Annals of Optimization Theory and Practice

Volume 1, Number 1, 59-68
February 2018

DOI: 10.22121/aotp.2018.106001.1002

A polynomial-time algorithm to determine BCC efficient frontier without solving a mathematical programming problem

Masoud Sanei[1] • Hamid Hassasi[2]✉

[1] Department of Mathematics, Central Tehran Branch, Islamic Azad University, Tehran, Iran

[2] Faculty of Management Sciences, Central Tehran Branch, Islamic Azad University, Tehran, Iran

✉ hamid.mesho@gmail.com

(Received: November 1, 2017 / Accepted: January 14, 2018)

Abstract In this paper, we restrict our attention to the efficient frontier of the BCC model, where the BCC model is a well-known basic model in Data Envelopment Analysis (DEA). We here assume that each Decision Making Unit (DMU) has one input and one output. In order to obtain BCC efficient frontier, the paper proposes a polynomial-time algorithm of complexity bonded by $O(n^3)$ to produce well-behaved affine functions. The produced functions are then used to determine a point-wise minimum of a finite number of affine functions. It will be shown that by finding this function, we in fact also determine the efficient frontier of the BCC model. The main advantage of this approach is ability to achieve the efficient frontier, without solving a mathematical programming problem. Also, all of the Pareto efficient DMUs, as BCC-efficient DMUs, can be easily obtained using the proposed algorithm. A numerical example is presented to explain the use and effectiveness of the proposed algorithm.

Keywords Data envelopment analysis; BCC model; Efficient frontier; Point-wise minimum; Pareto efficient DMUs.

1. Introduction

Data envelopment analysis (DEA), introduced by Charnes *et al.* (1978) is a non-parametric approach to measure the relative efficiency score of a decision making unit (DMU). A collection T of pairs $(x.y)$, where $x = (x_1. \dots . x_m)$ is a vector of quantities of m inputs and $y = (y_1. \dots . y_s)$ is a vector of quantities of s outputs, is an empirical production set with the property that the input x can produce the output y. Based on inputs and outputs of the units, DEA forms an empirical efficient frontier. In this regard, because of the identification of the nature of returns to scale (RTS), investigating the strong defining hyperplanes of the empirical production possibility set (PPS) is an important task. Also, efficient hyperplanes can be used for determining rates of change of outputs with change in inputs.

The investigation of the efficient frontier and the identification of the strong defining hyperplanes of the PPS are frequently encountered in the literature in the DEA context; but, to our knowledge, only few of the existing works in DEA have been written on this subject. One of the first papers extended to search the efficient frontier in DEA was an article by Yu *et al.* (1996), where the authors analyzed the construction of all DEA efficient frontiers in generalized data envelopment analysis. Also, searching of the efficient frontier in DEA has been considered by Korhonen (1997). Jahanshahloo *et al.* (2005) tried to find the piecewise linear frontier of the production function which identifies the efficient frontier DMUs in DEA. In addition, an approach for finding strong defining hyperplanes of PPS in DEA has been suggested by Jahanshahloo *et al.* (2007). Furthermore, an alternative approach for determining these hyperplanes was proposed by Jahanshahloo *et al.* (2009). Amirteimoori and Kordrostami (2012) proposed a method for generating all linearly independent strong defining hyperplanes of the PPS passing through a specific DMU. Lotfi *et al.* (2011) suggested a method for finding efficient hyperplanes with variable returns to scale the technology in DEA by using the multiple objective linear programming (MOLP) structure.

This paper, tries to determine the efficient frontier of the BCC model Banker *et al.* (1984), without solving a mathematical programming problem. We here assume that each DMU has one input and one output. In order to obtain BCC efficient frontier, the paper proposes a polynomial-time algorithm of complexity bonded by $O(n^3)$ to produce well-behaved affine functions. The produced functions are then used to determine a point-wise minimum of a finite number of affine functions. It will be shown that by finding this function, we in fact also determine the strong defining hyperplanes of PPS. All of the BCC-efficient DMUs can be easily obtained using the proposed algorithm.

The rest of the paper is organized as follows: In the next Section, some basic concepts on PPS and the structure of its defining hyperplanes are reviewed. The third Section of the paper tries to propose a new algorithm for finding all efficient hyperplanes. In order to explain the details of the proposed approach, a numerical example is solved in fourth Section. Finally, the last section of the paper submits our concluding observations and further directions.

2. Preliminaries

In this section, we intend to describe the structure of the PPS and its strong defining hyperplanes. Toward this, we present some basic definitions, models and concepts that will be used in the next sections. Most of them are taken from Cooper *et al.* (2006).

The notation $(x_j . y_j)$ is employed for the observed vectors of the input and output, respectively, for the DMU_j $(1 \leq j \leq n)$. The empirical PPS T_v, defined to be the convex hull of these observed points, is stated as follows:

$$T_v = \left\{ (x.y) \,\middle|\, x \geq \sum_{j=1}^{n} \lambda_j x_j \,.\, y \leq \sum_{j=1}^{n} \lambda_j y_j \,.\, \sum_{j=1}^{n} \lambda_j = 1. \lambda_j . j \in \{1. \dots . n\}. x \in \mathbb{R}^m . y \in \mathbb{R}^s \right\}$$

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

T_v is constructed axiomatically and it is a closed and convex set. A DMU is efficient if it lies on the frontier of T_v; otherwise, it is inefficient. $DMU_O \in T_v$ is said to be extremely efficient in T_v if and only if it is an extreme efficient point in T_v.

Consider the following (BCC) fractional programming problem for evaluating the relative efficiency score of DMU_O $(O \in \{1.2. \dots .n\})$:

$$\begin{aligned} &Max\ \frac{uy_O + u_O}{vx_O} \\ &\text{subject to} \\ &\frac{uy_j + u_O}{vx_j} \leq 1 \qquad (j = 1.2. \dots .n) \\ &u \geq 0,\ v \geq 0,\ u_O \text{ free.} \end{aligned} \tag{1}$$

The foregoing problem can be equivalently rewritten as follows:

$$\begin{aligned} &Max\ uy_O + u_O \\ &\text{subject to} \\ &vx_O = 1 \\ &uy_j - vx_j + u_O \leq 0 \\ &u \geq 0,\ v \geq 0,\ u_O \text{ free.} \end{aligned} \tag{2}$$

This problem is referred to as the multiplier form of the input-oriented BCC model Cooper *et al.* (2006).

Definition 1. If a DMU is BCC-efficient, then it is called Strong Efficient (Pareto efficient).

Definition 2. Let $(x.y) \in T_v$. A hyperplane $\text{H} = \{(\bar{x}.\bar{y}) | u(y - \bar{y}) - v(x - \bar{x}) = 0. u \geq 0. v \geq 0\}$ is called a supporting hyperplane of T_v at $(\bar{x}.\bar{y}) \in \text{H}$ if for each $(\bar{x}.\bar{y}) \in T_v$, $u(y - \bar{y}) - v(x - \bar{x}) \leq 0$.

A collection of vectors $Z_1.Z_2. \dots .Z_k$ of dimension n is called affine independent if $\{Z_2 - Z_1. \dots .Z_k - Z_1\}$ is linear independent Murty (1983).

Definition 3. A hyperplane H is a strong defining hyperplane of T_v if it is supporting and there exists at least one affine independent set with $m + s$ elements of strongly efficient DMUs that lie on H.

3. The proposed Algorithm

At this moment, we are ready to propose a new algorithm for finding the frontiers of the BCC model by the results which we intend to discuss soon. Toward this end, suppose there are n DMUs: DMU_1, $\text{DMU}_2, \dots$, DMU_n, each with one input and one output, i.e., $\text{DMU}_j = (x_j.y_j)$ for every $1 \leq j \leq n$. In the next step, we sort the DMUs according to increasing input value. That is, we sort the DMUs in such a way that DMU_1 has the smallest input value and DMU_n has the largest one. If there are two or more DMUs with the same input value, then keep just the DMU with the largest output value among all such DMUs and discard others from the process.

Remark 1: To be noted in the foregoing discussion is the fact that from the sorted list of the DMUs and as we keep just the DMU with the largest output value among all DMUs with the same input value and discard others from the process, $x_j - x_l$ is positive and bounded away from zero for all $l+1 \le j \le n$, where $l \in \{1.2.\dots.n\}$. That is the condition $x_j - x_l > 0$ always holds true.

Now, let indices $l.p.q \in \{1.\dots.n\}$ be such that $l \le p \le q$, and let, for these indices, we have the following expressions:

$$
\begin{aligned}
C^t &= \frac{y_p - y_l}{x_p - x_l} = \max\left\{0.\max_{l+1\le j\le n}\left\{\frac{y_j - y_l}{x_j - x_l}\right\}\right\} \ge 0 \\
C^{t+1} &= \frac{y_q - y_p}{x_q - x_p} = \max\left\{0.\max_{p+1\le j\le n}\left\{\frac{y_j - y_p}{x_j - x_p}\right\}\right\} \ge 0
\end{aligned} \tag{3}
$$

Remark 2: From Remark 1, we know that the denominator in the above formulas is positive and bounded away from zero. Therefore, based on the formulas in (3), we have

$$\frac{y_p - y_l}{x_p - x_l} \ge \frac{y_q - y_p}{x_q - x_p} \tag{4}$$

The first formula in (3) can be used to obtain a hyperplane passing through DMU_l and DMU_p, as follows:

$$(y - y_l) = C^t(x - x_l) = \frac{y_P - y_l}{x_p - x_l}(x - x_l)$$

In other words, this hyperplane can be expressed by the equation

H: $(x_p - x_l)(y - y_l) - (y_P - y_l)(x - x_l) = 0$

or

H: $(x_p - x_l)y - (y_P - y_l)x + x_l y_P - x_p y_l = 0$

More specifically, if we let

$$u^* = (x_p - x_l),\ v^* = (y_P - y_l) \text{ and } u_O^* = x_l y_P - x_p y_l \tag{5}$$

then the hyperplane H: $u^* y - v^* x + u_O^* = 0$ is a hyperplane passing through DMU_p and DMU_l. Similarly, the second formula in (3) can be used to obtain a hyperplane passing through DMU_p and DMU_q.

From the foregoing discussion, we intend to show that the hyperplane $u^* y - v^* x + u_O^* = 0$ is a strong defining hyperplane of T_v. Toward this end, it is necessary to show that H: $u^* y - v^* x + u_O^* = 0$ is supporting and strong. Regarding this subject, we have the following two theorems.

Note that in the evaluation of DMU_O ($O \in \{1.2.\dots.n\}$), if $(u^*.v^*.u_O^*)$ is an optimal solution to Model (2), then $u^* y - v^* x + u_O^* = 0$ is a supporting hyperplane Cooper *et*

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

al. (2006) on the PPS. Under this assumption, the supporting condition can be equivalently expressed in terms of the optimal solution of Model (2) as follows:

Theorem 1: Let the hyperplane H: $u^*y - v^*x + u_O^* = 0$ be a hyperplane passing through DMU_l and DMU_p, where $(u^*.v^*.u_O^*)$ is given by Equation (5). Then, in the evaluation of DMU_l and DMU_p, the solution $(u^*.v^*.u_O^*)$ is an optimal solution to the model (2).

Proof. We only prove the assertion for DMU_l. The other case can be easily deduced as a similar manner. First of all, we must show that $(u^*.v^*.u_O^*)$ is a feasible solution to the model (2). Since the model (2) is a reformulated form of the model (1), we prove this using Model (1). The sign restrictions on the decision variables $u^* \geq 0$, $v^* \geq 0$ can be easily deduced from Remark 1. Hence, it is sufficient to show that $\frac{u^*y_j+u_O^*}{v^*x_j} \leq 1$, for each $j = 1.2.\dots.n$. On the one hand, from two formulas in (3) and the expression in (4), the following inequality holds for each $l+1 \leq j \leq n$ under the assumptions $l \in \{1.\dots.n\}$ and $l \leq p$:

$$\frac{y_P - y_l}{x_p - x_l} \geq \frac{y_j - y_l}{x_j - x_l} \qquad (6)$$

On the other hand, from Remark 1, we have $x_j - x_l > 0$, for each $j \in \{1.\dots.n\}$. Therefore, we can equivalently rewrite (6) as follows:

$$(y_P - y_l)(x_j - x_l) \geq (x_p - x_l)(y_j - y_l)$$

or

$$(y_P - y_l)x_j - (y_P - y_l)x_l \geq (x_p - x_l)y_j - (x_p - x_l)y_l$$

and so we have

$$(y_P - y_l)x_j \geq (x_p - x_l)y_j + x_l y_P - x_p y_l$$

Therefore, the expressions in (5) imply that $u^*y_j + u_O^* \leq v^*x_j$, for each $j \in \{1.\dots.n\}$. This yields

$$\frac{u^*y_j + u_O^*}{v^*x_j} \leq 1$$

In order to prove that $(\mathrm{u}^*.\mathrm{v}^*.\mathrm{u}_O^*)$ is indeed optimal, we must show that $\frac{u^*y_l+u_O^*}{v^*x_l} = 1$.

Toward this end, from (5), we can write

$$\frac{u^*y_l + u_O^*}{v^*x_l} = \frac{(x_p - x_l)y_l + x_l y_P - x_p y_l}{(y_P - y_l)x_l} = \frac{(y_P - y_l)x_l}{(y_P - y_l)x_l} = 1$$

This completes the proof.

It is worthwhile to note that the theorem is true, in general, for each point on the line segment joining DMU_l and DMU_p, i.e., for any convex combination of the two points.

Theorem 2: The hyperplane H: $u^*y - v^*x + u_O^* = 0$ is the strong defining hyperplane of T_v.

Proof. It is obvious from the previous discussion and from the theorem 1 that the hyperplane is supporting. Therefore, it is sufficient to show that there exists at least one affine independent set with $m + s$ elements of strongly efficient DMUs that lie on H. Note here that because of the single input and single output case, $m + s$ is equal to 2. On the one hand, since $(y_P - y_l) > 0$ and $(x_p - x_l) > 0$, so from the fact that $\{(x_l . y_l) - (x_p . y_p)\} \neq \{(0.0)\}$ one can conclude that DMU_l and DMU_p are affine independent. On the other hands, we know from Theorem 1 and Definition 1 that DMU_l and DMU_p are strongly efficient DMUs that lie on H.

The following corollaries are immediately concluded from Theorems 1 and 2.

Corollary 1. DMU_l and DMU_p are extreme efficient DMUs and any activity $(x_j . y_j)$ on the line segment joining DMU_l and DMU_p is strongly (BCC) efficient .

Corollary 2. The functions $f_t(x) = C^t(x - x_l) + y_l$ are the strong defining hyperplanes of T_v. If the number of these functions is equal to h, then the function $f(x) = \underset{1 \leq t \leq h}{min} \{f_t(x)\}$ $(x \in R)$, is the frontier of the BCC model.

Remark 2: It will be noted that the functions $f_t(x)$ $(1 \leq t \leq h)$, are affine functions. This implies that $f(x)$, is the point-wise minimum of a finite number of affine functions (see, for example Murty (1983)).

The foregoing discussion suggests a constructive algorithm for actually describing the frontier of the BCC model in terms of no more than n observed activities, using a number of elementary computational operations such as additions, comparisons and also a finite number of affine functions, which are bounded above by a polynomial in n of degree $\leq$ 3. That is, it suggests a polynomial-time algorithm of complexity bounded by $O(n^3)$ to obtain such a description, as follows:

An algorithm for finding the efficient frontier of the BCC model:

Input: n,

DMUs: DMU_1, DMU_2, …, DMU_n each with one input and one output, i.e., $\text{DMU}_j = (x_j . y_j)$ for every $1 \leq j \leq n$.

Step 0: Sort the DMUs in such a way that DMU_1 has the smallest input value and DMU_n has the largest one. If there are two or more DMUs with the same input value, then keep just the DMU with the largest output value among all such DMUs and discard others from the process.

Step 1: Set $s_0 = 1$, $l = s_0$, $t = 1$ and go to the next step.

Step 2: Let $C^t = \max\{0. \max_{l+1 \leq j \leq n} \left\{\frac{y_j - y_l}{x_j - x_l}\right\}\}$, and determine the function

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

$f_t(x) = C^t(x - x_l) + y_l$

Step 3: Let s_t $(l + 1 \leq s_t \leq n)$ be the largest index such that coordinates of DMU$_{s_t}$, i.e., $(x_{s_t}. y_{s_t})$ satisfy $f_t(x)$.

If either $s_t = n$ or else $C^t = 0$, then go to the next step. Otherwise, set $t = t + 1$, $l = s_t$ and return to the previous step.

Step 4: Suppose that the number of function $f_t(x)$ is equal to h, in step 3. After re-indexing the functions, if necessary, let

$f(x) = \min_{1 \leq t \leq h} \{f_t(x)\} . x_1 \leq x \leq x_{s_m}$ $(x \in R)$

Output: The frontiers of the BCC model

From the fact that the slope of the affine functions are strictly decreasing and also owing to the finite number of DMUs, the validity of the algorithm is obvious.

It is clear that, if we use this new algorithm, then all of the Pareto efficient DMUs, as well as the efficient frontier of the BCC model, can be obtained without solving a mathematical programming problem according to Steps 1 and 3 of the algorithm.

We can modify the algorithm to determine the efficient frontier of the CCR model, which is defined as follows Cooper *et al.* (2006).

$$
\begin{aligned}
&Max\ uy_O \\
&\text{subject to} \\
&vx_O = 1 \\
&uy_j - vx_j \leq 0 \\
&u \geq 0, v \geq 0
\end{aligned}
\tag{7}
$$

In other words, with the following slight modification, the proposed algorithm can be used to obtain the efficient frontier of the CCR model.

A slight modification of the proposed algorithm for finding the efficient frontier of the CCR model:

Input: n,

DMUs: DMU$_1$, DMU$_2$, …, DMU$_n$ each with one input and one output, i.e., DMU$_j = (x_j. y_j)$ for every $1 \leq j \leq n$.

Step 1: Let $C = \max\{0. \max_{1 \leq j \leq n} \left\{\frac{y_j - 0}{x_j - 0}\right\}\}$, and determine the function

$f(x) = Cx$

Step 2: Let s $(1 \leq s \leq n)$ be the index such that coordinates of DMU$_s$, i.e., $(x_s. y_s)$ satisfy $f(x)$. Then, DMU$_s$ is a CCR-efficient DMU.

Output: The frontiers of the CCR model as $f(x) = Cx$ $\ (x \in R)$

4. Numerical example

In this section, in order to show how the proposed approach can successfully be used to determine the frontiers of the CCR and BCC models, we apply the new algorithm to the following simple example from Cooper *et al.* (2006) and comment on the results. For this purpose, we use Table 1, which shows 8 DMUs with 1 input and 1 output. We label the DMUs from A to H at the head of each column in Table 1.

Table 1. Data of the example.

DMU	A	B	C	D	E	F	G	H
Input	2	3	3	4	5	5	6	8
Output	1	3	2	3	4.5	2	3	5

i) The identification of the efficient frontier of the BCC model:

From the step 0 of the proposed algorithm, we sort the DMUs according to increasing input value as shown in Table 2. Since the DMUs B and C have the same input value, then we keep just B with the largest output value 3 and discard C from the process. This is also done for the DMUs E and F with the same input value. Therefore, after re-indexing the DMUs, we have Table 2.

Table 2. Modified data of the example.

DMU	DMU_1	DMU_2	DMU_3	DMU_4	DMU_5	DMU_6
Input	2	3	4	5	6	8
Output	1	3	3	4.5	3	5

Step 1: $s_0 = 1, l = 1, t = 1.$

Step 2: $C^1 = \max\left\{0. \max\left\{\frac{3-1}{3-2}. \frac{3-1}{4-2}. \frac{4.5-1}{5-2}. \frac{3-1}{6-2}. \frac{5-1}{8-2}\right\}\right\} = \{0. \max\left\{2.1. \frac{7}{6}. \frac{1}{2}. \frac{3}{2}\right\} = 2$

$f_1(x) = 2(x-2) + 1 = 2x - 3$

Step 3: Since $s_1 = 2 \neq 8$, we set $t = 2, l = 2$ and return to the previous step.

Step 2: $C^2 = \max\left\{0. \max\left\{\frac{3-3}{4-3}. \frac{4.5-3}{5-3}. \frac{3-3}{6-3}. \frac{5-3}{8-3}\right\}\right\} = \frac{3}{4}$

$$f_2(x) = \frac{3}{4}x + \frac{3}{4}$$

Step 3: Since $s_2 = 4 \neq 8$, we set $t = 3, l = 4$ and return to the step 2.

Step 2: $C^3 = \max\left\{0. \max\left\{\frac{3-4.5}{6-5}. \frac{5-4.5}{8-5}\right\}\right\} = \frac{1}{6}$

$$f_3(x) = \frac{1}{6}x + \frac{11}{3}$$

Step 3: Since $s_3 = 8$, we go to the next step.

Step 4: We have the following function as the efficient frontier of the BCC model:

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

$$f(x) = \min\{2x - 3.\frac{3}{4}x + \frac{3}{4}.\frac{1}{6}x + \frac{11}{3}\}.2 \le x \le 8\ (x \in R)$$

As depicted in figure 1, the DMUs A, B, E and H are extreme efficient DMUs and so they are strongly BCC-efficient.

ii) The identification of the efficient frontier of the CCR model:

Using the slightly modified algorithm, one can determine the efficient frontier of the CCR as follows.

Step 1: $C = \max\left\{0.\max\left\{\frac{1}{2}.\frac{3}{3}.\frac{2}{3}.\frac{3}{4}.\frac{4.5}{5}.\frac{2}{5}.\frac{3}{6}.\frac{5}{8}\right\}\right\} = 1$. Then, as can be easily seen in figure 1, the efficient frontier of the CCR model is $f(x) = x$. Also, Figure 1 gives B as the only CCR-efficient DMU.

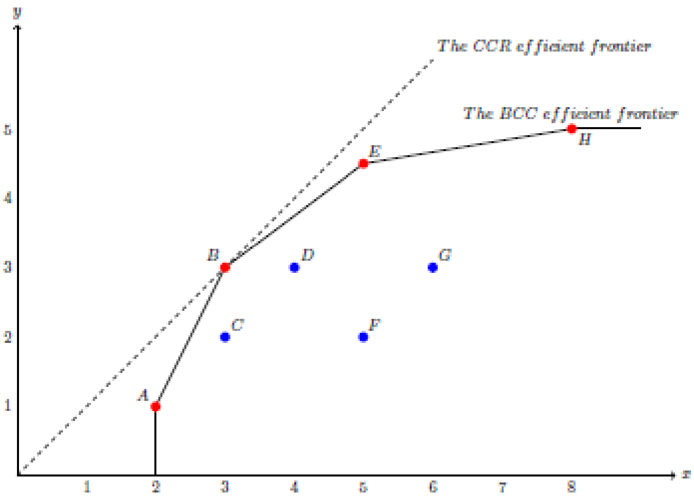

Figure 1. The Efficient Frontier of the CCR and BCC models in the given example.

5. Conclusions

One of the main concerns in DEA is trying to obtain the efficient frontier of the production possibility set. However, in the literature, there are few studies on the subject of finding this efficient frontier. In this paper, we have tried to determine the frontiers of the BCC model. Here, it has been assumed that each DMU has one input and one output. In order to obtain BCC efficient frontier, the paper proposes a new algorithm to produce well-behaved affine functions. The produced functions are then used to determine a point-wise minimum function. It has been shown that by finding this function, we in fact also determine the efficient frontier of the BCC model. All of the Pareto efficient DMUs, as BCC-efficient DMUs, can be easily obtained using proposed algorithm, without solving a mathematical programming problem. A numerical example has been presented to explain the use and effectiveness of the proposed algorithm.

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran

Authors hope that the proposed approach can arouse more interest in this research. They believe that the generalization of the proposed algorithm to the case of multiple-inputs and multiple-outputs can be an interesting subject for further researches.

References

1. Amirteimoori, A., & Kordrostami, S. (2012). Generating strong defining hyperplanes of the production possibility set in data envelopment analysis. *Applied Mathematics Letters*, *25*(3), 605-609.
2. Banker, R. D., Charnes, A., & Cooper, W. W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management science*, *30*(9), 1078-1092.
3. Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European journal of operational research*, *2*(6), 429-444.
4. Cooper, W. W., Seiford, L. M., & Tone, K. (2006). *Introduction to Data Envelopment Analysis, Applications References and DEA-Solver Software*. Kluwer Academic Publishers.
5. Lotfi, F. H., Jahanshahloo, G. R., Mozaffari, M. R., & Gerami, J. (2011). Finding DEA-efficient hyperplanes using MOLP efficient faces. *Journal of Computational and Applied Mathematics*, *235*(5), 1227-1231.
6. Jahanshahloo, G. R., Lotfi, F. H., & Zohrehbandian, M. (2005). Finding the piecewise linear frontier production function in data envelopment analysis. *Applied Mathematics and Computation*, *163*(1), 483-488.
7. Jahanshahloo, G. R., Lotfi, F. H., Rezai, H. Z., & Balf, F. R. (2007). Finding strong defining hyperplanes of production possibility set. *European Journal of Operational Research*, *177*(1), 42-54. G.R.
8. Jahanshahloo, G. R., Shirzadi, A., & Mirdehghan, S. M. (2009). Finding strong defining hyperplanes of PPS using multiplier form. *European Journal of Operational Research*, *194*(3), 933-938. G.R.
9. Korhonen, P. (1997). *Searching the efficient frontier in data envelopment analysis. IIASA*. IR-97-79. P.
10. Murty, K. G. Linear programming. john Wily 1983.
11. Yu, G., Wei, Q., Brockett, P., & Zhou, L. (1996). Construction of all DEA efficient surfaces of the production possibility set under the generalized data envelopment analysis model. *European Journal of Operational Research*, *95*(3), 491-510.

Published by Firouzabad Institute of Higher Education, Firouzabad, Fars, Iran