Using the Outer Approximation Algorithm for Generating all Efficient Extreme Points of DEA

J. Gerami
Shiraz Branch, Islamic Azad University

Abstract. Identifying the efficient extreme units in a production possibility set is a very important matter in data envelopment analysis, as these observed, real units have the best performances. In this paper, we proposed a multiple objective programming model, in which the feasible region is the production possibility set under the assumption of variable returns to scale and the objective function consists of input and output variables. As we know, by increasing the dimensions of the problem, the set of efficient points would increase as well; thus, using the multiple objective linear programming problem-solving methods in a decision set would lead to computational problems and it would be much easier to work in the outcome set instead of the decision set. In this research, we show that the efficient points in the outcome set of the suggested multiple objective linear programming problems correspond with the efficient extreme points in data envelopment analysis. An outer approximation algorithm is presented for production of all efficient extreme points in the outcome set. This algorithm provides us with the equations for all efficient surfaces. In the outcome set, this algorithm would use few calculations to produce all the extreme points. Finally, we demonstrate the presented approach through numerical examples.

AMS Subject Classification: 65K99
Keywords and Phrases: Data Envelopment analysis, multiple objective, linear programming, outer approximation

1.   Introduction

Data envelopment analysis (DEA) developed by Charnes et al. [12] has become one of the most widely used methods in operations research/management science. A reason for this success is that DEA is a task-oriented approach and focuses on an important task: to evaluate the relative (technical) efficiency of comparable Decision Making Units (DMUs) essentially performing the same task. Based on information about existing data on the performance of the units and some preliminary assumptions, the purpose of DEA is to empirically characterize the so-called efficient frontier (surface) based on the set of available DMUs and to project all DMUs onto this frontier. If a DMU lies on the frontier, it is referred to as an efficient unit, otherwise inefficient. Efficiency evaluation is based on the data available without taking into account the decision-makers (DM) preferences. All efficient DMUs are considered equally good. However, if the efficient units are not equally preferred by the decision-makers it is necessary to somehow incorporate the decision-maker's judgments or a priori knowledge into the analysis. A straightforward and widely used method has been to restrict possible values of the multipliers of so-called dual DEA models. Approach is to explicitly or implicitly gather direct preference information about the desirable input and output-values of DMUs, and insert that information in a form or another into the analysis. DEA is a technique based on mathematical programming for evaluating the relative efficiency of a set of decision-making units (DMUs). The efficiency of each DMU is determined by the efficiency frontier. The units on the efficiency frontier are assumed efficient; otherwise, they are considered as inefficient. In fact, DEA sets up a production possibility set and considers its frontier as the efficient frontier made according to the non-domination condition, see, for instance DE Witte and Marques [19].

For this approach, some ideas can be adopted from research carried out in the field of Multiple Criteria Decision Making (MCDM), especially in Multiple Objective Linear Programming (MOLP). In MCDM /MOLP, one of the key issues is to provide a decision-makers with a tool, which makes it possible to evaluate points lying on the efficient frontier. It has been shown that the MOLP and DEA models have a similar structure, see, for instance Hosseinzadeh Lotfi et al. [29]. Thus, theory and approaches developed in MOLP for evaluating solutions on the efficient frontier can also be applied in DEA.

This is important that because the dimension of the outcome set is smaller than m+s and the dimension of the decision set is n+m+s-1, generating all or portion of outcome set is expected, in general, to be less the demanding computationally than generating all or portions of the decision set. The identification of DEA efficient units under various DEA models is equivalent to the identification of the lowest input and the highest output solutions within the

production possibility set for the corresponding multi-objective programming problem. The DEA-efficient DMU corresponds to the pareto efficient solution (or non-dominated solution). From this point of view, just as in the discussion of multi-objective programming, the set of all extreme points of variable returns to scale (VRS) models in DEA have significant values in the field of DEA, See, for instance Benson [7], Rockafellar [40]. In this article, we the use outer approximation algorithm for generating the set of all efficient extreme points of models DEA with VRS as proposed by Benson [8] and to do so, we use all efficient extreme points of the outcome set of the MOLP problem. The organization of this paper is as follows, in Section 2, we present literature review. In Section 3, we present MOLP problem and its relation to models DEA with VRS, and we provide the theoretical foundation of the outer approximation procedure. We summarize some relevant results concerning efficient extreme points of the MOLP problem. Section 4, provides a detailed statement of the algorithm; additionally, a small example problem is solved for illustration purpose. Section 6, provides a computational experiment and statistical analysis. Some concluding remarks are given in the last section.

2. Literature Review

In recent years, there have been a number of studies discussing the relationship between DEA and MOLP models. In their article, Doyle and Green [20] showed that DEA is an MCDM method. Alene et al. [1] used MOLP problem solving methods to apply the decision makers a priori knowledge in DEA problems. Golany [26] presented a data envelopment analysis model with a MOLP structure and used interactive MOLP methods to solve the model. Their model helped the decision maker (DM) to allocate a set of inputs, such as resources, on the efficiency frontier based on the level of outputs. Joro et al. [34] revealed that DEA problems have a similar structure to MOLP models; therefore, to solve the DEA models, we can use the corresponding reference point models in MOLP.

Wong et al. [46] proposed an equivalent model between DEA and MOLP and demonstrated how to solve a DEA problem interactively, without any prior judgment, by transforming a MOLP formula. Using interactive MOLP methods, they searched for the most preferred solutions (MPS) points on the efficiency frontier along with resource allocation and target setting according to the DMs a priori knowledge; then, they used interactive approaches such as G-D-F, Steam and Stom to solve the model and finally, engaged in a comparison of results. Yang et al. [47] attempted to demonstrate the use of interactive MOLP methods for target setting in DEA and illustrated the relationship between output-oriented DEA dual models and formulation of maxmin preferred

points in MOLP models; they used the interactive projected gradient approach
to identify the efficient units. Malekmohammadi et al. [39] focused on the topic
of target setting in DEA using MOLP problems; they extended the models pre-
sented by Yong et al [47] to simultaneously reduce the final inputs and increase
the final outputs and showed that instead of solving n models, we can set our
targets according to the DMs preferences by solving only one model.

Hosseinzadeh et al. [28] evaluated the relationship between output-oriented
dual models in DEA and MOLP models. In their study, they showed how a
DEA model can be solved interactively by transforming a MOLP formula; in
this regard, they used the Z-W approach to apply the DMs a priori knowledge
in the performance process. Ebrahimnejad et al. [21] proposed an interactive
MOLP method to identify the target units in DEA models in the presence of
undesirable outputs; they extended the relationship between BCC models and
the reference point model in MOLP toward a simultaneous and interactive in-
crease in desirable outputs and decrease in undesirable final outputs based on
MOLP models.

The main purpose of MOLP problems is to find the set of efficient solu-
tions. These solutions are Pareto optimal solutions that can simultaneously
optimize all objective functions. Among these units, the efficient extreme units
are the most important ones; these would be observed, real units and their
performance would determine the performance of other units in the system.

We may search for solutions also on the efficient frontier in DEA. Since the
outcome set has a much simpler structure and smaller size than the decision
set, a handful of researchers in recent years have begun to turn their attention
to the mathematics and tools for generating all or portions of the efficient out-
come set, rather than the efficient decision set, for the MOLP problem. See, for
instance, Banker et al. [2], Banker et al. [3], Benson [4], Benson and Sayin [10],
Dauer and Liu [17], Dauer and Saleh [18], Dauer and Gallagher [16], Dauer [14,
15]. Various methods have been presented for identification of these units, out
of which we can mention the approaches proposed by Chon [13], Evans [22],
Goicoechea et al. [25], Luc [38], Sawaragi et al. [41], Steuer [43], Yu [48] and
Zeleny [51]. One of these approaches is the vector maximization method, see
Kuhn and Tucker [37]. We can use this method to determine all efficient points
in a decision set, see Benson [5], Isermann [31], Bitran [11], Villarreal and Kar-
wan [45], Kostreva and Wiecek [36]. The problem with all those methods of
determining efficient points in the feasible region and the decision set was too
many calculations and the presented approaches were not convergent in most
cases; in this relation, when the problems dimensions, variables and constraints
increase, the set of efficient points would expand in MOLP problems and we
would face a difficult process for finding the efficient set. Since the outcome set
has a simpler form and a smaller region compared to the decision set, it would

be easier to find the efficient points in the outcome, see Steuer [44], Dauer and Liu [17], Dauer and Saleh [18], Benson [4, 6], Dauer [14, 15], Gallagher and Saleh [24], Dauer and Gallagher [16], Horst et al. [30].

Therefore, instead of directly solving the DEA models, we present a DEA model with a MOLP structure and use the MOLP models outcome set to specify the efficient extreme units. To find the set of efficient units in the outcome set, we can employ methods such as the outer approximation algorithm [8] and the weight set decomposition algorithm [9]. In this research, we make use of the outer approximation algorithm, which is a convergent algorithm using little calculations based on linear searching and linear programming techniques. The following methods have been proposed to find DEA efficient points using efficient surfaces in MOLP.

Jahanshahloo et al. [33] presented a method for finding the piecewise linear frontier of the production function in data envelopment analysis. Korhonen [35] introduced another method to search for the efficiency frontier in DEA. In another study, Jahanshahloo et al. [32] proposed an approach for finding strongly efficient hyperplanes of the production possibility set (PPS) in data envelopment analysis. Sayin [42] presented an algorithm for determining efficient faces in DEA. Hosseinzadeh et al. [27] proposed a new method for finding the set of efficient surfaces in DEA based on MOLP models; in this relation, they introduced a linear programming model that could find the efficient defining hyperplanes of the production possibility set. The approach proposed in the present paper is a new and distinguished method comparing to previous approaches. The advantage to our approach is that this method can determine all efficient extreme points of the production possibility set in the outcome set through little calculations.

3. Structural Similarities Between MOLP and DEA

Assume that we have n observed decision-making units as $DMU_j, j = 1, \ldots, n$, where each DMU consumes an m-vector input to produce an s-vector output. Suppose that $X^j = (x_1^j, x_2^j, \ldots, x_m^j)^t$ and $Y^j = (y_1^j, y_2^j, \ldots, y_s^j)^t$ are the vectors of inputs and outputs, respectively, for $DMU_j, j = 1, \ldots, n$, in which it has been assumed that $X^j \geqslant 0$, $X^j \neq 0$ and $Y^j \geqslant 0$, $Y^j \neq 0$. λ_j is the reference weight for $DMU_j, j = 1, \ldots, n$. $X = (x_1, x_2, \ldots, x_m) \in E^m$ represents the input variable vector. $Y = (y_1, y_2, \ldots, y_s) \in E^s$ shows the output variable vactor. We define the production possibility set of data envelopment analysis with VRS as follows:

$$T_v = \{(X, Y) = (x_1, \ldots, x_m, y_1, \ldots, y_s) \mid \sum_{j=1}^{n} \lambda_j y_r^j \geqslant y_r, r = 1, \ldots, s, \sum_{j=1}^{n} \lambda_j x_i^j \leqslant$$

$$x_i, i = 1, \ldots, m, \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \geqslant 0, j = 1, \ldots, n\}.$$

We must note that the T_v set includes all input and output vectors (X, Y) that apply to the sets constraints.

Definition 3.1. $DMU_o = (X^o, Y^o) \in T_v$ is called an efficient point if and only if there is not an $(X, Y) \in T_v$ such that $(X, -Y)^t \leqslant (X^o, -Y^o)^t$ and $(X, -Y)^t \neq (-X^o, Y^o)^t$.

Definition 3.2. $DMU_o = (X^o, Y^o) \in T_v$ is called a weak efficient point if and only if there is not an $(X, Y) \in T_v$ such that $(X, -Y)^t < (X^o, -Y^o)^t$.

Consider the following MOLP problem

$$\begin{aligned} \min \quad & C^t Z \\ s.t. \quad & Z \in R = \{Z \mid AZ \leqslant b, Z \geqslant 0\}. \end{aligned} \tag{1}$$

where $C = (C_1^T, C_2^T, \ldots, C_p^T)$ is a $p \times n$ matrix, $c_i^T = (c_{i1}, c_{i2}, \ldots, c_{ip}) \in E^n$, $i = 1, \ldots, p$, represent the multiples of the i-th objective function in the MOLP problem. E^n shows the Euclidean space. A is the technology matrix including all variable multiples in problem (1). A is an $m \times n$ matrix, $n \geqslant m$ and rank(A)=m,

$b = (b_1, b_2, \ldots, b_m) \in E^m$. $Z = (z_1^T, z_2^T, \ldots, z_n^T) \in E^n$, $Z \in R^n$ represents the decision-making variable vector in the MOLP problem and R shows the feasible region of the MOLP problem. The Pareto solution and weak Pareto solution of (1) are defined as follows:

Definition 3.3. $\bar{Z} \in R$ is called a Pareto solution of (1) if there does not exist $Z \in R$ such that $C^T Z \leqslant C^T \bar{Z}$, $C^T Z \neq C^T \bar{Z}$.

Definition 3.4. $\bar{Z} \in R$ is called a weak Pareto solution of (1) if there does not exist $Z \in R$ so that $C^t Z < C^t \bar{Z}$.

Put $Z = (X, Y, \lambda) = (x_1, \ldots, x_m, y_1, \ldots, y_s, \lambda_1, \ldots, \lambda_n)$, $X \in E^m$, $Y \in E^s$, $\lambda \in E^n$, $C_j^T = -e_j^T, j = 1, \ldots, m$, $C_j^T = e_j^T$, $j = m + 1, \ldots, m + s$, $e_j^T \in E^{m+s+n}$ is a vector whose jth element is one and other elements are zero, $C = (C_1^T, C_2^T, \ldots, C_{m+s}^T)$, and

$$R = \{(x_1, \ldots, x_m, y_1, \ldots, y_s, \lambda_1, \ldots, \lambda_n) \mid \sum_{j=1}^{n} \lambda_j y_r^j \geqslant y_r, r = 1, \ldots, s, \sum_{j=1}^{n} \lambda_j x_i^j \leqslant$$

$$x_i, i = 1, \ldots, m, \sum_{j=1}^{n} \lambda_j = 1, \lambda_j \geqslant 0, j = 1, \ldots, n\}.$$

Then problem (1) is converted to

$$
\begin{aligned}
\max \ & \{-x_1, \ldots, -x_m, y_1, \ldots, y_s\} \\
s.t. \quad & \sum_{j=1}^{n} \lambda_j y_r^j \geqslant y_r, \qquad \text{r=1,\ldots,s,} \\
& \sum_{j=1}^{n} \lambda_j x_i^j \leqslant x_i, \qquad \text{i=1,\ldots,m,} \\
& \sum_{j=1}^{n} \lambda_j = 1, \\
& \lambda_j \geqslant 0, \quad j = 1, \ldots, n, \\
& x_i \geqslant 0, \quad i = 1, \ldots, m, \quad y_r \geqslant 0, \quad r = 1, \ldots, s.
\end{aligned}
\tag{2}
$$

In model (2), vector$(x_1, \ldots, x_m, y_1, \ldots, y_s)$ is the variable vector for inputs and outputs; we can obtain the values of this vector by solving model (2). Note (X, Y, λ) is a feasible solution of problem (2) while $(-X, Y)$ is a vector belong to objective function space of problem (2). By considering Definition 3.3 $(X^\star, Y^\star, \lambda^\star)$ is called a pareto solution of (2) if there does not exist (X, Y, λ) such that $(-X^\star, Y^\star) \leqslant (-X, Y)$ and $(-X^\star, Y^\star) \neq (-X, Y)$.

Theorem 3.5. *Let $(X^\star, Y^\star) \in T_v$ then*
(i) $(X^\star, Y^\star, \lambda^\star)$ is a Pareto solution of (2) if and only if $(X^\star, Y^\star)$ is an efficient unit in T_v.
(ii) $(X^\star, Y^\star, \lambda^\star)$ is a weak Pareto solution of (2) if and only if $(X^\star, Y^\star)$ is a weak efficient unit in T_v.

Proof. (i) Let $(X^\star, Y^\star, \lambda^\star)$ be a Pareto solution of (2). We show that $(X^\star, Y^\star)$ is an efficient point in T_v. By contradiction, suppose $(X^\star, Y^\star)$ is not an efficient point in T_v, then there is an $(\bar{X}, \bar{Y}) \in T_v$ such that $(-\bar{X}, \bar{Y}) \geqslant (-X^\star, Y^\star)$ and $(-\bar{X}, \bar{Y}) \neq (-X^\star, Y^\star)$. Since $(\bar{X}, \bar{Y}) \in T_v$, there is a $\bar{\lambda} \in E^n$ such that $(\bar{X}, \bar{Y}, \bar{\lambda})$ is a feasible solution of (2). Since $(-\bar{X}, \bar{Y}) \geqslant (-X^\star, Y^\star)$ and $(-\bar{X}, \bar{Y}) \neq (-X^\star, Y^\star)$, then we have a contradiction; therefore, $(X^\star, Y^\star)$ is an efficient point in T_v,
Now suppose $(X^\star, Y^\star)$ is an efficient point in T_v. Since $(X^\star, Y^\star) \in T_v$, there is a $\lambda^\star \in R^n$ such that $(X^\star, Y^\star, \lambda^\star)$ is a feasible solution of (2). AS $(X^\star, Y^\star)$ is an efficient point in T_v, there is no $(\bar{X}, \bar{Y}) \in T_v$ such that $(-\bar{X}, \bar{Y}) \geqslant (-X^\star, Y^\star)$ and $(-\bar{X}, \bar{Y}) \neq (-X^\star, Y^\star)$. Since, there is a vector $\bar{\lambda}$ for $(\bar{X}, \bar{Y}) \in T_v$, such that $(\bar{X}, \bar{Y}, \bar{\lambda})$ is a feasible solution of (2). Regarding the above relations there is no $(\bar{X}, \bar{Y}, \bar{\lambda})$ that is a feasible solution of (2) such that $(-\bar{X}, \bar{Y}) \geqslant (-X^\star, Y^\star)$ and $(-\bar{X}, \bar{Y}) \neq (-X^\star, Y^\star)$. Therefore $(X^\star, Y^\star, \lambda^\star)$ is a Pareto solution of (2), and the proof iscompleted.
(ii) Proof is similar to (i). □

Theorem 3.6. *Let* $V^= = \{(-x_1, \ldots, -x_m, y_1, \ldots, y_s) \mid$
$(x_1, \ldots, x_m, y_1, \ldots, y_s) \in T_v\}$, *(* $V^=$ *is called the outcome set for MOLP) then*
$dim\ (V^=) = m + s$.

Proof. Since $C = (C_1^T, C_2^T, \ldots, C_{m+s}^T)$ and $C_j^T = -e_j^T$, $j = 1, \ldots, m$, $C_j^T = -e_j^T$, $j = m+1, \ldots, m+s$. Rank$\{-e_1, \ldots, -e_m, e_{m+1}, \ldots, e_{s+m}\} = m + s$ and $T_v \neq \emptyset$, then dim $(V^=) \leqslant m + s$. $\square$

Theorem 3.7. *The optimal values of problem (2) are finite.*

Proof. Since $\sum_{j=1}^{n} \lambda_j y_r^j \geqslant y_r$, $r = 1, \ldots, s$, and $\sum_{j=1}^{n} \lambda_j = 1$, $\lambda_j \geqslant 0$, $j = 1, \ldots, n$,

then $y_r, r = 1, \ldots, s$ are finite. Similary $-\sum_{j=1}^{n} \lambda_j x_i^j \geqslant -x_i$, $i = 1, \ldots, m$, and

$\sum_{j=1}^{n} \lambda_j = 1$, $\lambda_j \geqslant 0$, $j = 1, \ldots, n$, then $(-x_i), i = 1, \ldots, m$ are finite, Therefore,

the optimal values of problem (2) are finite. $\square$

By using the observed DMUs, For each $i = 1, \ldots, m$ and $r = 1, \ldots, s$, we put
$v_i^{AL} = -\max\{x_i^j \mid (x_1, \ldots, x_m, y_1, \ldots, y_s)^j \in T_v, j = 1, \ldots, n\}$.
$v_{r+m}^{AL} = \min\{y_r^j \mid (x_1, \ldots, x_m, y_1, \ldots, y_s)^j \in T_v, j = 1, \ldots, n\}$.
Vector $v^{AL} = (v_1^{AL}, v_2^{AL}, \ldots, v_m^{AL}, V_{m+1}^{AL}, v_{m+2}^{AL}, \ldots, v_{m+s}^{AL}) \in R^{m+s}$ is called the
anti-ideal point of outcome set for problem (2). Let $\hat{v} \in R^{m+s}$ satisfy $\hat{v} < v^{AL}$,
we define V as follows:
$V = \{v = (v_1, v_2, \ldots, v_m, v_{m+1}, \ldots, v_{m+s}) \mid \hat{v} \leqslant v \leqslant$
$(-x_1, \ldots, -x_m, y_1, \ldots, y_s)$, for some $(x_1, \ldots, x_m, y_1, \ldots, y_s) \in T_v\}$.

Theorem 3.8. *Set* V *is a nonempty, bounded polyhedron in* R^{m+s} *of dimension*
$m + s$.

Proof. Since $\hat{v} < v^{AL} < (-X, Y), (X, Y) \in T_v, T_v \neq \emptyset$, by Theorem(3.3), the
definition of V implies that V is a nonempty, bounded set in R^{m+s}. We may
write $V = V_1 \cap \{V^= + V_2\}$, where
$V_1 = \{v \in R^{m+s} \mid v \geqslant \hat{v}\}$ and $V_2 = \{Z \in R^{m+s} \mid Z \leqslant 0\}$, Since V_1, V_2,
$V^=$ are polyhedral sets as proposed by Dauer and Gallagher [10], therefore V
is a polyhedral set. Since $\hat{v} < (-X, Y), (-X, Y) \in T_v, IntV \neq \emptyset (IntV$ show
interior points set of V), by Theorem (3.2) the dimension of V is $m + s$, and
the proof is complete. $\square$

A point $v^o \in V$ is called an efficient (or admissible) point of V when no $v \in V$
exist such that $v \geqslant v^o$ and $v \neq v^o$. When no $v \in V$ exist such that $v > v^o$, then
v^o is called a weakly efficient (or weakly admissible) point of V. Let V_E and

V_{WE} denote the set all efficient and weakly efficient points, respectively of V.

Theorem 3.9. *Let $V_E^=$ be the set of efficient point of $V^=$ then $V_E^= = V_E$.*

Proof. Suppose that $(-X, Y) \in V_E^=$ but $(-X, Y)$ does not belong to V_E, then by the definition of V_E there exists a point $v' \in V$ such that $v' \geqslant (-X, Y)$. Since $v' \in V$, there exists a point $(X', Y') \in T_v$ such that $v' \leqslant (-X', Y')$, therefore $(-X, Y) < (-X', Y')$. This contradicts $(-X, Y) \in V_E^=$, therefore $v = (-X, Y) \in V_E$.

Now suppose $v \in V_E$, to show that $v \in V_E^=$, we show that $v = (-X, Y)$ for some $(X, Y) \in T_v$ and $(-X, Y) \in EF$ (efficient points of outcome set for MOLP). Since $v \in V$, therefore $v \leqslant (-X, Y)$ for some $(X, Y) \in T_v$, since $v \in V_E$ and $(-X, Y) \in V$ then $v = (-X, Y)$. Let $(X', Y') \in T_v$ satisfy $(-X, Y) < (-X', Y')$, $(-X, Y) \neq (-X', Y')$, then by the definition of V, since $v^{AL} < (-X', Y')$, we have $(-X', Y') \in V$, then $v = (-X, Y)$ does not belong to V_E, but this is a contradiction, therefore $v \in V_E^=$ and the proof is completed. $\square$

Let

$$\beta = \max \sum_{r=1}^{s} y_r - \sum_{i=1}^{m} x_i \tag{3}$$
$$s.t. \quad (x_1, x_2, \ldots, x_{+m}, y_1, y_2, \ldots, y_s) \in T_v.$$

By Theorem(3.3) β is a finite number. If $(\bar{X}, -\bar{Y})$ is an optimal solution of (3), then $(\bar{X}, -\bar{Y})$ is an optimal solution of (2), (we solve problem (2) by the wighthed-sum problem method (by choosing $w_j = 1$, $j = 1, \ldots, m + s$), see, Zeleny [48, 49]).

For $i = 1, \ldots, m + s, j = 1, \ldots, m + s$, we put $q^0 = \hat{v} = (\hat{v}_1, \ldots, \hat{v}_{m+s}) = (-\hat{X}, \hat{Y})$, $\hat{v} < v^{AL}$ and $q^j = (q_1^j, q_2^j, \ldots, q_i^j, \ldots, q_{m+s}^j)$ such that $q_i^j = \hat{v}_i$ for $i \neq j$ and $q_i^j = \beta + \hat{v}_j - \sum_{r=1}^{m+s} \hat{v}_j = \beta + \hat{v}_j - (\sum_{j=1}^{s} \hat{y}_r - \sum_{i=1}^{m} \hat{x}_i)$ for $i = j$.

Theorem 3.10. *The Convex hull of $\{q^0, q1, \ldots, q^{m+s}\}$ is an $m+s$–dimensional simplex and contains V.*

Proof. First we show that $\{q^0, q1, \ldots, q^{m+s}\}$ is a affinely independent, since $(q^j - q^o)_i = 0$ for $i \neq j$ and $(q^j - q^o)_i = \beta - \sum_{r=1}^{s} \hat{y}_r + \sum_{i=1}^{m} \hat{x}_i$ for $i = j$, $i = 1, \ldots, m + s, j = 1, \ldots, m + s$, we put $\gamma = \beta - \sum_{r=1}^{s} \hat{y}_r + \sum_{i=1}^{m} \hat{x}_i$, therefore, $\gamma > 0$ (by the definition of β, this is evident), Let $\sum_{j=1}^{m+s} c_j (q^j - q^0) = 0$ then

$$\sum_{j=1}^{m+s} c_j(q^j - q^o) = (c_1\gamma, c_2\gamma, \ldots, c_{m+s}\gamma) = 0;$$ since $\gamma \neq 0$, therefore, $c_j = 0$,

$j = 1, \ldots, m+s$ and $\{q^0, q1, \ldots, q^{m+s}\}$ is a affinely independent.

To show that the convex hull contains V, suppose $\bar{v} \in V$. Since $\{q1 - q^0, \ldots, q^{m+s} - q^0\}$ is a lineary independent set, hence it is a basis for R^{m+s}; therefore, there is $\alpha^T = (\alpha_1, \ldots, \alpha_{m+s})$ such that $\sum_{j=1}^{m+s} \alpha_j = 1$, $\alpha_j \geqslant 0$,

$j = 1, \ldots, m+s$ and $\bar{v} - q^0 = \sum_{j=1}^{m+s} \alpha_j(q^j - q^0)$. If $\sum_{j=1}^{m+s} \alpha_j > 1$, we have $\sum_{j=1}^{m+s}(\bar{v} - q^0)_j = \sum_{j=1}^{m+s} \alpha_j\gamma = \gamma \sum_{j=1}^{m+s} \alpha_j > \gamma$, but we have $\sum_{j=1}^{m+s}(\bar{v} - q^0)_j \leqslant \max \sum_{j=1}^{m+s}(v - q^0)_j = \max(\sum_{j=1}^{m+s} v_j - \sum_{j=1}^{m+s} \hat{v}_j) = \beta - \sum_{r=1}^{s} \hat{y}_r + \sum_{i=1}^{m} \hat{x}_i = \gamma$, which contradicts the previous paragraph. Hence $\sum_{j=1}^{m+s} \alpha_j \leqslant 1$ and $\bar{v} = (1 - \sum_{j=1}^{m+s} \alpha_j)q^0 + \sum_{j=1}^{m+s} \alpha_j q^j$, therefore, $\bar{v} \in S$ (S is the convex hull of $\{q^0, q1, \ldots, q^{m+s}\}$). Then we showed that $V \subseteq S$ and the proof is completed. $\square$

Theorem 3.11. S may also be written as following.

$$S = \{(-X, Y) \in E^{m+s} \mid (-\hat{X}, \hat{Y}) < (-X, Y), \sum_{r=1}^{s} y_r - \sum_{i=1}^{m} x_i \leqslant \beta\}.$$

Proof. Suppose that $\bar{v} \in S$, then

$$\bar{v} = \sum_{j=0}^{m+s} \alpha_j q_j = (1 - \sum_{j=1}^{m+s} \alpha_j)q^0 + \sum_{j=1}^{m+s} \alpha_j q^j = q^0 + \sum_{j=1}^{m+s} \alpha_j(q^j - q^0) = (-\hat{X}, \hat{Y}) +$$

$(\alpha_1\gamma, \alpha_2\gamma, \ldots, \alpha_{m+s}\gamma) > (-\hat{X}, \hat{Y}) = q^0$.

Therefore $\bar{v} > (-\hat{X}, \hat{Y})$. On other hand, we havae

$$\sum_{r=1}^{m+s} \bar{v}_j = \sum_{i=1}^{m+s} \alpha_j\gamma = \gamma \leqslant \beta, \text{ therefore}$$

$$\bar{v} \in \{(-X, Y) \in E^{m+s} \mid (-X, Y) > (-\hat{X}, \hat{Y}), \sum_{r=1}^{s} y_r - \sum_{i=1}^{m} x_i \leqslant \beta\}.$$

Now suppose that $(-\bar{X}, \bar{Y}) \in \{(-X, Y) \in R^{m+s} \mid (-X, Y) > (-\hat{X}, \hat{Y}),$

$\sum_{r=1}^{s} y_r - \sum_{i=1}^{m} x_i \leqslant \beta\}$. Let $\alpha_j = (\frac{-\bar{x}_j + \hat{x}_j}{\gamma}), j = 1, \ldots, m$, $\alpha_{m+j} = (\frac{\bar{y}_j - \hat{y}_j}{\gamma})$, $j = 1, \ldots, s$, $0 \leqslant \alpha_j \leqslant 1$, then $\sum_{j=1}^{m+s} \alpha_j = \sum_{j=1}^{m}(\frac{-\bar{x}_j + \hat{x}_j}{\gamma}) + \sum_{j=1}^{s}(\frac{\bar{y}_j - \hat{y}_j}{\gamma})$. We have

$-\bar{x}_j + \hat{x}_j > 0$, $\bar{y}_j - \hat{y}_j > 0$, but $\sum_{j=1}^{s} \bar{y}_r - \sum_{j=1}^{m} \bar{x}_j - (\sum_{j=1}^{s} \hat{y}_r - \sum_{j=1}^{m} \hat{x}_j) < \beta -$

$(\sum_{j=1}^{s} \hat{y}_r - \sum_{j=1}^{m} \hat{x}_j) = \gamma$, therefore $\sum_{j=1}^{m+s} \alpha_j = \sum_{j=1}^{m} (\frac{-\bar{x}_j + \hat{x}_j}{\gamma}) + \sum_{j=1}^{s} (\frac{\bar{y}_j - \hat{y}_j}{\gamma}) < 1$.

By the definition of q^j, $j = 1, \ldots, m + s$, we have $q^0 + \sum_{j=1}^{m+s} \alpha_j (q^j - q^0) =$

$q^0 + (\alpha_1\gamma, \alpha_2\gamma, \ldots, \alpha_{m+s}\gamma) = (-\hat{X}, \hat{Y}) +$
$\gamma(\frac{-\bar{x}_1 + \hat{x}_1}{\gamma}, \ldots, \frac{-\bar{x}_m + \hat{x}_m}{\gamma}, \frac{\bar{y}_1 - \hat{y}_1}{\gamma}, \ldots, \frac{\bar{y}_s - \hat{y}_s}{\gamma}) == (-\hat{X}, \hat{Y}) + (-\bar{X}, \bar{Y}) +$

$(-\hat{X}, \hat{Y}) = (-\bar{X}, \bar{Y})$, then $(-\bar{X}, \bar{Y}) = \sum_{j=0}^{m+s} \alpha_j q_j = (1 - \sum_{j=1}^{m+s} \alpha_j)q^0 + \sum_{j=1}^{m+s} \alpha_j q_j$,

therefore $\bar{v} = (-\bar{X}, \bar{Y}) \in S$. $\square$

Theorem 3.12. Let $(-X_p, Y_p) \in Int(V)$ and suppose $(-X, Y) > (-\hat{X}, \hat{Y})$, $(-X, Y)$ does not belong to V and $(-X^w, Y^w) = (-(X + \lambda^*(X_p - X)), Y + \lambda^*(Y_p - Y))$, where λ^* is the solution of problem (4) then $(-X^w, Y^w) \in V_{WE}$.

$$\max \lambda$$
$$s.t. \quad (-(X + \lambda(X_p - X)), Y + \lambda(Y_p - Y)) \in V. \tag{4}$$

Proof. Suppose $(-X^w, Y^w)$ does not belong to V_{WE}. Then we may choose a point $(-X_0, Y_0) \in V$ such that $(-X_0, Y_0) > (-X^w, Y^w)$. Since $(-X_p, Y_p) \in IntV$, then $(-X_p, Y_p) > (-\hat{X}, \hat{Y})$, on other hand, we have $(-X, Y) > (-\hat{X}, \hat{Y})$, then $(-X^w, Y^w) > (-\hat{X}, \hat{Y})$.
Put $d_1 = \min\{(y_{ro} - y_r^w), (-x_{io} + x_i^w) \mid r = 1, \ldots, s, , i = 1, \ldots, m\}$,
$d_2 = \min\{(y_r^w - \hat{y}_r), (-x_i^w + \hat{x}_i) \mid r = 1, \ldots, s, i = 1, \ldots, m\}$.
Choose $\epsilon > 0$ such that $\epsilon < d_1$ and $\epsilon < d_2$. Let $N_\epsilon(-X^w, Y^w) = \{(-X, Y) \in R^{m+s} \mid \|(-X, Y) - (-X^w, Y^w)\| < \epsilon\}$. Suppose
$Z \in N_\epsilon(-X^w, Y^w)$ then $-\epsilon < z_r - y_r^w < \epsilon$ and $y_r^w - \epsilon < z_r < y_r^w + \epsilon$. Since $\epsilon < d_1$ then $y_{ro} - y_r^w > \epsilon$, $y_r^w - \hat{y}_r < \epsilon$ then, $\hat{y}_r < z_r < y_{ro}$, $r = 1, \ldots, s, i = 1, \ldots, m$.
Therefore $Z \in V$. Similarly we show that $-\hat{x}_i < z_r < -x_i^0$, $i = 1, \ldots, m$. We conclude that $N_\epsilon(-X^w, Y^w) \subseteq V$. Since $\epsilon > 0$, this contradicts the fact that $(-X^w, Y^w)$ belong to the boundary of V(consider problem(4)) so the proof is complete. $\square$

From Rockafellar [40] and Yu [48] and the weighted-sum problem, F is a face of V if and only if F is equal to the set of the optimal solution set of following problem.

$$\max \sum_{r=1}^{s} u_r y_r - \sum_{i=1}^{m} u_{i+s} x_i \qquad (5)$$
$$s.t. \quad (x_1, x_2, \ldots, x_m, y_1, y_2, \ldots, y_s) \in T_v,$$

for some $(u_1, u_2, \ldots, u_s, u_{s+1}, u_{s+2}, \ldots, u_{s+m}) \in R^{s+m}$.
The variable vector $(u_1, u_2, \ldots, u_s, u_{s+1}, u_{s+2}, \ldots, u_{s+m})$ expresses the corresponding multiples of the output and input vector
$(y_1, y_2, \ldots, y_s, x_1, x_2, \ldots, x_m)$. We know that $(-X^w, Y^w)$ is weak efficient if and only if the optimal value of the following problem is zero. (It is the clear)

$$\max \; t$$
$$s.t. \quad (-X, Y) - et \geqslant (-X^w, Y^w), \qquad (6)$$
$$(X, Y) \in T_v, t \geqslant 0.$$

The dual of the linear program (6) is as follows:

$$\min \; -\sum_{r=1}^{s} u_r y_r^w + \sum_{i=1}^{m} u_{i+s} x_i^w + v_{m+s+1}$$
$$s.t. \quad -u_r + v_r \geqslant 0, \qquad\qquad r = 1, \ldots, s,$$
$$u_{s+i} - v_{s+i} \geqslant 0, \qquad\quad i = 1, \ldots, m,$$
$$\sum_{r=1}^{s} u_r + \sum_{i=1}^{m} u_{i+s} \geqslant 1 \qquad\qquad (7)$$
$$-\sum_{r=1}^{s} v_r y_r^j + \sum_{i=1}^{m} v_{i+s} x_i^j + v_{m+s+1} \geqslant 0,$$
$$u_j \geqslant 0, \quad v_j \geqslant 0, \quad j = 1, \ldots, m+s.$$

By the duality Theorem of linear programing, since the optimal value of (6) is zero, problem (7) also has an optimal value equal to zero, therefore $\sum_{r=1}^{s} u_r^* y_r^w - \sum_{i=1}^{m} u_{i+s}^* x_i^w - v_{m+s+1}^* = 0$ and
$(u_1^*, u_2^*, \ldots, u_s^*, u_{s+1}^*, u_{s+2}^*, \ldots, u_{s+m}^*) \geqslant 0$,
$(u_1^*, u_2^*, \ldots, u_s^*, u_{s+1}^*, u_{s+2}^*, \ldots, u_{s+m}^*) \neq 0$, From Falk and Hoffman [23], we know that the optimal values of problem (5) correspond to weakly efficient faces of V for $(u_1, u_2, \ldots, u_s, u_{s+1}, u_{s+2}, \ldots, u_{s+m}) = (u_1^*, u_2^*, \ldots, u_s^*, u_{s+1}^*, u_{s+2}^*, \ldots, u_{s+m}^*)$. Inequality $\sum_{r=1}^{s} u_r^* y_r - \sum_{i=1}^{m} u_{i+s}^* x_i - v_{m+s+1}^* \leqslant 0$, construct inequality cuts needed in the outer approximation algorithm for generating all efficient extreme points of V.

4. Generating All Efficient Extreme Points of the production possibility set

We apply the outer approximation algorithm for generating all efficient extreme points of the outcome set of problem (2). In what follows, all efficient extreme points of the production possibility set of the DEA with VRS are essentially immediately available upon termination of the algorithm, by converting points $(-X, Y)$ in the outcome set of problem (2) to equivalent points (X, Y) in T_v .

The Outer Approximation Algorithm applied as follows:

The Initialization step. Compute a point $(-X_p, Y_p) \in Int(V)$.
$(-X_p, Y_p) \in E^{m+s}$ may be set equal to any strict convex combination of v^{AL} and $(-X^*, Y^*)$, where $(-X^*, Y^*)$ is any optimal solution to the linear program (6) with $(-X^w, Y^w) = (-X^{AL}, Y^{AL}) = v^{AL}$ and construct the $m + s$-dimensional simplex $S^0 = S$ containing described in Theorems (3.6) and (3.7). Set S is a m+s-dimensional simplex consisting of the vertices of V, as described in Theorems (3.6). Store both the vertex set S given in Theorem (3.6) inequality representation of $S^0 = S$ given in the Theorem (3.7). Set $k = 0$ and go to iteration k. Iteration k, $k \geqslant 0$, See Steps 1 through 4 below.

Step 1). If, all vertices of S^k belong to V, then stop $S^k = V$. Otherwise, choose any vertex of S^k such that, it does not belong to V, for example $(-X^k, Y^k)$ and continue (To test a given $(-X^k, Y^k)$ is membership in V, one may apply the phase-1 procedure of the simplex method to problem (6) by putting $(-X^w, Y^w) = (-X^k, Y^k)$).

Step 2). Compute $(-X^w, Y^w)$ description in the Theorem (3.8) by putting $(-X, Y) = (-X^k, Y^k)$.

Step 3). Set $S^{k+1} = S^k \cap \{(-X, Y) \in R^{m+s} \mid \sum_{r=1}^{s} u_r^* y_r - \sum_{i=1}^{m} u_{i+s}^* x_i \leqslant v_{m+s+1}^* \}$
where $(u_1^*, u_2^*, \ldots, u_s^*, u_{s+1}^*, u_{s+2}^*, \ldots, u_{s+m}^*)$ is any dual optimal solution to the linear programing (7) with $(-X^w, Y^w)$ that have been calculated in the step (2).

Step 4). Using vertex of S^k and method that it supposed by Falk and Hoffman [23] and definition of S^{k+1} given in Step (3), determine all vertex of S^{k+1}, set $k = k + 1$ and go to iteration k.

By definition of S^{k+1} in step (3) since $(-X^k, Y^k)$ don't belong to S^{k+1} and $V \subseteq S^{k+1}$, we conclude that algorithm generates distinct polyhedral S^j, $j = 0, 1, \ldots k$, so that $V \subset S^k \subset S^{k-1}, \ldots, \subset S^1 \subset S^0$. This implies that the algorithm must be finite and it must terminate in some iteration $k \geqslant 0$. S^k is

a $m + s$-dimensional simplex including the vertices of V, formed in each stage k.

Theorem 4.1. Let $k \geqslant 0$ denotetheiteration number in which $S^k = V$ and the outerapproximation algorithm terminate. Let $E = \{(-X, Y) \mid (-X, Y)$ belong to vertex set of S^k and $(-X, Y) > (-\hat{X}, \hat{Y})\}$ then E is identical to the set of all efficient extreme points of $V^=$.

Proof. From before we have $S^k = V = \{(-X, Y) \in R^{m+s} \mid (-X, Y) > (-\hat{X}, \hat{Y}), \sum_{r=1}^{s} y_r - \sum_{i=1}^{m} x_i \leqslant \beta\} \cap (\cap_{n=0}^{k-1} H_n)$ and for $n = 0, 1, \ldots, k-1$, $H_n = \{(-X, Y) \in R^{m+s} \mid \sum_{r=1}^{s} u_r^n y_r - \sum_{i=1}^{m} u_{s+i}^n x_i - v_{m+s+1}^n \leqslant 0, (X, Y) \in T_v\}$.

$(u_1^n, \ldots, u_s^n, u_{s+1}^n, \ldots, u_{s+m}^n)$ is any dual optimal solution to the linear programming (6) with $(-X^w, Y^w)$ in step 3. Notice also that

$$\{(-X, Y) \in R^{m+s} \mid \sum_{r=1}^{s} u_r^n y_r - \sum_{i=1}^{m} u_{s+i}^n x_i - v_{m+s+1}^n < 0, (X, Y) \in T_v\} \subseteq V_{WE}.$$

Suppose that $(-\bar{X}, \bar{Y}) \in E$, then $(-\bar{X}, \bar{Y})$ are belong to vertex set of $S^k = V$ and $(-\bar{X}, \bar{Y}) > (-\hat{X}, \hat{Y})$, therefore at last, $m + s$ of the inequalities below, must hold as equations at $(-X, Y) = (-\bar{X}, \bar{Y})$.

$$\sum_{r=1}^{s} y_r - \sum_{i=1}^{m} x_i \leqslant \beta, \sum_{r=1}^{s} u_r^n y_r - \sum_{i=1}^{m} u_{s+i}^n x_i \leqslant v_{m+s+1}, n = 0, \ldots, k-1.$$

That implies that $(-\bar{X}, \bar{Y}) \in V_{WE}$. We show $(-\bar{X}, \bar{Y}) \in V_E$, by contradiction, suppose $(-\bar{X}, \bar{Y})$ dose not belong to V_E, therefore, there is $(-X, Y)$ such that $(-\bar{X}, \bar{Y}) \leqslant (-X, Y), (-\bar{X}, \bar{Y}) \neq (-X, Y)$.

Let $I_{11} = \{j \mid y_j = \bar{y}_j\}$, $I_{12} = \{j \mid y_j \neq \bar{y}_j\}$, $I_{21} = \{j \mid x_j = \bar{x}_j\}$, $I_{22} = \{j \mid x_j \neq \bar{x}_j\}$. For $j \in I_{12}$ let $n_j = y_j - \bar{y}_j$ and for $j \in I_{22}$ let $m_j = -x_j + \bar{x}_j > 0$.

We choose $M > 0$ such that $\bar{y}_j - \frac{n_j}{M} > \hat{y}_j$, $\bar{x}_j + \frac{m_j}{M} < \hat{x}_j$ and define $y_j^{new} = \bar{y}_j$ for $j \in I_{11}$ and $y_j^{new} = \bar{y}_j - \frac{n_j}{M}$ for $j \in I_{12}$ and $x_j^{new} = \bar{x}_j$ for $j \in I_{21}$ and $x_j^{new} = \bar{x}_j + \frac{m_j}{M}$ for $j \in I_{22}$.

Then $y_j = n_j + \bar{y}_j > 0$ for $j \in I_{11}$, $y_j = \bar{y}_j$ for $j \in I_{12}$ and $x_j = -m_j + \bar{x}_j > 0$ for $j \in I_{21}$, $x_j = \bar{x}_j$ for $j \in I_{22}$.

Then $\bar{y}_j = \frac{M}{M+1} y_j^{new} + \frac{1}{M+1} y_j$, $\bar{x}_j = \frac{M}{M+1} x_j^{new} + \frac{1}{M+1} x_j$ therefore $(-\bar{X}, \bar{Y}) = \frac{1}{M+1}(-X, Y) + \frac{M}{M+1}(-X^{new}, Y^{new})$. We have $0 < \frac{1}{M+1} < 1$ then $(-\bar{X}, \bar{Y})$ is a strict convex combination of $(-\bar{X}, \bar{Y}) \in V$ and $(-X^{new}, Y^{new}) \in V$, that is contradiction (because $\bar{v} = (-\bar{X}, \bar{Y})$ is belong to vertex set of $V = S^k$). Therefore $(-\bar{X}, \bar{Y}) \in V_E$, from Theorem (3.5), we have $(-\bar{X}, \bar{Y}) \in V_E^=$. Since $V^= \subset V$ then $(-\bar{X}, \bar{Y})$ is an efficient extreme point of $V^=$.

Now suppose that $(-X, Y)$ is a efficient extreme point of $V^=$ and $(-X, Y)$

don't belong to all efficient extreme points of S^k, therefore, we choose $(-X^1, Y^1), (-X^2, Y^2) \in V$ and $\alpha \in R, 0 < \alpha < 1$ so that $(-X, Y) = \alpha(-X^1, Y^1) + (1-\alpha)(-X^2, Y^2)$. By solving problem (2) by wighthed-sum problem method (see, Zeleny [49]), and from Theorem (3) that it proposed by Benson [7], we conclude that $(-X, Y)$ is an efficient extreme point of the polyhedron $V^=$.

We may select a point $(u_1, u_2, \ldots, u_s, u_{s+1}, u_{s+2}, \ldots, u_{s+m}) \in R^{m+s}$ $(u_1, u_2, \ldots, u_s, u_{s+1}, u_{s+2}, \ldots, u_{s+m}) > 0$, such that $(-X, Y)$ is the unique optimal solution to the problem

$$\max \sum_{r=1}^{s} u_r y_r - \sum_{i=1}^{m} u_{i+s} x_i \tag{8}$$
$$s.t. \quad (-x_1, -x_2, \ldots, -x_m, y_1, y_2, \ldots, y_s) \in V.$$

From the definition of V, this implies that $(-X, Y)$ is also the unique optimal solution to the problem (8). since $(-X^1, Y^1), (-X^2, Y^2) \in V$, therefore

$$\sum_{r=1}^{s} u_r y_r^1 - \sum_{i=1}^{m} u_{i+s} x_i^1 \leqslant \sum_{r=1}^{s} u_r y_r - \sum_{i=1}^{m} u_{i+s} x_i \text{ and}$$

$$\sum_{r=1}^{s} u_r y_r^2 - \sum_{i=1}^{m} u_{i+s} x_i^2 \leqslant \sum_{r=1}^{s} u_r y_r - \sum_{i=1}^{m} u_{i+s} x_i.$$

Since $0 < \alpha < 1$ these inequalities imply that

$$\alpha(\sum_{r=1}^{s} u_r y_r^1 - \sum_{i=1}^{m} u_{i+s} x_i^1) + (1-\alpha)(\sum_{r=1}^{s} u_r y_r^2 - \sum_{i=1}^{m} u_{i+s} x_i^2) \leqslant$$

$$\sum_{r=1}^{s} u_r y_r - \sum_{i=1}^{m} u_{i+s} x_i. \text{ Since } (-X, Y) = \alpha(-X^1, Y^1) + (1-\alpha)(-X^2, Y^2) \text{ the}$$

left-hand-side of the previous inequality equals $\sum_{r=1}^{s} u_r y_r - \sum_{i=1}^{m} u_{i+s} x_i$, yielding a contradiction. Therefore, $(-X, Y)$ belong to all efficient extreme points of S^k must be true, so that the proof is complete. $\square$

5.   Application and Discussion

In this section, we illustrate the problem by two numerical example.

Example 5.1. Consider the case where there are seven units with an input and an output whose details have been given in the following Table.

Table 1: The data of the seven DMUs.

	DMU_1	DMU_2	DMU_3	DMU_4	DMU_5	DMU_6	DMU_7
Inputs	1	1	2	4	5	6	3
Outputs	1	3	5	6	7	8	2
θ^*_{BCC}	1	1	1	0.83	0.93	1	0.33
Efficiency status	non-extreme	extreme	extreme	-	-	extreme	-
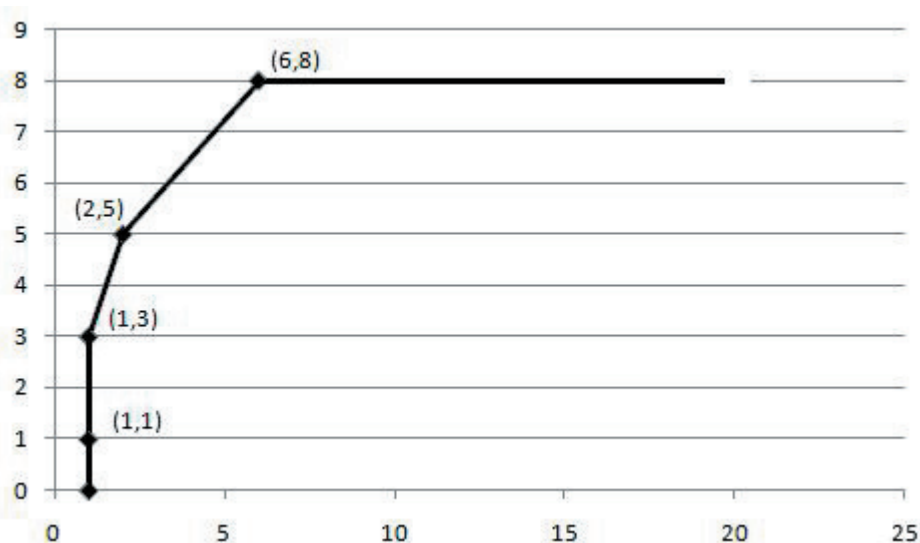
Figure 1. The efficiency frontier of production possibility set.

The corresponding MOLP is

$$
\begin{aligned}
\max \ & \{-x_1, y_1\} \\
s.t \quad & \lambda_1 + \lambda_2 + 2\lambda_3 + 4\lambda_4 + 5\lambda_5 + 6\lambda_6 + 3\lambda_7 \leqslant x_1 \\
& \lambda_1 + 3\lambda_2 + 5\lambda_3 + 6\lambda_4 + 7\lambda_5 + 8\lambda_6 + 2\lambda_7 \geqslant y_1 \\
& \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 + \lambda_6 + \lambda_7 = 1 \\
& \lambda_1 \geqslant 0, \lambda_2 \geqslant 0, \lambda_3 \geqslant 0, \lambda_4 \geqslant 0, \lambda_5 \geqslant 0, \\
& \lambda_6 \geqslant 0, \lambda_7 \geqslant 0, x_1 \geqslant 0, y_1 \geqslant 0.
\end{aligned}
\tag{9}
$$

Where the efficiency frontier of production possibility set of above example is shown in Figure 1.

In the initialization step of the algorithm have

$v_1^{AL} = -max\{x_1^j \mid j = 1,,\ldots, 7\} = -6,$

$v_2^{AL} = min\{y_1^j \mid j = 1,,\ldots, 7\} = 1$ then $v^{AL}=(v_1, v_2) = (-6, 1).$

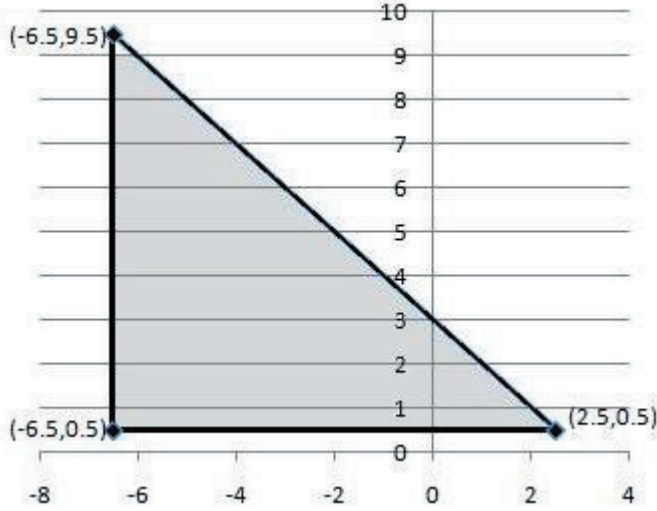We select $(-\hat{X}, \hat{Y}) = (-6.5, 0.5).$ By definishition of V, we have

$V = \{(v_1, v_2) \mid 0.5 \leqslant v_2 \leqslant y_1, -6.5 \leqslant v_1 \leqslant -x_1,$ for some (x_1, y_1) belonging to the feasible region of problem (9)$\}$.

Put $k = 1$. If we solve problem (3), we would obtain $\beta = 3$.

Therefore $S^0 = \{(-x_1, y_1) \mid 0.5 \leqslant y_1, -6.5 \leqslant -x_1, y_1 - x_1 \leqslant 3\}$. As shown in Figure 2.

The verte:
$\{(-6.5, 0.5$

Figure 2: The S^0 set in (x_1, y_1) space.

The vertex set of S^0 given in Theorem (3.6) are
$\{(-6.5, 0.5), (-6.5, 9.5), (2.5, 0.5)\}$ which are computed as follows:
$q^0 = (-6.5, 0.5)$, $q_1^1 = 3 - 6.5 + 6 = 2.5$, $q_2^1 = 0.5$, $q_1^2 = -6.5$, $q_2^2 = 3 + 0.5 + 6 = 9.5$, therefore $q^1 = (2.5, 0.5)$ and $q^2 = (-6.5, 9.5)$.

In step (1) of the algorithm, since (-6.5,9.5) does not belong to V, we put $(-X^1, Y^1) = (-6.5, 9.5)$.

We go to step (2). If we solved problem (6) by $(-X^w, Y^w) = v^{AL} = (v_1, v_2) = (-6, 1)$ we would obtain $(X^*, Y^*, t^*) = (2, 5, 4)$.
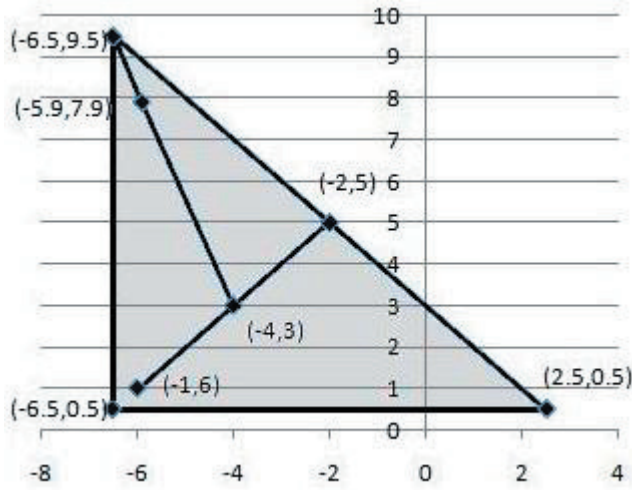
We put $(-X_p, Y_p) = 0.5$ $(-X^*, Y^*) + 0.5(v_1^{AL}, v_2^{AL}) = 0.5(-2,5) + 0.5(-6,1)$, then $(-X^p, Y^p) = (-4, 3)$.

Now we solved problem (4), we put $(-X, Y) = (-X_1, Y_1) = (-6.5, 9.5)$ and $(-X_p, Y_p) = (-4, 3)$, we would obtain $\lambda^* = 0.76$ and $(-X^w, Y^w) = (-5.9, 7.92)$. As shown in Figure 3.

We solved problem (7) by $(-X^w, Y^w) = (-5.9, 7.92)$. , we would obtain $u_1^* = 0.57$, $u_2^* = 0.43$, $v_3^* = 2$, then the inequality cut is as follows.
$0.57y_1 - 0.43x_1 \leqslant 2$.

We go to step (3) and we organize

Figure 3. The $(-X_p, Y_p)$, $(-X^w, Y^w)$, and v^{AL} in (x_1, y_1) space.

$S^1 = \{(-x_1, y_1) \mid 0.5 \leqslant y_1, -6.5 \leqslant -x_1, y_1 - x_1 \leqslant 3, 0.57y_1 - 0.43x_1 \leqslant 2\}$.
As shown in Figures 5.4 and 5.5. The vertex set of S^1 given in the Theorem
(3.6) are $\{(-6.5, 0.5), (-2, 5), (2.5, 0.5), (-6.5, 8.375)\}$. We put $k = 2$. Since
(-6.5,8.35) does not belong to V, we put $(-X^2, Y^2) = (-6.5, 8.375)$.
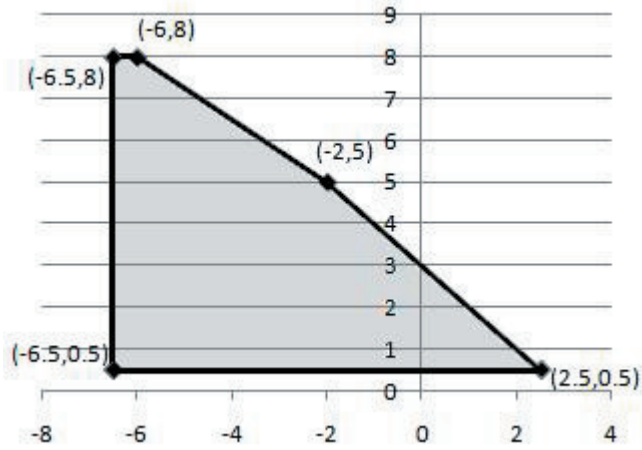

Figure 4. The S^1 in (x_1, y_1) space.

Figure 5. The S^2 in (x_1, y_1) space.
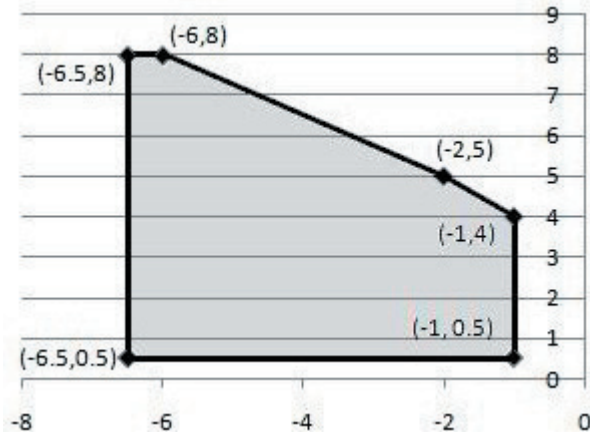
Figure 6. The S^3 in (x_1, y_1) space.

In the next step, we solved problem (4), we put $(-X, Y) = (-X^2, Y^2) = (-6.5, 8.375)$ and $(-X_p, Y_p) = (-4, 3)$, we obtain $\lambda^* = 0.924$ and $(-X^w, Y^w) = (-6.326, 8)$. We solved problem (7) by $(-X^w, Y^w) = (-6.326, 8)$, we would obtain $u_1^* = 1, u_2^* = 0, v_3^* = 8$, then the inequality cut is as follows.
$y_1 \leqslant 8$.
We go to step (3) and we organize
$S^2 = \{(-x_1, y_1) \mid 0.5 \leqslant y_1, -6.5 \leqslant -x_1, y_1 - x_1 \leqslant 3, 0.57y_1 - 0.43x_1 \leqslant 2, y_1 \leqslant 8\}$.

The vertexs set of S^2 given in Theorem (3.6) are
$\{(-6.5, 0.5), (-2, 5), (2.5, 0.5), (-6.5, 8), (-6, 8)\}$.As shown in Figure 5.5. We
put $k = 3$. Since (2.5, 0.5) dose not belong to V, we put
$(-X^3, Y^3)=(2.5, 0.5)$. In the next step, we solved problem (4), we put $(-X, Y) = (-X^3, Y^3) = (2.5, 0.5)$ and $(-X_p, Y_p) = (-4, 3)$, we would obtain $\lambda^* = 0.462$
and $(-X^w, Y^w) = (-1, 1.8462)$. We solved problem (7) by $(-X^w, Y^w) = (-1, 1.8462)$, we would obtain $u_1^* = 0, u_2^* = 1, v_3^* = -1$, then the inequality cut
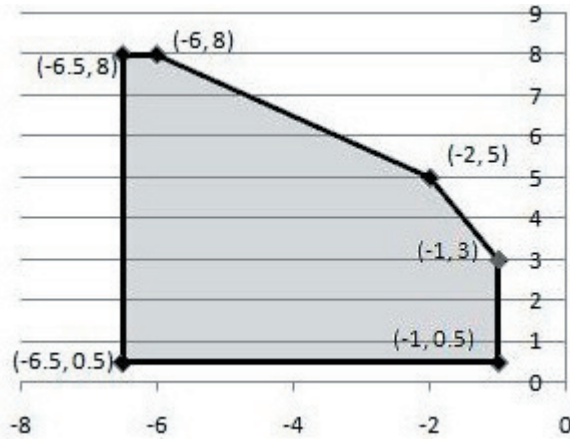is as follows.
$x_1 \leqslant -1$.

Figure 7. The S^4 in (x_1, y_1) space.

We go to step (3) and we organize $S^3 = \{(-x_1, y_1) \mid 0.5 \leqslant y_1, -6.5 \leqslant -x_1, y_1 - x_1 \leqslant 3, 0.57y_1 - 0.43x_1 \leqslant 2, y_1 \leqslant 8, x_1 \leqslant -1\}$.
As shown in Fig 6. The vertexes set of S^3 given in Theorem (3.6) are
$\{(-6.5, 0.5), (-1, 4), (-1, 0.5), (-2, 5), (-6, 8), (-6.5, 8)\}$.
We put $k = 4$. Since (-1,4) does not belong to V, we put $(-X^4, Y^4) = (-1, 4)$.

In the next step, we solved problem (4), we put $(-X^4, Y^4) = (-1, 4)$ and
$(-X_p, Y_p) = (-4, 3)$, we would obtain $\lambda^* = 0.857$ and $(-X^w, Y^w) = (-1.429, 3.857)$.
We solved problem (7) by $(-X^w, Y^w) = (-1.429, 3.857)$, we would obtain
$u_1^* = 0.3333, u_2^* = 0.6666, v_3^* = 0.3333$, then the inequality cut is as follows.
$y_1 - 2x_1 \leqslant 1$.

We go to step (3) and we organize $S^4 = \{(-x_1, y_1) \mid 0.5 \leqslant y_1, -6.5 \leqslant -x_1, y_1 - x_1 \leqslant 3, 0.57y_1 - 0.43x_1 \leqslant 2, y_1 \leqslant 8, x_1 \leqslant -1, y_1 - 2x_1 \leqslant 1\}$.
As shown in Figure 7, the vertexes set of S^4 given in Theorem (3.6) are
$\{(-6.5, 0.5), (-1, 3), (-1, 0.5), (-2, 5), (-6, 8), (-6.5, 8)\}$.

Since (-6.5,0.5), (-1,0.5) and (-6.5,8) have the same components to $\hat{V} = (-\hat{X}, \hat{Y}) = (-6.5, 0.5)$, then they do not belong to $V^=$. Therefore, the vertexes set of $V_E^=$ are as follows.
$\{(-1, 3), (-2, 5), (-6, 8)\}$. We obtain the vertexes set of T_v by converting $(-x_1, y_1)$ to (x_1, y_1) as follows:
$(1, 3), (2, 5), (6, 8)$.

The algorithm is terminated.

Example 5.2. Consider the case where there are five units with an input and two outputs whose details have been given in the following Table.

Table 2: The input and outputs of the DMUs.

	DMU_1	DMU_2	DMU_3	DMU_4	DMU_5
Inputs	1	1	1	1	1
Output1	6	5	2	3	2
Output2	2	3.5	5	3.5	2
θ_{BCC}^*	1	1	1	0.833	0.5
Efficiency status	extreme	extreme	extreme	-	-

The corresponding MOLP is

$$\max \{-x_1, y_1, y_2\}$$
$$s.t \quad \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \leqslant x_1,$$
$$6\lambda_1 + 5\lambda_2 + 2\lambda_3 + 3\lambda_4 + 2\lambda_5 \geqslant y_1,$$
$$2\lambda_1 + 3.5\lambda_2 + 5\lambda_3 + 3.5\lambda_4 + 2\lambda_5 \geqslant y_2, \qquad (10)$$
$$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 = 1,$$
$$\lambda_1 \geqslant 0, \lambda_2 \geqslant 0, \lambda_3 \geqslant 0, \lambda_4 \geqslant 0, \lambda_5 \geqslant 0,$$
$$x_1 \geqslant 0, y_1 \geqslant 0, y_2 \geqslant 0.$$

In the initialization step of the algorithm have
$v_1^{AL} = -max\{x_1^j \mid j = 1,,\ldots,5\} = -1, v_2^{AL} = min\{y_1^j \mid j = 1,,\ldots,5\} = 2,$
$v_3^{AL} = min\{y_2^j \mid j = 1,,\ldots,5\} = 2$ then $v^{AL}=(v_1, v_2, v_3) = (-1, 2, 2)$.
We select $(-\hat{X}, \hat{Y}) = (-1.5, 1.5, 1.5)$. By definishition of V, we have
$V = \{(v_1, v_2, v_3) \mid -1.5 \leqslant v_1 \leqslant -x_1, 1.5 \leqslant v_2 \leqslant y_1, 1.5 \leqslant v_3 \leqslant y_2,$
for some (x_1, y_1, y_2) belonging to the feasible region of problem (10)$\}$.
Put $k = 1$. If we solve problem (3), we would obtain $\beta = 7.5$.
Therefore $S^0 = \{(-x_1, y_1, y_2) \mid 1.5 \leqslant y_1, 1.5 \leqslant y_2, -1.5 \leqslant -x_1, y_1 + y_2 - x_1 \leqslant 7.5\}$.

The vertexs set of S^0 given in Theorem (3.6) are
$\{(-1.5, 1.5, 1.5), (-4.5, 1.5, 1.5), (-1.5, 7.5, 1.5), (-1.5, 1.5, 7.5)\}$ which are computed as follows:
$q^0 = (-1.5, 1.5, 1.5), q_1^1 = 7.5 - 1.5 - 1.5 = 4.5, q_2^1 = 1.5, q_3^1 = 1.5, q_1^2 =$

$-1.5, q_2^2 = 7.5 + 1.5 - (1.5) = 7.5, q_3^2 = 1.5, q_1^3 = -1.5, q_2^3 = 1.5, q_3^3 = 7.5 +$
$1.5 - (1.5) = 7.5$, therefore $q^1 = (-4.5, 1.5, 1.5)$, $q^2 = (-1.5.7.5, 1.5)$ and
$q^3 = (-1.5.1.5, 7.5)$.

In step (1) of the algorithm, since $(-1.5.7.5, 1.5)$ does not belong to V, we put
$(-X^1, Y^1) = (-1.5.7.5, 1.5)$.

We go to step (2). If we solved problem (6) by $(-X^w, Y^w) = v^{AL} =$
$(v_1, v_2, v_3) = (-1, 2, 2)$ we would obtain $(X^*, Y^*, t^*) = (-1, 6, 2, 0)$.
We put $(-X_p, Y_p) = 0.5 (-X^*, Y^*) + 0.5 (v_1^{AL}, v_2^{AL}, v_3^{AL}) = 0.5(-1, 6, 2) + 0.5(-1, 2, 2)$,
then $(-X^p, Y^p) = (-1, 4, 2)$. Now we solved problem (4),

we put $(-X, Y) = (-X^1, Y^1) = (-1.5, 7.5, 1.5)$ and $(-X_p, Y_p) = (-1, 4, 2)$, we
would obtain $\lambda^* = 0.571$ and $(-X^w, Y^w) = (-1.286, 6, 1.715)$.
We solved problem (7) by $(-X^w, Y^w) = (-1.286, 6, 1.715)$, we would obtain
$u_1^* = 1, u_2^* = 0, u_3^* = 0, v_4^* = 6$, then the inequality cut is as follows.
$y_1 \leqslant 6$.

We go to step (3) and we organize
$S^1 = \{(-x_1, y_1, y_2) \mid 1.5 \leqslant y_1, 1.5 \leqslant y_2, -1.5 \leqslant -x_1, y_1 + y_2 - x_1 \leqslant 7.5, y_1 \leqslant 6\}$.
The vertex set of S^1 given in the Theorem (3.6) are
$\{(-1.5, 1.5, 1.5), (0, 6, 1.5), (-1.5, 1.5, 7.5)\}$. We put $k = 2$. Since $(-1.5, 1.5, 7.5)$ does not belong to V, we put $(-X^2, Y^2) = (-1.5, 1.5, 7.5)$. In the next
step, we solved problem (4), we put $(-X, Y) = (-X^2, Y^2) = (-1.5, 1.5, 7.5)$
and $(-X_p, Y_p) = (-1, 4, 2)$. We would obtain $\lambda^* = 0.471$ and $(-X^w, Y^w) = (-1.236, 2.823, 4.591)$.

We solved problem (7) by $(-X^w, Y^w) = (-1.236, 2.823, 4.591)$. We would obtain $u_1^* = \frac{1}{3}, u_2^* = \frac{2}{3}, u_3^* = 0, v_4^* = 4$, then the inequality cut is as follows.
$y_1 + 2y_2 \leqslant 12$.
We go to step (3) and we organize $S^2 = \{(-x_1, y_1, y_2) \mid 1.5 \leqslant y_1, 1.5 \leqslant y_2, -1.5 \leqslant -x_1, y_1 + y_2 - x_1 \leqslant 7.5, y_1 \leqslant 6, y_1 + 2y_2 \leqslant 12\}$.

The vertexs set of S^2 given in Theorem (3.6) are
$\{(-1.5, 1.5, 1.5), (-1.5, 6, 3), (0, 1.5, 5.25), (0, 3.5, 4.5)\}$. We put $k = 3$. Since $(-1.5, 6, 3)$ dose not belong to V, we put
$(-X^3, Y^3) = (-1.5, 6, 3)$. In the next step, we solved problem (4), we put $(-X, Y) = (-X^3, Y^3) = (-1.5, 6, 3)$ and $(-X_p, Y_p) = (-1, 4, 2)$, we would obtain $\lambda^* = 0.75$ and $(-X^w, Y^w) = (-1.375, 5.5, 2.75)$.

We solved problem (7) by $(-X^w, Y^w) = (-1.375, 5.5, 2.75)$, we would obtain
$u_1^* = 0.6, u_2^* = 0.4, u_3^* = 0, v_4^* = 4.4$, then the inequality cut is as follows.

$0.6y_1 + 0.4y_2 \leqslant 4.4$.

We go to step (3) and we organize
$S^3 = \{(-x_1, y_1, y_2) \mid 1.5 \leqslant y_1, 1.5 \leqslant y_2, -1.5 \leqslant -x_1, y_1 + y_2 - x_1 \leqslant 7.5, y_1 \leqslant 6, y_1 + 2y_2 \leqslant 12, 0.6y_1 + 0.4y_2 \leqslant 4.4\}$. The vertexes set of S^3 given in Theorem (3.6) are
$\{(-1, 5, 3.5), (-0.5, 6, 2), (0, 1.5, 5.25), (0, 3, 4.5), (-1, 5, 1.5, 1.5)\}$.
We put $k = 4$. Since $(0, 1.5, 5.25)$ does not belong to V, we put $(-X^4, Y^4) = (0, 1.5, 5.25)$. In the next step, we solved problem (4), we put $(-X^4, Y^4) = (0, 1.5, 5.25)$ and $(-X_p, Y_p) = (-1, 4, 2)$,
we would obtain $\lambda^* = 0$ and $(-X^w, Y^w) = (-1, 4, 2)$. We solved problem (7) by $(-X^w, Y^w) = (-1, 4, 2)$, we would obtain $u_1^* = 0, u_2^* = 0, u_3^* = 1, v_4^* = -1$, then the inequality cut is as follows.
$-x_1 \leqslant -1$.

We go to step (4) and we organize $S^4 = \{(-x_1, y_1, y_2) \mid 1.5 \leqslant y_1, 1.5 \leqslant y_2, -1.5 \leqslant -x_1, y_1 + y_2 - x_1 \leqslant 7.5, y_1 \leqslant 6, y_1 + 2y_2 \leqslant 12, 0.6y_1 + 0.4y_2 \leqslant 4.4, -x_1 \leqslant -1\}$. The vertexes set of S^4 given in Theorem (3.6) are
$\{(-1, 5, 3.5), (-1, 6, 2), (-1, 1.5, 5.25), (-1, 5, 1.5, 1.5)\}$.
We put $k = 5$. Since $(-1, 1.5, 5.25)$ does not belong to V, we put $(-X^5, Y^5) = (-1, 1.5, 5.25)$. In the next step, we solved problem (4), we put $(-X^5, Y^5) = (-1, 1.5, 5.25)$ and $(-X_p, Y_p) = (-1, 4, 2)$, we would obtain $\lambda^* = 0.923$ and $(-X^w, Y^w) = (-1, 1.693, 5)$.

We solved problem (7) by $(-X^w, Y^w) = (-1, 1.693, 5)$, we would obtain $u_1^* = 0, u_2^* = 1, u_3^* = 0, v_4^* = 2$, then the inequality cut is as follows.
$y_2 \leqslant 2$.

We go to step (5) and we organize
$S^5 = \{(-x_1, y_1, y_2) \mid 1.5 \leqslant y_1, 1.5 \leqslant y_2, -1.5 \leqslant -x_1, y_1 + y_2 - x_1 \leqslant 7.5, y_1 \leqslant 6, y_1 + 2y_2 \leqslant 12, 0.6y_1 + 0.4y_2 \leqslant 4.4, -x_1 \leqslant -1, y_2 \leqslant 2\}$. The vertexes set of S^5 given in Theorem (3.6) are
$\{(-1, 5, 3.5), (-1, 6, 2), (-1, 2, 5), (-1, 5, 1.5, 1.5)\}$.

Since $(-1.5, 1.5, 1.5)$ hase the same components toto
$\hat{V} = (-\hat{X}, \hat{Y}) = (-1.5, 1.5, 1.5)$, then they do not belong to $V^=$. Therefore, the vertexes set of $V_E^=$ are as follows:
$\{(-1, 5, 3.5), (-1, 6, 2), (-1, 2, 5)\}$. We obtain the vertexes set of T_v by converting $(-x_1, y_1, y_2)$ to (x_1, y_1, y_2) as follows:
$\{(1, 5, 3.5), (1, 6, 2), (1, 2, 5)\}$.

The algorithm is terminated.

6. Computational Experiment and Statisical Analysis

To conduct a preliminary computational experiment for our proposed approach, we can use the preliminary VS-Fortran code to execute the outer linear approximation algorithm, see Benson [8]. The Horst-ThoaiDe Vries method [29] is used to execute the fourth step of the algorithm; the linear bisection method is used for our univariate search in the second step, and to solve the linear programming problem, we use the simplex algorithm; as implemented by the subroutines of IMSL. [51]. Benson [8] has provided the number of iterations and efficient extreme points and the CPU introduction times for thirty multiple objective linear programming problems with different dimensions. In the present research, we use the Gams software to solve our DEA models and the Lindo software is used for solving the linear programming problems. Note that in order to determine the efficient extreme points using traditional DEA models, we need to solve at least n models, which is difficult to do; it would also be quite difficult to obtain information related to the efficient surfaces. However, in this article, we arrive at all the efficient extreme points by solving only one MOLP model, and the model is not dependent on the unit under evaluation. The m+s+n model is variable in the decision set and the number of m+s is variable in the outcome set. Now, the outcome set is smaller and we can convert the efficient extreme points in this set to efficient extreme points in the decision set through a simple calculation; thus, using the presented algorithm, we can determine the efficient extreme points of the production possibility set and its efficient surfaces by solving one model and a few iterations of the algorithm. In the numerical example provided, we use the model to evaluate seven decision-making units (DMUs) under VRS technology, each having one input and one output. In the one example, there are 9 variables in the decision set and 2 variables in the outcome set; we obtained all efficient extreme points after four iterations of the algorithm. In the third step of the algorithm, we solve a linear programming problem to find the optimal values of λ^*, and to form the cutting-plane equations, a linear programming problem is solved in each stage. As can be observed, this method involves fewer calculations for finding the extreme points compared to traditional DEA models, which require solving n models for the same purpose. For the example 2, we have similar interparation. The statisical analysis of examples discriped in Table 6.1.

Table 6.1: The statisical analysis of examples.

Example	The number of Inputs and outputs
Example 1	2
Example 2	3
Example	The number of DMUs
Example 1	7
Example 2	5
Example	The number of variables in feasible region
Example 1	9
Example 2	8
Example	The number of efficient extreme points in feasible region
Example 1	84
Example 2	70
Example	The number of efficient extreme points in outcome space
Example 1	3
Example 2	3
Example	The number of algorithm iterations
Example 1	4
Example 2	5
Example	The number of Solved LPs
Example 1	9
Example 2	12

The presented algorithm has many useful computational advantages to previous approaches for determining the efficient extreme points of the production possibility set:

1. Since the algorithm produces all efficient extreme points of the outcome set based on the decision set and the outcome set is smaller than the decision set, fewer calculations are needed for finding these points.

2. The proposed algorithm is precise and finite; thus, through solving one MOLP model and a number of iterations, we can arrive at all efficient extreme points and efficient surfaces.

3. This algorithm does not face the issues of previous algorithms in producing the efficient extreme points, such as infeasibility and degeneracy.

4. The presented approach makes a new connection between DEA and MOLP problems; in this regard, we can identify all efficient surfaces by solving one MOLP problem and multiple iterations of the algorithm.

5. The presented approach can be a new method for obtaining all efficient extreme points.

7.   Conclusion

The purpose of this paper was to develop a new method for generating efficient

extreme points of the production possibility set with VRS. We proposed an MOLP problem whose feasible region same of is production possibility set. We applied the outer approximation algorithm for generating the efficient extreme points of MOLP problem. Since the average number of efficient extreme points in the outcome set is less than the average number of efficient extreme points in the decision set, the method proposed is pretty fit. We obtain the efficient frontier by solving an MOLP problem, the outer approximation algorithm can be implemented relatively easily by using search methods, linear programming techniques, and any one of several special methods from the global optimization literature for generating new vertexes set as linear inequality cuts are added to the containing polyhedral generated by the algorithm. This algorithm used few calculations to produce all the extreme points. The results of this paper can be used on various DEA-related application problems. The results proposed a way for extending the analysis of production efficiency to further path. In this method, we obtain the efficient frontier by solving an MOLP problem. For future research, we suggest extending our presented approach to determine the units return to scale class; furthermore, we can solve the proposed MOLP problem using other methods for obtaining extreme points, such as vector maximization and make a comparison of results.

References

[1] R. Allen, A. D. Athanassopoulos, R. G. Dyson, and E. Thanassoulis, Weights restrictions and value judgements in data envelopment analysis: evolution, development and future directions, *Ann. Oper. Res.,* 73 (1997), 13-34.

[2] R. Banker, H. Chang, and W. W. Cooper, Equivalence and implementation of alternative methods for determining return to scale in data envelopment analysis, *Eur. J. Oper. Res.,* 89 (1996), 473-481.

[3] R. Banker, A. Charnes, and W. W. Cooper, Some models for estimating technical and scale inefficiencies in data envelopment analysis, *Manage. Sci.,* 30 (9) (1984), 1078-1092.

[4] H. P. Benson, A Geometrical Analysis of the Efficient Outcome Set in Multiple-Objective Convex Programs with Linear Criterion Functions, *J. Global Optim.,* 6 (1995), 231-251.

[5] H. P. Benson, Vector Maximization with Two Objective Functions, *J. Optim Theory. Appl.,* 28 (1979), 253-257.

[6] H. P. Benson, Generating the Efficient Outcome Set in Multiple Objective Linear Programs: The Bicriteria Case, *Acta Math. Vietnam,* 22 (1997), 29-51.

[7] H. P. Benson, Admissible Points of a Convex Polyhedron, *J. Optim Theory Appl.,* 38 (1982), 341-361.

[8] H. P Benson, An Outer Approximation Algorithm for Generating All Efficient Extreme Points in the Outcome Set of a Multiple Objective Linear Programming Problem, *J. Global Optim.,* 13 (1998), 1-24.

[9] H. P. Benson and S. Erjiang, Continuous Optimization A weight set decomposition algorithm for finding all efficient extreme points in the outcome set of a multiple objective linear program, *Eur. J. Oper. Res.,* 139 (2002), 26-41.

[10] H. P. Benson and S. Sayin, Towards Finding Global Representations of the Efficient Set in Multiple Objective Mathematical Programming, *NAV RES LOG.,* 44 (1997), 47-67.

[11] G. R. Bitran, Theory and Algorithms for Linear Multiple Objective Programs with Zero one Variables, *Math. Program.,* 17 (1979), 362-390.

[12] A. Charnes, W. W. Cooper, and E. Rhodes, Measuring the efficiency of decision making units, *Eur. J. Oper. Res.,* 2 (6) (1978), 429-444.

[13] J. L. Cohon, *Multiobjective Programming and Planning,* Academic Press, New York, 1978.

[14] J. P. Dauer, Analysis of the Objective Space inMultiple Objective Linear Programming, *J. Math. Anal. Appl.,* 126 (1987), 579-593.

[15] J. P. Dauer, On Degeneracy and Collapsing in the Construction of the Set of Objective Values in a Multiple Objective Linear Program, *Ann. Oper. Res.,* 47 (1993), 279-292.

[16] J. P. Dauer and R. J. Gallagher, A Combined Constraint-Space, Objective-Space Approach for Determining High-Dimensional Maximal Efficient Faces of Multiple Objective Linear Programs, *Eur. J. Oper. Res.,* 88 (1996), 368-381.

[17] J. P. Dauer and Y. H. Liu, Solving Multiple Objective Linear Programs in Objective Space, *Eur. J. Oper. Res.,* 46 (1990), 350-357.

[18] J. P. Dauer and O. A. Saleh, Constructing the Set of Efficient Objective Values in Multiple Objective Linear Programs, *Eur. J. Oper. Res.,* 46 (1990), 358-357.

[19] K. De Witte and R. Marques, Influential observations in frontier models, a robust non-oriented approach to the water sector, *Ann. Oper. Res.,* 181 (2010), 377-392.

[20] J. Doyle and R. Green, Data envelopment analysis and multiple criteria decision making, *Omega,* 21 (6) (1993), 713-715.

[21] A. Ebrahimnejad and M. Tavana, An interactive MOLP method for identifying target units in output-oriented DEA models: The NATO enlargement problem, *Measurement,* 52 (2014), 124-134.

[22] G. W. Evans, An Overview of Techniques for Solving Multiobjective Mathematical Programs, *Manage. Sci.,* 30 (1984), 1268-1282.

[23] J. E. Falk and K. L. Hoffman, A Successive Underestimating Method for Concave Minimization Problems, *Math. Oper. Res.,* 1 (1976), 251-259.

[24] R. J. Gallagher and O. A. Saleh, A Representation of an Efficiency Equivalent Polyhedron for the Objective Set of a Multiple Objective Linear Program, *Eur. J. Oper. Res.,* 80 (1995), 204-212.

[25] A. Goicoechea, D. R. Hansen, and L. Duckstein, Multiobjective Decision Analysis with Engineering and Business Applications, *John Wiley and Sons,* New York, 1982.

[26] B. Golany, An interactive MOLP procedure for the extension of DEA to effectiveness analysis, *J. Oper. Res. Soc.,* 39 (8) (1988), 725-734.

[27] F. Hosseinzadeh Lotfi, G. R. Jahanshahloo, M. R. Mozaffari, and J. Gerami, Finding DEA-efficient hyperplanes using MOLP efficient faces, *J. Comput. Appl. Math.,* 235 (2011), 1227-1231.

[28] F. Hosseinzadeh Lotfi, G. R. Jahanshaloo, M. Soltanifar, A. Ebrahimnejad, and S. M. Manosourzadeh, Relationship between MOLP and DEA based on output-orientated CCR dual model, *Expert Syst. Appl.,* 37 (6) (2010), 4331-4336.

[29] F. Hosseinzadeh Lotfi, A. A. Noora, G. R. Jahanshahloo, J. Jablonsky, M. R. Mozaffari, and J. Gerami, An MOLP based procedure for finding efficient units in DEA models, *Cent Eur J Oper Res.,* 17 (2009), 1-11.

[30] R. Horst, N. V. Thoai, and J. Devries, On Finding the New Vertices and Redundant Constraints in Cutting Plane Algorithms for Global Optimization, *Oper. Res. Lett.,* 7 (1988), 85-90.

[31] H. Isermann, The Enumeration of the Set of All Efficient Solutions for a Linear Multiple Objective Program, *Oper. Res. Q.,* 28 (1977), 711-725.

[32] G. R. Jahanshahloo, F. Hosseinzadeh Lotfi, H. Zhiani Rezai, and F. Rezai Balf, Finding strong defining hyperplanes of Production Possibility Set, *Eur. J. Oper. Res.,* 177 (2007), 42-54.

[33] G. R. Jahanshahloo, F. Hosseinzadeh Lotfi, and M. Zohrehbandian, Finding the piecewise linear frontier production function in Data Envelopment Analysis, *Appl. Math. Comput.,* 163 (1) (2005), 483-488.

[34] R. Joro, P. Korhonen, and J. Wallenius, Structural comparison of data envelopment analysis and multiple objective linear programming, *Manage. Sci.,* 44 (7) (1998), 962-970.

[35] P. Korhonen, Searching the efficient frontier in Data Envelopment Analysis, *IIASA, IR.,* (1997), 97-79.

[36] M. M. Kostreva and M. M. Wiecek, Time Dependency in Multiple Objective Dynamic Programming, *J. Math. Anal. Appl.,* 173 (1993), 289-307.

[37] H. W. Kuhn and A. W. Tucker, *Nonlinear Programming, in J. Neyman (ed.),* Proceedings of the 2nd Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, Berkeley, California, (1950), 481-492.

[38] D. T. Luc, *Theory of Vector Optimization,* Springer Verlag, Berlin/New York, 1989.

[39] N. Malekmohammadi, F. Hosseinzadeh Lotfi, and A. B. Jaafar, Target setting in data envelopment analysis using MOLP, *Appl. Math. Model.,* 35 (2011), 328-338.

[40] R. T. Rockafellar, *Convex Analysis,* pinceton University Press, Princeton, New Jersey, 49 (1970).

[41] Y. Sawaragi, H. Nakayama, and T. Tanino, *Theory of Multiobjective Optimization,* Academic Press, Orlando, Florida, 1985.

[42] S. Sayin, An algorithm based on facial decomposition for finding the efficient set in multiple objective linear programming, *Oper. Res. Lett.,* 19 (1996), 87-94.

[43] R. E. Steuer, Multiple Criteria Optimization: Theory, Computation, and Application, *John Wiley and Sons,* New York, 1986.

[44] R. E. Steuer, *ADBASE Multiple Objective Linear Programming Package,* University of Georgia, Athens, Georgia, 1989.

[45] B. Villarreal and M. H. Karwan, Multicriteria Integer Programming: A (Hybrid) Dynamic Programming Recursive Approach, *Math. Program,* 21 (1981), 204-223.

[46] B. Y. H. Wong, M. Luque, and J. B. Yang, Using interactive multiobjective methods to solve DEA problems with value judgements, *Comput. Oper. Res.,* 36 (2009), 623-636.

[47] J. B. Yang, B. Y. H. Wong, Xu. Dong-Ling, and T.J. Stewart, Integrating DEA-oriented performance assessment and target setting using interactive MOLP methods, *Eur. J. Oper. Res.,* 195 (2009), 205-222.

[48] P. L. Yu, *Multiple Criteria Decision Making,* Plenum, New York, 1985.

[49] P. L Yu and M. Zeleny, The set of all nondominated solutions in linear cases and multicriteria simplex method, *J. Math. Anal. Appl.,* 49 (1975), 430-468.

[50] M. Zeleny, *Linear Multiobjective Programming,* Springer Verlag, Berlin New York, 1974.

[51] M. Zeleny, *Multiple Criteria Decision Making,* McGraw-Hill, New York, 1982.

[52] International Mathematical and Statistical Libraries, Inc., *The IMSL Library Reference Manual,* IMSL, Houston, Texas, 1991.

Javad Gerami
Assistant Professor of Mathematics
Department of Mathematics
Shiraz Branch, Islamic Azad University
Shiraz, Iran
E-mail: geramijavad@gmail.com