Journal of Mathematical Extension
Vol. 16, No. 10, (2022) (9)1-27
URL: https://doi.org/10.30495/JME.2022.2186
ISSN: 1735-8299
Original Research Paper

Finding All the Strong and Weak Defining Hyperplanes of PPS Without Solving any LPs

S. Oskoueian
Mashhad Branch, Islamic Azad University

H. Zhiani Rezai *
Mashhad Branch, Islamic Azad University

Abstract. The production possibility set (PPS) is defined as the set of all inputs and outputs of a system in which inputs can produce outputs. The frontier the production possibility set can be partitioned to strong defining hyperplanes and weak defining hyperplanes. These hyperplanes are useful in sensitivity and stability analysis, identifying the status of returns to scale of a DMU, incorporating performance into the efficient frontier analysis, and so on. In this paper, by using the basic concepts of Linear Algebra, we propose an algorithm for finding all strong and weak defining hyperplanes of PPS without solving any linear programming problems. The proposed method is applicable to both, PPS under constant and variable returns-to-scale assumptions. Two numerical examples are presented to explain the usage and effectiveness of the proposed algorithm. Our method can be easily implemented using existing packages for mathematical algorithm, such as python.

AMS Subject Classification: MSC code1; MSC code 2; more
Keywords and Phrases: Data envelopment analysis, Production possibility set, Strong defining hyperplane, Weak defining hyperplane

Received: September 2021; Accepted: January 2022
* Corresponding Author

1 Introduction

Data Envelopment Analysis (DEA), developed by Charnes et al. [5], is the most widely used method for estimating production frontier and evaluating the efficiency of Decision Making Units (DMUs). The Production Possibility Set (PPS) is a convex polyhedral set with a portion of its boundary constituting the efficient frontier. The frontier consists of two types of defining hyperplanes, strong and weak defining hyperplanes. The defining hyperplanes determine the nature of returns to scale and also is important for defining a suitable pattern for inefficient DMUs. Thus, devising a method for identifying all the strong and weak defining hyperplanes becomes a significant issue. Over the past years, many papers have been written on the subject of finding the efficient frontier. Jahanshahloo et al. [12] provided an algorithm for finding all the strong defining hyperplanes of the PPS. Their method is based on the identification of all the coplanar strong efficient DMUs and applicable to both DEA models under constant and variable returns-to-scale assumptions. Jahanshahloo et al. [11] proved that the hyperplane which corresponds to an extreme optimal solution of the multiplier form (in evaluating an efficient DMU) and whose components corresponding to inputs and outputs are nonzero is a strong defining hyperplanes of the PPS. Jahanshahloo et al. [13] provided a method to obtain efficient frontier by using 0-1 integer programming. Sueyoshi et al. [16] proposed a linear programming problem to identify all efficient DMUs that consist of a reference set; meanwhile, under special conditions optimal solutions of this model can be considered as the gradient of the defining hyperplane. Wei et al. [17, 18] investigated the properties of a K-cone and studied the problem of constructing all the “DEA-efficient surface” of the production possibility set under generalized DEA model. By introducing a variety of the supper-efficient model, Jahanshahloo et al. [9] proposed a method for finding all the weak defining hyperplanes of PPS with variable returns-to scale. By using a MOLP problem Jahanshahloo et al. [10] provided an algorithm to find the gradient of efficient hyperplanes which characterizes the efficient face. Ghazi et al. [8] proposed an algorithm to generate the strong defining hyperplanes for the PPS with the VRS technology. Their algorithm is based on a MOLP problem in DEA. Amirteimoori et al. [1] presented a method for generating all linearly independent strong defining hyperplanes (LISDHs) of the PPS passing through a specific DMU. Olesen and Petersen. [14] studied the characteristics of the production possibility set and discussed the utilization of given surface structure information.

But, in all of the above papers linear programming problems has been used for finding defining hyperplanes. It seems using linear programing models and theory of DEA is the most common method of finding hyperplanes of PPS. By using these methods, hyperplanes cannot be find easily. A few exceptions are Dula et al. [7] and Ranjbar et al. [15]. The procedures in [7] are in a category of preprocessors, uncover efficient DMUs through using translation and rotating hyperplanes. These two preprocessors, without using LPs and under special conditions, yield some strong defining hyperplanes, but not all of them. Ranjbar et al. [15] provided an algorithm for finding defining hyperplanes of variable returns- to- scale technology by using the basic concepts of Linear Algebra.

In this paper, an algorithm is developed by using the method proposed by Jahanshahloo et al. [12] so that the strong and weak hyperplanes are found without solving any linear programming problems. The proposed algorithm is simple and quick to solve.

The paper is organized as follows: In section 2, we present the definitions and concepts in DEA. Section 3, illustrates some characteristics of the hyperplanes of PPS. In the next section, the proposed method which finds the strong defining hyperplanes is introduced. Section 5 deals with the introduction of the proposed method for finding the weak defining hyperplanes. Section 6 contains the summary of the suggested algorithm for finding defining hyperplanes. In section 7, some numerical examples are provided. The comparison between the method presented in this paper and the previous approaches presented in this field provided in section 8 and finally a conclusion is summarized in the last section. All the proofs are given in Appendix A. In Appendix B, the mathematical interpretations of Theorems are illustrated through an example.

2 Mathematical Preliminary

In DEA, it is assumed that there are n DMUs such that each $DMU_j(j = 1, ..., n)$ uses a column vector of inputs $(\mathbf{x_j})$ to produce a column vector of outputs $(\mathbf{y_j})$, where $\mathbf{x}_j = (x_{1j}, ..., x_{mj})^t$, $\mathbf{y}_j = (y_{1j}, ..., y_{sj})^t$. It is also assumed that $\mathbf{x}_j \geqslant \mathbf{0}, \mathbf{y}_j \geqslant \mathbf{0}, \mathbf{x}_j \neq \mathbf{0}$, and $\mathbf{y}_j \neq \mathbf{0}$ for every $j = 1, ..., n$. Furthermore, we name $\mathbf{z}_j \equiv (\mathbf{x_j}, \mathbf{y_j})$. The production possibility set (PPS) is defined as follows:

$$T = \{(\mathbf{x}, \mathbf{y}) \,|\mathbf{y} \text{ can be produced by } \mathbf{x}\} .$$

T is characterized by the five postulates (see [6]).

T_c is built on the assumption of constant return to scale (see [5]).

$$T_c = \left\{ (\mathbf{x}, \mathbf{y}) \,|\mathbf{x} \geqslant \sum_{j=1}^{n} \mathbf{x}_j \lambda_j, \mathbf{y} \leqslant \sum_{j=1}^{n} \mathbf{y}_j \lambda_j, \lambda_j \geqslant 0, j = 1, ..., n \right\} .$$

T_v is built on the assumption of variable return to scale (see [3]). It can be defined as follows:

$$T_v = \left\{ (\mathbf{x}, \mathbf{y}) \,|\mathbf{x} \geqslant \sum_{j=1}^{n} \mathbf{x}_j \lambda_j, \mathbf{y} \leqslant \sum_{j=1}^{n} \mathbf{y}_j \lambda_j, \sum_{j=1}^{n} \lambda_j = 1 \ , \quad \lambda_j \geqslant 0, j = 1, ..., n. \right\}$$

T_c and T_v are the production possibility sets of CCR and BCC models respectively. For evaluating $DMU_k \, (k \in \{1, ..., n\})$, we can use the multiplier form of BCC model that is formulated as follows:

$$\begin{array}{ll} \text{Max} & \boldsymbol{u}\boldsymbol{y}_k + u_0 \\ \text{s.t} & \boldsymbol{u}\boldsymbol{y}_j - \boldsymbol{v}\boldsymbol{x}_j + u_0 \leq 0 \quad j = 1, \dots, n \\ & \boldsymbol{v}\boldsymbol{x}_k = 1 \\ & \boldsymbol{u} \geq 0, \boldsymbol{v} \geq 0 \\ & u_0 \quad \text{fiee} \end{array} \tag{1}$$

where $\mathbf{u} = (u_1, ..., u_s)$ and $\mathbf{v} = (v_1, ..., v_m)$ are s -vector and m -vector respectively. Suppose (u^*, v^*, u_0^*) is the optimal solution of model (1). The following definitions introduce strong and weak efficient DMUs.

***Definition* 2.1.** DMU_k is called strong efficient, if $u^*y_k + u_0^* = 1$ and there exists at least one optimal solution (u^*, v^*), with $u^* > 0$, $v^* > 0$.

***Definition* 2.2.** DMU_k is called weak efficient, if $u^*y_k + u_0^* = 1$ and at least one element of (u^*, v^*) is zero for every optimal solution of (1).

Note. If u_0 is omitted from (1), the CCR model is obtained. The definition of efficiency is the same (see [2]).

***Definition* 2.3.** DMU_k is called non-dominated, if and only if there is no $\left(\mathbf{x}_j^t, \mathbf{y}_j^t\right) \in PPS$ such that

$$\begin{pmatrix} -\mathbf{x}_j \\ \mathbf{y}_j \end{pmatrix} \geqslant \begin{pmatrix} -\mathbf{x}_k \\ \mathbf{y}_k \end{pmatrix} \; and \; \begin{pmatrix} -\mathbf{x}_j \\ \mathbf{y}_j \end{pmatrix} \neq \begin{pmatrix} -\mathbf{x}_k \\ \mathbf{y}_k \end{pmatrix}.$$

Otherwise, we say DMU_j dominates DMU_k.

3 Foundation of Hyperplanes in DEA

A production possibility set is enveloped by hyperplanes which are forming the efficient frontier. For DMUs with m inputs and s outputs, the PPS is a subset of $\mathbb{R}^{m+s}$. Since the set of strong[3] defining hyperplanes of the PPS is unique (it has been assumed that the redundant hyperplanes are omitted), each strong defining hyperplane of PPS passes through at least $m + s$ DMUs which are affine independent[4]. Otherwise, the hyperplane is not unique. As an example, consider a set of six DMUs in $\mathbb{R}^{2+1}$ (Table 1 and Fig 1). A, D, G lie on the same line; hence, the hyperplane passing through them is not unique. since A, D, E, G are affine independent, therefore only one hyperplane passes through them.

Remark 3.1. The results of this section are based on the following geometric fact:

[3] Hyperplane $\{x; p'x = a, p \geq 0\}$ is strong if none of components of the p are zero and it is weak if some of the components of p are zero.

[4] A collection of vectors $a_1, \ldots, a_{k+1}$ of dimension n is called affine independent if $\{a_2 - a_1, \ldots, a_{k+1} - a_1\}$ is linear independent ([4]).

let $F = \left\{z_{j_1}, ..., z_{j_{m+s}}\right\} \subseteq R^{m+s}$ be the collection of arbitrary $m+s$ DMUs. $z = (x_1, ..., x_m, y_1, ..., y_s)$ lies on the hyperplane H, passing through members of F, if and only if the determinant of the components of the vectors $\overrightarrow{z_{j_1}z}, \overrightarrow{z_{j_1}z_{j_2}}, ..., \overrightarrow{z_{j_1}z_{j_{m+s}}}$ are equal to zero i.e.,

$$\begin{vmatrix} x_1 - x_{1j_1} & \dots & x_m - x_{mj_1} & y_1 - y_{1j_1} & \dots & y_s - y_{sj_1} \\ x_{1j_2} - x_{1j_1} & \dots & x_{mj_2} - x_{mj_1} & y_{1j_2} - y_{1j_1} & \dots & y_{sj_2} - y_{sj_1} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ x_{1j_{m+s}} - x_{1j_1} & \dots & x_{mj_{m+s}} - x_{mj_1} & y_{1j_{m+s}} - y_{1j_1} & \dots & y_{sj_{m+s}} - y_{sj_1} \end{vmatrix} = 0 \quad (2)$$

where $x_1, ..., x_m, y_1, ..., y_s$ are variables, $x_{pj_t}(p = 1, ..., m, t = 1, ..., m+s)$ is the p -th input of DMU_{j_t} and y_{qj_t} $(q = 1, ..., s, t = 1, ..., m+s)$ is the q -th output of DMU_{j_t}.

The important point to note is the fact that the gradient (normal vector) of H, which is obtained by Eq. (2), is orthogonal to the $m+s-1$ vector in the form of $\overrightarrow{z_{j_2}z_{j_1}}, \overrightarrow{z_{j_3}z_{j_1}}, ..., \overrightarrow{z_{j_{m+s}}z_{j_1}}$. Needless to say, the gradient of H is also orthogonal to all the vectors that lie on H. In other words, the rows of the determinant of Eq. (2) can be considered as the vectors that are the difference between any two arbitrary points of F i.e.,

$$\overrightarrow{z_{j_t}z_{j_{t'}}} \qquad s.t \qquad t, t' \in \{1, ..., m+s\} \;,\; t \neq t'.$$

Remark 3.2. If the $m+s$ points, through which the hyperplane H passes, are not affine independent, then the Eq. (2) becomes trivia (leads to 0=0). It means that the hyperplane passing through these points is not unique.

For example, consider a system of six DMUs with variable return to scale assumption as in Table 1.

It is easily found in Fig. 1 that the set $\{A, D, G\}$ is not affine independent, since $\{D - A, G - A\}$ is not linear independent. We now apply Eq. (2) for $\{A, D, G\}$, hence it follows that 0=0 i.e.,

$$\begin{vmatrix} x_1 - 2 & x_2 - 2 & y - 4 \\ 1 - 2 & 5 - 2 & 6 - 4 \\ 1.5 - 2 & 3.5 - 2 & 5 - 4 \end{vmatrix} = 0.$$

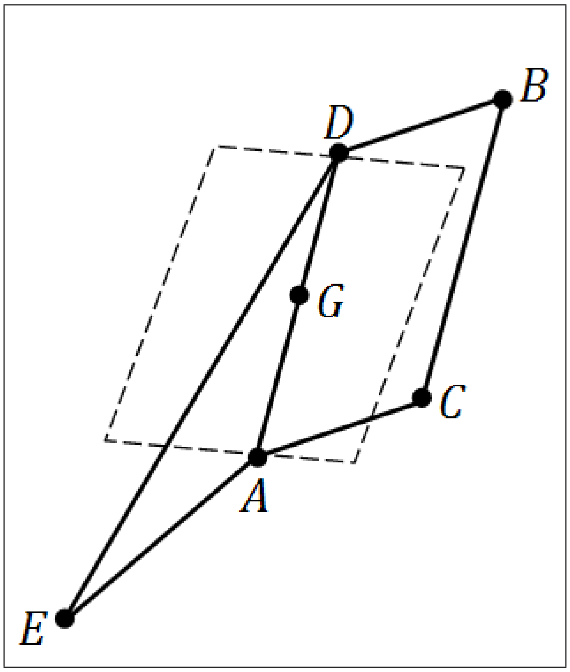

Figure 1: Remark 3.2.

Table 1: An illustrative numerical example for Fig. 1

	DUM					
	A	B	C	D	E	G
x_1	2	4	5	1	0.5	1.5
x_2	2	4	1	5	1	3.5
y	4	8	6	6	1	5

This implies that the hyperplane passing through $\{A, D, G\}$ is not unique.

4 Finding The Strong Defining Hyperplane of $\mathbf{T_c}$ And $\mathbf{T_v}$

In this section, Definition 4.1 is presented as a criterion for identifying all the strong defining hyperplanes.

***Definition* 4.1.** The hyperplane $H = \{(\mathbf{x}, \mathbf{y})\,|\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0,\ (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0})\,,\ (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \neq (\mathbf{0}, \mathbf{0})\}$ is a strong defining hyperplane of PPS, if and only if:

1. At least $m + s$ actual DMUs of PPS lie on H.
2. It is supporting.
3. All of the components of its gradient are strictly positive i.e., $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) > (\mathbf{0}, \mathbf{0})$.

Now, suppose that $J = \{1, ..., n\}$ is a set of indices of all DMUs. We define F as the set of indices of DMUs which are non-dominated.

$F = \{j | DMU_j$ can not be dominated by any observed $DMUs\}$.

Let $|F| = n_1 \leqslant |J| = n$.

According to Definition 4.1, we choose an arbitrary $m + s$ members of F. Again, one should note that when we deal with T_c, one of these $m + s$ DMUs must be the origin; therefore, only $m + s - 1$ members of F are chosen. As it was mentioned, because of the ray unboundedness postulate in T_c, all the strong and weak defining hyperplanes must contain the origin. We call this set $D = \{j_1, ..., j_{m+s}\}$.

By using D, a hyperplane can be generated through Eq. (2).

4.1 Supporting conditions

Suppose that the equation of the obtained hyperplane from Eq. (2) is in the form of $H : \bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0$.

Remark 4.2. If $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0})$, then H has the potential to be supporting. In what follows, the conditions of Theorem 4.3 are going to be surveyed so that the supporting of H is proved.

If $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) \leqslant (\mathbf{0}, \mathbf{0})$, then the hyperplane H must be generated by $(-\bar{\mathbf{u}}, -\bar{\mathbf{v}})$, and then we can survey the conditions of Theorem 4.3. (Note that in T_v, H must be generated by $(-\bar{\mathbf{u}}, -\bar{\mathbf{v}}, -\bar{u}_0)$).

If some of the gradient components of H corresponding to input vector are nonnegative or some of its gradient components corresponding to output vector are nonpositive, then the gradient of H is not in the direction of a supporting hyperplane.

***Theorem* 4.3.** *Suppose* $H = \{(\mathbf{x}, \mathbf{y}) \,|\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0, (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0}), (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \neq (\mathbf{0}, \mathbf{0})\}$, $\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0$ *is formulated from Eq.* (2)*. If*

$$\bar{\mathbf{u}}\mathbf{y}_j - \bar{\mathbf{v}}\mathbf{x}_j + \bar{u}_0 = 0 \qquad j \in D$$
$$\bar{\mathbf{u}}\mathbf{y}_j - \bar{\mathbf{v}}\mathbf{x}_j + \bar{u}_0 \leqslant 0 j \in J - D,$$

then H is supporting.

Proof. See Appendix A. □

5 Identifying Equations of Weak Defining Hyperplanes

The idea to find weak defining hyperplanes is straightforward by adding artificial weak DMUs.

Remark 5.1. In T_c we call each point on the input/output axes as "artificial DMU", and in T_v we call each virtual DMU on the weak hyperplanes, as "artificial DMU" hereafter. It should be noted that the weak hyperplanes are parallel to the axis of input or output.

***Definition* 5.2.** The hyperplane $H = \{(\mathbf{x}, \mathbf{y}) \,|\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0,\ (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0}),\ (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \neq (\mathbf{0}, \mathbf{0})\}$ is weak defining hyperplane of PPS, if and only if:

1. At least $m + s$ actual and artificial DMUs of PPS lie on H.
2. It is supporting.
3. At least one component of its gradient (normal vector) is zero.

5.1 Identifying artificial DMUs in $\mathbf{T_c}$

We know that the weak defining hyperplanes are parallel to at least one of the input or output axes. But in T_c owing to its special structure, all the hyperplanes of PPS must contain the origin; thus, the weak defining hyperplanes should necessarily contain entirely an input or output axis that the hyperplane is parallel to. It means that all points on the input/output axes should satisfy equation of weak hyperplanes. To this end, artificial DMUs is added on the input-output axes in the following form

$$\begin{aligned} &\gamma_l = (0,0,...,0,\alpha,0,...,0) \in \mathbb{R}^{m+s} \qquad l = 1,...,m \\ &\gamma_{m+q} = (0,0,...,0,-\alpha,0,...,0) \in \mathbb{R}^{m+s} \qquad q = 1,...,s. \end{aligned} \tag{3}$$

In which $\alpha > 0$. The vector γ_l $(l = 1,...,m)$ is an artificial DMU that lies on the l -th input axis where α appears in the l -th position. Likewise, γ_{m+q} $(q = 1,...,s)$ is an artificial DMU that lies on the q -th output axis where $-\alpha$ appears in the $(m+q)$ -th position (the minus sign shows that these artificial DMUs are dominated by origin).

5.2 Identifying artificial DMUs in T_v

We know that artificial DMUs through which a weak hyperplane passes must be satisfied in its equation. In the absence of the ray unboundedness postulate in T_v, weak defining hyperplanes may not contain input/output axes. In other words, we try to identify other artificial DMUs such that they satisfy equations of weak defining hyperplanes of T_v. In this case, we add $m+s$ artificial DMUs for each actual DMU which lie on the weak hyperplanes that are parallel to the input or output axis.

These artificial DMUs are as follows:

$$\begin{aligned} z_j^l =&(x_{1j},x_{2j},...,x_{(l-1)j},x_{lj}+\alpha,x_{(l+1)j},...,x_{mj},y_{1j},...,y_{sj}) \\ &j=1,...,n,\ l=1,...,m \\ z_j^{m+q} =&(x_{1j},x_{2j},...,x_{mj},y_{1j},...,y_{(q-1)j},y_{qj}-\alpha,y_{(q+1)j},...y_{sj}) \\ &j=1,...,n,\ q=1,...,s. \end{aligned} \tag{4}$$

It should be noted that each $z_j^l\,(l = 1,...,m)$ and $z_j^{m+q}\,(q = 1,...,s)$ are dominated by z_j.

Let us illustrate concepts of these artificial DMUs with a simple example. Fig. 2 exhibits three actual DMUs z_1, z_2, z_3, each with two inputs and one output. Note that there is no strong defining hyperplane.

It is easy to see that the artificial DMUs z_1^1, z_2^1 and z_3^1 lie on the weak hyperplanes that are parallel to the first axis of input and the z_1^2, z_2^2 and z_3^2 lie on the weak hyperplanes that are parallel to the second axis of input and the artificial DMU z_1^3 lies on the weak hyperplanes that are parallel to the first axis of output.

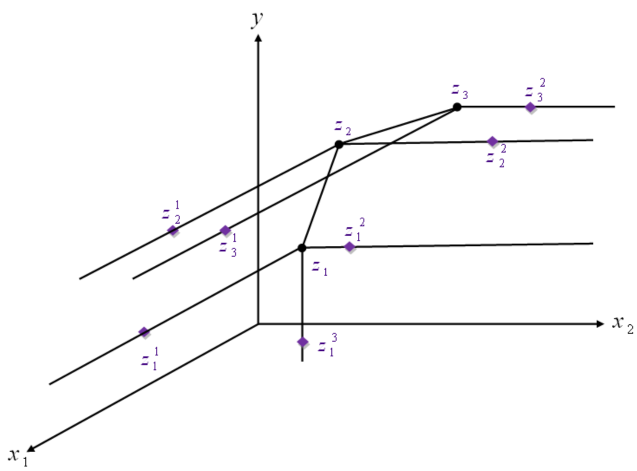

Figure 2: The presentation of actual and artificial DMUs.

Theorem 5.3 will show that in T_v, adding $n\times(m+s)$ artificial DMUs in the form of (4) is equivalent to the adding $m+s$ artificial DMUs in the form of (3). Hence, with the result of Theorem 5.3 only few artificial DMUs must be added.

***Theorem* 5.3.** *Assume H is a hyperplane that is generated with a set of $m+s$ actual and artificial DMUs which are in the form of (4), then H can also be generated with the set of $m+s$ actual and artificial DMUs which are in the form of (3) and the equation (2) is replaced with the following equation:*

$$\begin{vmatrix} x_1 - x_{1j_1} & \cdots & x_m - x_{mj_1} & & y_1 - y_{j_1} & \cdots & y_s - y_{sj_1} \\ x_{1j_2} - x_{1j_1} & \cdots & x_{mj_2} - x_{mj_1} & & y_{1j_2} - y_{j_1} & \cdots & y_{sj_2} - y_{sj_1} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ x_{1j_k} - x_{1j_1} & \cdots & x_{mj_k} - x_{mj_1} & & y_{1j_k} - y_{j_1} & \cdots & y_{sjk} - y_{sj_1} \\ & & & \gamma_{t_1} & & & \\ & & & \vdots & & & \\ & & & \gamma_{t_{k'}} & & & \end{vmatrix} = 0 \quad (5)$$

where k is the number of actual DMUs and k' is the number of artificial DMUs such that $k + k' = m + s$, $k \geqslant 1, k' \geqslant 1$ and in which $\gamma_{t_1}, ..., \gamma_{t_{k'}}$ are k' arbitrary artificial DMUs among $m+s$ artificial DMUs from (3).

Proof. See Appendix A. □

Now, for both T_c and T_v, we can put $m + s$ artificial DMUs in the form of (3) and actual DMUs in a new set in the following form:

$$E = \left\{z_{j_1}, ..., z_{j_{n_1}}, \gamma_l, \gamma_{m+q}\right\} \ l = 1, ..., m, \ q = 1, ..., s.$$

where n_1 is the number of members of F. We construct the set D by selecting $m + s$ arbitrary $DMUs$ from E such that it consists of at least one artificial DMU and at least one actual DMU. To this end, we generate the hyperplane passing on D by using Eq. (5).

Remark 5.4. For D in T_c the following conditions must be met:

1. The origin belongs to D.
2. At least one artificial DMU and one actual DMU belong to D.

Using what has been mentioned up to now, for finding all the strong and weak defining hyperplanes of the PPS, an algorithm is going to be presented in the next section.

6 Summary of the Algorithm for Finding All the Strong and Weak Defining Hyperplanes

Using what has been mentioned up to now, an algorithm for finding all strong and weak defining hyperplanes of the PPS is presented as follows:

Let $J = \{1, ..., n\}$ be an index set of all DMUs.

Step 1 (purge). Define F as the set of DMUs which are non-dominated.

$F = \{j | DMU_j \text{cannot be dominated by any observed } DMUs\}$

where $|F| = n_1 \leqslant |J| = n$.

Step 2. Consider $m + s$ artificial DMUs as:

$$\gamma_l = (0, ..., 0, \alpha, 0, ..., 0) \in \mathbb{R}^{m+s} \ , \ l = 1, ..., m$$
$$\gamma_{m+q} = (0, ..., 0, -\alpha, 0, ..., 0) \in \mathbb{R}^{m+s} \ , \ q = 1, ..., s$$

where $\alpha > 0$ is an arbitrary scalar. Put indices of both non-dominated and artificial DMUs in E.

Step 3. Choose $m + s$ arbitrary members of E. Call this set $D = \{j_1, ..., j_{m+s}\}$. In dealing with T_c, one of these $m + s$ DMUs must be the origin.

Construct a hyperplane using Eq. (2). If at least one of the $m + s$ members of D is artificial, then use Eq. (5). Suppose that the equation of the obtained hyperplane is in the form of $H : \bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0$.

Step 4. If some of the components of its gradient which are corresponding to the output vector are nonpositive or those corresponding to the input vector are nonnegative, then go to step 7.

If $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0})$, then H may be supporting and go to the next step.

If $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) \leqslant (\mathbf{0}, \mathbf{0})$, then construct the hyperplane H with $(-\bar{\mathbf{u}}, -\bar{\mathbf{v}})$ and go to the next step.

Step 5. If

$$\bar{\mathbf{u}}\mathbf{y}_j - \bar{\mathbf{v}}\mathbf{x}_j + \bar{u}_0 = 0 \ j \in D$$
$$\bar{\mathbf{u}}\mathbf{y}_j - \bar{\mathbf{v}}\mathbf{x}_j + \bar{u}_0 \leqslant 0 \ j \in J - D,$$

then H is supporting. Otherwise, go to step 7.

Step 6. If at least one of the m+s members of D is an artificial DMU, then H is a weak defining hyperplane. Otherwise, if all the components of its gradient are strictly positive, then H is a strong defining hyperplane.

Step 7. If another subset of E with $m + s$ members can be found, go to step 3, else stop.

6.1 Convergence of algorithm

The mentioned algorithm stops in a finite number of iterations, with all the strong and weak defining hyperplanes of PPS under constant and variable returns to scale assumptions. The third step will be repeat until all the hyperplanes passing through D are found, that D is subset of m+s members of E. On the other hand, the number of iterations of third step is less than $\binom{m+s+n_1}{m+s}$. Thus, algorithm is finite-time convergent.

7 Numerical Examples

Here, we present two examples. In Example 7.1, all defining hyperplanes of a system with CRS technology is found. Example 7.2 is devoted to a system of DMUs in VRS technology.

Example 7.1. ($\mathbf{T_c}$−**CRS**). Consider a system of five DMUs with two inputs and one output as in Table 2. All DMUs have been depicted in Fig. 3.

Table 2: Data on numerical example 7.1

	DUM				
	a	b	c	d	e
x_1	1	2	5	6	1.5
x_2	4	2	1	1	3
y	1	1	1	1	1

Since $\mathbf{x}_d \geqslant \mathbf{x}_c$, $y_d \leqslant y_c$ then DMU_d is dominated and $F = \{a, b, c, e\}$.
Let $\alpha = 2$, we have:
$\gamma_1 = (2, 0, 0)$, $\gamma_2 = (0, 2, 0)$, $\gamma_3 = (0, 0, -2)$
$E = \{a, b, c, e, \gamma_1, \gamma_2, \gamma_3\}$.

The first strong hyperplane is constructed by using Eq. (2) on $D = \{o, a, b\}$ where o is the origin.

$$\begin{vmatrix} x_1 & x_2 & y \\ 1 & 4 & 1 \\ 2 & 2 & 1 \end{vmatrix} = -6y + 2x_1 + x_2, \qquad H_1 : -6y + 2x_1 + x_2 = 0.$$

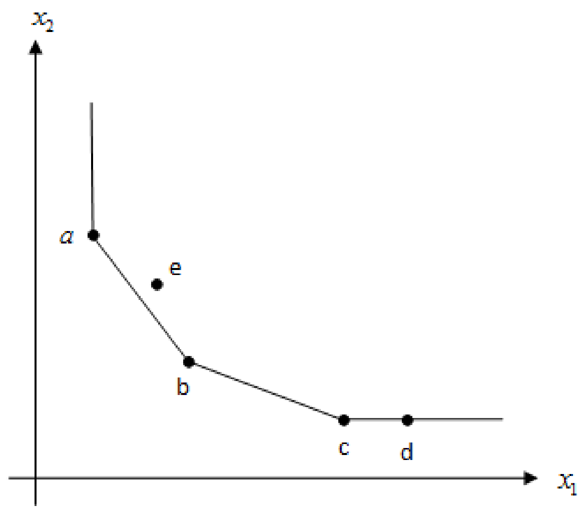

Figure 3: PPS of numerical Example 7.1.

By using Remark 4.2 and corresponding with $\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0$, we have $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) = (-6, -2, -1) < 0$; hence, construct the hyperplane with $(-\bar{u}, -\bar{v})$ that leads to $H_1 : 6y - 2x_1 - x_2 = 0$.

It can be easily verified that conditions of Theorem 4.3 are held and H_1 is defining because:

$a: \; 6\,(1) - 2(1) - (4) = 0$

$b: \; 6\,(1) - 2(2) - (2) = 0$

$c: \; 6\,(1) - 2(5) - (1) < 0$

$e: \; 6\,(1) - 2(1.5) - (3) < 0.$

On $D_2 = \{o, a, c\}$ we obtain $H_2 : 19y - 3x_1 - 4x_2 = 0$, but this is not supporting since by considering conditions of Theorem4.3 for DMU_b we have:

$b : 19\,(1) - 3(2) - 4(2)) > 0.$

On $D_3 = \{o, a, \gamma_2\}$, because one of the three members of D_3 is artificial DMU, therefore by using Eq. (5) we have:

$$\begin{vmatrix} x_1 & x_2 & y \\ 1 & 4 & 1 \\ 0 & 2 & 0 \end{vmatrix} = y - x_1, \qquad H_3 : y - x_1 = 0.$$

Note that conditions of Theorem 4.3 are held and H_3 is a weak defining hyperplane. In a similar manner, the other equations of the strong and weak defining hyperplanes are obtained. Table 3 summarizes the results related to the strong and weak defining hyperplanes of the Example 7.1.

Table 3: The results of the supposed algorithm for Example 7.1

D	Equation of hyperplanes	Strong or weak
$\{o, a, b\}$	$6y - 2x_1 - x_2 = 0$	*strong*
$\{o, b, c\}$	$8y - x_1 - 3x_2 = 0$	*strong*
$\{o, a, \gamma_2\}$	$y - x_1 = 0$	*weak*
$\{o, c, \gamma_1\}$	$y - x_2 = 0$	*weak*

Example 7.2. ($\mathbf{T_v}$−**VRS**). Consider a system of four DMUs with one input and one output as it is shown in Fig. 4. Data is given in Table 4.

Table 4: Data of numerical Example 7.2

	DUM			
	a	b	c	d
x	1	2	4	3
y	1	3	5	3

Since $x_b \geqslant x_d$, $y_d \leqslant y_b$, the DMU_d is dominated and $F = \{a, b, c\}$. Let $\alpha = 2$, we have $\gamma_1 = (2, 0)$, $\gamma_2 = (0, -2)$; then $E = \{a, b, c, \gamma_1, \gamma_2\}$. On $D = \{a, b\}$ and by using Eq. (2) we get:

$$\begin{vmatrix} x-1 & y-1 \\ 2-1 & 3-1 \end{vmatrix} = -y + 2x - 1, \qquad H_1 : -y + 2x - 1 = 0.$$

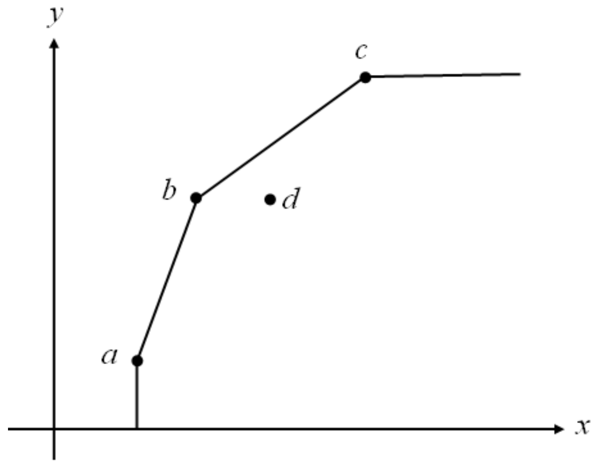

Figure 4: PPS of numerical Example 7.2.

By using Remark 4.2 and corresponding $\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0$, we have $(\bar{u}, \bar{v}) = (-1, -2) < 0$; hence, we construct the hyperplane H_1 with $(-\bar{u}, -\bar{v})$ that leads to $H_1 : y - 2x + 1 = 0$.

It can be easily verified that the conditions of Theorem 4.3 are held and H_1 is defining because:

a $1 - 2\,(1) + 1 = 0$

b $3 - 2\,(2) + 1 = 0$

c $5 - 2\,(4) + 1 < 0$.

Table 5 summarizes the results of applying the algorithm for detecting strong and weak defining hyperplanes of the Example 7.2.

8 Comparative Analysis

In this section, we have made a comparison between the method presented in this paper and the previous ones presented with the subject of finding defining hyperplanes.

Most of the articles presented in this field, find only strong defining hyperplanes (Say [1, 8, 10, 11, 12, 13]). But it is important to generate the weak defining hyperplanes, specially when the frontier of the PPS is

Table 5: The results of the supposed algorithm for Example 7.2

D	HYPERPLANES	Is supporting?	Strong or weak
$\{A, B\}$	$y - 2x + 1 = 0$	Yes	strong
$\{A, C\}$	$-4x + 3y + 1 = 0$	No	–
$\{B, C\}$	$y - x - 1 = 0$	Yes	strong
$\{A, \gamma_1\}$	$y - 1 = 0$	No	–
$\{A, \gamma_2\}$	$-x + 1 = 0$	Yes	weak
$\{B, \gamma_1\}$	$y - 3 = 0$	No	–
$\{B, \gamma_2\}$	$-x + 2 = 0$	No	–
$\{C, \gamma_1\}$	$y - 5 = 0$	Yes	weak
$\{C, \gamma_2\}$	$-x + 4 = 0$	No	–

constructed only of weak facets.

Also, most of the provided methods are applicable to PPS, either constant returns to scale (CRS) or variable returns to scale (VRS). (Say [1, 8, 9, 13, 15]).

Finally, in all of the mentioned papers, methods of finding defining hyperplanes solve many linear programming problems. This matter requires more computational effort.

The presented algorithm (in section 6) has the following strengths:

i) It Calculates both strong and weak defining hyperplanes.

iii) It's applicable to both, PPS under constant and variable returns to scale.

iii) It finds all the defining hyperplanes without solving any LPs.

Our method can be easily implemented using existing packages for mathematical algorithm, such as python.

9 Conclusions

Data Envelopment Analysis (DEA) has a strong connection with Linear Programming (LP). The structure of PPS is studied since early years of DEA. Finding strong and weak defining hyperplanes of PPS is surveyed in many papers that are mostly baes on solving LPs. Today, with advanced computers and softwares solving a linear programming (or even hundreds of LPs) is not time consuming. But existence of alternative solution for a LP is a big issue in the subject. It causes many difficulties in determining the maximal facets (and also defining hyperplanes) PPS.

In this paper, we have proposed an algorithm for finding all the strong and weak defining hyperplanes without solving any linear programming problems. By means of the equations of hyperplanes which construct the efficient frontier, we can identify returns to scale of DMUs and all efficient DMUs that consist of a reference set. Furthermore, the method of moving inefficient DMUs to efficient frontier can be studied in more details. With all defining hyperplanes of PPS in hand, many issues in DEA can be explored simpler.

Appendix A

Proofs of Theorems

Proof of Theorem 4.3. Since H passes through all members of D, then $H \cap T_v \neq \emptyset$. Also, we have:

$$\mathbf{u}\mathbf{y}_j - \mathbf{v}\mathbf{x}_j + u_0 \leqslant 0 \qquad j = 1, ..., n.$$

Suppose that for $j = 1, ..., n$, $\lambda_j \geqslant 0$ are scalars such that $\sum_{j=1}^{n} \lambda_j = 1$, then we have:

$$\bar{\mathbf{u}} \left(\sum_{j=1}^{n} \lambda_j y_j \right) - \bar{\mathbf{v}} \left(\sum_{j=1}^{n} \lambda_j x_j \right) + \bar{u}_0 \left(\sum_{j=1}^{n} \lambda_j \right) \leqslant 0.$$

Again, with regard to the structure of T_v we have

$$-\bar{\mathbf{v}}\mathbf{x} \leqslant -\bar{\mathbf{v}} \sum_{j=1}^{n} \lambda_j \mathbf{x}_j$$

$$\bar{\mathbf{u}}\mathbf{x} \leqslant \bar{\mathbf{u}} \sum_{j=1}^{n} \lambda_j \mathbf{y}_j$$

$$\bar{u}_0 \leqslant \bar{u}_0.$$

Therefore, for each $(\mathbf{x}, \mathbf{y}) \in T_v$ we get:

$$\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 \leqslant \bar{\mathbf{u}} \sum_{j=1}^{n} \lambda_j \mathbf{y}_j - \bar{\mathbf{v}} \sum_{j=1}^{n} \lambda_j \mathbf{x}_j + \bar{u}_0 \sum_{j=1}^{n} \lambda_j \leqslant 0.$$

It means that all the PPS (T_v) is on one side of H. In other words, H is supporting.

Note that by omitting u_0, the proof is concluded for T_c straightforward.

Proof of Theorem 5.3. It is known that each hyperplane can be generated by using a set of $m + s$ points (DMUs). We will investigate different possible cases in which the artificial DMUs in the form of (4), are effective in constructing a hyperplane. Then, it will be shown that these hyperplanes can also be generated by using artificial DMUs in the form of (3).

Let D consists of both actual and artificial DMUs. Suppose D contains k arbitrary actual and k arbitrary artificial DMUs such that $k + k = m + s$ and $k \geq 1, k \geq 1$.

Case 1. All the k' artificial DMUs has the same index as the actual DMUs that are members of D ; i.e.

$$D = \left\{ z_{j_1}, ..., z_{j_k}, z^*_{j_{t_1}}, ..., z^*_{j_{t_{k'}}} \right\} \text{ and } \left\{ j_{t_1}, ..., j_{t_{k'}} \right\} \subseteq \{j_1, ..., j_k\}.$$

where $*$ can be either l ($l \in \{1, ..., m\}$) or q ($q \in \{1, ..., s\}$).

Now let H be the hyperplane that is generated by using the members of D and Eq. (2). Therefore, the gradient of H is orthogonal to vectors $\overrightarrow{z_{j_1} z_{j_2}}, ..., \overrightarrow{z_{j_1} z_{j_k}}, \overrightarrow{z_{j_t} z^*_{j_{t_\beta}}}$ such that $\beta \in \{1, ..., k'\}$. On the other hand, Remark 3.2 implies that the gradient of H is also orthogonal to $m+s-1$ arbitrary vectors as follows:

$$\overrightarrow{z_{j_1} z_{j_2}}, ..., \overrightarrow{z_{j_1} z_{j_k}}, \overrightarrow{z_{j_t} z^*_{j_{t_\beta}}} \text{ s.t } j_{t_\beta} = j_t \ , \ \beta \in \left\{1, ..., k'\right\}.$$

It means that, in the determinant of Eq. (2), rows $(k + 1)$ to $(k + k')$ are the differences between actual and artificial DMUs which have the same indices. But these differences are the vectors introduced in (3). So, the determinant of Eq. (2) transforms to the determinant of Eq. (5).

Case 2. The k' artificial DMUs are of two types. The first type are artificial DMUs that have the indices same as some actual DMUs of set D ; and the second type are artificial DMUs that don't have the indices same as some actual DMUs of set D.Suppose $k'' + k''' = k'$ where k'' is the number of the first type and k''' is the number of the second type. Then $D = \left\{ z_{j_1}, ..., z_{j_k}, z^*_{j_{t_1}}, ..., z^*_{j_{t_{k''}}}, z^*_{j_{t'_1}}, ..., z^*_{j_{t'_{k'''}}} \right\}$ and $\{t_1, ..., t_{k''}\} \subseteq \{1, ..., k\}$, $\{t'_1, ..., t'_{k'''}\} \subseteq J - \{1, ..., k\}$.

In this case, we will show that the obtained hyperplane using the above set is either not supporting (therefore it is not defining) or the set D can be transformed to a set which is a subset of case 1, and in this case the hyperplane can also be generated by Eq. (5). Corresponding to Remark 3.2, we know that the supporting hyperplanes have a form as follows:

$$H = \{(\mathbf{x}, \mathbf{y}) \,|\bar{\mathbf{u}}\mathbf{y} - \bar{\mathbf{v}}\mathbf{x} + \bar{u}_0 = 0,\ (\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0})\}$$

in which $(\bar{\mathbf{u}}, \bar{\mathbf{v}})$ is gradient of H. Without loss of generality, suppose that for an arbitrary artificial DMU of D, say $z^*_{j_t} \in D$, we have $z_{j_t} \notin D$ and also suppose $* = l$ and $l \in \{1, ..., m\}$. since $z^l_{j_t}$ is dominated by z_j we have:

$$\begin{aligned}&(-x_{1j_t}, -x_{2j_t}, ..., -x_{lj_t}, y_{1j_t}, ..., y_{sj_t}) \geqslant \\ &(-x_{1j_t}, -x_{2j_t}, ..., -x_{lj_t} - \alpha, y_{1j_t}, ..., y_{sj_t}).\end{aligned}$$

And since $(\bar{\mathbf{u}}, \bar{\mathbf{v}}) \geqslant (\mathbf{0}, \mathbf{0})$, then:

$$\begin{aligned}&u_1 y_{1j_t} + \cdots + u_s y_{sj_t} - v_1 x_{1j_t} - \cdots - v_l x_{lj_t} \cdots - v_m x_{mj_t} + u_0 \geqslant \\ &u_1 y_{1j_t} + \cdots + u_s y_{sj_t} - v_1 x_{1j_t} - \cdots - v_l (x_{1j_t} + \alpha) - \cdots - v_m x_{mj_t} + u_0.\end{aligned}$$

Because H is binding in $z^l_{j_t} \in D$, the right-hand side of the above inequality will be zero i.e.,

$$u_1 y_{1j_t} + \cdots + u_s y_{sj_t} - v_1 x_{1j_t} - \cdots - v_l x_{lj_t} \cdots - v_m x_{mj_t} + u_0 \geqslant 0.$$

Subcase 2a. If $u_1 y_{1j_{t'_1}} + \cdots + u_s y_{sj_{t'_1}} - v_1 x_{1j_{t'_1}} - \cdots - v_l x_{lj_{t'_1}} \cdots - v_m x_{mj_{t'_1}} + u_0 > 0$, then clearly H is not a supporting hyperplane, because

H does not pass through z_{j_t}. So, in this case, the set D will not lead to the construction of a supporting hyperplane.

Subcase 2b. If $u_1 y_{1j_{t'_1}} + \cdots + u_s y_{sj_{t'_1}} - v_1 x_{1j_{t'_1}} - \cdots - v_l x_{lj_{t'_1}} \cdots - v_m x_{mj_{t'_1}} + u_0 = 0$, then H passes through z_{j_t}. Therefore, this hyperplane can be constructed by eliminating artificial DMU $z_{j_t}^l$ from D and substituting the actual same-indexed DMU. In this case we have a subset of case 1.

Finally we can conclude, for each artificial DMU that there doesn't exist any same-index actual DMU in D, either the hyperplane is not supporting or the hyperplane can be obtained by other forms of D which are subsets of case 1.

Appendix B

Illustration of Theorem 5.3 with an example

Consider a system contain 7 DMUs with two inputs and one output (Table 6). The production possibility set (T_v) is shown in Fig. 5. These seven DMUs span four strong defining hyperplanes and six weak defining hyperplanes. The dotted regions and the shaded regions of the production possibility set are the strong and weak efficient frontier respectively. Some artificial DMUs in the form of (4) have also been depicted in Fig. 5.

We have:

$$z_2^1 = (5+\alpha, 1, 6)\,, \quad z_3^1 = (4+\alpha, 4, 8)\,, \quad \text{and} \quad z_5^2 = (0.5, 1+\alpha, 1)$$

in all of which $\alpha > 0$, for example $\alpha = 1$.

Table 6: An illustrative numerical example

	DUM						
	z_1	z_2	z_3	z_4	z_5	z_6	z_7
x_1	2	5	4	1	0.5	1	1.5
x_2	2	1	4	5	1	0.5	3.5
y	4	6	8	6	1	1	5

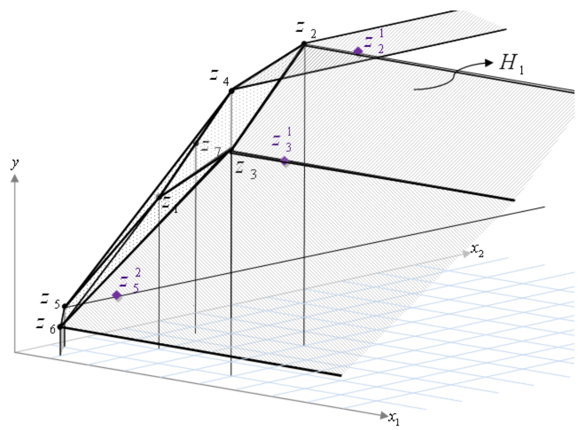

Figure 5: he production possibility set of the example.

The detailed analysis proof of Theorem 5.3 will be provided in the following instances.

Case 1. Choose $D_1 = \{z_2, z_3, z_2^1\}$. The set D_1 contains two actual DMUs and one artificial DMU. The Artificial DMU z_2^1 has the same index as z_2. By using Eq. (2) we have:

$$\begin{vmatrix} x_1 - 5 & x_2 - 1 & y - 6 \\ 4 - 5 & 4 - 1 & 8 - 6 \\ 5 - 5 & 4 - 1 & 8 - 6 \end{vmatrix} = 0 \ \rightarrow \quad H_1: \quad 3y - 2x_2 - 16 = 0.$$

The obtained weak hyperplane is associated with H_1 in Fig. 3 which is parallel to the first axis of input. Note that the Remark 3.1 implies that the gradient of H_1 is also orthogonal to vectors $\overrightarrow{z_2z_3}, \overrightarrow{z_3z_3^1}$, in which $\overrightarrow{z_3z_3^1} = (1, 0, 0)$. Consequently, the equation of H_1 is can be generated by using the following relation:

$$\begin{vmatrix} x_1 - 5 & x_2 - 1 & y - 6 \\ 4 - 5 & 4 - 1 & 8 - 6 \\ 5 - 4 & 4 - 4 & 8 - 8 \end{vmatrix} = \begin{vmatrix} x_1 - 5 & x_2 - 1 & y - 6 \\ -1 & 3 & 2 \\ 1 & 0 & 0 \end{vmatrix}$$

$$\Longrightarrow \ H_1: \quad 3y - 2x_2 - 16 = 0.$$

Obviously, the above relation is associated with Eq. (5). This expresses the equivalence between the set $D_1 = \left\{z_2, z_3, z_3^1\right\}$ and the set $D = \{z_2, z_3, \gamma_1\}$ in which γ_1 is an artificial DMU in the form of (3).

Case 2. Choose the set $D_2 = \left\{z_3, z_3^1, z_2^1\right\}$, that contains one actual DMU and two artificial DMUs. The artificial DMU z_3^1 has the same index as z_3 (Actual DMU), but there is no actual DMU with same index as z_2^1 . DMU z_2^1 is dominated by z_2. Let us now summarize the different possible cases that may arise.

Subcase 2a. The hyperplane that passes through members of D_2 also passes through z_2. Therefore the obtained hyperplane can be generated by the set $\left\{z_3, z_3^1, z_2\right\}$.

Subcase 2b. The obtained hyperplane when using D_2 is not supporting.

Due to the Fig. 1, it is clear that the subcase 2a is occurred. Therefore, in regard of case 1, the set $D_2 = \left\{z_3, z_3^1, z_2^1\right\}$ is equivalent to the set $D = \{z_3, \gamma_1, z_2\}$.

Now, choose the set $D_3 = \left\{z_3, z_3^1, z_5^2\right\}$ in which D_3 contains one actual DMU and two artificial DMUs. The artificial DMU z_3^1 has the same index as z_3 (Actual DMU), but there is no actual DMU with same index as z_5^2 . Here subcase 2b is occurred. Thus, the obtained hyperplane using D_3 and Eq. (4) is not supporting and z_5 is not on one side of it; i.e.,

$$\begin{vmatrix} x_1 - 4 & x_2 - 4 & y - 8 \\ 5 - 4 & 4 - 4 & 8 - 8 \\ 0.5 - 4 & 2 - 4 & 1 - 8 \end{vmatrix} = 0 \ \rightarrow \quad 2y - 6x_2 + 8 = 0.$$

And for z_5 we have:

$$\begin{aligned} &2\,(1) - 6\,(1) + 8 > 0 \\ &\forall j \ 2(y_j) - 6\,(x_{2j}) + 8 \leqslant 0. \end{aligned}$$

Therefore, this hyperplane passes through PPS and is not supporting.

The above example implies that the outcomes of adding artificial DMUs in the form of (4) and using Eq. (2) is similar to outcomes of adding artificial DMUs in the form of (3) and using Eq. (5). Therefore, all the weak defining hyperplanes of a system is generable by actual DMUs and artificial DMUs in the form of (3) and by using Eq. (5).

References

[1] A. Amirteimoori, S. Kordrosami. Generating strong defining hyperplanes of the production possibility set in data envelopment analysis, *Applied Mathematics Letters* 25(2012) 605-609.

[2] R.D. Banker, A. Charnes and W.W.Cooper. Some models for estimating technical and scale inefficiencies in Data Envelopment Analysis, *Management Science* 30 (1984) 1078-1092.

[3] R.D. Banker and R.M. Thrall. Estimation of return to scale using Data Envelopment Analysis, *European Journal of Operational Research* 62/1(1992)74-84.

[4] M. Bazaraa, H. Sherali, and C.M. Shetty. *Linear Programming And Network Flows*, Wiley New York 1990.

[5] A. Charnes, W.W. Cooper and E. Rhodes. Measuring the efficiency of decision making units, *European Journal of Operational Research* 2(6) (1978) 429-444.

[6] W.W. Cooper, L.M. Seiford, and K.Tone. *Data Envelopment Analysis.* Springer 2006. 2nd edition.

[7] J. Dula and F. Lopez. Preprocessing DEA, *Computers & Operations Research* 36(2009) 1204-1220.

[8] A. Ghazi, F. Hosseinzadeh Lotfi, M. Sanei. Finding the strong efficient frontier and strong defining hyperplanes of production possibility set using multiple objective linear programming, *Operational Research* (2020).

[9] G.R. Jahanshahloo, F. Hosseinzadeh Lotfi and D. Akbarian. Finding weak defining hyperplanes of PPS of BCC model, *Applied Mathematical modeling* 34 (2010) 3321-3332.

[10] G.R. Jahanshahloo, F. Hosseinzadeh Lotfi, M.R. Mozaffari and J.Gerami. Finding DEA-efficient hyperplanes using MOLP efficient faces, *Journal of Computational and Applied Mathematic*s 235(2011) 1227-1231.

[11] G.R. Jahanshahloo, A. Shirzadi and S.M. Mirdehghan. Finding strong defining hyperplanes of PPS using multiplier form, *European Journal of Operational Research* 194 (2009) 933-938.

[12] G.R. Jahanshahloo, F. Hosseinzadeh Lotfi, H. Zhiani Rezai and F. Rezai Balf. Finding strong defining hyperplanes of Production Possibility Set, *European Journal of Operational Research* 177 (2007) 42-54.

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

[14] O.B. Olesen and N.C. Petersen. Identification and use of efficient faces and facets in DEA, *Journal of Productivity Analysis* 20 (2003) 323-360.

[15] H. Ranjbar, F. Hosseinzadeh Lotfi, M.R. Mozaffari and J. Gerami. Finding defining hyperplanes with variable return to scale technology, *International Journal of Mathematical Analysis.* 3 (17) (2009)943-954.

[16] S. Sueyoshi and K. Sekitani. The measurement of returns to scale under a simultaneous occurrence of multiple solutions in a reference set and a supporting hyperplane, *European Journal of Operational Research* 181(2007)549-570.

[17] Q. Wei and G. Yu. Analyzing properties of K-cones in the generalized data envelopment analysis model, *Journal of Econometrics* 80(1997)63-84.

[18] G. Yu, Q. Wei, P. Brocket and L. Zhu. Construction of all DEA efficient surfaces of the production possibility set under the Gen-

eralized Data Envelopment Analysis Model, *European Journal of Operational Research* 95(1996) 491-510.

Sara Oskoueian
Department of Mathematics
Mashhad branch, Islamic Azad University.
Mashhad, Iran.
E-mail: sara.oskoueian@hotmail.com

Hamed Zhiani Rezai
Department of Mathematics
Associate Professor of Mathematics
Mashhad branch, Islamic Azad University.
Mashhad, Iran.
E-mail: hz_rezai@hotmail.com