the above linear program is written as

Maximize $F(x) = c^T x$
Subject to

$$x \in X = \{x,\ Ax \leq b\} \qquad (2.8)$$

This problem can also be written in the form:

Maximize $F(x) = c^T x$
Subject to

$$a_i^T\ x \leq bi,\ i = 1, 2..., m + n. \qquad (2.9)$$

Here a_i^T represents the i th row of the given matrix A, then we have in the non degenerate case an extreme point (vertex) of X lies on some n linearly independent subset of X. We shall give an iterative method for solving this problem and our task is to find the optimal extreme point for this program (first efficient extreme point), this method starts with an initial feasible point then a sequence of feasible directions toward optimality is generated to find this optimal extreme of this programming, in general if x^{k-1} is a feasible point obtained at iteration k-1 (k = 1, 2 ...) then at iteration k our procedure finds a new feasible point xk given by

$$x^k = x^{k-1} + \alpha_{k-1}\ d^{k-1} \qquad (2.10)$$

Where d^{k-1} is the direction vector along which we move and given by

$$d^{k-1} = H_{k-1} c \qquad (2.11)$$

Here H_{k-1} is an n x n symmetric matrix given by

$$H_{k-1} = \begin{cases} I, & \text{for } k=1 \\ H_{k-1}^q & \text{If } k>1 \end{cases} \qquad (2.12)$$

In (2.12) we have I is an nxn identity matrix and q is the number of active constraints at the current point while H_{k-1}^q is defined as follows, for each active constraint s; s = 1, 2,..., q.

$$H_{k-1}^q = H_{k-1}^{s-1} - \frac{H_{k-1}^{s-1} a_s a_s^T H_{k-1}^{s-1}}{a_s^T H_{k-1}^{s-1} a_s} \qquad (2.13)$$

With H_{k-1}^q = I. Then H_{k-1} is given by H_{k-1} = H. The step length α_{k-1} is given by

$$\alpha_{k-1} = \min_{i=1,\dots,m+n} \{g_i / g_i = \frac{b_i - a_i^T x^{k-1}}{a_i^T d^{k-1}}, \text{and } g_i > 0\} \qquad (2.14)$$

This relation states that α_{k-1} is always positive. Proposition 2-2 below shows that such a positive value must exist if a feasible point exists. We have due to the well known Kuhn-Tuker condition[5,8] for the point x^k to be an optimal solution of the linear program (2.8) their must exist $u \geq 0$ such that

$$A_r^T u = c, \text{ or simply } u = (A_r A_r^T)^{-1} A_r c \qquad (2.15)$$

Here A_r is a sub matrix of the given matrix A containing only the coefficients of the set of active constraints at the current point x^k. This fact will act as a stopping rule of our proposed algorithm, also we have to point out that the matrix $H_{k-1} = H_{k-1}^2$ through the following proposition.

Proposition 2-1: For H_{k-1} defined by relation (2.5) above we have $(H_{k-1})^2 = H_{k-1}$

Proof: This can be proved by induction, define a matrix

$Q_1 = \frac{a_1 a_1^T}{a_1^T a_1}$ and since $H_{k-1}^1 = (I - \frac{a_1 a_1^T}{a_1^T a_1})$

then

$H_{k-1}^1\ Q_1 = 0$, $Q_1^2 = Q_1$, $(H_{k-1})^2 = H_{k-1}^1$ and is an orthongal projective matrix. Also, if we define

$Q_2 = \frac{a_2 a_2^T H_{k-1}^1}{a_2^T H_{k-1}^1 a_2}$ and $H_{k-1}^* = (I - \frac{a_2 a_2^T H_{k-1}^1}{a_2^T H_{k-1}^1 a_2})$ then since

$H_{k-1}^2 = H_{k-1}^1 (I - \frac{a_2 a_2^T H_{k-1}^1}{a_2^T H_{k-1}^1 a_2})$, we have $H_{k-1}^* Q_2 = 0$

$Q_2^2 = Q_2$ and $(H_{k-1}^1)^2 = H_{k-1}^1$. Now, since $H_{k-1}^2 = H_{k-1}^1 H_{k-1}^*$

and both matrices H_{k-1}^1 and H_{k-1}^* are orthogonal projective, then H_{k-1}^2 is orthogonal projective matrix and we have $(H_{k-1}^2)^2 = H_{k-1}^2$

Applying the same argument, we conclude that $H_{k-1} = H^{+}_{k-1}$ is an orthogonal projective matrix such that $(H_{k-1})^2 = H_{k-1}$.

Proposition 2-2: Any solution xk given by equation (2.10) is feasible and increases the objective function value.

Proof:

$$\begin{aligned} F(x^k) - F(x^{k-1}) &= c^T x^k - c^T x^{k-1} \\ &= c^T \alpha_{k-1} d^{k-1} \\ &= \alpha_{k-1} c^T H_{k-1} c \\ &= \alpha_{k-1} c^T H^2_{k-1} c \\ &= \alpha_{k-1} \|H_{k-1} c\| > 0 \end{aligned}$$

This proves that x^k increases the objective function. Next, we shall prove that x^k is a feasible point. For x^k to be a feasible point it must satisfy all constraints of problem (2-1), then

$$a_i^T (x^{k-1} + \alpha_{k-1} d^{k-1}) \le b_i$$

Must hold for all i ∈ {1, 2... m+n} which can be written

$$a_i^T \alpha_{k-1} d^{k-1} \le b_i - a_i^T x^{k-1}, i=1,2...m+n$$

And this is valid for any i since if there is p ∈ {1, 2... m+n} such that

$$a_p^T d^{k-1} > 0 \text{ and } a_p^T d^{k-1} > b_p - a_p^T x^{k-1}, \text{ then } \frac{b_p - a_p^T x^{k-1}}{a_p^T d^{k-1}} < \alpha_{k-1}$$

That will contradict our definition of α_{k-1}. Next, we shall give a result that guarantees the existence of α_{k-1} defined by relation (2.14) above

Proposition 2-3: At any iteration k if a feasible point that will increase the objective function exists then α_{k-1} as defined by relation (2.14) must exist.

Proof: To prove this result it is enough to prove that

$$a_i^T d^{k-1} \le 0 \qquad (2.16)$$

Cannot be true for all i ∈ {1, 2... m+n}. Now suppose that relation (2.16) is true for i ∈ {1,2..., m+n} then writing (2.16) in matrix form and multiplying both sides by $u^T \ge 0$, we get

$$u^T A\, d^{k-1} \le 0$$

$$\text{i.e.,} \quad u^T A\, H_{k-1}\, c \le 0 \qquad (2.17)$$

Since the constraints of the dual problem for the linear programming problem (2-1) can be written in the form $u^T A = c^T$, $u \ge 0$, then (2-10) can be written as:

$$c^T H^2_{k-1} c \le 0, \text{since } H_{k-1} = H^2_{k-1}$$

$$\text{i.e.,} \quad \|H_{k-1} c\| \le 0$$

This contradicts the fact that the norm must be positive, which implies that relation (2.14) cannot be true for all, i ∈ {1, 2... m+n}. Thus if a feasible point x^k exists then α_{k-1} as defined by relation (2.14) must exist. Based on the above results we shall give in the next section a full description of our algorithm for solving the multiple objective linear programming problems (MOLP) to find all efficient extreme points in two phases as follows:

3- New Algorithm for Solving (MOLP) Problems:

Phase I: Find an initial efficient extreme point through the following steps.

Step 0: set k=1, Ho=I, d^0=c, let x^0 be an initial feasible point and applies relations (2.14) to compute α_0.

Step 1: Apply relation (2.10) to find a new solution x^k.

Step 2: Apply relation (2.15) to compute u, if $u \ge 0$ stop. The current solution x^k is the optimal solution otherwise go to step 3.

Step 3: Set k = k+1, apply relations (2.12), (2.11) and (2.14) to compute H_{k-1}, d^{k-1} and α_{k-1} respectively and go to step 1.

Given an initial feasible point x^0 and a vector c, step 0 computes α_0 in O(m+n) steps. Computing x^k in step 1 requires O(n) steps while testing the optimality of the current solution x^k, in step 2 requires $O(n^3)$ steps. Step 3 of the algorithm requires $O(n^3)$ steps to compute H_{k-1} while computing d^{k-1}, the feasible direction that increase the value of the objective function, requires $O(n^2)$ steps, finally to compute α_{k-1} requires O(m+n) steps. Hence the application of each iteration of our algorithm requires $O(\max\{m+n, n^3\})$ steps.

Proposition 3-1 below states that the above algorithm solves the (LP) problem to find an initial efficient extreme point in at most m+n iteration.

It has to be clear that at all iterations we have to update the projective matrix according to the number of the active constraints, if a^T_s, s = 1,2,…q, corresponding to the active constraints, then adding the $(q+1)^{th}$ constraint we have to update the projective matrix given by relation (2.5) to be in the form.

$$H^{q+1}_{k+1} = H^q_k - \frac{H^q_k a_{q+1} a^T_{q+1} H^q_k}{a^T_{q+1} H^q_k a_{q+1}}.$$

Remark 3-1: Assuming that the number of active constraints at point x^k denoted by q,if q<n and relation (2-15) is satisfied this indicates that x^k is an optimal non-extreme point (efficient non extreme satisfying (2.7)), in this case the objective function can not be improved through any feasible direction and we have $H_k c=0$ at this point x^k. We note that although the matrix H_k is singular it does not cause the breakdown of this algorithm but indicate that all subsequent search directions d^{k+1} will be orthogonal to c.

Remark 3-2: If q=n and relation (2-15) is satisfied this indicates that x^k is an efficient extreme point the columns of H_k has to be computed via a subset $\bar{q}$ of these active constraints at x^k such that $\bar{q}$ <n satisfying (2.7).

Suppose at iteration k-1, we have x^{k-1} is an extreme non optimal (i.e. q=n and relation (2-15) is not satisfied), then a move has to be made through a direction d^{k-1} lies in the nullity of a subset of the set of the active constrains at x^{k-1}. Each constraint in this subset satisfies relation (2-14). Also we have to show that if an active constraint at any iteration becomes inactive for the next iteration then it will never be active again at any subsequent iterations this can be shown as follows. Let x^{k-2}, x^{k-1}, x^k represents three successive points generated by relation (2-10) and suppose more that x^{k-2} is an extreme non optimal point, then at xk-2 we have

$$a^T_s x^{k-2} = b_s \qquad s = 1,2,\ldots,n$$
$$a^T_j x^{k-2} < b \qquad j = n+1,\ldots,m+n$$

If the t^{th} active constrain of the form

$$a^T_t x^{k-2} = b_t$$

is now dropped from the above set of the active constraints (the one with most negative ui in relation (2-15) and a feasible direction d^{k-2} is defined via the remaining subset of the active constraints, we note that this subset will remains active for the next iteration correspond to the point x^{k-1} together with one constraint from the above non active constraint satisfies relation (2-14) will also be active at x^{k-1}, the t^{th} active constraint at x^{k-2} will be inactive at the point x^{k-1} together with the remaining subset of the non active constraints at x^{k-2}, then if the t^{th} constraint will become active again for the point x^k,we have

$$a^T_t x^k = b_t, \text{ which gives}$$
$$a^T_t(\alpha_{k-2} d^{k-2} + \alpha_{k-1} d^{k-1}) = 0$$

and hence a direction $d = \alpha_{k-2} d^{k-2} + \alpha_{k-1} d^{k-1}$ with $\alpha_{k-1} \geq 0$ and $\alpha_{k-2} \geq 0$, can be defined such that $a^T_t d = 0$ and this contradicts that the feasible direction d^{k-2} is the only direction of movement defined for all s=1,…,n and t ≠ s, hence if an active constraint at a given extreme non optimal becomes inactive for the next iteration it will never be active again at any iteration of the algorithm. Next, we shall prove that the number of iterations that our algorithm requires to solve the (LP) problem is limited by m+n iterations.

Proposition 3-1: Our algorithm solves the (L P) problem in at most m+n iterations.

Proof: For this algorithm at least one constraint is added at a time starting with $H^0_0 = I$, then an optimal extreme point may be reached in n steps and the algorithm terminate in at most n iterations. On the other hand if at a given iteration we have non optimal extreme point and at least one constraint has to be dropped from the set of active constraints, this constraint can not be active again at any subsequent iterations of the algorithm. Since our allowed directions (given by 2-11) that improve the value of the objective function lies in the nullity of a subset of the given set of constraints, then we are moving in the direction parallel to a certain subset of the (m+n) constraints and hence in the worst case the maximum number of iterations required to reach the optimal point is limited by m+n.

In our analysis to find all efficient extreme points in multiple objective linear program we do this by proceeding from a given efficient point defined by phase I to its adjacent efficient extremes. By defining a frame for Cone (H) denoted by F, called a minimal spanning system. For a n x n matrix H denote the set of indices of the columns of H by IdH. Hence if $H = (h^1,\ldots.h^n)$, then $I\ d_H=\{1,2\ldots n\}$. For a matrix H we define the positive cone spanned by the columns of H (called a conical or positive hull by Stoer and Witzgall [10]) as

$$\text{Cone}(H) = \text{Cone}(h^i; i \in Id_H) = \{h \in R^n; h = \sum_{i \in Id_H} \beta_i h^i, \beta_i \geq 0\}$$

A frame F of Cone (H) is a collection of columns of H such that

Cone (h^i; $i \in Id_H$) = Cone (H) and for each $j \in Id_H$, we have Cone (h^i; $i \in Id_H \backslash \{j\}$) $\neq$ Cone (H)

Based on the above definitions, we start phase II, to find all efficient extreme points of (MOLP) problem through a finite number of steps as follows

Phase II:

Step 1: Let x^k be an efficient point, compute H_k correspond to this point x^k

Step 2: Construct a frame F of cone H_k using e.g.. the method of Wets and Witzgall [12].

Step 3: for each $h^i \in F$ determine α* obtained by solving the system of linear inequalities of the form $\alpha A h^i \leq b - A x^k$, the boundary points of this interval gives α*. Then $x^* = x^k + {}^*h^i$, is an efficient extreme point for this (MOLP) problem and go to step 1.

Example:

Maximize $z_1(x) = x_1 + 3x_2 + x_3$

$z_2(x) = 2x_1 + x2 + 3x_3$

Subject to: $2x_1 + 2x_2 + x^3 \leq 4$

$x^1 \geq 0,\ x^2 \geq 0,\ x^3 \geq 0$

Phase I: The first efficient extreme point is obtained by solving the (LP) problem Maximize $z = 3x^1 + 4x^2 + 4x^3$,

Subject to: $2x_1 + 2x^2 + x^3 \leq 4$

$x^1 \geq 0,\ x^2 \geq 0,\ x^3 \geq 0$

Step 0: k=1, $H_0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, d^0 = \begin{pmatrix} 3 \\ 4 \\ 4 \end{pmatrix}$

Let $x^0 = \begin{pmatrix} 1/2 \\ 1/2 \\ 1 \end{pmatrix}$ be an initial feasible point, then (2.14) gives $\alpha_0 = 1/18$ and we go to step 1.

Step 1: apply relation (2.10) to get

$$x^1 = \begin{pmatrix} 1/2 \\ 1/2 \\ 1 \end{pmatrix} + 1/18 \begin{pmatrix} 3 \\ 4 \\ 4 \end{pmatrix} = \begin{pmatrix} 12/18 \\ 13/18 \\ 22/18 \end{pmatrix}$$

and we go to step 2

Step 2: for this point x^1 the first constraint is the only active constraint and since relation (2.15) is not satisfied indicates that this point is not optimal, we go to step3.

Step 3: set k = 2, then

$$H_1 = \begin{pmatrix} 5/9 & -4/9 & -2/9 \\ -4/9 & 5/9 & -2/9 \\ -2/9 & -2/9 & 8/9 \end{pmatrix}, d^1 = \begin{pmatrix} -1 \\ 0 \\ 2 \end{pmatrix} \text{ and } \alpha_1 = 2/3$$

And we go to step 1, to get

$$x^2 = \begin{pmatrix} 12/18 \\ 13/18 \\ 22/18 \end{pmatrix} + 2/3 \begin{pmatrix} -1 \\ 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ 13/18 \\ 46/18 \end{pmatrix}$$

For this point the first and the second constraints are the only active constraints, since (2.15) is not satisfied we go to step 3 to get

$$H_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1/5 & -2/5 \\ 0 & -2/5 & 4/5 \end{pmatrix}, d^2 = \begin{pmatrix} 0 \\ -4/5 \\ 8/5 \end{pmatrix} \text{ and } \alpha_2 = 65/72$$

and again we go to step 1 to get

$$x^3 = \begin{pmatrix} 0 \\ 13/18 \\ 46/18 \end{pmatrix} + 65/72 \begin{pmatrix} 0 \\ -4/5 \\ 8/5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 4 \end{pmatrix}$$

For this point x^3 (2.15) is satisfied with u = [4 5 4], indicating that this point is optimal for this linear programming and we start phase II.

Phase II: For this multiple objective linear program we have:

$T_1 = \begin{pmatrix} -2/90 & 14/90 \\ 34/90 & -18/90 \\ -10/90 & 25/90 \end{pmatrix}, T_2 = \begin{pmatrix} -8/5 \\ 1/5 \\ 1 \end{pmatrix}$, nd the matrix P is in the form

$$P=\begin{pmatrix} 8/17 & 9/17 & 0 & 0 \\ 0 & 1/9 & 8/9 & 0 \\ 0 & 5/13 & 0 & 8/13 \end{pmatrix}, PAT_1=\begin{pmatrix} 5/17 & 1/17 \\ -3/9 & 1/9 \\ 1/13 & -3/13 \end{pmatrix}$$

We have λ = [5 1] is the only positive weight defined for this problem such that on solving the linear program (2.4) the set of efficient solution can be explored.

At the point x3 a subset of the set of the active constraints such that

$$[5\ 1]\begin{pmatrix} 1 & 3 & 1 \\ 2 & 1 & 3 \end{pmatrix}=[u_1\ u_2]\begin{pmatrix} 2 & 2 & 1 \\ -1 & 0 & 0 \end{pmatrix}$$

Has a solution u_1=8, u_2 = 9 is defined to compute

$$H_3=\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1/5 & -2/5 \\ 0 & -2/5 & 4/5 \end{pmatrix}$$

A frame of the columns of H_3 is used as feasible directions to find the adjacent extreme point for x^3 by solving the system of linear iniquities

$$\begin{pmatrix} 2 & 2 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}\begin{pmatrix} 0 \\ 1/5 \\ -2/5 \end{pmatrix}\alpha \le \begin{pmatrix} 0 \\ 0 \\ 0 \\ 4 \end{pmatrix}$$

Vith α^* = 10, we have an adjacment

Efficient extreme point of the form

$$x_1{}^*=\begin{pmatrix} 0 \\ 0 \\ 4 \end{pmatrix}+10\begin{pmatrix} 0 \\ 1/5 \\ -2/5 \end{pmatrix}=\begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}$$

Is explored, if we repeat the above steps for this point we get x3 is the adjacent point of the form

$$x^3=\begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}-10\begin{pmatrix} 0 \\ 1/5 \\ -2/5 \end{pmatrix}=\begin{pmatrix} 0 \\ 0 \\ 4 \end{pmatrix}$$

We conclude that the two efficient extreme points $[0\ 0\ 4]^T$ and $[0\ 2\ 0]^T$ are the only efficient extremes points for this multiple objective linear programming problem.

4-Conclusion: In this paper we suggest a new method to find all efficient extreme points of multiple objective linear programming (MOLP) problems based on using a feasible direction of movement from an efficient extreme point to its adjacent one. This method is based on modified conjugate gradient projection methods used to solve non linear programming problems with linear constraints [6, 7] to solve the multiple objective linear programming problem.

REFERENCES

1. Ecker, J.G. and I.A. Kauda, 1978. finding all efficient extreme points for multiple objective inear programming, mathematical programming, 14: 249-261.
2. Ecker, J.G., H.S. Hegren and I.A. Kauda, 1980. generating maximal efficient faces for multiple objective linear programs, journal of optimization theory and application, pp: 353-361.
3. Evans, J.P. and R.F. Steven, 1973. Generating efficient extreme points in linear multiple objective programming two algorithms and computing experience in Cochran and Zeleny, multiple criteria decision making, university of South Carolina press.
4. Gal, T., 1977. A general method for determining the set of all efficient solution to a linear vector maximum problem, European journal of operational research, 1: 307-322.
5. Greig, D.M., 1980. Optimization. Longman. London and New York.
6. Goldfarb, D., 1969. extension of Davidson's variable metric method to maximization under linear inequality and equality constraints. SIAM J. Appl. Math., 17: 739-764.
7. Goldfarb, D. and L. Lapiduo, 1968. Conjugate gradient method for non linear programming problems with linear constraints, Ind & eng. Chem. Fund., 7: 148-151.
8. Hillier, F.S. and G.J. Lieberman, 1990. introduction to operations research, 5th edition, McGraw-Hill publishing Company.
9. Isermann, H., 1977. The enumeration of the set of all efficient solution for a linear multiple objective program oper. Res. quart, 28: 711-725.
10. Stoer, J. and C. Witzgall, 1970. Convexity and Optimization in finite dimension I, Springer. Verlag, Berlin.
11. Tamura, K. and S. Mura, 1977. On linear vector maximization problems, J. oper. Res. Soc. Japan, pp: 139-149.
12. Wets, R.J.B. and Witzgall, 1967. Algorithms for frames and linearity Spaces of cones, Journal of research of the National Bureau of Standard 71B.1-7.
13. Zeleny, M., 1974. linear Multi objective programming, springer-verlag, New York.