Journal of Nonlinear and Convex Analysis
Volume 12, Number 1, 2011, 29–48

Yokohama Publishers
ISSN 1880-5221 ONLINE JOURNAL

IMPROVEMENT INDICES KEEPING THE FEASIBILITY IN DATA ENVELOPMENT ANALYSIS

SATOSHI WASHIO, SYUUJI YAMADA, TAMAKI TANAKA, AND TETSUZO TANINO

Abstract. In this study, we propose a computational algorithm to determine the equations simultaneously, which define the efficient frontiers of the four types of DEA(Data Envelopment Analysis) models(the CRS, VRS, IRS and DRS models). By applying the algorithm, we calculate each efficiency score of all DMUs(Decision Making Units) and classify efficient DMUs into three types of RTS(Returns To Scale). Moreover, for each DMU which is not efficient in the CRS model, we present two kinds of improvements. One is for becoming an efficient unit in the CRS model. The other is for becoming an efficient unit in the CRS model over the PP(Production Possibility) set of the VRS model. For each type of improvements, we use an order matrix defined by the decision maker to calculate an improvement target contained in the PP sets of some models.

1. Introduction

DEA is a method to estimate a relative efficiency of DMU performing similar tasks in a production system that consumes inputs to produce outputs. DEA was suggested by Charns, Cooper and Rhodes [3]. DEA can analyze the efficiency for several inputs and outputs. The CRS model which was suggested by Charnes, Cooper and Rhodes [3] evaluates the ratio between a weighted sums of multiple inputs and outputs. The VRS model which was suggested by Banker, Charnes and Cooper [1] has the feasible set defined by adding an equality condition of weight variables to the feasible set of the CRS model. Moreover, the VRS model can classify efficient DMUs into three types of RTS. The IRS and DRS models suggested by Seiford and Thrall [9], and Fare and Grosskopf [5] respectively, have the feasible set defined by adding an inequality condition of weight variables to the feasible set of the CRS model. By solving such models for each DMU, we can obtain the evaluated score of the efficiency. Moreover, for each inefficient DMU, an improvement to become an efficient unit is presented by using the efficiency score. However, since the improvements obtained by solving the almost conventional models forces each DMU to decrease all inputs or to increase all outputs at the same rate, it is often difficult to improve the values of all inputs or all outputs according to such the improvements. Because, it may be difficult to vary all inputs or all outputs at the same rate. In order to overcome such the difficulty, slacks-based measure [11] using the slack variables and the improvement with the inner-product norm [10] have been proposed.

2010 *Mathematics Subject Classification.* Primary 46N10; Secondary 93B15.

Key words and phrases. Efficient frontier, production possibility set.

This work is based on research 21540121 supported by Grant-in-Aid for Scientific Research (C) from Japan Society for the Promotion of Science.

In this paper, we propose an algorithm for obtaining the linear systems forming the efficient frontiers of the CRS, VRS, IRS, DRS models. By classifying the equations obtained by the algorithm into the ones forming the efficient frontier of the four models, we calculate efficiency scores of all DMUs for the four models. By using the equations forming the efficient frontier of the VRS model, we classify efficient DMUs in the VRS model into three types of RTS. Moreover, we present two kinds of the improvements by using the norm based on an order matrix defined by the decision maker. In the second improvement, we confine the improvement target to not only the PP set of the CRS model but also the PP set of the other model. Hence, all improvement rates for inputs (or outputs) do not always coincide. Furthermore, we present two kinds of algorithms for calculating the improvements. All improvement are obtained by solving quadratic mathematical problems with the use of the equations forming the efficient frontiers of the four models.

The constitution of this paper is as follows. In Section 2, we introduce the DEA models with the convex PP sets. In Section 3, we suggest an algorithm for constructing the equations forming the efficient frontiers. Moreover, we classify the equations obtained by the algorithm into the ones forming the efficient frontier of the four models. In Section 4, by using the equations calculated by the algorithm, we analyze DMUs and propose the improvements to become an efficient unit in the CRS model.

Through this paper, we use the following notation : Let $\mathbb{R}^n$ be an n-dimensional Euclidean space. For each vector $a \in \mathbb{R}^n, a^\top$ denotes the transposed vector of a. Let I_m be the unit matrix on $\mathbb{R}^m$. For each subset $S \subset \mathbb{R}^n$, $\dim S$ denotes the dimension number of S. For each vector $a \in \mathbb{R}^n, ||a||$ denotes the Euclidean norm of a. For each subset $S \subset \mathbb{R}^n$, $\mathrm{int}\, S$ denotes the interior of S. For each convex polyhedral set $S \subset \mathbb{R}^n$, $V(S)$ denotes the set of all vertices of S. For nutural numbers a and b $(a \geqslant b)$, ${}_aC_b := \frac{a!}{b!(a-b)!}$.

2. DEA models with the convex PP sets

Through this paper, n denotes the number of DMUs. Each DMU consumes m different inputs to produces s different outputs. Specifically, for each j $(j \in \{1, \dots, n\})$, DMU$_j$ has an input vector $x_j := (x_{1j}, \dots, x_{mj})^\top$ and an output vector $y_j := (y_{1j}, \dots, y_{sj})^\top$. Then, we assume the following conditions.

(A1): $x_j > 0$, $y_j > 0$ for each $j \in \{1, \dots, n\}$.
(A2): $(x_{j_1}, y_{j_1}) \neq (x_{j_2}, y_{j_2})$ for each $j_1, j_2 \in \{1, \dots, n\}$ $(j_1 \neq j_2)$.
(A3): $n > m + s$.
(A4): $\dim\{(x_j, y_j) : j = 1, \dots, n\} = m + s$.

2.1. GRS model.

GRS model. In order to calculate an efficiency of DMU$_k (1 \leqslant k \leqslant n)$, the GRS model is formulated as follows:

$$
(\mathrm{GRS}_k)\left\{\begin{array}{ll}
\text{minimize} & \theta \\
\text{subject to} & \theta x_{ik}-\displaystyle\sum_{j=1}^{n}\lambda_j x_{ij}\geqslant 0 \ i=1,\dots,m, \\
& \displaystyle\sum_{j=1}^{n}\lambda_j y_{rj}-y_{rk}\geqslant 0 \ r=1,\dots,s, \\
& L\leqslant \displaystyle\sum_{j=1}^{n}\lambda_j\leqslant U, \\
& \theta\in\mathbb{R},\ \lambda_j\geqslant 0 \ j=1,\dots,n,
\end{array}\right.
$$

where, $L \leqslant 1$ and $U \geqslant 1$. The GRS model unifies the four models by introducing parameters on intensity vector λ. If $L = 0$ and $U = \infty$, then the model is called the CRS model. Similarly, if $L = U = 1$, then the model is called the VRS model. If $L = 1$ and $U = \infty$, then the model is called the IRS model. If $L = 0$ and $U = 1$, then the model is called the DRS model. The efficiency of DMU_k in (GRS_k) is given by the following definition.

Definition 2.1. DMU_k is said to be GRS-efficient if the optimal value of (GRS_k) equals one. Otherwise, DMU_k is said to be GRS-inefficient.

Let θ^* denotes the optimal value of (GRS_k). From the definition of the constraint conditions of (GRS_k), it is obvious that $0 < \theta^* \leqslant 1$. Then, the PP set of the GRS model is defined as follows.

$$
\mathrm{T}(L, U) := \left\{(x,y) : x \geqslant \sum_{j=1}^{n}\lambda_j x_j, 0 \leqslant y \leqslant \sum_{j=1}^{n}\lambda_j y_j,\ \exists\lambda\in\Lambda(L, U)\right\},
$$

$$
\Lambda(L, U) := \left\{\lambda\in\mathbb{R}^n : L\leqslant\sum_{j=1}^{n}\lambda_j\leqslant U,\ \lambda\geqslant 0\right\}.
$$

It is clear that $\Lambda(L, U)$ is a closed convex set for each $L \leqslant 1$ and $U \geqslant 1$. Moreover, the following theorem holds.

Theorem 2.2. *For each $L \leqslant 1$ and $U \geqslant 1$, $\mathrm{T}(L, U)$ is a closed convex set.*

Proof. Firstly, we shall show that $\mathrm{T}(L, U)$ is convex. For each $(x^1, y^1), (x^2, y^2) \in \mathrm{T}(L, U)$, there exist $\lambda^1, \lambda^2 \in \Lambda(L, U)$ such that $x^1 \geqslant \sum_{j=1}^n \lambda_j^1 x_j$, $0 \leqslant y^1 \leqslant \sum_{j=1}^n \lambda_j^1 y_j$, $x^2 \geqslant \sum_{j=1}^n \lambda_j^2 x_j$ and $0 \leqslant y^2 \leqslant \sum_{j=1}^n \lambda_j^2 y_j$. For each $0 \leqslant \alpha \leqslant 1$, $\alpha\lambda^1 + (1-\alpha)\lambda^2 \in \Lambda(L, U)$. Moreover, $\alpha x^1 + (1-\alpha)x^2 \geqslant \sum_{j=1}^n (\alpha\lambda_j^1 + (1-\alpha)\lambda_j^2)x_j$ and $0 \leqslant \alpha y^1 + (1-\alpha)y^2 \leqslant \sum_{j=1}^n (\alpha\lambda_j^1 + (1-\alpha)\lambda_j^2)y_j$. Therefore, $\mathrm{T}(L, U)$ is a convex set.

Secondly, we shall show that $\mathrm{T}(L, U)$ is closed. Let $\{(x^k, y^k)\} \subset \mathrm{T}(L, U)$ satisfy $(x^k, y^k) \to (\bar{x}, \bar{y})$ as $k \to \infty$. Let $\epsilon > 0$. Since $x^k \to \bar{x}$ as $k \to \infty$, there exists $l \in \mathbb{N}$ such that $||x^k|| \leqslant ||\bar{x}|| + \epsilon$ for each $k > l$. Let $\delta := \max\{||\bar{x}|| + \epsilon, \max\{||x^k|| : k = 1, \dots, l\}\}$. Then, $||x^k|| \leqslant \delta$ for each $k \in \mathbb{N}$. Since $x_j > 0$ for each $j = 1, \dots, n$, $\delta' := \min\{x_{ij} : i = 1, \dots, m, j = 1, \dots, n\} > 0$. For each $k \in \mathbb{N}$, $\alpha \in \{\alpha \in \mathbb{R}^n : \sum_{j=1}^n \alpha_j = 1, \alpha_j \geqslant 0\, j = 1, \dots, n\}$, we have $x^k \leqslant \sum_{j=1}^n \alpha_j \frac{\delta}{\delta'} x_j$. Hence,

$\lambda_j \leqslant \frac{\delta}{\delta'}\alpha_j \leqslant \frac{\delta}{\delta'}$. For each $k \in \mathbb{N}$, there exists $\lambda^k \in \Lambda(L, U) \cap \{\lambda \in \mathbb{R}^n : 0 \leqslant \lambda_j \leqslant \frac{\delta}{\delta'}, j = 1, \ldots, n\}$ such that $x^k \geqslant \sum_{j=1}^n \lambda_j^k x_j$, $0 \leqslant y^k \leqslant \sum_{j=1}^n \lambda_j^k y_j$. Since $\{\lambda \in \mathbb{R}^n : 0 \leqslant \lambda_j \leqslant \frac{\delta}{\delta'}, j = 1, \ldots, n\}$ is compact, without loss of generality, we can assume that $\lambda^k \to \bar{\lambda}$ as $k \to \infty$. Then, from the closeness of $\Lambda(L, U)$, $\bar{\lambda} \in \Lambda(L, U)$. Hence, $\bar{x} = \lim_{k\to\infty} x^k \leqslant \lim_{k\to\infty} \sum_{j=1}^n \lambda_j^k x_j = \sum_{j=1}^n \bar{\lambda}_j x_j$, $0 \leqslant \bar{y} = \lim_{k\to\infty} y^k \leqslant \lim_{k\to\infty} \sum_{j=1}^n \lambda_j^k y_j = \sum_{j=1}^n \bar{\lambda}_j y_j$. Therefore $(\bar{x}, \bar{y}) \in \mathrm{T}(L, U)$. Hence, $\mathrm{T}(L, U)$ is closed. Consequently, $\mathrm{T}(L, U)$ is a closed convex set. □

In this paper, we denote the PP sets of the previous four models $\mathrm{T}_{\mathrm{CRS}}$, $\mathrm{T}_{\mathrm{VRS}}$, $\mathrm{T}_{\mathrm{IRS}}$ and $\mathrm{T}_{\mathrm{DRS}}$, respectively.

3. Algorithm for constructing the equations forming the efficient frontiers

3.1. Basic definitions and theorems in convex analysis.

The algorithm proposed in the next subsection utilizes the basic techniques in convex analysis. In this sebsection, we show several definitions and theorems in convex analysis.

Definition 3.1 (polar set)**.** Let E be a nonempty subset in $\mathbb{R}^n$. Then, E^* is said to be the polar set of E if E^* is defined as follows.

$$E^* := \{y \in \mathbb{R}^n : y^\top x \leqslant 1 \ \forall x \in E\}.$$

Definition 3.2 (convex hull)**.** Let E be a nonempty subset in $\mathbb{R}^n$. Then, $\mathrm{co}(E)$ is said to be the convex hull of E if $\mathrm{co}(E)$ is defined as follows.

$$\mathrm{co}(E) := \left\{ x \in \mathbb{R}^n : x = \sum_{j=1}^n \lambda_j x_j,\ \sum_{j=1}^n \lambda_j = 1,\ x_j \in E,\ \lambda_j \geqslant 0 \ j = 1, \ldots, n \right\}.$$

Definition 3.3 (Facet)**.** Let E be a polytope in $\mathbb{R}^n$. Then, $F := E \cap \{x \in \mathbb{R}^n : a^\top x = b\}$ is called the facet of E if $a^\top x \leqslant b$ for each $x \in E$ and $\dim F = n - 1$.

Theorem 3.4. *Let E be a nonempty set in $\mathbb{R}^n$. Then E^* is a closed convex set.*

Proof. Firstly, we shall show that E^* is convex. For all $x, y \in E^*$ and $0 \leqslant \lambda \leqslant 1$, $(\lambda x + (1-\lambda)y)^\top z = \lambda x^\top z + (1-\lambda)y^\top z \leqslant \lambda + (1-\lambda) = 1$ for all $z \in E$. Hence, $(\lambda x + (1-\lambda)y) \in E^*$. Therefore, E^* is convex. Secondly, we shall show that E^* is closed. By the definition of the polar set, E^* is the intersection of closed halfspaces. Hence, E^* is closed. □

Theorem 3.5 (Konno, Thach and Tuy [8], Proposition 2.6)**.** *Let E be a nonempty closed convex set in $\mathbb{R}^n$ and $\mathbf{0} \in E$. Then $E^{**} = E$.*

Theorem 3.6. *Let E be a polytope in $\mathbb{R}^n$. Then, $E^* = (V(E))^*$.*

Proof. Let $V(E) = \{a^1, \ldots, a^m\}$. Since E is a polytope,

$$E = \{x : x = \sum_{i=1}^m \lambda_i a^i,\ \sum_{i=1}^m \lambda_i = 1,\ \lambda_i \geqslant 0 \ i = 1, \ldots, m\}. \tag{3.1}$$

Obviously $E \supset V(E)$. From the principle of the polar set, $E^* \subset (V(E))^*$. Hence, we shall show that $E^* \supset (V(E))^*$. Let $y \in (V(E))^*$. Then, for each $i = 1, \ldots m$,

$(a^i)^\top y \leqslant 1$. By (3.1), for each $x \in E$, $x^\top y = (\sum_{i=1}^m \lambda_i a^i)^\top y = \sum_{i=1}^m \lambda_i (a^i)^\top y \leqslant \sum_{i=1}^m \lambda_i = 1$. Therefore, $y \in E^*$. Consequently, $E^* = (V(E))^*$. □

Theorem 3.7 (Jonathan, M.B., Adrian, A.L. [7])**.** *Let E be a nonempty subset in $\mathbb{R}^n$. Then E is bounded if and only if $\mathbf{0} \in \operatorname{int} E^*$.*

Theorem 3.8. *Let E be a polytope in $\mathbb{R}^n$ and $\mathbf{0} \in \operatorname{int} E$. Then, $E = (V(E^*))^*$.*

Proof. From Theorem 3.5, $E = E^{**}$. By Theorems 3.6 and 3.7, E^* is a polytope if E is a polytope satisfying $\mathbf{0} \in \operatorname{int} E$. From Theorem 3.6, $E = E^{**} = (V(E^*))^*$. □

We note that the convex hull of all DMUs is nonempty bounded closed convex set. Let $P_j = (x_j^\top, y_j^\top)^\top$, $j = 1, \dots, n$, where x_j and y_j are inputs and outputs of DMU$_j$, respectively. Let $P := \operatorname{co}(\{P_1, \dots, P_n\})$ and $T = \frac{1}{n}(P_1 + \cdots + P_n)$. Then by Assumption (A4), $P - T$ is a compact convex set satisfying $\mathbf{0} \in \operatorname{int}(P - T)$. Moreover, all hyperplanes forming the efficient frontiers of the four models are contained in that of P or the convex conical hull of all DMUs. We can calculate all equations forming P and the convex conical hull of all DMUs by using Theorem 3.8. Hence, we can obtain all equations forming them.

3.2. **Algorithm for calculating equations forming the efficient frontiers.** We need to clarify the equations forming the efficient frontiers to calculate the improvements. Therefore, in this section, we propose an algorithm for constructing the equations forming them. Since P is a polytope, by translating all DMUs, we construct a polytope including $\mathbf{0}$. Moreover all vertices of P are DMUs. By calculating all vertices of a polytope, we can clarify the equations forming the polar set of it. Therefore, by utilizing the properties of polar sets, we calculate all equations forming them.

Algorithm FFA:

Step 0:

Set $n' := 2n$ and $\bar{n} := n' + m + s$. Set $P_i\ (i = 1, \dots, n')$ and $P_i'\ (i = 1, \dots, \bar{n})$ as follows.

$$P_i := \begin{cases} (x_i^\top, y_i^\top)^\top & \text{if } i \in \{1, \dots, n\}, \\ 2P_{i-n} & \text{if } i \in \{n+1, \dots, n'\}. \end{cases} \tag{3.2}$$

$$P_i' := \begin{cases} P_i - T' & \text{if } i \in \{1, \dots, n'\}, \\ e^{i-n'} & \text{if } i \in \{n'+1, \dots, \bar{n}\}, \end{cases} \tag{3.3}$$

where $T' = \frac{1}{n'}(P_1 + \cdots + P_{n'})$ and e^j is a vector of $\mathbb{R}^{m+s}$ satisfying $e_j^j = 1$ and $e_i^j = 0$ for each $j \in \{1, \dots, m+s\}$ and $i \in \{1, \dots, m+s\} \backslash \{j\}$. Let $c_i := i\ (i = 1, \dots, m+s)$. Set $t := 0$ and go to Step 1.

Step 1:

If $\dim \{P_{c_i}' : i = 1, \dots, m+s\} = m+s$, then go to Step 2. Otherwise, go to Step 4.

Step 2:

Calculate W satisfying the following linear system:

$$\left\{\begin{array}{l}(P'_{c_1})^\top W = \alpha(c_1),\\ \quad\vdots\\ (P'_{c_{m+s}})^\top W = \alpha(c_{m+s}),\end{array}\right. \tag{3.4}$$

where, $\alpha(c_i)\,(i = 1, \ldots, m + s)$ are as follows.

$$\alpha(c_i) := \left\{\begin{array}{ll} 1 & \text{if } c_i \in \{1, \ldots, n'\},\\ 0 & \text{if } c_i \in \{n' + 1, \ldots, \bar{n}\}.\end{array}\right.$$

Go to Step 3.

Step 3:

If $(P'_j)^\top W \leqslant 1 (j = 1, \ldots, n')$, then W is a vertex of $(\mathrm{co}(\{P'_1, \ldots, P'_{n'}\}))^*$. Furthermore, if $W_1 \leqslant 0, \ldots, W_m \leqslant 0, W_{m+1} \geqslant 0, \ldots, W_{m+s} \geqslant 0$, then $t \leftarrow t + 1, V_t := W$. Go to Step 4.

Step 4:

If $c_1 = \bar{n} - m - s + 1$, go to Step 5. Otherwise,

Step 4–0:

Set $c_{m+s} \leftarrow c_{m+s} + 1$ and $j := m + s$. Go to Step 4–1.

Step 4–1:

If $c_j \leqslant \bar{n} - m - s + j$, set $c_{j'} \leftarrow c_j + j' - j$ for every $j' > j$. Go to Step 1. Otherwise, set $c_{j-1} \leftarrow c_{j-1} + 1, j \leftarrow j - 1$ and go to Step 4–1.

Step 5:

For each $i = 1, \ldots, t$, hyperplane forming the efficient frontier is as follows. If $\max\{|Y_{i,1}|, \ldots, |Y_{i,s}|\} > 0$ and $\frac{1+(X_i^\top, Y_i^\top)T'}{\max\{Y_{i,1}, \ldots, Y_{i,s}\}} > 0$, then hyperplane forming the efficient frontier is as follows.

$$H_i := \left\{(x, y) : X_i^\top x + Y_i^\top y = \frac{(1 + (X_i^\top, Y_i^\top)T')}{2}\right\}.$$

Otherwise,

$$H_i := \{(x, y) : X_i^\top x + Y_i^\top y = 1 + (X_i^\top, Y_i^\top)T'\},$$

where $(X_i^\top, Y_i^\top)^\top := V_i,\ X_i \in \mathbb{R}^m,\ Y_i \in \mathbb{R}^s$. Stop the algorithm.

At Step 0, in order to obtain all efficient frontiers of the CRS model, for each $i = 1, \ldots, n$, P_{i+n} is added. Let $\bar{P} := \mathrm{co}(\{P'_1, \ldots, P'_{n'}\})$. To calculate all vertices of $(\bar{P})^* \cap \hat{\mathbb{R}}^{m+s} := \{x \in \mathbb{R}^{m+s} : x_i \leqslant 0 (1 \leqslant i \leqslant m),\ x_i \geqslant 0 (m + 1 \leqslant i \leqslant m + s)\}$, all combinations of $\{P'_1, \ldots, P'_{\bar{n}}\}$ are considered. At Step 1, to examine whether there exists a solution of linear system (3.4), $\dim\{P'_{c_i} : i = 1, \ldots, m + s\}$ is calculated. At Step 3, to examine whether W obtained at Step 2 is a vertex of $(\bar{P})^*$, $W^\top P'_1, \ldots, W^\top P'_{\bar{n}}$ are calculated. If all values are less than or equal to one, then W is a vertex of $(\bar{P})^*$. At Step 4, to select all combinations of choosing $m + s$ numbers from $\{1, \ldots, \bar{n}\}$, $c_1, \ldots, c_{m+s}$ are updated. At Step 5, for each $i \in \{1, \ldots, t\}$, the necessity of H_i to construct the efficient frontier is examined.

Example 3.9. We illustrate Algorithm FFA in the case of $m = s = 1$. The data of DMUs is listed in Table 1 and illustrated in Figure 1. By executing Algorithm

FFA, we can calculate all equations forming the efficient frontier based on the data in TABLE 1 as follows:

TABLE 1. The data of four DMUs

DMU	A	B	C	D
Input	2	4	4	6
Output	1	2	3	2

Step 0: Since $n = 4$, set $n' := 2n = 8$. According to (3.2), $P_i\,(i = 1, \dots, 8)$ are calculated as follows(FIGURE 2):

$$P_1 = (2,1)^\top, P_2 = (4,2)^\top, P_3 = (4,3)^\top, P_4 = (6,2)^\top,$$

$$P_5 = 2P_1 = (4,2)^\top, P_6 = (8,4)^\top, P_7 = (8,6)^\top, P_8 = (12,4)^\top.$$

Then, $T' = \frac{1}{n'}(P_1 + \cdots + P_{n'}) = \frac{1}{8}(48,24)^\top = (6,3)^\top$. According to (3.3), $P_i'\,(i = 1, \dots, 10)$ are calculated as follows(FIGURE 3):

$$P_1' = (-4,-2)^\top, P_2' = (-2,-1)^\top, P_3' = (-2,0)^\top, P_4' = (0,-1)^\top, P_5' = (-2,-1)^\top,$$

$$P_6' = (2,1)^\top, P_7' = (2,3)^\top, P_8' = (6,1)^\top, P_9' = (1,0)^\top, P_{10}' = (0,1)^\top.$$

Set $c_1 := 1, c_2 := 2$, $\bar{n} := 10$ and $t := 0$. Go to Step 1.

Step 1: Since $\dim\{(-4,-2)^\top, (-2,-1)^\top\} = 1$, go to Step 4.

Step 4: Set $c_1 := 1, c_2 := 3$. Go to Step 1.

Step 1: Since $\dim\{(-4,-2)^\top, (-2,0)^\top\} = 2$, go to Step 2.

Step 2: Calculate W satisfying the following linear system:

$$\begin{cases} (-4,-2)^\top W = 1, \\ (-2,0)^\top W = 1. \end{cases}$$

Then, $W = (-\frac{1}{2}, \frac{1}{2})^\top$. Go to Step 3.

Step 3: We examine whether W is a vertex of $\bar{P} := \mathrm{co}(\{P_1', \dots, P_8'\})$. Since, $(P_j')^\top(-\frac{1}{2}, \frac{1}{2}) \leqslant 1 (j = 1, \dots, 8)$, $(-\frac{1}{2}, \frac{1}{2})^\top$ is a vertex of $(\bar{P})^*$. By Theorem 3.8 and the coordinate transformation moving T' to the origin, we can obtain all equations forming $P := \mathrm{co}(\{P_1, \dots, P_8\})$. In order to obtain only the efficient facets of P, we consider the vertices contained in $\{W \in \mathbb{R}^2 : W_1 \leqslant 0,\ W_2 \geqslant 0\}$. Since, $W_1 = -\frac{1}{2} \leqslant 0$ and $W_2 = \frac{1}{2} \geqslant 0$, set $V_1 := (-\frac{1}{2}, \frac{1}{2})^\top$ which is a vertex of polytope Q in FIGURE 3.2. Go to Step 4.

Step 4: Set $c_1 := 1, c_2 := 4$. Go to Step 1.

We repeat this operation to $c_1 = 9, c_2 = 10$. Then, $t = 4$. $V_1 = (-\frac{1}{2}, \frac{1}{2})^\top$, $V_2 = (-\frac{1}{4}, 0)^\top$, $V_3 = (-\frac{1}{2}, \frac{2}{3})^\top$ and $V_4 = (0, \frac{1}{3})^\top$ are all vertices except the origin of Q. Go to Step 5.

Step 5: For $t = 1$, $-\frac{1}{2}x + \frac{1}{2}y = 1 + (-\frac{1}{2}, \frac{1}{2})^\top(6,3) = -\frac{1}{2}$. Hence, $H_1 := \{(x,y) : -x + y = -1\}$. Similarly, $H_2 := \{(x,y) : x = 2\}$ and $H_3 := \{(x,y) : -\frac{1}{2}x + \frac{2}{3}y = 0\}$. For $t = 4$, since $|Y_{4,1}| = \frac{1}{3} > 0$ and $\frac{1+(0,\frac{1}{3})(6,3)}{Y_{4,1}} = 6 > 0$, $\frac{1}{3}y = \frac{1+(0,\frac{1}{3})^\top(6,3)}{2} = 1$. Hence, $H_4 := \{(x,y) : y = 3\}$. $H_1, \dots, H_4$ are all efficient facets of $\mathrm{co}(\{P_1, \dots, P_4\})$. Stop the algorithm.

By Algorithm FFA, we can obtain four vertices of Q as follows(FIGURE 4).

$$V_1 = \left(-\frac{1}{2}, \frac{1}{2}\right)^\top, V_2 = \left(-\frac{1}{4}, 0\right)^\top, V_3 = \left(-\frac{1}{2}, \frac{2}{3}\right)^\top, V_4 = \left(0, \frac{1}{3}\right)^\top.$$

By Theorem 3.6, Q^* are formed by four equations as follows(FIGURE 5).

$$H_1^{'} : \left(-\frac{1}{2}, \frac{1}{2}\right)^\top (x, y) = 1, H_2^{'} : \left(-\frac{1}{4}, 0\right)^\top (x, y) = 1,$$

$$H_3^{'} : \left(-\frac{1}{2}, \frac{2}{3}\right)^\top (x, y) = 1, H_4^{'} : \left(0, \frac{1}{3}\right)^\top (x, y) = 1.$$

By the coordinate transformation, we obtain four equations as follows(FIGURE 6).

$$H_1^{''} : \left(-\frac{1}{2}, \frac{1}{2}\right)^\top (x, y) = 1 + \left(-\frac{1}{2}, \frac{1}{2}\right)^\top (6, 3) = -\frac{1}{2},$$

$$H_2^{''} : \left(-\frac{1}{4}, 0\right)^\top (x, y) = 1 + \left(-\frac{1}{4}, 0\right)^\top (6, 3) = -\frac{1}{2},$$

$$H_3^{''} : \left(-\frac{1}{2}, \frac{2}{3}\right)^\top (x, y) = 1 + \left(-\frac{1}{2}, \frac{2}{3}\right)^\top (6, 3) = 0,$$

$$H_4^{''} : \left(0, \frac{1}{3}\right)^\top (x, y) = 1 + \left(0, \frac{1}{3}\right)^\top (6, 3) = 2.$$

By the operation at Step 5, we obtain four equations as follows(FIGURE 7).

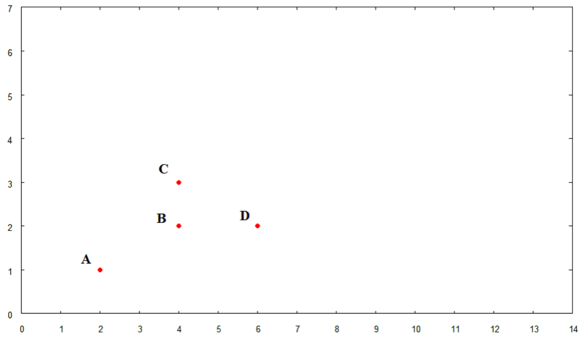
$$H_1 : \left(-\frac{1}{2}, \frac{1}{2}\right)^\top (x, y) = -\frac{1}{2}, H_2 : \left(-\frac{1}{4}, 0\right)^\top (x, y) = -\frac{1}{2},$$

$$H_3 : \left(-\frac{1}{2}, \frac{2}{3}\right)^\top (x, y) = 0, H_4 : \left(0, \frac{1}{3}\right)^\top (x, y) = \frac{1 + (0, \frac{1}{3})^\top (6, 3)}{2} = 1.$$

Then, $H_1, \ldots, H_4$ form the efficient frontiers.

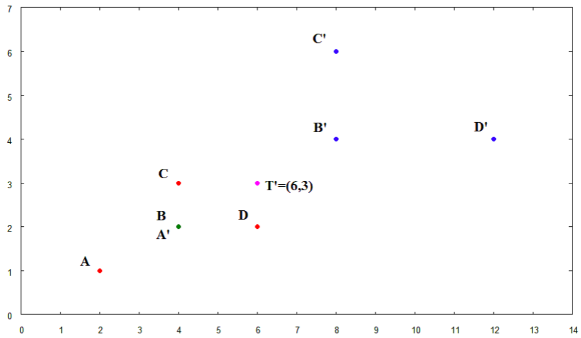
Since the PP set of the CRS model is closed convex cone, by the operation of Step 0, we can always calculate all equations of the CRS model. Moreover, the origin is contained in $\mathrm{co}(\{P_1^{'}, \ldots, P_{n'}^{'}\})$. FIGURE 4 shows the hyperplane $\{(x, y) : (P_j^{'})^\top (x, y) = 1\}$ for each $j = 1, \ldots, n^{'}$. Polytope Q means intersection of $\hat{\mathbb{R}}^{m+s}$ and $(\bar{P})^*$. We calculate all vertices of Q by performing from Step 1 to Step 4. FIGURE 5 shows the hyperplane $\{(x, y) : v^\top (x, y) = 1\}$ for each vertex($:= v$) except the origin of polytope Q in FIGURE 4. In other words, it is the polar set of polytope Q in FIGURE 4. By the coordinate transformation moving $T^{'}$ to the origin, we get Figure 6. FIGURE 7 shows all the hyperplanes calculated by Algorithm FFA. By the operation at Step 5, the hyperplane consisting of only DMUs generated at Step 0 is replaced by that of original DMUs.

By the definition of $\bar{P}$ and Assumption (A4), $\bar{P}$ is a polytope and $\mathbf{0} \in \mathrm{int}\, \bar{P}$. Hence, $(\bar{P})^*$ is a polytope and $\mathbf{0} \in \mathrm{int}\, (\bar{P})^*$. Of course, $\hat{\mathbb{R}}^{m+s}$ is a closed convex polyhedral set containing $\mathbf{0}$. Thus, the intersection of $\hat{\mathbb{R}}^{m+s}$ and $(\bar{P})^*$ is a polytope containing $\mathbf{0}$.



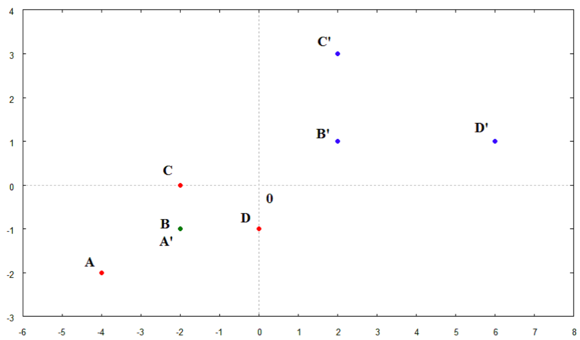
$$P_1 := A$$
$$P_2 := B$$
$$P_3 := C$$
$$P_4 := D$$

FIGURE 1. Illustration of all DMUs



$$P_5 := 2P_1 = A'$$
$$P_6 := 2P_2 = B'$$
$$P_7 := 2P_3 = C'$$
$$P_8 := 2P_4 = D'$$

FIGURE 2. We add two times the original DMUs



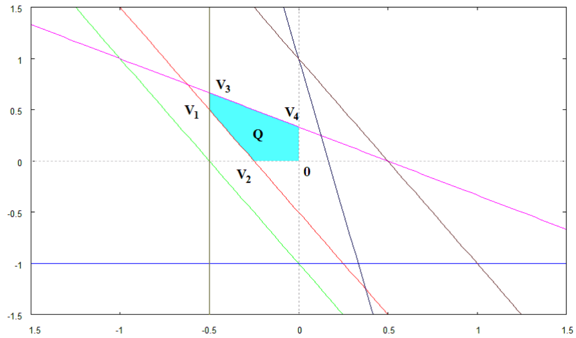
Generation of $P_1', \ldots, P_8'$.

$$P_1' := P_1 - T'$$
$$P_2' := P_2 - T'$$
$$P_3' := P_3 - T'$$
$$P_4' := P_4 - T'$$
$$P_5' := P_5 - T'$$
$$P_6' := P_6 - T'$$
$$P_7' := P_7 - T'$$
$$P_8' := P_8 - T'$$

FIGURE 3. The coordinate transformation

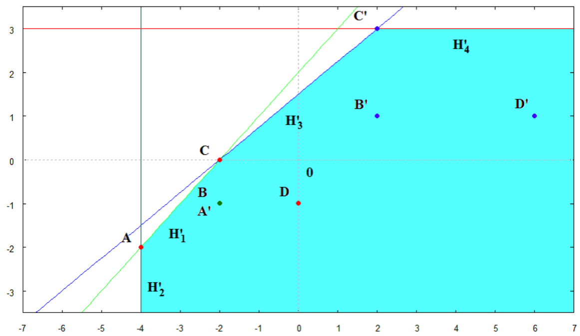
We note that $\bar{P}$ satisfies the conditions of Theorem 3.8. Moreover, the number of vertices of the intersection of $\hat{\mathbb{R}}^{m+s}$ and $(\bar{P})^*$ is a finite number. Thus, Algorithm FFA terminates within finite number of iterations. In particular, at Step 4, all combinations of choosing $m+s$ numbers as $c_1, \ldots, c_{m+s}$ from $\{1, \ldots, 2n+m+s\}$ are considered. Thus, Algorithm FFA terminates within ${}_{2n+m+s}C_{m+s}$ iterations.

3.3. **Classification of the equations.** By Algorithm FFA, we obtain all equations forming the efficient frontiers of the four models. We classify the equations by using the following lemma and theorems.



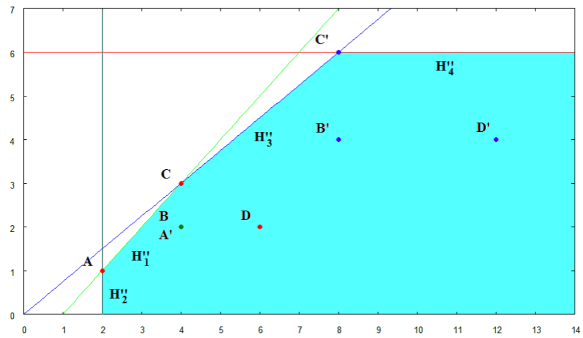
$$V_1 := (-\tfrac{1}{2}, \tfrac{1}{2})$$
$$V_2 := (-\tfrac{1}{4}, 0)$$
$$V_3 := (-\tfrac{1}{2}, \tfrac{2}{3})$$
$$V_4 := (0, \tfrac{1}{3})$$

FIGURE 4. Hyperplane that inner product of each DMU and (x, y) equals one



$$H_1' : (-\tfrac{1}{2}, \tfrac{1}{2})^\top (x, y) = 1$$
$$H_2' : (-\tfrac{1}{4}, 0)^\top (x, y) = 1$$
$$H_3' : (-\tfrac{1}{2}, \tfrac{2}{3})^\top (x, y) = 1$$
$$H_4' : (0, \tfrac{1}{3})^\top (x, y) = 1$$

FIGURE 5. The polar set of Q in FIGURE 4



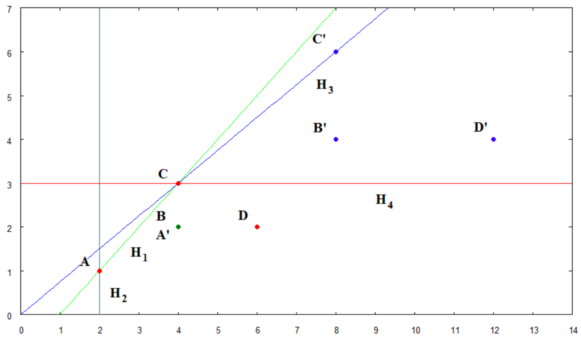
$$H_1'' : (-\tfrac{1}{2}, \tfrac{1}{2})^\top (x, y) = -\tfrac{1}{2}$$
$$H_2'' : (-\tfrac{1}{4}, 0)^\top (x, y) = -\tfrac{1}{2}$$
$$H_3'' : (-\tfrac{1}{2}, \tfrac{2}{3})^\top (x, y) = 0$$
$$H_4'' : (0, \tfrac{1}{3})^\top (x, y) = 2$$

FIGURE 6. The coordinate transformation moving T' to the origin

Lemma 3.10. *Let* $E \subset \mathbb{R}^n$ *be a polytope satisfying* $\mathbf{0} \in \operatorname{int} E$. *Then, for each* $a \in V(E)$, $\dim(E^* \cap \{x \in \mathbb{R}^n : a^\top x = 1\}) = n - 1$.

Proof. Since E is bounded, by Theorem 3.7, $\mathbf{0} \in \operatorname{int} E^*$. This implies that $\dim E^* = n$. Moreover, since $\mathbf{0} \in \operatorname{int} E$ and $a \in V(E) \subset \operatorname{bd} E$, $a \neq \mathbf{0}$ and hence $\dim(E^* \cap \{x : a^\top x = 1\}) \leqslant n - 1$. Futhermore, since E^* is a polytope and $a \in E$,

$$E^* = \operatorname{co} V(E^*) \subset \{x : a^\top x \leqslant 1\} \tag{3.5}$$



$$H_1 : (-\tfrac{1}{2}, \tfrac{1}{2})^\top(x,y) = -\tfrac{1}{2}$$
$$H_2 : (-\tfrac{1}{4}, 0)^\top(x,y) = -\tfrac{1}{2}$$
$$H_3 : (-\tfrac{1}{2}, \tfrac{2}{3})^\top(x,y) = 0$$
$$H_4 : (0, \tfrac{1}{3})^\top(x,y) = 1$$

FIGURE 7. All hyperplanes obtained by Algorithm FFA

In order to obtain a contradiction, we suppose that

$$l := \dim(E^* \cap \{x : a^\top x = 1\}) \leqslant n - 2.$$

Then, by (5), there exists $b^1, \ldots, b^{l+1} \in (V(E^*) \cap \{x : a^\top x = 1\})$ such that $b^1, \ldots, b^{l+1}$ are affine independent. Then, $\dim\{b^1, \ldots, b^{l+1}\} \leqslant n-1$. Therefore, there exists $b \in \mathbb{R}^n \backslash 0$ such that $b^\top b^i = 0\, i = 1, \ldots, l+1$. We note that $v^\top a < 1$ for each $v \in V(E^*)\backslash\{b^1, \ldots, b^{l+1}\}$. Hence, for each $v \in V(E^*)\backslash\{b^1, \ldots, b^{l+1}\}$, let

$$\alpha^v := \begin{cases} 1 & \text{if } v^\top b = 0, \\ \dfrac{1 - v^\top a}{|v^\top b|} & \text{if } v^\top b \neq 0. \end{cases}$$

Then, by setting $\bar{\alpha} := \min\{\alpha^v : v \in V(E^*)\backslash\{b^1, \ldots, b^{l+1}\}\}$, we have $v^\top(a \pm \bar{\alpha} b) = v^\top a \pm \bar{\alpha} v^\top b \leqslant v^\top a + \bar{\alpha}|v^\top b| \leqslant 1$ for each $v \in V(E^*)\backslash\{b^1, \ldots, b^{l+1}\}$. Moreover, for each b^i $(i = 1, \ldots, l+1)$, ${b^i}^\top(a \pm \bar{\alpha} b) = {b^i}^\top a \pm \bar{\alpha}({b^i}^\top b) = {b^i}^\top a = 1$. This implies that $a - \bar{\alpha} b,\, a + \bar{\alpha} b \in (V(E^*))^* = E$. Since $a = \frac{1}{2}(a - \bar{\alpha} b + a + \bar{\alpha} b)$, this contradicts $a \in V(E)$. Consequently, $\dim(E^* \cap \{x : a^\top x = 1\}) = n - 1$. $\square$

Theorem 3.11. *Assume that $H_{p,q,c} = \{(x,y) \in \mathbb{R}^{m+s} : q^\top y - p^\top x = c\}$ are calculated by Algorithm* FFA. *If $c = 0$, then $H_{p,q,c} \cap \mathrm{T}_{\mathrm{CRS}}$ is a facet of $\mathrm{T}_{\mathrm{CRS}}$.*

Proof. Since p, q and c are constructed at Step 2 of Algorithm FFA, $\dim H_{p,q,c} = m + s - 1$. By Assumption (A4), $\dim \mathrm{T}_{\mathrm{CRS}} = m + s$. By Lemma 3.10, $\dim H_{p,q,c} \cap (\bar{P} + T) = m + s - 1$. By the definition of $\mathrm{T}_{\mathrm{CRS}}$, $\bar{P} + T \subset \mathrm{T}_{\mathrm{CRS}}$. Therefore, $\dim H_{p,q,c} \cap \mathrm{T}_{\mathrm{CRS}} = m+s-1$. By Step 2 of Algorithm FFA, $(-p^\top, q^\top)^\top(x_j^\top, y_j^\top) \leqslant 0$ $(j = 1, \ldots, n)$. For each $(x^\top, y^\top)^\top \in \mathrm{T}_{\mathrm{CRS}}$, there exists $\lambda' \geqslant 0$ such that $x \geqslant \sum_{j=1}^n \lambda'_j x_j$, $0 \leqslant y \leqslant \sum_{j=1}^n \lambda'_j y_j$. Then $(-p^\top, q^\top)(x^\top, y^\top)^\top = -p^\top x + q^\top y \leqslant -p^\top \sum_{j=1}^n \lambda'_j x_j + q^\top \sum_{j=1}^n \lambda'_j y_j \leqslant 0$. Consequently, $H_{p,q,c} \cap \mathrm{T}_{\mathrm{CRS}}$ is a facet of $\mathrm{T}_{\mathrm{CRS}}$. $\square$

Theorem 3.12. *If $c \neq 0$, then $H_{p,q,c} \cap \mathrm{T}_{\mathrm{VRS}}$ is a facet of $\mathrm{T}_{\mathrm{VRS}}$.*

Proof. By Assumption (A4), $\dim \mathrm{T}_{\mathrm{VRS}} = m + s$. By Lemma 3.10 and Step 4 of Algorithm FFA, $\dim H_{p,q,c} \cap \mathrm{co}\{P_1, \ldots, P_n\} = m+s-1$. By the definition of $\mathrm{T}_{\mathrm{VRS}}$, $\mathrm{co}\{P_1, \ldots, P_n\} \subset \mathrm{T}_{\mathrm{VRS}}$. Therefore, $\dim H_{p,q,c} \cap \mathrm{T}_{\mathrm{VRS}} = m + s - 1$. By Step 2 of Algorithm FFA, $(-p^\top, q^\top)^\top(x_j^\top, y_j^\top) \leqslant c$ $(j = 1, \ldots, n)$. For each $(x^\top, y^\top)^\top \in$

$\mathrm{T_{VRS}}$, there exists $\lambda' \geqslant 0$ such that $x \geqslant \sum_{j=1}^{n} \lambda_j' x_j,\ 0 \leqslant y \leqslant \sum_{j=1}^{n} \lambda_j' y_j,\ \sum_{j=1}^{n} \lambda_j' = 1$. Then $(-p^\top, q^\top)(x^\top, y^\top)^\top = -p^\top x + q^\top y \leqslant -p^\top \sum_{j=1}^{n} \lambda_j' x_j + q^\top \sum_{j=1}^{n} \lambda_j' y_j \leqslant c$. Consequently, $H_{p,q,c} \cap \mathrm{T_{VRS}}$ is a facet of $\mathrm{T_{VRS}}$. □

Theorem 3.13. *If $c \leqslant 0$, then $H_{p,q,c} \cap \mathrm{T_{IRS}}$ is a facet of $\mathrm{T_{IRS}}$.*

Proof. We can complete the proof in a way similar to Theorem 3.12. □

Theorem 3.14. *If $c \geqslant 0$, then $H_{p,q,c} \cap \mathrm{T_{DRS}}$ is a facet of $\mathrm{T_{DRS}}$.*

Proof. We can complete the proof in a way similar to Theorem 3.12. □

Theorem 3.15. *If $c = 0$ and $\dim(\{(x_i^\top, y_i^\top)^\top : i = 1, \dots, n\} \cap H_{p,q,c}) = m + s - 1$, then $H_{p,q,c} \cap \mathrm{T_{VRS}}$ is a facet of $\mathrm{T_{VRS}}$.*

Proof. Since $\dim(\{(x_i^\top, y_i^\top)^\top : i = 1, \dots, n\} \cap H_{p,q,c}) = m+s-1$, $\dim H_{p,q,c} \cap \mathrm{T_{VRS}} = m + s - 1$. By Step 2 of Algorithm FFA, $(-p^\top, q^\top)^\top (x_j^\top, y_j^\top) \leqslant 0$ $(j = 1, \dots, n)$. For each $(x^\top, y^\top)^\top \in \mathrm{T_{VRS}}$, there exists $\lambda' \geqslant 0$ such that $x \geqslant \sum_{j=1}^{n} \lambda_j' x_j,\ 0 \leqslant y \leqslant \sum_{j=1}^{n} \lambda_j' y_j,\ \sum_{j=1}^{n} \lambda_j' = 1$. Then $(-p^\top, q^\top)(x^\top, y^\top)^\top = -p^\top x + q^\top y \leqslant -p^\top \sum_{j=1}^{n} \lambda_j' x_j + q^\top \sum_{j=1}^{n} \lambda_j' y_j \leqslant 0$. Therefore, $H_{p,q,c} \cap \mathrm{T_{VRS}}$ is a facet of $\mathrm{T_{VRS}}$. □

4. Applications of the Algorithm FFA

In this Section, by applying Algorithm FFA, we identify the properties of DMUs. In Section 4.1, we we calculate efficiency scores of all DMUs and classify efficient DMUs which are efficient in the VRS model into three types of RTS. In Section 4.2, we propose an algorithm for calculating an improvement to become an efficient unit in the CRS model by the smallest change. The improvement with the inner-product norm proposed in [10] is a special case of this one. In Section 4.3, we propose an algorithm for calculating an improvement to become an efficient unit in the CRS model by the smallest change and containing in $\mathrm{T_{VRS}}$. In Section 4.4, we perform a numerical analysis by utilizing algorithms provided in this paper.

4.1. **Efficiency score and RTS.** Since we can classify equations by using Theorems 3.11, 3.12, 3.13, 3.14 and 3.15, we can obtain efficiency scores of the four models easily by substituting the input and output values of each DMU as follows.

Theorem 4.1 (Jahanshahloo, Lotfi and Zohrehbandian [6])**.** *Let H_j be a hyperplane forming the efficient frontier of the CRS model for each $j \in S_c$, where $\mathrm{H}_j := \{(x, y) : U_j^\top y - V_j^\top x = 0\}$, then the efficiency score of DMU_k in the CRS model is obtained as follows.*

$$\mathrm{Eff}(\mathrm{DMU}_k) = \max \left\{ \frac{U_j^\top y_k}{V_j^\top x_k} : j \in S_c \right\}.$$

Theorem 4.2 (Jahanshahloo, Lotfi and Zohrehbandian [6])**.** *Let H_j be a hyperplane forming the efficient frontiers of the VRS, DRS, IRS models for each $j \in S_v,\ S_i,\ S_d$, where $\mathrm{H}_j := \{(x, y) : U_j^\top y - V_j^\top x + U_{jk} = 0\}$, then the efficiency scores of DMU_k*

in the VRS, DRS, IRS *models are obtained as follows.*

$$\mathrm{Eff}(\mathrm{DMU}_k) = \max\left\{\frac{U_j^\top y_k + U_{jk}}{V_j^\top x_k} : V_j^\top x_k \neq 0,\, j \in S_v,\ S_i,\ S_d\right\}.$$

The RTS express a type of the efficiency by the change of the scale about the activity of DMUs. There exist three types of the RTS, that is, the increasing RTS, the decreasing RTS and the constant RTS. The increasing RTS improves the efficiency by expanding the scale. Likewise, the decreasing RTS improves the efficiency by contracting the scale. The constant RTS means that it is desirable to maintain the present scale.

By Jahanshahloo, Lotfi, Zohrehbandian [6], the RTS are given as follows. Let h be the number of the hyperplanes calculated by Algorithm FFA. For each $j = 1, ..., h$, H_j be the hyperplane defined by $\{(x, y) : U_j^\top y - V_j^\top x = U_{jk}\}$. Let S_c be the index set of all hyperplanes of the CRS model. Similarly, let $S_v,\ S_i,\ S_d$ be the index sets of all hyperplanes of the VRS, IRS, DRS models, respectively.

Let θ_k be an optimal solution (VRS_k) for DMU_k and

$$S_k := \left\{U_{jk} : \frac{U_j^\top y + U_{jk}}{V_j^\top x} = \theta_k,\, V_j^\top x \neq 0,\, j \in S_v\right\}.$$

Theorem 4.3 (Jahanshahloo, Lotfi and Zohrehbandian [6]). *The RTS is classified as the following.*

(i): DMU_k is said to be the increasing RTS if $\min\{S_k\} < \max\{S_k\} \leqslant 0$ or $\min\{S_k\} = \max\{S_k\} < 0$.

(ii): DMU_k is said to be the decreasing RTS if $0 \leqslant \min\{S_k\} < \max\{S_o\}$ or $0 < \min\{S_k\} = \max\{S_k\}$.

(iii): DMU_k is said to be the constant RTS if $\min\{S_k\} < 0 < \max\{S_k\}$ or $\min\{S_k\} = \max\{S_k\} = 0$.

For the data in Table 1, we identify the RTS by Theorem 3.11. In general, the RTS is considered only VRS-efficient DMUs, but by Theorem 3.11, the RTS can be considered for all DMUs. Table 2 shows the classification of the RTS with the data in Table 1.

Table 2. Classification of the RTS

DMU	$\min\{S_{\mathrm{DMU}}\}$	$\max\{S_{\mathrm{DMU}}\}$	RTS
A	-2	-1	increasing
B	-1	-1	increasing
C	-1	3	constant
D	-1	-1	increasing

4.2. **Improvement over an efficient frontier of the CRS model.** We propose an improvement of DMU_k which is inefficient in the CRS model. Let $\{(x,y) : U_j^\top y - V_j^\top x = 0\},\ j \in S_c$ be the all hyperplanes of the CRS model. We denote the efficient frontier of the CRS model $\mathrm{F}_{\mathrm{CRS}}$. For each $j \in S_c$, let $w_j = (-V_j^\top\ U_j^\top)^\top$ and $z = (x^\top, y^\top)^\top$. Then we can formulate the hyperplanes as $H_j := \{z : z^\top w_j = 0\}$ for each $j \in S_c$. We define the norm by a symmetric positive semidefinite matrix $A \in \mathbb{R}^{(m+s)\times(m+s)}$ as follows.

$$||z - P_k||_A := \sqrt{(z-P_k)^\top A (z-P_k)}.$$

The improvement of DMU_k over an efficient frontier of the CRS model is obtained by solving the following problem.

$$(\mathrm{C}_k) \begin{cases} \text{minimize} & ||z - P_k||_A \\ \text{subject to} & z \in \mathrm{F}_{\mathrm{CRS}}. \end{cases}$$

Since we can calculate all hyperplanes of the CRS model by Algorithm FFA, we can solve Problem (C_k). A solution of Problem (C_k) is a point having the shortest norm from P_k over an efficient frontier of the CRS model.

Example 4.4. If $A = I_{m+s}$, then a solution of Problem (C_k) is a point having the shortest norm from P_k over an efficient frontier of the CRS model. In addition, if

$$A = A_k := \begin{pmatrix} \left(\frac{1}{P_{1,k}}\right)^2 & & 0 \\ & \ddots & \\ 0 & & \left(\frac{1}{P_{m+s,k}}\right)^2 \end{pmatrix},$$

then a solution of Problem (C_k) is a point considered the ratio of inputs and outputs and over an efficient frontier of the CRS model.

Let z^* be an optimal solution of Problem (C_k), then $IC_k := z^* - P_k$ is the improvement such that DMU_k become an efficient unit in the CRS model by the smallest change. We propose following algorithm for solving Problem (C_k). An improvement for inefficient DMU_k in the CRS model is obtained by the following algorithm:

Algorithm ICRS:

Step 1:
Let $OC_{k,j}$ be an optimal solution of the following quadratic problem.

$$(\mathrm{C}_{k,j}) \begin{cases} \text{minimize} & ||z - P_k||_A \\ \text{subject to} & z^\top w_j = 0. \end{cases}$$

Go to Step 2.

Step 2:
Select $j' \in \arg\min\{||OC_{k,j} - P_k||_A : j \in S_c\}$. Let $IC_k := OC_{k,j'} - P_k$ be the improvement and $OC_k := OC_{k,j'}$. This algorithm terminates.

Problem ($C_{k,j}$) is a standard quadratic programming problem. Hence, ($C_{k,j}$) can be solved by the nonlinear optimization techniques (e.g. [2]). For each $j \in S_c$, $OC_{k,j}$ is contained in H_j. Thus, OC_k is a CRS-efficient point.

Theorem 4.5. *Let OC_k be an optimal solution of Algorithm* ICRS. *Then, $OC_k \in$* $\mathrm{F}_{\mathrm{CRS}}$.

Proof. In order to obtain a contradiction, we suppose that $OC_k \notin \mathrm{F}_{\mathrm{CRS}}$. Then, there exists $j \in S_c$ such that $OC_k^\top w_j > 0$. Since $P_k^\top w_j < 0$, $(\alpha OC_k + (1-\alpha)P_k)^\top w_j = 0$ for some $\alpha \in (0,1)$. Then, from the definition of $OC_{k,j}$, we have the following inequality: $||OC_{k,j} - P_k||_A \leqslant ||\alpha OC_k + (1-\alpha)P_k - P_k||_A < ||OC_k - P_k||_A$. This contradicts the optimality of OC_k for Algorithm ICRS. Consequently, $OC_k \in \mathrm{F}_{\mathrm{CRS}}$. □

We propose IC_k as an improvement for DMU_k. If $IC_{i,k} < 0$ $(i = 1, \ldots, m)$ then input i should be decreased to become an efficient unit. If $IC_{i,k} > 0$ $(i = m+1, \ldots, m+s)$ then output i should be increased to become an efficient unit. Sometimes, the improvement may not be allowed in the actual situations. So, we present another improvement based on careful study of multiple models in next subsection.

4.3. **Improvement contained in** $\mathrm{T}_{\mathrm{VRS}}$**.** We propose an improvement of DMU_k which is inefficient in the CRS model. Let $\{(x,y) : U_j^\top y - V_j^\top x = 0\},\ j \in S_c$ be all the hyperplanes of the CRS model. For each $j \in S_c$, let $w_j = (-V_j^\top\ U_j^\top)^\top$ and $z = (x^\top, y^\top)^\top$. Then we can formulate the hyperplanes as $H_j := \{z : z^\top w_j = 0\}$ for each $j \in S_c$. Let $\{(x,y) : S_i^\top y - T_i^\top x + S_{ik} = 0\},\ i \in S_v$ be all the hyperplanes of the VRS model. The improvement of DMU_k contained in $\mathrm{T}_{\mathrm{VRS}}$ is obtained by solving the following problem.

$$(\mathrm{B}_k)\begin{cases} \text{minimize} & ||z - P_k||_A \\ \text{subject to} & z \in \mathrm{F}_{\mathrm{CRS}},\ x \in \mathrm{T}_{\mathrm{VRS}}. \end{cases}$$

Let z^* be an optimal solution of Problem (B_k), then $IB_k := z^* - P_k$ is the improvement so that DMU_k become an efficient unit in the CRS model and is contained in $\mathrm{T}_{\mathrm{VRS}}$. We can consider the improvement which is contained in $\mathrm{T}_{\mathrm{IRS}}$ or $\mathrm{T}_{\mathrm{DRS}}$ similarly. We propose following algorithm for solving Problem (B_k). An improvement for inefficient DMU_k in the CRS model is obtained by the following algorithm:

Algorithm IVRS:

Step 1:
Let $OB_{k,j}$ be an optimal solution of the following quadratic problem.

$$(\mathrm{B}_{k,j})\begin{cases} \text{minimize} & ||z - P_k||_A \\ \text{subject to} & z^\top w_j = 0, \\ & S_i^\top y - T_i^\top x + S_{ik} \leqslant 0,\ \forall i \in S_v. \end{cases}$$

Go to Step 2.

Step 2:
Select $j^{'} \in \arg\min\{||OB_{k,j} - P_k||_A : j \in S_c\}$. Let $IB_k := OB_{k,j'} - P_k$ be the improvement and $OB_k := OB_{k,j'}$. This algorithm terminates.

Problem ($\mathrm{B}_{k,j}$) is a standard quadratic programming problem. Hence, ($\mathrm{B}_{k,j}$) can be solved by the nonlinear optimization techniques (e.g. [2]). For each $j \in S_c$, $OB_{k,j}$ is contained in the intersection of H_j and $\mathrm{T}_{\mathrm{VRS}}$. Thus, OB_k is a CRS-efficient point which is contained in $\mathrm{T}_{\mathrm{VRS}}$.

Theorem 4.6. *Let OB_k be an optimal solution of Algorithm* IVRS. *Then, $OB_k \in \mathrm{F}_{\mathrm{CRS}}$ and $OB_k \in \mathrm{T}_{\mathrm{VRS}}$.*

Proof. We can complete the proof in a way similar to Theorem 4.1. □

We propose IB_k as an improvement contained in $\mathrm{T}_{\mathrm{VRS}}$ for DMU_k. Then, the improved unit according to the improvement from P_k belongs to the $\mathrm{T}_{\mathrm{VRS}}$. If $IB_{i,k} < 0$ $(i = 1, \ldots, m + s)$ then we should decrease ith element of P_k to become efficient. Moreover, if $IB_{i,k} > 0$ $(i = 1, \ldots, m + s)$ then we should increase ith element of P_k to become efficient.

4.4. **Numerical analysis.** Now, we perform a numerical analysis for 10 Japanese banks by utilizing algorithms provided in this paper. As shown in Table 3, each bank has the ordinary profit as the single output. The number of employees and total assets are the two inputs used to generate the output.

Table 3. Inputs and Output values for 10 Japanese banks, 2008

Bank	Input1 (persons)	Input2 (one hundred million Japanese yen)	Output (one hundred million Japanese yen)
B1	3701	119895	3179
B2	3675	98359	2688
B3	3659	80955	2180
B4	3004	59600	1563
B5	2887	66373	1477
B6	2872	90984	2450
B7	2752	60770	1852
B8	2506	49008	1137
B9	2268	41151	1148
B10	2148	41158	1124

The efficiency scores and the RTS are shown in the Table 4. All efficiency scores are calculated by using Theorems 4.1 and 4.2, and the RTS are obtained by using Theorem 4.3.

Three banks are CRS-efficient and they do not have to think the improvement. Another bank's improvements are given by Table 5-8. The improvement over an efficient frontier of CRS model ($A = A_k$) is given in Table 5. Improvements contained in $\mathrm{T}_{\mathrm{VRS}}$, $\mathrm{T}_{\mathrm{IRS}}$ and $\mathrm{T}_{\mathrm{DRS}}$ ($A = A_k$) are given in Tables 6, 7 and 8, respectively. The improvement over an efficient frontier of CRS model think decreasing inputs and increasing outputs. In contrast, other improvements might increasing inputs or decreasing outputs. This means that the DMU is impossible to become CRS-efficient in the PP set of the other models by decreasing inputs. Similarly, the

TABLE 4. DEA analysis for 10 Japanese banks, 2008

Bank	CRS	VRS	IRS	DRS	RTS
B1	1.000000	1.000000	1.000000	1.000000	C
B2	0.961359	0.996536	0.961359	0.996536	-
B3	0.884268	0.931186	0.884268	0.931186	-
B4	0.860520	0.884500	0.884500	0.860520	-
B5	0.741447	0.814268	0.814268	0.741447	-
B6	1.000000	1.000000	1.000000	1.000000	C
B7	1.000000	1.000000	1.000000	1.000000	C
B8	0.761275	0.859975	0.859975	0.761275	-
B9	0.915398	1.000000	1.000000	0.915398	I
B10	0.896108	1.000000	1.000000	0.896108	I

DMU is impossible to become CRS-efficient in the PP set of the other models by increasing outputs.

TABLE 5. Improvement over $\mathrm{F}_{\mathrm{CRS}}$ $(A = A_k)$

Bank	Input1	Input2	Output
B1	-	-	-
B2	-43.28	-358.63	90.72
B3	0.00	-5290.05	125.93
B4	0.00	-4769.59	107.99
B5	0.00	-11664.62	190.27
B6	-	-	-
B7	-	-	-
B8	0.00	-7415.07	130.57
B9	0.00	-1895.63	48.33
B10	0.00	-2368.56	58.13

TABLE 6. Improvement contained in $\mathrm{T}_{\mathrm{VRS}}$ $(A = A_k)$

Bank	Input1	Input2	Output
B1	-	-	-
B2	-802.17	-7345.86	-237.27
B3	-789.48	-20185.00	-328.00
B4	-91.89	1170.00	289.00
B5	27.46	-5603.00	375.00
B6	-	-	-
B7	-	-	-
B8	246.09	11785.65	715.47
B9	637.19	19619.00	704.00
B10	604.85	19826.76	732.25

TABLE 7. Improvement contained in T_{IRS} $(A = A_k)$

Bank	Input1	Input2	Output
B1	-	-	-
B2	-782.55	-6663.43	-220.03
B3	-766.14	10754.95	288.33
B4	-246.99	1175.20	289.16
B5	216.46	2166.65	611.68
B6	-	-	-
B7	-	-	-
B8	276.21	11833.95	717.19
B9	485.72	19635.57	704.51
B10	613.94	19628.56	728.51

TABLE 8. Improvement contained in T_{DRS} $(A = A_k)$

Bank	Input1	Input2	Output
B1	-	-	-
B2	-1196.07	-17582.18	-558.71
B3	-1203.75	-1025.48	-71.04
B4	-1347.67	-7127.44	-150.04
B5	-1256.57	-14720.64	-86.13
B6	-	-	-
B7	-	-	-
B8	-874.90	2665.70	254.44
B9	-644.53	10281.06	236.93
B10	-555.89	9280.97	234.19

The number of iterations of Algorithm FFA increases sharply as the number of inputs, outputs and DMUs increase. For example, in the case where $m = 2$, $s = 1$ and $n = 10$, the number of iterations of Algorithm FFA is 1,771. In the case where $m = 3$, $s = 2$ and $n = 20$, the number of iterations of Algorithm FFA is 1,221,759. In TABLE 9, we show the averages of the computational times for 100 test problems to calculate efficiency scores of the four models for all DMUs. The input and output values are randomly-determined. In the case of relatively small scale, the computational time of proposed algorithm is faster than that of solving linear programming problems. In the case of large scale, there exist some efforts to decrease the number of iterations. Since the efficient frontier is formed by efficient DMUs, we can execute Algorithm FFA by using only VRS-efficient DMUs. Then, in the case of TABLE 3, the number of iterations decrease from 1,771 to 286.

5. CONCLUSION

In this paper, we have proposed Algorithm FFA for constructing all equations simultaneously forming the efficient frontiers of the CRS, VRS, IRS and DRS models. These models can utilize the properties of the polar sets, since the PP sets of these

TABLE 9. Comparison of the computational time (second)

Situation	Algorithm FFA	Conventional method
2 Inputs and 1 Output		
10 DMUs	0.0157	0.1155
20 DMUs	0.0470	0.5320
3 Inputs and 2 Outputs		
10 DMUs	0.3270	0.3322
20 DMUs	7.3769	0.8377

models are convex. By calculating all equations forming the efficient frontiers of the four models, the efficiency score for each model can be obtained easily. The computational time of Algorithm FFA is inferior to solving linear programming problems. However, we can obtain not only the efficiency scores of the four models but also the RTS and improvements by using the equations.

Moreover, we have presented two kinds of the improvements by applying Algorithm FFA. The improvement proposed in Section 4.2 turns to the closest point over $\mathrm{F}_{\mathrm{CRS}}$ by using the norm based on an order matrix defined by the decision maker. Furthermore, the improvement proposed in Section 4.3 is the direction to the target contained in the intersection of $\mathrm{F}_{\mathrm{CRS}}$ and $\mathrm{T}_{\mathrm{VRS}}$. The improvement target contained in the PP sets of multiple models is first calculated by clarifying all equations forming the multiple efficient frontiers.

REFERENCES

[1] 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.

[2] M. S. Bazaraa, H. D. Sherali and C. M. Shetty, *Nonlinear Programming - Theory and Algorithms*, 2nd ed., John Wiley, New York, 1993. .

[3] A. Charns, W. W. Cooper and E. Rhodes, *Measuring Efficiency of Decision Making Units*, European Journal of Operations Research **2** (1978), 429–444.

[4] A. Charnes, W. W. Cooper and R. M. Thrall, *A Structure for Classifying and Characterizing Efficiency and Inefficiency in Data Envelopment Analysis*, The Journal of Productivity Analysis **2** (1991), 197–237.

[5] R. Fare and S. Grosskopt, *A nonparametric cost approach to scale efficiency* Scand. J. of Economics **87** (1985), 594–604.

[6] G. R. Jahanshahloo, F. H. Lotfi and M. Zohrehbandian, *Finding the piecewise linear frontier production function in data envelopment analysis*, Applied Mathematics and Computation **163** (2005), 483–488.

[7] M. B. Jonathan and A. L. Adrian, *Convex Analysis and Nonlinear Optimization Theory and Examples*, 2nd ed., Springer, New York, 2006.

[8] H. Konno, P. T. Thach and H. Tuy, *Optimization on Low Rank Nonconvex Structures*, Kluwer Academic Publishers, Dordrecht, 1997.

[9] L. M. Seiford and R. M. Thrall, *Recent developments in DEA, the mathematical programming approach to frontier analysis*, Journal of Econometrics **46** (1990), 7–38.

[10] A. Takeda and H. Nishino, *On measuring the inefficiency with the inner-product norm in data envelopment analysis*, European Journal of Operational Research **133** (2001), 377–393.

[11] K. Tone, *A slacks-based measure of efficiency in data envelopment analysis*, European Journal of Operational Research **130** (2001), 498–509.

Manuscript received May 31, 2009
revised April 10, 2010

S. Washio
Deparment of Infomation Science and Engineering Graduate School of Science and Technology, Niigata University, Niigata 950-2181, Japan
E-mail address: `satoshi@m.sc.niigata-u.ac.jp`

S. Yamada
Deparment of Infomation Science and Engineering Graduate School of Science and Technology, Niigata University, Niigata 950-2181, Japan
E-mail address: `yamada@math.sc.niigata-u.ac.jp`

T. Tanaka
Deparment of Infomation Science and Engineering Graduate School of Science and Technology, Niigata University, Niigata 950-2181, Japan
E-mail address: `tamaki@math.sc.niigata-u.ac.jp`

T. Tanino
Deparment of Electronics and Information Systems, Osaka University, Suita, Osaka 565-0871, Japan
E-mail address: `tanino@eei.eng.osaka-u.ac.jp`