Characterizing and finding full dimensional efficient facets of PPS with constant returns to scale technology

G. R. Jahanshahloo *, I. Roshdi †‡, M. Davtalab-Olyaie §

Abstract

In DEA (Data Envelopment Analysis), the Full Dimensional Efficient Facets (FDEFs) of PPS (Production Possibility Set) play a significant role and have many useful applications. In this research, we, first, provide a detailed characterization of the structure of FDEFs of the PPS with constant returns to scale technology, using basic concepts of the polyhedral sets theory. Then, using the mentioned characterization together with a mixed integer linear programming, we propose an effective algorithm for finding all of the FDEFs of the PPS. We will elaborate on our algorithm by an illustrative example.

Keywords: Data envelopment analysis; Production possibility set; Weights and multipliers; Full dimensional efficient facet; Polyhedral set; Mixed integer linear programming.

1 Introduction

DEA is a mathematical programming method for evaluating the relative efficiency of Decision Making Units (DMUs) with multiple inputs and outputs. The relative comparison in DEA is performed within a production possibility set (PPS), which is empirically constructed from the observations by assuming several postulates (see [16]) and DEA forms an empirical frontier. The efficient frontier is the main part of the PPS’s frontier. The FDEFs of the PPS form the efficient frontier, which play a significant role and have many useful applications in DEA, e.g.,

1. Uniquely identifying a reference set for a given DMU that contains the maximum number of efficient DMUs. In this way, if we consider the efficient projection of the given DMU, we can simply determine the set of all FDEFs on which this efficient projection is lying. Consequently, the reference set of DMU will be the set of the efficient DMUs that are lying on the intersection of these FDEFs.

2. Sensitive and stability analysis in DEA [8].

3. Finding the closest target for a given inefficient DMU. For this purpose as mentioned in [10, 12], one can simply measure the minimum distance between given DMU from each of the FDEFs, and at last determine the minimum value of these distances.

So far, about finding FDEFs and their structure, several papers have been written while none of them has directly and comprehensively discussed about them. Yu et al. [9] studied the structural properties of DEA efficient surfaces of the PPS under the generalized DEA model. Olsen and Petersen [14, 15] provided an outline
of possible uses of complete information on the facial structure of PPS. Also they proposed two algorithms based on the set so-called E_{j_o}. They stated that: “The only candidates for spanning an FDEF including any given DMU_{j_o}; j_o \in E, are those that can be termed efficient along with DMU_{j_o} itself. Let E_{j_o} denote an index set for this set of extreme efficient DMUs.” Here, the question is that how it is possible to determine E_{j_o} without having all of FDEFs on which DMU_{j_o} is lying. In fact, to determine E_{j_o} we need all FDEFs on which DMU is lying. Therefore, the foundation of their algorithms has this serious problem and for implementing their algorithms, a method is required to determine E_{j_o} without having FDEFs. Recently, Jahanshahloo et al. [6] proposed an algorithm for finding the “strong defining hyperplanes” of PPS. They proved that in evaluating an extreme CCR-efficient DMU the hyperplanes that are corresponding to the extreme optimal and strictly positive solutions of the multiplier form of the CCR model, are strong defining hyperplanes. Their approach has several computational difficulties that are summarized as follows:

1. They perform the algorithm for all CCR-efficient (extreme and non-extreme) DMUs, while the number of these DMUs may be a large one.

2. To employ their proposed algorithm, the multiplier form should be solved via the simplex method. Moreover, all the extreme (basic feasible) optimal solutions should be found, while there does not exist an effective method for performing this task. Furthermore, none of these solutions is necessarily strictly positive and the algorithm may be yielded some extreme (basic feasible) optimal solutions that have the zero components.

3. Through implementation of the algorithm for different DMUs, many iterated strong defining hyperplanes may be generated where their algorithm is unable to prevent this.

In this paper, first, using basic concepts of the polyhedral sets theory, we seek to provide a detailed characterization of the structure of FDEFs, and secondly, we propose an effective algorithm for finding all FDEFs of PPS. As we will demonstrate in Section 4, our algorithm is computationally better than those algorithms mentioned above.

The paper unfolds as follows: The primal and dual descriptions of the PPS with focus on the representation of the constant returns to scale technology are reported in Section 2. An applied model for identifying the extreme-efficient DMUs is presented in this section, as well. Section 3 includes the characteristics and structure of FDEFs of PPS. In Section 4, we will develop the new method for finding all of the FDEFs. An illustrative example is documented in Section 5, which intuitively describes the new algorithm. The conclusion and future directions for research are summarized in the last section.

2 Background

Consider n observed DMUs, DMU_{j}, j = 1, 2, ..., n, which use the same number, m, of inputs, x_{ij}, i = 1, 2, ..., m, to produce the same number, s, of outputs, y_{rj}, r = 1, 2, ..., s. The input and output vectors of DMU_{j} respectively are denoted by x_{j} and y_{j}; we assume that they are nonnegative and neither one is equal to zero. We use (x_{j}, y_{j}) to describe DMU_{j}, and consider DMU_{0} as the DMU under evaluation. Further, as stated in [2], “We say that a data domain is in reduced form if for no pair (j, k) with j \neq k and the scalar \( \alpha \) is \( DMU_{j} = \alpha DMU_{k} \).” We assume that the data domain is in reduced form.

Under the standard assumptions of Inclusion of observations, convexity, constant returns to scale and free disposability of inputs and outputs, the unique non-empty PPS is generated from a set of n observed DMUs, DMU_{j}, (j = 1, 2, ..., n), is as follows:

\[
T_{c} = \{(x, y) \in \mathbb{R}_{\geq 0}^{m+n} \mid x \geq \sum_{j=1}^{n} \lambda_{j} x_{j}, \ y \leq \sum_{j=1}^{n} \lambda_{j} y_{j}, \ \lambda_{j} \geq 0, \ j = 1, 2, ..., n\}.
\]

Charnes et al. [1], relative to T_{c}, introduced the following model for measuring the efficiency.
of DMU\(_o\) of DMU\(_o\):

\[
\text{Min } \theta \\
\text{s.t. } \sum_{j=1}^{n} \lambda_j x_j \leq \theta x_o, \\
\sum_{j=1}^{n} \lambda_j y_j \geq y_o, \\
\lambda_j \geq 0, \ j = 1, 2, ..., n.
\]  

This program is called the envelopment form (input-oriented) of the CCR model. DMU\(_o\) is radial efficient if and only if the optimal objective of model (2.1) (called Additive model) is equal to zero:

\[
\begin{align*}
\text{Max } & \sum_{i=1}^{m} s_i^- + \sum_{r=1}^{s} s_r^+ \\
\text{s.t. } & \sum_{j=1}^{n} \lambda_j x_{ij} + s_i^- = x_{io}, \ i = 1, 2, ..., m, \\
& \sum_{j=1}^{n} \lambda_j y_{jr} - s_r^+ = y_{ro}, \ r = 1, 2, ..., s, \\
& \lambda_j \geq 0, \ s_i^- \geq 0, \ s_r^+ \geq 0, \\
& j = 1, 2, ..., n, \ i = 1, 2, ..., m, \ r = 1, 2, ..., s.
\end{align*}
\]  

(2.2)

The dual form of model (2.1) is called the multiplier form of the CCR model and is as follows:

\[
\begin{align*}
\text{Max } & U^t y_o \\
\text{s.t. } & V^t x_o = 1, \\
& U^t y_j - V^t x_j \leq 0, \ j = 1, 2, ..., n, \\
& U \geq 0, \ V \geq 0.
\end{align*}
\]  

(2.3)

Where \( U^t = (u_1, u_2, ..., u_s) \) and \( V^t = (v_1, v_2, ..., v_m) \) are the \( s \)-vector and \( m \)-vector, respectively. It can be easily verified that, with reference to (2.3), DMU\(_o\) is CCR-efficient if and only if there exists some optimal solution, \((U^*, V^*)\), such that \((U^*, V^*) > 0\) and \(U^t y_o = 1\).

The set of all DMUs corresponding to positive \( \lambda_j^* \)'s is called the reference set to DMU\(_o\) and is denoted by \( R_o \), i.e., \( R_o = \{ \text{DMU} | \lambda_j^* > 0 \text{ in some optimal solution of (2)} \} \). Charnes et al. [2] introduced a nice classification of DMUs in CCR model. They classified the radial efficient DMUs into the categories \( E, E' \) and \( F \). Similar to them, we call the elements of \( E, E' \) and \( F \), respectively extreme CCR-efficient, non-extreme CCR-efficient and CCR weak-efficient. They also provided a procedure based on DEA computations to do the mentioned classification. To implement our algorithm, which will be presented in Section 4, we need to determine the elements of \( E \). The following model simply provides an alternative test to find all the extreme CCR-efficient DMUs without any preliminary DEA computations:

\[
\begin{align*}
\text{Max } & \gamma_o = \sum_{j=1}^{n} \lambda_j \\
\text{s.t. } & \sum_{j=1}^{n} \lambda_j x_j \leq x_o, \\
& \sum_{j=1}^{n} \lambda_j y_j \geq y_o, \\
& \lambda_j \geq 0, \ j = 1, 2, ..., n.
\end{align*}
\]  

(2.4)

**Lemma 2.1** A DMU\(_o\) is extreme CCR-efficient if and only if the optimal objective of model (2.4), \( \gamma_o^* \), is zero.

**Proof.** We first note that, by the above categorization of DMUs, DMU\(_o\) is extreme CCR-efficient if and only if the solution \( \lambda_j = 0, \ j = 1, 2, ..., n \). Then, \( \gamma_o^* = 1 \) is the unique feasible solution of the model (2.1). Now suppose that DMU\(_o\) is extreme CCR-efficient. By contradiction, if \( \gamma_o^* > 0 \), then there exists an optimal solution, \( \lambda^* \), of (2.4) such that for at least some index \( t \), \( t \neq o, \lambda_t^* > 0 \). This solution is also a feasible solution of the model (2.1) with \( \theta = 1 \). This is a contradiction. On the other hand, suppose that DMU\(_o\) is not extreme CCR-efficient. Then there exists a feasible solution \((\theta, \lambda)\) of the model (2.1), such that for at least some index \( t \), \( t \neq o, \lambda_t > 0 \). Either if \( \theta = 1 \) or \( \theta < 1 \), then the solution, \( \lambda \), is a feasible solution of (2.4), and so \( \gamma_o^* > 0 \).

### 3 Characteristics and structures of FDEFs of \( T_c \)

Let \( P \subseteq \mathbb{R}^d \) be a convex set. A linear inequality \( cx \leq c_0 \) is valid for \( P \) if it is satisfied for all \( x \in P \). A face of \( P \) is any set of the form \( F = P \cap \{ x \in \mathbb{R}^d : cx = c_0 \} \) where \( cx \leq c_0 \) is a valid inequality for \( P \). The dimension of a face is the dimension of its affine hull: \( \dim(F) = \dim(\text{aff}(F)) \). The face of dimension \( \dim(P) - 1 \) is called facet. Thus, the facets are the maximal proper faces. For DMUs with \( m \) inputs and \( s \) outputs, \( T_c \) is a convex subset of \( \mathbb{R}^{m+s} \). So the dimension of each facet of \( T_c \) is \( m+s-1 \). Therefore, each facet of \( T_c \) contains at least \( m+s \) DMUs that are affine independent.

In the evaluation of DMU\(_o\) \((o \in \{1, 2, \ldots, n\})\), if \((U^*, V^*)\) be an optimal solution of the model

\[
\text{Min } \theta \\
\text{s.t. } \sum_{j=1}^{n} \lambda_j x_j \leq x_o, \\
\sum_{j=1}^{n} \lambda_j y_j \geq y_o, \\
\lambda_j \geq 0, \ j = 1, 2, ..., n.
\]  

1A set of vectors \( \{a_1, a_2, ..., a_{k+1}\} \) of dimension \( n \) is called affine independent if \( \{a_2 - a_1, a_3 - a_1, ..., a_{k+1} - a_1\} \) is linear independent.

www.SID.ir
(2.3), then $H : U^s y - V^s x = 0$ is a supporting hyperplane of the $T_c$ (see [17]), i.e., the inequality $U^s y - V^s x \leq 0$ is valid for $T_c$. So the set

$$F = T \cap \left\{(x, y) \in \mathbb{R}^{m+s} : U^s y - V^s x = 0\right\} = H \cap T_c$$

is a face of $T_c$. If $(U^s, V^s) > 0$, then $H$ is called strong supporting and the corresponding face, $H \cap T_c$, is called strong face. Consider DMU$_0$ in Figure 1. Using model (2.3), it can be seen that there are alternative optimal solutions which define an infinite number of supporting hyperplanes passing through DMU$_0$, of which only two hyperplanes ($H_1$ and $H_2$) are strong and $F_1 = H_1 \cap T_c$, $F_2 = H_2 \cap T_c$ are the strong facets of $T_c$. We propose an algorithm for determining all of the FDEFs of $T_c$. To completely characterize the structure of the FDEF of $T_c$, we need the following definitions and preliminaries:

**Definition 3.1** Suppose that $H : U^s y - V^s x = 0$ is a supporting hyperplane of $T_c$. $F = H \cap T_c$ is called a Full Dimensional Efficient Facet (FDEF) of $T_c$, if (i) there exists at least one affine independent set with $m+s$ elements of CCR-efficient DMUs lying on $F = H \cap T_c$, and (ii) all multipliers are strictly positive, i.e. $(U^s, V^s) > 0$.

The hyperplane that satisfies the above definition is called Strong Defining Hyperplane (SDH) of $T_c$.

Suppose that $H : U^s y - V^s x = 0$ is a supporting hyperplane of $T_c$. If the DMUs (observed or virtual) $(\bar{x}, \bar{y})$ and $(\bar{x}, \bar{y})$ belong to $T_c$ and lie on $H$, then the DMU, $\mu (\bar{x}, \bar{y}) + \eta (\bar{x}, \bar{y})$, belongs to $T_c$ and lies on $H$ for any positive scalars $\mu$ and $\eta$. Therefore, the intersection of each supporting hyperplane of $T_c$ with it, is a convex polyhedral cone. Each convex polyhedral cone is completely characterized by its extreme directions$^2$. So we have the following definitions:

In a given convex set, a nonzero vector $d$ is called a (recession) direction of the set if for each $x_o$ in the set: the ray $\{x_o + \lambda d | \lambda \geq 0\}$ also belongs to the set. Hence starting at any point $x_o$ in the set, one can recede along $d$ for any step length $\lambda \geq 0$ and remain within the set. An extreme direction of a convex set is a direction of the set that cannot be represented as a positive combination of two distinct directions of the set. Two vectors, $\bar{d}$ and $\tilde{d}$, are said to be distinct or not equivalent, if $\bar{d}$ cannot be represented as a positive multiple of $d$.

The following lemma characterizes the extreme directions of the strong face, $F = H \cap T_c$, where $H$ is a strong supporting hyperplane of $T_c$.

**Lemma 3.1** Suppose that $H : U^s y - V^s x = 0$ is a strong supporting hyperplane of $T_c$. An $(m+s)-$vector, $d$, is an extreme direction of the strong face, $H \cap T_c$, if and only if it is an extreme CCR-efficient DMU lying on $H$.

**Proof.** Let DMU$_0 = (x_o, y_o)$ be an extreme CCR-efficient DMU that lies on $H$. Since the set $H \cap T_c$ is a convex polyhedral cone, the point $(\bar{x}, \bar{y}) + \lambda (x_o, y_o)$ belongs to the set $H \cap T_c$ for any point $(\bar{x}, \bar{y})$ in the set $H \cap T_c$ and any positive scalar $\lambda$. Therefore $(x_o, y_o)$ is a recession direction of the set $H \cap T_c$. By contradiction, we prove that it is also extreme. Otherwise, there exist two distinct recession directions of the set $H \cap T_c$ (i.e., two distinct points of the set $H \cap T_c$), namely, $\bar{d}$ and $\tilde{d}$, such that:

$$(x_o, y_o) = \bar{d} = \tilde{d} + \alpha \bar{d} \quad , \quad \bar{d} > 0.$$

Since $\bar{d}$ and $\tilde{d}$ belong to the $H \cap T_c$, by the structure of $T_c$, there exist nonnegative vectors $\lambda$ and $\bar{x}$ such that

$$\bar{d} = \sum_{j=1}^{n} \bar{x}_j (x_j, y_j)$$

and

$$\tilde{d} = \sum_{j=1}^{n} \tilde{x}_j (x_j, y_j) \quad , \quad \tilde{x}_j, \bar{x}_j \geq 0, \quad \text{for all} \ j.$$

$^2$For more details see [8, 13].
Hence,

\[(x_o, y_o) = \alpha \sum_{j=1}^{n} \lambda_j (x_j, y_j) + \alpha \sum_{j=1}^{n} \tilde{\lambda}_j (x_j, y_j)\]

In other words, \(\sum_{j=1}^{n} \lambda_j x_j = x_o\) and \(\sum_{j=1}^{n} \lambda_j y_j = y_o\) where \(\tilde{\lambda}_j := \alpha \lambda_j + \alpha \tilde{\lambda}_j\), \(j = 1, 2, \ldots, n\). This relations show that \((\theta = 1, \tilde{\lambda})\) is a feasible solution of model (2.1) in evaluating DMU\(_o\), in which for at least some index, \(t \neq o\), \(\tilde{\lambda}_t > 0\). This is a contradiction.

On the other hand, suppose that \(\tilde{d}\) is an extreme recession direction of the set \(H \cap T_c\). Let \(\tilde{d} = (d_x, d_y)\) where \(d_x \in \mathbb{R}^n_+\) and \(d_y \in \mathbb{R}^s_+\). Since the point \(\tilde{d}\) lies on the hyperplane \(H\), we have \(U^t d_y - V^t d_x = 0\). Since \((U, V) > 0\), without loss of generality, we can assume that the coefficient vectors \((\bar{U}, \bar{V})\) has been normalized with respect to \(\tilde{d}\), i.e., \(U^t \bar{d}_y = V^t \bar{d}_x = 1\). Therefore, \(\tilde{d}\) is CCR-efficient. We evaluate \(\tilde{d}\) by model (2.1): In each optimal solution, \(\lambda^t\), we have

\[\sum_{j \in R_d} \lambda^t_j x_j = \tilde{d}_x,\]
\[\sum_{j \in R_d} \lambda^t_j y_j = \tilde{d}_y,\]
\[\lambda^t_j \geq 0, j \in R_d\]

Since \(\tilde{d}\) belongs to \(H\), by the above relations, DMU\(_j\), \(j \in R_d\) belongs to the set \(H \cap T_c\). So each DMU\(_j\), \(j \in R_d\) is also a recession direction of the set \(H \cap T_c\). We claim that \(\tilde{d}\) is equal to exactly one DMU\(_t\), \(t \in R_d\). In other words, equivalently exactly for one index \(t \in R_d\) we have \(\lambda^t_t > 0\). Otherwise, if there is more than one index \(j \in R_d\) such that \(\lambda^t_j > 0\), then the extreme recession direction is written as a nonnegative combination of at least two distinct recession directions of \(H \cap T_c\) and this is a contradiction.

We can go further and prove the following theorem:

**Theorem 3.1** Suppose that \(H : U^t y - V^t x = 0\) is a SDH of \(T_c\), i.e., \(H \cap T_c\) is a FDEF of \(T_c\), then there exists at least one linear independent set with \(m + s - 1\) elements of extreme CCR-efficient DMUs in the set \(H \cap T_c\).

**Proof.** Since the set \(H \cap T_c\) is a FDEF of \(T_c\), there exists at least one linear independent set with \(m + s - 1\) elements of CCR-efficient DMUs in the set \(H \cap T_c\). Therefore, the set \(H \cap T_c\) is an \((m + s - 1)\)-dimensional convex polyhedral cone. Suppose that \(D = \{\text{DMU}_1, \ldots, \text{DMU}_k\}\) is the set of all extreme recession directions of \(H \cap T_c\). So \(H \cap T_c\) is equal to all nonnegative combinations of the elements of the set \(D\), i.e., \(H \cap T_c = \text{Pos}(D) := \{\sum_{j=1}^{k} \lambda_j \text{DMU}_j | \lambda_j \geq 0, j = 1, \ldots, k\}\). By Lemma 3.1, each DMU\(_j\) is an extreme CCR-efficient DMU. It is clear that \(k \geq m + s - 1\). Since the set \(H \cap T_c\) is an \((m + s - 1)\)-dimensional convex polyhedral cone, there is some linearly independent \((m + s - 1)\)-subset of \(D\), and the result is in hand.

Here we open a question: Suppose that \(H\) is an SDH of \(T_c\). Is the set \(H \cap T_c\) a \((m + s - 1)\)-simplicial cone? i.e., does exist exactly one linear independent set with \(m + s - 1\) elements of the extreme CCR-efficient DMUs lying on \(H\)? The answer is negative. The following counterexample illustrates this fact (see Figure 2).

**Counterexample**

Consider four DMUs are given in Table 1. Units A, B, C and D in Table 1 use two inputs to produce two outputs. By using model (2.4), we can verify that all these DMUs are extreme CCR-efficient. Moreover, all of them lie on the SDH \(H : 16y_1 + 9y_2 - 7x_1 - 7x_2 = 0\). The intersection of \(H\) and the PPS constructed by these DMUs is a 3-dimensional FDEF of PPS. As depicted in Figure 2, indeed, these DMUs are the four extreme directions of this FDEF. This figure visually describes a section at a given output level, say \(y_2 = 1\).

![Figure 2: Counterexample](www.SID.ir)
Table 1: Data of counterexample

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>x₁</td>
<td>4</td>
<td>11</td>
<td>17</td>
<td>6</td>
</tr>
<tr>
<td>x₂</td>
<td>11</td>
<td>4</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>y₁</td>
<td>6</td>
<td>6</td>
<td>9.5</td>
<td>9.5</td>
</tr>
<tr>
<td>y₂</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Formulation for Identifying a FDEF:** Suppose that \( H : U^*y - V^*x = 0 \) is a strong supporting hyperplane of \( T_c \) passing through \( DMU_o \). As we mentioned above, the set \( H \cap T_c \) is a convex polyhedral cone that is generated by its extreme recession directions. Let \( D = \{ D_1, ..., D_k \} \) be the set of all extreme recession directions of the face \( H \cap T_c \), then \( H \cap T_c = Pos(D) := \{ \sum_{j=1}^{k} \lambda_j DMU_j | \lambda_j \geq 0, j = 1, ..., k \} \). Thus, for finding a SDH passing through \( DMU_o \), we should find a strong supporting hyperplane as \( H : U^*y - V^*x = 0 \) that it is passing through maximum number of extreme CCR-efficient DMUs.

To do this, we use the following mixed integer linear programming problem, presented in Cooper et al. [18] with small changes:

\[
\begin{aligned}
\text{Min} & \quad I_o = \sum_{j \in E} I_j \\
\text{s.t.} & \quad V^t x_o = 1, \\
& \quad U^t y_j - V^t x_j + t_j = 0, j \in E, \\
& \quad t_j - M I_j \leq 0, j \in E, \\
& \quad I_j \in \{0,1\}, j \in E, \\
& \quad t_j \geq 0, j \in E, \\
& \quad U \geq 0, V \geq 0,
\end{aligned}
\tag{3.5}
\]

where the set \( E \) is the set of all extreme CCR-efficient observed units and \( M \) is a sufficiently large positive quantity.

Note that \( I_j = 0 \) if and only if \( t_j = 0 \), i.e., \( DMU_j \) belongs to the hyperplane \( H : U^*y - V^*x + u = 0 \). Then, since we are minimizing \( \sum_{j \in E} I_j \) and \( I_j \in \{0,1\} \), model (3.5) will be directed toward finding optimal solutions with as many \( I_j^* = 0 \) as possible, i.e., with as many possible \( t_j^* = 0 \); Equivalently with as many possible extreme recession directions which \( H \cap T_c \) has.

**Theorem 3.2** Suppose that \( DMU_o \) is an extreme CCR-efficient DMU. If there exists at least one SDH passing through \( DMU_o \) and \((U^*, V^*)\) is an optimal solution of model (3.5) in which \((U^*, V^*) > 0\), then \( H_o : U^*y - V^*x = 0 \) is a SDH of \( T_c \).

**Proof.** Suppose that \((U^*, V^*)\) is an optimal solution of model (3.5) in which \((U^*, V^*) > 0\). Since there exists at least one SDH passing through \( DMU_o \), by Theorem 3.1, we have \( I_o^* = |E| - k \leq |E| - (m + s - 1) \). Consider the following model:

\[
\begin{aligned}
\text{Max} & \quad U^t y_o \\
\text{s.t.} & \quad V^t x_o = 1, \\
& \quad U^t y_j - V^t x_j \leq 0, j \in E, \\
& \quad U \geq 0, V \geq 0.
\end{aligned}
\tag{3.6}
\]

In fact model (3.6) is the multiplier form with constraint restricted to \( j \in E \). It is apparent that \((U^*, V^*)\) is an optimal solution of model (3.6). There are two cases:

Case (I), \((U^*, V^*)\) is an extreme (basic feasible) optimal solution of model (3.6). Then, because \((U^*, V^*) > 0\), there exist \( m + s \) linearly independent constraints of \( U^t y_j - V^t x_j \leq 0, j \in E \) binding at \((U^*, V^*)\). Suppose that \( U^t y_j - V^t x_j = 0, i = 1, ..., m + s \) are these linearly independent constraints binding at \((U^*, V^*)\). Therefore, the following matrix is row full rank:

\[
\begin{pmatrix}
-x_{j1} & y_{j1} \\
-x_{j2} & y_{j2} \\
\vdots & \vdots \\
-x_{jm+s} & y_{jm+s}
\end{pmatrix}
\]

and it is row equivalent with the following matrix:

\[
\begin{pmatrix}
-x_{j1} & y_{j1} \\
x_{j1} - x_{j2} & y_{j2} - y_{j1} \\
\vdots & \vdots \\
x_{j1} - x_{jm+s} & y_{jm+s} - y_{j1}
\end{pmatrix}
\]

so the set \( \{DMU_j \mid DMU_{j_i}, i = 2, \ldots, m+s \} \) is linear independent. Hence there exist \( m + s \) affinely independent of extreme CCR-efficient DMUs lying on \( H^* \cap T_c \). Therefore, by

**Definition 3.2** \( H^* \cap T_c \) is a FDEF of \( T_c \).

Case (II), \((U^*, V^*)\) is not an extreme (basic feasible) optimal solution of model (3.6).
We prove that this cannot take place. Suppose that \((\overline{U}^i, \overline{V}^i), \ldots, (\overline{U}^h, \overline{V}^h)\) are the gradients of all the SDH passing through \(DMU_o\), where \((\overline{U}^i, \overline{V}^i) > 0, \ i = 1, \ldots, h\) and \((\hat{U}^i, \hat{V}^i), \ldots, (\hat{U}^l, \hat{V}^l)\) are the gradients of all the weak defining hyperplane passing through \(DMU_o\). It is clear that \((U^i, V^i) > 0, \ i = 1, \ldots, h\) and \((\hat{U}^i, \hat{V}^i), \ i = 1, \ldots, l\) are all the extreme optimal solutions (Basic optimal feasible) of model (3.6). Since \((U^*, V^*)\) is not an extreme optimal solution of model (3.6), it can be represented as a convex combination of vectors \((\overline{U}^i, \overline{V}^i) > 0, \ i = 1, \ldots, h\) and \((\hat{U}^i, \hat{V}^i), \ i = 1, \ldots, l\). In other words:

\[
(U^*, V^*) = \sum_{i=1}^{h} \overline{\lambda}_i (U^i, \overline{V}^i) + \sum_{i=1}^{l} \hat{\lambda}_i (\hat{U}^i, \hat{V}^i)
\]

\[
\sum_{i=1}^{h} \overline{\lambda}_i + \sum_{i=1}^{l} \hat{\lambda}_i = 1, \ \overline{\lambda}_i \geq 0,
\]

\[
i = 1, \ldots, h, \hat{\lambda}_i \geq 0, \ i = 1, \ldots, l.
\]

(3.7)

There are two cases:

Case (I). There exists such a combination as (3.7) in which for some index \(r \in \{1, \ldots, h\}, \overline{\lambda}_r \neq 0\). Then, all of the extreme CCR-efficient DMUs lying on \(H^*_r\) are also lying on \(H^*: U^r y - V^r x = 0\) and these DMUs are the only extreme efficient DMUs that are lying on \(H^*\). Because if there exists another extreme CCR-efficient DMU lying on \(H^*\), in addition to these extreme CCR-efficient DMUs lying on \(H^*_r\), then \((\overline{U}^r, \overline{V}^r)\) is a feasible solution to model (3.8) where its objective value is less than \(I^*_o\), and this is a contradiction.

Case (II). There is no combination as (3.7) in which \(\overline{\lambda}_i > 0\) for any index \(i \in \{1, \ldots, h\}\). In other words, in any combination such as (3.7), \(\overline{\lambda}_i = 0\) for all indices \(i\). Therefore, the strong face, \(H^*_o \cap T_c\), is not contained in any FDEF of \(T_c\). Again, this is a contradiction.

Thus, \((U^*, V^*)\) is an optimal extreme solution of model (3.6) and so is an extreme optimal solution of model (2.3).

**Conclusion 1.** Suppose that \(DMU_o\) is an extreme CCR-efficient DMU and the vector \((U^*, V^*) > 0\) is an optimal solution of model (3.5), then it is an extreme (basic feasible) optimal solution of model (2.3) via the simplex method.

### 4 The proposed algorithm for finding all SDHs of \(T_c\)

In this section, using the characterization of structure of FDEFs that is completed in the above exploration, we propose an algorithm for finding all of the FDEFs of \(T_c\).

#### 4.1 The proposed algorithm

Our algorithm performs the following procedure for each extreme CCR-efficient DMU in each stage. **Main procedure.** Consider extreme CCR-efficient observed unit, \(DMU_o\), and evaluate it by model (3.5). Recalling Theorem 3.2, if there exists at least one FDEF containing \(DMU_o\) (equivalently if there exists at least one SDH passing through \(DMU_o\)), then the optimal solution of model (3.5) will be the gradient of a SDH passing through \(DMU_o\) and it is positive for variables \(U\) and \(V\). If the optimal solution of model (3.5) is not positive for variables \(U\) and \(V\), then the procedure will be terminated for \(DMU_o\). The procedure implements the following step till the optimal solution of model (3.5) is positive for variables \(U\) and \(V\).

**Main step.** Suppose that \(I^*_o = |E| - k\) and \((U^*, V^*) > 0\) respectively are the optimal objective and optimal solution of model (3.5). Let \(H^*_o = U^* y - V^* x = 0\) and \(F_o = H^*_o \cap T_c\). We save \(H^*_o\) as a SDH of \(T_c\) and set \(J_o = \{j : I^*_j = 0\}\). In fact, the set \(J_o\) is the indices of all extreme CCR-efficient DMUs lying on \(H^*_o\). Next, we add the following constraint to the constraints of model (3.5):

\[
\sum_{j \in J_o} |I_j| - \sum_{j \in J_o} |I_j| \leq I^*_o - 1
\]

and again we evaluate \(DMU_o\) by model (3.5). If there exists another SDH except \(H^*_o\) passing through \(DMU_o\), then Theorem 3.2 together with new added constraint, (4.8), will give the gradient of alternative SDH passing through \(DMU_o\) as an alternative optimal solution. We save this SDH and construct the set \(J_o\), which are corresponding to it.

If there does not exist another SDH except \(H^*_o\) passing through \(DMU_o\), then the procedure will be terminated for \(DMU_o\).
Suppose that the implementation of the procedure is repeated $t$ steps for DMU$_o$. Therefore, $t$ SDHs are determined. Note that in final step, model (3.5) will have exactly $t$ new added constraints corresponding to $t$ SDHs that have been determined in the previous steps. Therefore, after the implementation of the procedure for DMU$_o$, all the SDHs of $T_c$ passing through it, and all the extreme CCR-efficient DMUs lying on these hyperplanes will be determined.

After termination of the procedure for DMU$_o$, the main procedure is performed for another extreme CCR-efficient DMU$^3$; In order to prevent the algorithm from giving the gradients of iterated SDHs that have been determined in the implementation of the algorithm for DMU$_o$, the constraint $I_o = 1$ always must be added to the constraints of model (3.5) in all subsequent stages.

In general, in the implementation of the main procedure for the $r$th extreme CCR-efficient DMU, the constraints $I_j = 1$, $j = 1, ..., r - 1$ corresponding to DMU$_j$, $j = 1, ..., r - 1$, that the algorithm has been implemented for them up to now, must be added to the constraints of model (3.5).

Note that, at the end of any stage, if the number of remaining extreme CCR-efficient DMUs, which the algorithm has not been implemented for them, is less than $m + s$, then the algorithm will be automatically terminated.$^4$

By considering the structure of the algorithm, the following theorem, Theorem 3.2, guarantees that the algorithm will give the gradients of all SDHs of $T_c$ before termination.

**Lemma 4.1** Suppose that DMU$_p$ and DMU$_q$ are two extreme CCR-efficient DMUs lying on two distinct SDHs namely $H_p$ and $H_q$ (exclude their intersection). Then each strict convex combination of DMU$_p$ and DMU$_q$ is strong efficient, if it is not radial inefficient.

**Proof.** Let DMU$_l = \lambda$DMU$_p + (1 - \lambda)$DMU$_q$ where $0 < \lambda < 1$. Suppose that DMU$_l$ is not radial inefficient, then it is not an interior point of $T_c$, so it lies on frontier of $T_c$. If this frontier is strong, we are done. Otherwise it lies on weak frontier. Since DMU$_p$ and DMU$_q$ also lie on this frontier, and they belong to the reference set of DMU$_l$. By Theorem 3.4 in [17]- each nonnegative combination of the elements of reference set is strong efficient- DMU$_l$ is strong efficient. This is a contradiction.$^5$

**Theorem 4.1** In the implementation of the mentioned algorithm for DMU$_o$ in set $E$, while there exists a SDH passing through DMU$_o$, the optimal solution of model (3.5) for variables $U$ and $V$ will be positive.

**Proof.** By virtue of the type of the added constraints through the implementation of algorithm, it is sufficient to prove that the optimal solution of model (3.5) for variables $U$ and $V$ will be positive. By contradiction suppose that the optimal solution of model (3.5), $\{U^*, V^*\}$, is not positive and $I^*_o = |E| - k$. Since there exists at least one SDH passing through DMU$_o$, $k \geq m + s - 1$. Let $R = \{DMU_1, ..., DMU_k\}$ be the set of all extreme CCR-efficient DMUs lying on $H_o^*$: $U^*y - V^*x = 0$. Since $H_o^*$ is a weak defining hyperplane, so each DMU$_j (j = 1, ..., k)$ lies on the intersection $H_o^*$ with at least one SDHs of $T_c$. Suppose that $H_1, ..., H_l$ are all these SDHs. There are two cases:

1. Case (I). There exists some index $t(t \in \{1, ..., l\})$ such that all DMU$_j (j = 1, ..., k)$ lie on $H_t$. Then by considering the optimal value of objective function, there does not exist another extreme CCR-efficient DMU lying on $H_t$. Since $H_t$ is a SDH passing through DMU$_o$, the set $R$ is affinely independent. This shows that there are two distinct $m + s - 1$ dimensional hyperplanes -$H_o^*$ and $H_t^*$ passing through the set $R$. This is a contradiction.

2. Case (II). There exists at least two DMUs in set $R$, namely DMU$_p$ and DMU$_q$ lying on the intersection of $H_o^*$ with two distinct SDHs of $T_c$ namely $H_p$ and $H_q$, respectively. Since each convex combination of DMU$_p$ and DMU$_q$ lies on $H_o^*$, then it is weak efficient. On the other hand by lemma 4.1 it is strong efficient. This is a contradiction.$^6$

### 4.2. Computational advantages of the algorithm

As mentioned in Section 1, the algorithms proposed by Olsen and Petersen [14] have a serious fundamental problem that the set $E_i$, should be
determined for each DMU before implementation of their algorithms while it is not an easy task without having all of the FDEFs of the PPS. Our proposed algorithm, in comparison with the algorithm presented by Jahanshahloo et al. [6], has several advantages that are summarized as follows:

1. For identifying all of the FDEF of $T_c$, in contrast with the algorithm presented in [6] which it should be implemented for all CCR-efficient (extreme and non-extreme) DMUs, our algorithm is just implemented for extreme CCR-efficient DMUs.

2. Adding the constraint $I_p = 1$ to the model (3.5) constraints makes the algorithm able to prevent from generating iterated hyperplanes that have been obtained through implementation of the procedure for DMU, and this progressively decreases the volume of computations stage by stage.

3. As mentioned in Conclusion 1, in the implementation of the procedure for DMU, in each step the procedure gives an extreme (basic feasible) optimal solution of model (3.5). Here, unlike the algorithm presented in [6], in which all the extreme optimal solutions (positive and nonegative) of model (3.5) must be found, the structure of our algorithm is such that, firstly, all the positive extreme optimal solutions of model (3.5) (corresponding to all of the SDHs that DMU lies on them) is obtained and then it automatically terminates for DMU.

4.3. Summary of the algorithm for identifying all of the SDHs of $T_c$

Suppose that we have n DMUs, DMU, $j = 1, 2, ..., n$, with input vector $x_j$ and output vector $y_j$. We evaluate these DMUs by model (2.4). Then, we put the extreme CCR-efficient DMUs in set E, i.e., $E := \{ \text{DMU}_j | \text{DMU}_j \text{ is extreme CCR-efficient DMU} \}$.

Set $E_T = \phi$ and $S = \phi$.

**Step 1.** Put $\text{DMU}_p \in E - E_T$ and set $S_p = \phi$.

**Step 2.** Evaluate $\text{DMU}_p$ with model (3.5). If the optimal solution, $(U^*, V^*)$, is not positive, then set $E_T := E_T \cup \{ \text{DMU}_p \}$, and go to Step 5.

**Step 3.** For the solution $(U^*, V^*)$, set $J_p = \{ j : I_p^* = 0 \}$ and $J_p^* = \{ j : I_p^* = 1 \}$. If $|J_p| < m + s - 1$, then set $E_T := E_T \cup \{ \text{DMU}_p \}$, and go to Step 5; Otherwise put the hyperplane $H_p : U^*y - V^*x = 0$ into set $S_p$. Set $S := S \cup S_p$.

**Step 4.** Construct the inequality $\sum_{j \in J_p} |I_j| - \sum_{j \in J_p^*} |I_j| \leq I_p^* - 1$, and add it to the constraints of model (3.5) and go to Step 2.

**Step 5.** Add the constraint $I_p = 1$ to the constraints of model (3.5) and If $|E_T| > |E| - (m + s - 1)$ go to Step 1, otherwise the algorithm is terminated and stop.

5. Illustrative example

Consider four DMUs, A, B, C and D are given in Table 1 that use two inputs to produce one output. The PPS $T_c$ constructed by these DMUs are shown in Figure 3. Clearly all units are extreme CCR-efficient except unit D. Therefore, $E_T = \{ A, B, C \}$. To give a detailed description of the algorithm, we implement it stage by stage:

**Stage 1:**

Let $E_T = \phi$ and put $A \in E$.

**Step 1-1.** Evaluate unit A by model (3.5). $I_A^* = 1$ and $(v_1^*, v_2^*, u^*) = (0.3, 0.6, 1)$ are respectively the optimal objective and the optimal solution of model (3.5). Since $(v_1^*, v_2^*, u^*) > 0$, $H_A : y - 0.3x_1 - 0.6x_2 = 0$ is the SDH of $T_c$; So $S = \{ H_A \}$.

Furthermore, $t_B^* = 0$, that is, $H_A$ passes through B, so set $J_A = \{ A, B \}$, $J_A^* = \{ C \}$ and construct the inequality $\sum_{j \in J_A} |I_j| - \sum_{j \in J_A^*} |I_j| \leq I_A^* - 1 = 0$ (a1).

**Step 1-2.** Add the constraint (a1) to the constraints of model (3.5) and again evaluate A by

Figure 3: Illustrative example
model (3.5). We have \( I_A^* = 2 \), so the cardinal of the new set \( J_A \) for A is less than \( m + s - 1 = 2 \) therefore the algorithm terminates for A. Hence \( E_T = \{ A \} \) and the set \( S \) doesn’t change.

Add the constraint \( I_A = 1 \) to the constraints of model (3.5) and perform the following stages:

**Stage 2:**

**Step 2-1.** Put unit \( B \in E - E_T \) and evaluate it by model (3.5). \( I_B^* = 1 \) and \((v^*_1, v^*_2, u^*) = (0.125, 0.375, 1)\) are respectively the optimal objective and the optimal solution of model (3.5). Since \((v^*_1, v^*_2, u^*) > 0\), \( H_B : y - 0.125x_1 - 0.375x_2 = 0 \) is the SDH of \( T_c \); So \( S = \{ H_A, H_B \} \). Furthermore, \( t^*_c = 0 \), that is, \( H_B \) passes through C, so set \( J_B = \{ B, C \} \), \( J_B^* = \{ A \} \) and construct the inequality \( \sum_{j \in J_B^*} |I_j| - \sum_{j \in J_B} |I_j| \leq I_B^* - 1 = 0 \) (b1).

**Step 2-2.** Add the constraint (b1) to the constraints of model (3.5) and again evaluate B by model (3.5). We have \( I_B^* = 2 \), so the cardinal of the new set \( J_B \) for B is less than \( m + s - 1 = 2 \), therefore the algorithm terminates for B. Hence \( E_T = \{ A, B \} \) and set \( S \) doesn’t change.

**Stage 3:** Since \(|E_T| > |E| - (m + s - 1) = 1\), the algorithm is totally terminated.

### 6 Conclusion

In this paper, a detailed characterization of FDEFs of \( T_c \) has been provided. We have demonstrated that each FDEF of \( T_c \) is a convex polyhedral cone which is generated by extreme CCR-efficient DMUs lying on it. In addition, we have proved that the model in Cooper et al. [18] can take part in finding FDEFs. Using this information, we have proposed an algorithm for identifying all FDEFs of \( T_c \). Furthermore, via the implementation of our algorithm, the extreme (basic feasible) optimal solutions of model (2.3) will be automatically generated. As discussed in Section 4, our algorithm is computationally better than those proposed in [6, 14]. FDEFs may be used in sensitivity and stability analysis, identifying the reference set of a DMU, incorporating performance information into the efficient frontier analysis and finding the closest target for a given inefficient DMU. Moreover, in the construction of the cross-efficiency matrix, the gradient of the SDH is the best weight for the CCR-efficient DMUs that lie on it and also for the inefficient DMUs that are projected on it.

### References


Golam Reza Jahanshahloo is a Professor of Applied Mathematics at Kharazmi (Tarbiat Moallem) University, Tehran, Iran. His research interests include linear programming, multi-objective linear programming problems, and data envelopment analysis. He has supervised many M.Sc and Ph.D students in these areas, and has published many papers in international journals.

Israfil Roshdi was born in the Iran, East Azerbaijan in 1982. He has got M.Sc degree in Applied Mathematics, Operations Research field in 2007 from Kharazmi (Tarbiat Moallem) University as the top student. He is now a PhD student of Applied Mathematics at Science and Research Branch, Islamic Azad University. He is working on linear programming, data envelopment analysis, and multi-objective linear programming problems.

Mostafa Davtalab-olyaie was born in the Iran, Qom in 1983. He has got M.Sc degree in Applied Mathematics, Operations Research field in 2007 from Kharazmi (Tarbiat Moallem) University. He is now a PhD student of Applied Mathematics at Kharazmi University. He is working on linear programming and data envelopment analysis.