数理解析研究所講究録
第 1685 巻 2010 年 200-209

Improvement indices based on careful study of the feasibility in DEA

鷲尾 哲 [1](Satoshi WASHIO), 山田 修司 [2](Syuuji YAMADA),
田中 環 [3](Tamaki TANAKA), 谷野 哲三 [4](Tetsuzo TANINO)
[1,2,3] 新潟大学大学院 自然科学研究科
Graduate School of Science and Technology, Niigata University
[4] 大阪大学大学院 工学研究科
Graduate School of Engineering, Osaka University

Abstract

In this study, we propose two kinds of improvement indices for making inefficient decision making units(DMUs) efficient in data envelopment analysis(DEA). Moreover, we propose an algorithm which calculates all equations forming the efficient frontiers of the CCR and BCC models. By utilizing the algorithm, we propose an algorithm for calculating such indices based on quadratic programming techniques.

1 Introduction

DEA is a non-parametric analytical methodology used for efficiency analysis of a DMU performing similar tasks in a production system that consumes inputs to produces outputs. The idea of DEA was introduced by Charns, Cooper and Rhodes [3]. The CCR model has its production frontier spanned by the convex cone of existing DMUs. The BCC model proposed by Banker, Charnes and Cooper [1] has its production frontier spanned by the convex hull of existing DMUs. By solving such models for each DMU, we can obtain the evaluated value of the efficiency. Moreover, DEA also provides improvement index, which can be used to improve the efficiency of the DMU. However, it is often diffcult to improve the values of inputs and outputs according to the index. For example, the indices obtained by the input-oriented CCR model or the input-oriented BCC model keep the output fixed at current level and improve the only input values. It is often difficult for some DMUs to accept. Therefore, we consider the change of input and output values at the same time. Consequently, we propose improvement indices based on careful study of the feasibility.

In this paper, we suggest two types of improvement indices for making inefficient DMUs efficient in the CCR model with the minimal change of input and output values. In the first index, we consider only the minimal change of input and output values. However, the improvement index is sometimes impossible in the actual situations. Hence, we present another improvement indices by focusing on the feasibility. Since the production possibility set of the BCC model can be identified as the feasible region of DMUs, the second index is constrained by the production possibility set of the BCC model.

The constitution of this paper is as follows. In Section 2, we introduce the DEA models with the convex production possibility sets. In Section 3, we suggest an algorithm for constructing the equations forming the efficient frontier. In Section 4, we propose two improvement indices to derive an efficient unit in the CCR model.

Through this paper, we use the following notation: Let R^n be the n-dimensional Euclidean space. For a vector $a \in R^n, a^\top$ denotes the transposed vector of a. Let I_n be the unit matrix on R^n. For a subset $S \subset R^n$, $\dim S$ denotes the dimension of S. For a vector $a \in R^n, ||a||$ denotes the Euclidean norm of a.

For subset $S \subset R^n$, bd S, cone S and conv S denote the boundary, conical hull and convex hull of S. For nutural numbers a and b $(a \geq b)$, ${}_aC_b := \frac{a!}{b!(a-b)!}$.

2 DEA models with convex production possibility sets

Through this paper, n denotes the number of DMUs. Each DMU consumes m different inputs to produce s different outputs. Specifically, for each $j \in \{1, \ldots, n\}$, DMU(j) has an input vector $x(j) := (x(j)_1, \ldots, x(j)_m)^\top$ and an output vector $y(j) := (y(j)_1, \ldots, y(j)_s)^\top$. Then, we assume the following conditions.

(A1) $x(j) > 0$, $y(j) > 0$ for each $j \in \{1, \ldots, n\}$.

(A2) $(x(j_1)^\top, y(j_1)^\top) \neq (x(j_2)^\top, y(j_2)^\top)$ for each $j_1, j_2 \in \{1, \ldots, n\}$ $(j_1 \neq j_2)$.

(A3) $n > m + s$.

(A4) $\dim(\{x(1), \ldots, x(n)\} \times \{y(1), \ldots, y(n)\}) = m + s$.

2.1 CCR model

In order to calculate an efficiency of DMU(k) for $1 \leq k \leq n$, the CCR model is formulated as follows:

$$
(\mathrm{CCR}(k)) \begin{cases} \text{minimize} & \theta \\ \text{subject to} & \theta x(k)_i - \sum_{j=1}^{n} \lambda_j x(j)_i \geq 0, \; i = 1, \ldots, m, \quad (1) \\ & \sum_{j=1}^{n} \lambda_j y(j)_r - y(k)_r \geq 0, \; r = 1, \ldots, s, \quad (2) \\ & \lambda_j \geq 0, \; j = 1, \ldots, n, \quad (3) \\ & \theta \in R. \end{cases}
$$

Let $\theta^*_{\mathrm{CCR}}(k)$ denote the optimal value of (CCR(k)). By (2) and (3), we have $\lambda := (\lambda_1, \ldots, \lambda_n)^\top \neq (0, \ldots, 0)^\top$ and hence $\lambda_{\hat{j}} > 0$ for some $\hat{j} \in \{1, \ldots, n\}$. From (1), we have $\theta^*_{\mathrm{CCR}} x(k)_i \geq \sum_{j=1}^{n} \lambda_j x(j)_i \geq \lambda_{\hat{j}} x(\hat{j})_i > 0$. This implies that $\theta^*_{\mathrm{CCR}}(k) > 0$. Moreover, we note that (λ', θ') is a feasible solution of (CCR(k)), if $\theta' = 1, \lambda'_k = 1$ and $\lambda'_j = 0$ for each $j \in \{1, \ldots, n\} \backslash \{k\}$. Therefore, $0 < \theta^*_{\mathrm{CCR}}(k) \leq 1$. By using the optimal value $\theta^*_{\mathrm{CCR}}(k)$ of (CCR(k)), the efficiency of DMU(k) for (CCR(k)) is defined in [3] as follows:

Definition 2.1 *DMU(k) is called CCR-efficient if $\theta^*_{\mathrm{CCR}}(k) = 1$. Otherwise, DMU($k$) is called CCR-inefficient.*

Let $\mathrm{T_{CCR}}$ be the production possibility set of the CCR model defined in [3] as follows:

$$
\mathrm{T_{CCR}} := \left\{ (x, y) \,\middle|\, x \geq \sum_{j=1}^{n} \lambda_j x(j), \; 0 \leq y \leq \sum_{j=1}^{n} \lambda_j y(j) \text{ for some } \lambda = (\lambda_1, \ldots, \lambda_n)^\top \geq 0 \right\}.
$$

This set is defined by setting $\theta := 0$ and adding a condition $y \geq 0$ for the constraint conditions of Problem (CCR(k)). Then the following theorem holds.

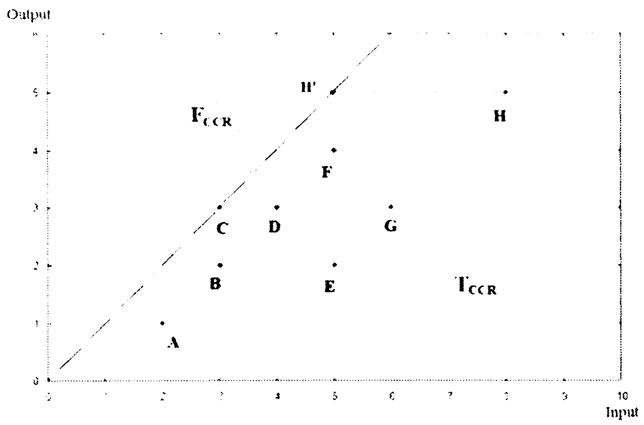

Figure 1: $\mathrm{T_{CCR}}$ for the data in Table 1

Theorem 2.1 $\mathrm{T_{CCR}}$ *is a polyhedral convex cone, that is,*

$$\mathrm{T_{CCR}} = (\mathrm{cc}\,\{(x(j), y(j)) |\, i = 1, \dots, n\} + (R^m_+ \times R^s_-)) \cap (R^m_+ \times R^s_+).$$

Hence, $\mathrm{T_{CCR}}$ *is closed.*

Example 2.1 *Assume that there are eight DMUs which consume one input to produce one output. The data of all DMUs is arranged in Table 1. In this case,* $\mathrm{T_{CCR}}$ *and* $\mathrm{F_{CCR}}$ *are shown in Figure 1.*

Table 1: The data of eight DMUs

DMU	A	B	C	D	E	F	G	H
Input	2	3	3	4	5	5	6	8
Output	1	2	3	3	2	4	3	5

In Example 2.1, C $\in \mathrm{F_{CCR}}$. Hence, C is CCR-efficient, that is, $\theta^*_{\mathrm{CCR}}(\mathrm{C}) = 1$. For other DMUs, to calculate the efficient values, we keep the output value fixed at current level and improve the input value within $\mathrm{T_{CCR}}$. For example, $\theta^*_{\mathrm{CCR}}(\mathrm{H}) = \frac{x(H')}{x(H)}$. This means that $(\theta^*_{\mathrm{CCR}}(\mathrm{H})x(H), y(H))$ is an efficient point. It is known that

$$\mathrm{F_{CCR}} = (\mathrm{bd}\,(\mathrm{T_{CCR}} + \{0^m\} \times \mathrm{R}^s_-)) \cap (\mathrm{R}^m_+ \times \mathrm{R}^s_+),$$

where $R^m_+ := \{x \in R^m |\, x \geq 0\}$, $R^s_+ := \{y \in R^s |\, y \geq 0\}$ and $R^s_- := \{y \in R^s |\, y \leq 0\}$. Obviously, $\mathrm{F_{CCR}}$ is the part of $\mathrm{bd}\,\mathrm{T_{CCR}}$ as shown in Figure 1. That is, $\mathrm{F_{CCR}}$ is constructed by all points which optimal solution of Problem (CCR(k)) equals to one.

2.2 BCC model

The BCC model is formulated as follows:

$$(\mathrm{BCC}(k))\begin{cases}\text{minimize} & \theta\\ \text{subject to} & \theta x(k)_i - \sum_{j=1}^{n}\lambda_j x(j)_i \geq 0,\ i=1,\ldots,m,\\ & \sum_{j=1}^{n}\lambda_j y(j)_r - y(k)_r \geq 0,\ r=1,\ldots,s,\\ & \sum_{j=1}^{n}\lambda_j = 1,\\ & \theta\in R,\ \lambda_j\geq 0,\ j=1,\ldots,n.\end{cases}$$

Let $\theta^*_{\mathrm{BCC}}(k)$ denotes the optimal value of (BCC(k)). From the definition of the constraint conditions of (BCC(k)), it is obvious that $0 < \theta^*_{\mathrm{BCC}}(k) \leq 1$. By using the optimal value $\theta^*_{\mathrm{BCC}}(k)$ of (BCC(k)), the efficiency of DMU(k) for (BCC(k)) is defined in [1] as follows:

Definition 2.2 *DMU(k) is called BCC-efficient if $\theta^*_{\mathrm{BCC}}(k) = 1$. Otherwise, DMU($k$) is called BCC-inefficient.*

Let $\mathrm{T_{BCC}}$ be the production possibility set of the BCC model as follows:

$$\mathrm{T_{BCC}} := \left\{ (x,y) \;\middle|\; x \geq \sum_{j=1}^{n}\lambda_j x(j),\ 0 \leq y \leq \sum_{j=1}^{n}\lambda_j y(j),\ \sum_{j=1}^{n}\lambda_j = 1 \text{ for some } \lambda = (\lambda_1,\ldots,\lambda_n)^{\mathrm{T}} \geq 0 \right\}.$$

This set is defined by setting $\theta := 0$ and adding a condition $y \geq 0$ for the constraint conditions of Problem (BCC(k)). Then the following theorem holds.

Theorem 2.2 $\mathrm{T_{BCC}}$ *is a closed convex set; indeed*

$$\mathrm{T_{BCC}} = (\mathrm{co}\,\{(x(j),y(j)) \mid i=1,\ldots,n\} + (R^m_+ \times R^s_-)) \cap (R^m_+ \times R^s_+).$$

Example 2.2 $\mathrm{T_{BCC}}$ *and* $\mathrm{F_{BCC}}$ *for the data in Table 1 are displayed in Figure 2.*

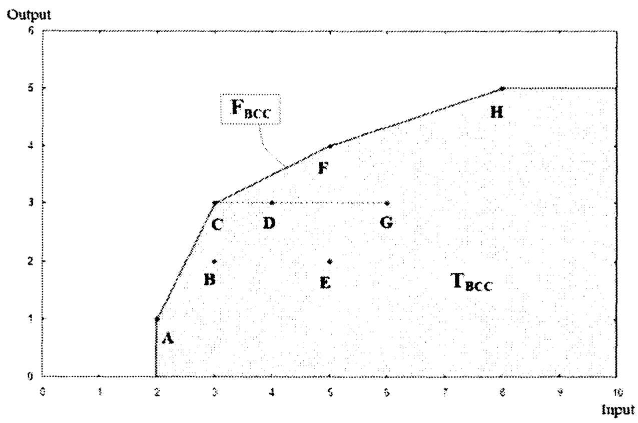

Figure 2: $\mathrm{T_{BCC}}$ for the data in Table 1

In Example 2.2, A,C,F,H $\in$ F_{BCC}. Hence, A,C,F and H are BCC-efficient and the other DMUs are BCC-inefficient. We keep the output value fixed at current level and improve the input value within T_{BCC} to calculate the efficient values for other DMUs. For example, $\theta^*_{BCC}(G) = \frac{x(C)}{x(G)}$. This means that $(\theta^*_{BCC}(G)x(G), y(G))$ is an efficient point. It is known that

$$F_{BCC} = (\mathrm{bd}\,(T_{BCC} + \{0^m\} \times R^s_-)) \cap (R^m_+ \times R^s_+).$$

Obviously, F_{BCC} is the part of $\mathrm{bd}\,T_{BCC}$ as shown in Figure 2. That is, F_{BCC} is constructed by all points which optimal solution of Problem (BCC(k)) equals to one.

3 Algorithm for constructing the equations forming the efficient frontiers

We need to clarify the equations forming F_{CCR} and F_{BCC} to calculate improvement indices. Therefore, in this section, we propose an algorithm for constructing the equations forming F_{CCR} and F_{BCC}. We compute the equations forming $\mathrm{bd}\,(T_{BCC} + \{0^m\} \times R^s_-)$ and $\mathrm{bd}\,(T_{CCR} + \{0^m\} \times R^s_-)$. Since $\mathrm{conv}\,\{(x_i^\top, y_i^\top)^\top \mid i = 1, \ldots, n\}$ is a polytope, by translating all DMUs, we construct a polytope including 0. Moreover all vertices of $\mathrm{conv}\,\{(x_i^\top, y_i^\top)^\top \mid i = 1, \ldots, n\}$ 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 $P(i)\,(i = 1, \ldots, 2n)$ and $P'(i)\,(i = 1, \ldots, 2n + m + s)$ as follows.

$$P(i) := \begin{cases} (x(i)^\top, y(i)^\top)^\top & \text{if } i \in \{1, \ldots, n\}, \\ 2P(i-n) & \text{if } i \in \{n+1, \ldots, 2n\}. \end{cases}$$

$$P'(i) := \begin{cases} P(i) - T & \text{if } i \in \{1, \ldots, 2n\}, \\ e^{i-2n} & \text{if } i \in \{2n+1, \ldots, 2n+m+s\}, \end{cases}$$

where $T := \frac{1}{2n}(P(1) + \cdots + P(2n))$ and e^j is a vector of R^{m+s} satisfying $e^j_j = 1$ and $e^j_i = 0$ for each $j \in \{1, \ldots, m+s\}$ and $i \in \{1, \ldots, m+s\} \backslash \{j\}$. Let $\mathcal{P} := \mathrm{co}\{P'(1), \ldots, P'(2n)\}$, $k_i := i$ for each $i \in \{1, \ldots, m+s\}$ and $\bar{n} := 2n + m + s$. Set $t = 1$ and go to Step 1.

Step 1

If $\dim\,\{P'(k_i) \mid i = 1, \ldots, m+s\} = m + s$, then go to Step 2. Otherwise, go to Step 3.

Step 2

Step 2–0

Calculate W solving the following system of linear equations:

$$\begin{cases} (P'(k_1))^\top W = 1, \\ \quad\vdots \\ (P'(k_{m+s}))^\top W = 1. \end{cases} \tag{1}$$

Step 2–1
If W calculated at Step 2–0 satisfies the following conditions, then $U_t := W$ and $t \leftarrow t+1$.

$$\begin{array}{l} (P'(j))^\top W \le 1 \; j = 1,\dots,2n, \\ W_i \le 0 \; i = 1,\dots,m, \\ W_i \ge 0 \; i = m+1,\dots,m+s. \end{array}$$

If $k_1 = 2n - m - s + 1$, go to Step 4. Otherwise, go to Step 3.

Step 3

Step 3–0
Set $k_{m+s} \leftarrow k_{m+s} + 1$ and $j := m+s$. Go to Step 3–1.

Step 3–1
If $k_j \le 2n - m - s + j$, set $k_j \leftarrow k_j + j' - j$ for every $j' > j$. Go to Step 1. Otherwise, set $k_{j-1} \leftarrow k_{j-1} + 1, j \leftarrow j - 1$ and go to Step 3–1.

Step 4
For each $i \in \{1,\dots,t-1\}$, let $p_i := \{-U_{1,i},\dots,-U_{m,i}\}^\top$ and $q_i := \{U_{m+1,i},\dots,U_{m+s,i}\}^\top$. For each $i = 1,\dots,t-1$, if $\max\{|q_{i,1}|,\dots,|q_{i,s}|\} > 0$ and $\frac{1+(-p_i^\top,q_i^\top)^\top T}{\max\{q_{i,1},\dots,q_{i,s}\}} > 0$, then $c_i := \frac{1+(-p_i^\top,q_i^\top)^\top T}{2}$. Otherwise, $c_i := 1 + (-p_i^\top, q_i^\top)^\top T$. Then, the hyperplane forming the efficient frontier is as follows.

$$H_{p_i,q_i,c_i} := \{(x,y) |\, q_i^\top y - p_i^\top x - c_i = 0\},$$

Stop the algorithm.

At Step 0 in Algorithm FFA, in order to obtain all equations forming the efficient frontier of the CCR model, $P(n+1),\dots,P(2n)$ are generated. To calculate all vertices of $\mathcal{P}^*$, all combinations of $\{P'(1),\dots,P'(\bar{n})\}$ are considered. At Step 1, to examine whether there exists a solution of linear system in Step 2–0, $\dim\{P'(k_i) |\, i = 1,\dots,m+s\}$ is calculated. At Step 2, all W calculated at Step 2–0 are checked to see whether they belong $\mathcal{P}^*$. We note that W is a vertex of $\mathcal{P}^*$ if W is contained in $\mathcal{P}^*$. At Step 3, to examine all combinations of choosing $m+s$ numbers from $\{1,\dots,\bar{n}\}$, $k_1,\dots,k_{m+s}$ are updated. At Step 4, for each $i \in \{1,\dots,t-1\}$, the necessity of H_i for constructing the efficient frontier is examined.

To show that Algorithm FFA terminates within finite number of iterations, we utilize the property of polytope.

Theorem 3.1 *The intersection of $R^m_- \times R^s_+$ and $\mathcal{P}^*$ is a polytope containing* 0.

By Theorem 3.1, the number of vertices of the intersection of $\hat{R}^{m+s}$ and $\mathcal{P}^*$ is finite. In particular, at Step 3, all combinations of $k_1,\dots,k_{m+s}$ from $\{1,\dots,\bar{n}\}$ are selected. Thus, Algorithm FFA terminates within ${}_{\bar{n}}C_{m+s}$ iterations. Let h be the number of hyperplanes H_{p_i,q_i,c_i} calculated by Algorithm FFA. For each $j = 1,...,h$, let

$$\begin{array}{l} W_j := (-p_j^\top, q_j^\top)^\top, \\ S_c := \{i |\, H_{p_i,q_i,c_i} \cap \mathrm{T_{CCR}} \subset \mathrm{F_{CCR}}\}, \\ S_b := \{i |\, H_{p_i,q_i,c_i} \cap \mathrm{T_{BCC}} \subset \mathrm{F_{BCC}}\}. \end{array}$$

Then, $\mathrm{T_{CCR}}$ and $\mathrm{T_{BCC}}$ can be represented by using equations forming the efficient frontier of each model as follows.

Theorem 3.2 $\mathrm{T_{CCR}} = \bigcap_{j \in S_c} \{Z |\, W_j^\top Z \le 0\}$ *and* $\mathrm{T_{BCC}} = \bigcap_{j \in S_b} \{Z |\, W_j^\top Z \le c_j\}$.

Moreover, $\mathrm{F_{CCR}}$ and $\mathrm{F_{BCC}}$ can be represented as follows.

Theorem 3.3 $\mathrm{F_{CCR}} = \left(\bigcup_{j \in S_c} \{Z \mid W_j^\top Z = 0\} \right) \cap \mathrm{T_{CCR}}$ *and* $\mathrm{F_{BCC}} = \left(\bigcup_{j \in S_b} \{Z \mid W_j^\top Z = c_j\} \right) \cap \mathrm{T_{BCC}}$.

Definition 3.1 (Facet) *Let E be a polytope in R^n. Then, $F := E \cap \{x \in R^n \mid a^\top x = b\}$ is called the facet of E if $a^\top x \leq b$ for each $x \in E$ and $\dim F = n - 1$.*

The equations calculated by Algorithm FFA are classified by following theorems. Since the efficient frontier of the CCR model include the origin, following theorem is given.

Theorem 3.4 *Assume that $H_{p,q,c} = \{(x, y) \in R^{m+s} \mid q^\top y - p^\top x - c = 0\}$ are calculated by Algorithm* FFA. *If $c = 0$, then $H_{p,q,c} \cap \mathrm{T_{CCR}}$ is a facet of $\mathrm{T_{CCR}}$.*

The efficient frontier which does not include the origin is not the efficient frontier of the CCR model, that is, this efficient frontier is it of the BCC model.

Theorem 3.5 *If $c \neq 0$, then $H_{p,q,c} \cap \mathrm{T_{BCC}}$ is a facet of $\mathrm{T_{BCC}}$.*

Moreover, the efficient frontier including the origin is discerned by adding following condition whether it is the efficient frontier of the BCC model.

Theorem 3.6 *If $c = 0$ and $\dim(\{(x(i)^\top, y(i)^\top)^\top \mid i = 1, \ldots, n\} \cap H_{p,q,c}) = m + s - 1$, then $H_{p,q,c} \cap \mathrm{T_{BCC}}$ is a facet of $\mathrm{T_{BCC}}$.*

We explain Algorithm FFA by the data in Table 1. Figure 3 shows the given eight DMUs. In Figure 4, the DMUs generated at Step 0 are added. By this operation, we can always calculate all equations of the CCR model. By subtracting vector T from each DMUs in Figure 4, we get Figure 5. Figure 6 shows the hyperplane $\{(x, y) \mid (P(j)')^\top (x^\top, y^\top)^\top = 1\}$ for each $j = 1, \ldots, 2n$. Polytope Q denotes the intersection of $\hat{R}_{m+s}$ and $\mathcal{P}^*$. We calculate all vertices of Q by performing from Step 1 to Step 3. Figure 7 shows the hyperplane calculated for each vertex of polytope Q in Figure 6. In other words, it is the polar set of polytope Q in Figure 6. By the coordinate transformation moving T to the origin, we get Figure 8. Figure 9 shows the all hyperplanes calculated by Algorithm FFA. By the operation at Step 4, the hyperplane consisting of only DMUs generated at Step 0 is removed.

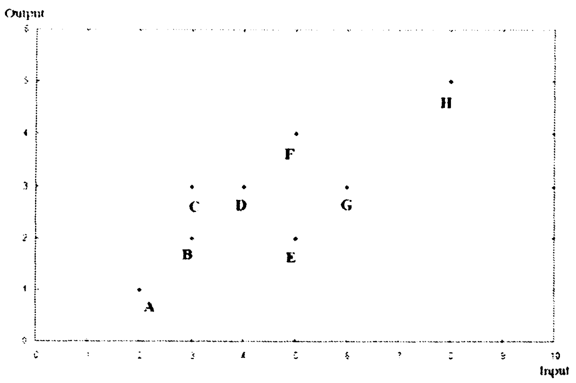

Figure 3: The data in TABLE 1

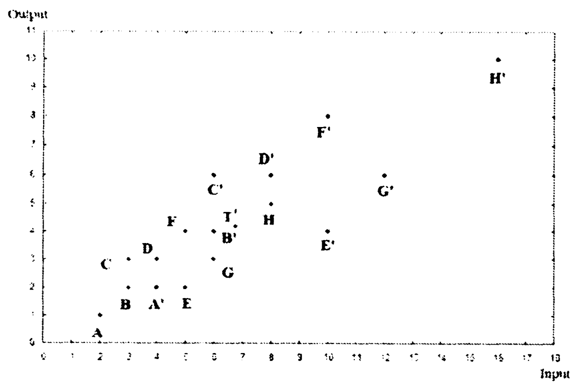

Figure 4: Added DMUs which two times

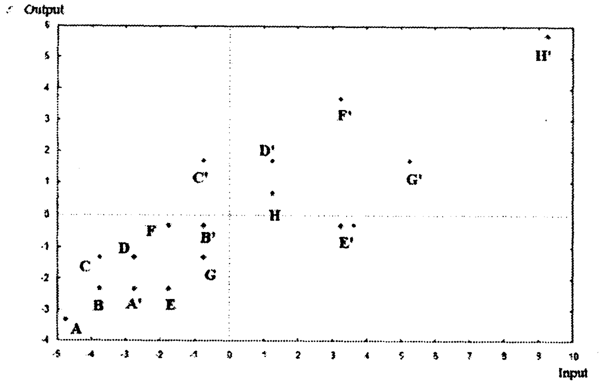

Figure 5: Subtracted vector T from each DMUs in FIGURE 4

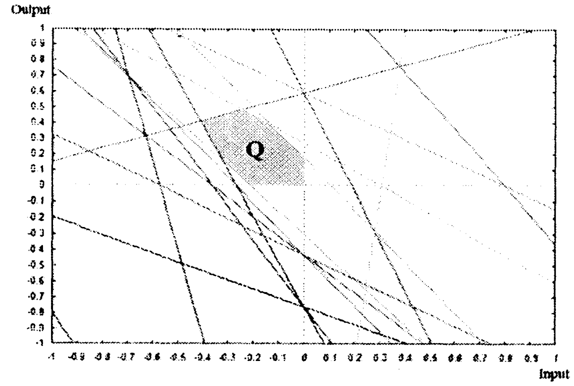

Figure 6: Hyperplane that inner product of each DMUs and (x, y) equals one

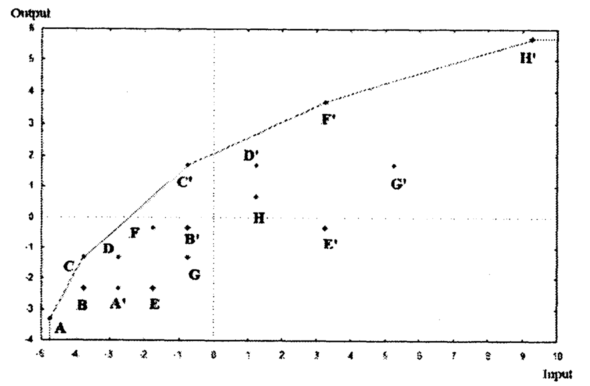

Figure 7: The polar set of polytope Q in FIGURE 6

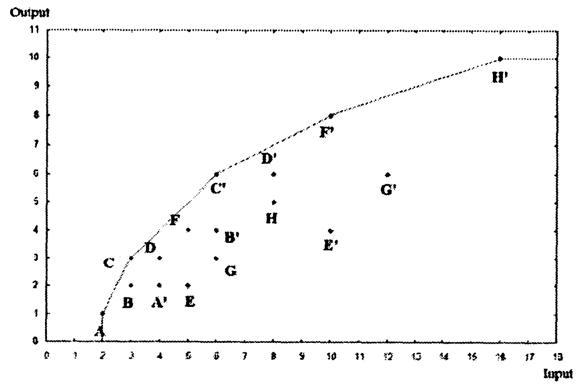

Figure 8: The coordinate transformation moving T to the origin

4 Improvement indices

In this section, we define two kinds of the improvement indices. Moreover we propose an algorithm for calculating those improvement indices.

4.1 Improvement indices in the CCR model

We propose two kinds of the improvement indices of DMU(k) which is inefficient in the CCR model. We define the norm depending on a symmetric positive semidefinite matrix $A \in R^{(m+s)\times(m+s)}$ as follows.

$$||Z||_A := \sqrt{Z^\top AZ}, \ Z \in R^{m+s}.$$

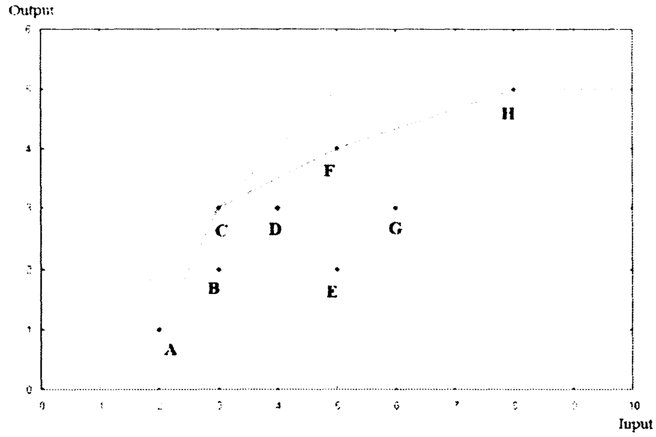

Figure 9: All hyperplanes

Example 4.1 *If $A = I_{m+s}$, then $||\cdot||_A$ means the Euclidean norm. If A is defined by*

$$A = M_k := \begin{pmatrix} \left(\frac{1}{P(k)_1}\right)^2 & & 0 \\ & \ddots & \\ 0 & & \left(\frac{1}{P(k)_{m+s}}\right)^2 \end{pmatrix},$$

then $||\cdot||_A$ means the norm which considered the scale transformation based on input and output values of DMU(k).

We define $d^i(k)(i = 1,2)$ as improvement indices for DMU(k), where $d^i(k)(i = 1,2)$ are the optimal solution of Problem $(\mathrm{ID}^i(k))(i = 1,2)$ defined as follows:

$$(\mathrm{ID}^i(k)) \begin{cases} \text{minimize} & ||Z||_A \\ \text{subject to} & Z \in B^i(k). \end{cases}$$

Here $B^1(k) := \mathrm{F_{CCR}} - P(k)$, $B^2(k) := (\mathrm{F_{CCR}} \cap \mathrm{T_{BCC}}) - P(k)$. Since $d^1(k)$ solves Problem $(\mathrm{ID}^1(k))$, $d^1(k) + P(k)$ has a minimal distance from $P(k)$ over $\mathrm{F_{CCR}}$. The feasible set $B^2(k)$ of Problem $(\mathrm{ID}^2(k))$ is the intersection of $B^1(k)$ and $\mathrm{T_{BCC}}$. By confining the feasible set to $\mathrm{T_{BCC}}$, $d^2(k)$ is more realistic than $d^1(k)$.

4.2 Algorithm for obtaining two improvement indices

We propose the following algorithm for obtaining two types of improvement indices $d^i(k)\,(i \in \{1,2\})$. Since we can not solve Problem $(\mathrm{ID}^i(k))(i = 1,2)$ directly, we calculate an optimal solution by solving the subproblems. For each subproblem, by the continuity of the objective function and confine the feasible region to compact set, we calculate an optimal solution. Let N_c be the number of elements of S_c. Improvement indices for DMU(k) are obtained by the following algorithm:

Algorithm ICCR

Step 0
Select $i \in \{1,2\}$. Set $j \leftarrow 1$ and go to Step 1.

Step 1

Let $d_j^i(k)$ be an optimal solution of Problem $(\mathrm{ID}_j^i(k))$ defined as follows:

$$(\mathrm{ID}_j^i(k)) \left\{ \begin{array}{ll} \text{minimize} & ||Z||_A \\ \text{subject to} & Z \in B_j^i(k), \end{array} \right.$$

where $B_j^1(k) := \{Z | (Z + P(k))^\top W_j = 0\}$,
$B_j^2(k) := \{Z | (Z + P(k))^\top W_j = 0, (Z + P(k))^\top W_l \leq c_l \text{ for each } l \in S_b\}$.
If $j = N_c$, then go to Step 2. Otherwise, set $j \leftarrow j + 1$ and go to Step 1.

Step 2

Select $j' \in \arg\min\{||d_j^i(k)||_A | j \in S_c\}$ and set $d^i(k) := d_{j'}^i(k)$. This algorithm terminates.

At Step 0 in Algorithm ICCR, we choose the type of the improvment index. At Step 1, we calculate a closest point over an hyperplane forming the efficient frontier of the CCR model from $P(k)$. At Step 2, we determine a closest point over $\mathrm{F_{CCR}}$ from $P(k)$.

We note that Problem $(\mathrm{ID}_j^i(\mathrm{k}))$ is a standard quadratic programming problem. Since $N_c < \infty$, Algorithm ICCR terminates within a finite number of iterations. An optimal solution of Algorithm ICCR is the improvement index for making inefficient DMUs efficient in the CCR model.

Theorem 4.1 *Let $d^i(k)$ $(i \in \{1, 2\})$ be the point calculated by Algorithm* ICCR. *Then, $P(k) + d^i(k) \in \mathrm{F_{CCR}}$.*

5 Conclusions

In this paper, we suggest Algorithm FFA for constructing all equations forming the efficient frontiers of the CCR and BCC models. These models can utilize the properties of the polar sets, since the production possibility set of these models are convex. By calculating all equations forming the efficient frontiers of two models, we can obtain the efficient values of all DMUs without solving (CCR(k)) and (BCC(k)) for each DMU.

Moreover, we propose two kinds of the improvement indices by analysing the efficient frontiers. To calculate the improvement index, all equations forming the efficient frontiers are used. For two types of indices, we consider a minimal distance from the DMU which is thinking about improvement. The first improvement index turns to the closest point over the efficient frontier of the CCR model. The second improvement index is a directioin to the intersection of $\mathrm{F_{CCR}}$ and $\mathrm{T_{BCC}}$. In this way, by adding conditions according to the situation, we can calculate improvement indices based on feasibility.

Futhermore, we suggest Algorithm ICCR to calculate two types of improvement indices. We can execute Algorithm ICCR using the existing nonlinear optimization techniques (e.g. [2]).

References

[1] Banker, R.D., Charnes, A., Cooper W.W.: "Some models for Estimating Technical and Scale Inefficiency in Data Envelopment Analysis," Management Science 30, pp.1078–1092, 1984.

[2] Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: "Nonlinear Programming - Theory and Algorithms," 1993.

[3] Charns, A., Cooper, W.W., Rhodes, E.: "Measuring Efficiency of Decision Making Units," European Journal of Operations Research 2, pp.429–444, 1978.