RAIRO-Oper. Res. 54 (2020) 53–68
https://doi.org/10.1051/ro/2018070

RAIRO Operations Research
www.rairo-ro.org

CLOSED FORM OF THE RESPONSE FUNCTION IN FDH TECHNOLOGIES: THEORY, COMPUTATION AND APPLICATION *

Amin Mostafaee[1] and Majid Soleimani-damaneh[2,**]

Abstract. Response function (RF), which gives the value of maximum feasible outputs in response to changing the inputs, has a crucial role in performance analysis and scale elasticity measurement. In this paper, a polynomial-time algorithm is provided which is able to obtain the closed form of the RF under (nonconvex) FDH productions technologies. Finite convergence of the presented algorithm is proved; and it is established that the algorithm is polynomial-time from a complexity standpoint. Moreover, an application of the proposed procedure with real-world data accompanying some experiment-based computational discussions are given.

Mathematics Subject Classification. 90C90, 90C26, 91B84.

Received October 2, 2017. Accepted September 4, 2018.

1. Introduction

Data Envelopment Analysis (DEA) is a non-parametric technique for evaluating Decision Making Units (DMUs) based on the Production Possibility Set (PPS). An important class of DEA models is that of Free Disposal Hull (FDH) models. These models, first presented by Deprins *et al.* [6], work under nonconvex technologies and analyze the efficiency of Decision Making Units (DMUs) considering the closest inner approximation of the true strongly disposable technology [5]. FDH models have been studied by many scholars, including Tulkens [24], Kerstens and Vanden Eeckaut [10], Cherchye *et al.* [5], Podinovski [15], Leleu [13, 14], Briec and Kerstens [2], Soleimani-damaneh *et al.* [19], Soleimani-damaneh and Reshadi [22], Soleimani-damaneh and Mostafaee [20, 21], Cesaroni and Giovannola [3], Cesaroni *et al.* [4], Diewert and Fox [7], Abdelsalam *et al.* [1]. See also Kerstens and Van de Woestyne [11] and Emrouznejad and Yang [8] for some reviews.

Marginal productivity and average productivity are two important measures which play crucial roles in performance analysis. In a problem with multiple-inputs and multiple-outputs, both aforementioned productivity measures are defined utilizing a Response Function (RF) which gives maximal proportion of outputs for a given proportion of inputs under feasibility; see Podinovski and Førsund [17]. Furthermore, Scale Elasticity (SE) at a given efficient DMU is defined with respect to the derivative of RF; see ([18], p. 150) and ([17], p. 1745). Thanks

Keywords. Data envelopment analysis, FDH technologies, response function, polynomial-time algorithm, real application, computational experiments.

* *The research of the second author was in part supported by a grant from the University of Tehran (No. 27836.1.11).*

[1] Department of Mathematics, College of Science, Tehran North Branch, Islamic Azad University, Tehran, Iran

[2] School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Enghelab Avenue, Tehran, Iran

**Corresponding author: `soleimani@khayam.ut.ac.ir`

to these important applications, RF is an important function in efficiency analysis, and a closed formula of this function helps the Decision Maker (DM) to have a better perspective about the behaviour of the DMU under assessment.

In the present work, we sketch a polynomial-time algorithm to obtain the RF in FDH models. The provided algorithm seeks to identify the proportional response of outputs with respect to the proportional change of inputs until reaching a unit with Most Productive Scale Size (MPSS). We establish the finite convergence of the presented algorithm, and moreover we show that it is polynomial-time from a complexity standpoint. In addition to the theoretical outcomes, we illustrate an application of the investigated procedure on real-world data derived from TIMSS study in Iran [12, 23]. Moreover, we address some experimental computational discussions.

The rest of the paper is organized as follows. Some preliminaries are given in Section 2. The main theoretical results are presented in Section 3. Section 4 is devoted to practical and computational investigations. Section 5 concludes the paper.

2. Preliminaries

Suppose that we have a set of n DMUs consisting of DMU_j; $j \in J = \{1, \ldots, n\}$. Each DMU_j consumes m positive inputs $x_{1j}, x_{2j}, \ldots, x_{mj}$ to produce s positive outputs $y_{1j}, y_{2j}, \ldots, y_{sj}$. Define $x_j := (x_{1j}, x_{2j}, \ldots, x_{mj})^t$ and $y_j := (y_{1j}, y_{2j}, \ldots, y_{sj})^t$ as input and output vectors of DMU_j, respectively. In this paper, the superscript t stands for transpose. Also, define $X := [x_1, x_2, \ldots, x_n]$ and $Y := [y_1, y_2, \ldots, y_n]$ as $m \times n$ and $s \times n$ matrices of inputs and outputs, respectively.

The FDH technologies under different Returns to Scale (RTS) assumptions can be represented as follows [10]:

$$
\begin{aligned}
T^{\text{FDH}_\Delta} = \{(x, y) \mid \sum_{j\in J} \lambda_j x_j \leq x,\ \ \sum_{j\in J} \lambda_j y_j \geq y \geq 0, \quad \lambda_j = \delta\mu_j;\ \forall j \in J,& \\
\sum_{j\in J} \mu_j = 1, \quad \mu \in (\{0,1\})^n, \quad \delta \in \Delta\}&
\end{aligned}
$$

where Δ, depending on the RTS assumption of the reference technology, is

$$
\begin{aligned}
\Delta^{\text{VRS}} &\equiv \{\delta \mid \delta = 1\}, \qquad \Delta^{\text{CRS}} \equiv \{\delta \mid \delta \geq 0\}, \\
\Delta^{\text{NIRS}} &\equiv \{\delta \mid 0 \leq \delta \leq 1\}, \qquad \Delta^{\text{NDRS}} \equiv \{\delta \mid \delta \geq 1\}.
\end{aligned}
$$

Here, VRS, CRS, NIRS, and NDRS stand for variable, constant, non-increasing, and nondecreasing RTS, respectively. Hereafter, for simplicity, we use notations FDH_V, FDH_C, FDH_NI, and FDH_ND, instead of FDH_VRS, FDH_CRS, FDH_NIRS, and FDH_NDRS, respectively.

Considering $\text{DMU}_o = (x_o, y_o)$, $o \in J$, as the unit under assessment, the input-oriented and output-oriented FDH radial efficiency measures of DMU_o are obtained by solving the following mixed-integer nonlinear programming problems, respectively [10]:

$$
\begin{aligned}
\theta_o^\Delta = \min\ & \theta \\
s.t.\ & \sum_{j\in J} \lambda_j x_j \leq \theta x_o, \\
& \sum_{j\in J} \lambda_j y_j \geq y_o, \\
& \lambda_j = \delta\mu_j, \quad \mu_j \in \{0,1\};\ \forall j \in J, \\
& \sum_{j\in J} \mu_j = 1,\ \ \delta \in \Delta,
\end{aligned}
\tag{2.1}
$$

$$
\begin{aligned}
\varphi_o^{\Delta} = \max\ & \varphi \\
s.t.\ & \sum_{j\in J} \lambda_j x_j \le x_o, \\
& \sum_{j\in J} \lambda_j y_j \ge \varphi y_o, \\
& \lambda_j = \delta\mu_j, \quad \mu_j \in \{0,1\};\ \forall j \in J, \\
& \sum_{j\in J} \mu_j = 1,\ \delta \in \Delta,
\end{aligned}
\tag{2.2}
$$

wherein $\Delta \in \{\mathrm{FDH_V}, \mathrm{FDH_C}, \mathrm{FDH_{NI}}, \mathrm{FDH_{ND}}\}$.

Definition 2.1. The $DMU_o = (x_o, y_o)$ is called Δ-efficient if there exists no $(x,y) \in T^{\Delta}$ such that $x \le x_o,\ y \ge y_o$ and $(x,y) \neq (x_o, y_o)$, where the vector inequalities are understood componentwise.

Definition 2.2. The $DMU_o = (x_o, y_o)$ is called an MPSS (i.e. a unit with Most Productive Scale Size) if $\theta_o^{\mathrm{FDH_C}} = 1$.

There are various papers in the DEA literature discussing the local RTS in $\Delta-$technologies. In a local sense, DMUs are categorized into three classes: the DMUs with Increasing RTS (IRS) status, the DMUs with Decreasing RTS (DRS) status, and the DMUs with Constant RTS (CRS) status. Podinovski [15] pointed out that local RTS is not a proper indicator of the direction in which MPSS in nonconvex Production Possibility Sets (PPSs) is achieved. He defined Global RTS (GRS) by adding a fourth case called Global Sub-Constant RTS (G-SCRS) under which MPSS can be achieved by both reducing and increasing the size of DMU.

Definition 2.3. [15,16] Let $\mathrm{DMU}_o = (x_o, y_o)$ be a $\mathrm{FDH_V}$-efficient unit. Then we have,
(i) G-CRS prevails at DMU_o if $\theta_o^{\mathrm{FDH_C}} = \theta_o^{\mathrm{FDH_{NI}}} = \theta_o^{\mathrm{FDH_{ND}}} = 1$.
(i) G-SCRS prevails at DMU_o if $\theta_o^{\mathrm{FDH_C}} = \theta_o^{\mathrm{FDH_{NI}}} = \theta_o^{\mathrm{FDH_{ND}}} < 1$.
(iii) G-IRS prevails at DMU_o if $\theta_o^{\mathrm{FDH_{ND}}} > \theta_o^{\mathrm{FDH_C}} = \theta_o^{\mathrm{FDH_{NI}}}$.
(iv) G-DRS prevails at DMU_o if $\theta_o^{\mathrm{FDH_{NI}}} > \theta_o^{\mathrm{FDH_C}} = \theta_o^{\mathrm{FDH_{ND}}}$.

3. Theoretical Results

Let $\mathrm{DMU}_o = (x_o, y_o)$ be the unit under consideration. The RF under $\mathrm{FDH_V}$ PPS is defined as

$$\beta_o(\alpha) := \max\{\beta \mid (\alpha x_o, \beta y_o) \in T^{\mathrm{FDH_V}}\}, \tag{3.1}$$

and its domain is

$$\Lambda := \{\alpha \in \mathbb{R} \mid \exists \beta \in \mathbb{R}\ \text{ s.t. } (\alpha x_o, \beta y_o) \in T^{\mathrm{FDH_V}}\}. \tag{3.2}$$

Hadjicostas and Soteriou [9] proved that Λ is a convex set under the convex-DEA technologies. See also ([15], p. 245). They utilized the convexity postulate to prove it. Therefore, their proof does not work here in FDH models. In the following lemma, we prove the convexity of Λ under nonconvex FDH technology. Our proof is based upon the possibility (free disposability) axiom.

Lemma 3.1. *The set Λ defined in (3.2) is a nonempty, closed, bounded below, unbounded above, and convex set in $\mathbb{R}$.*

Proof. The set Λ is nonempty because $1 \in \Lambda$. This set is clearly bounded below due to the first constraints of $T^{\mathrm{FDH_V}}$, and unbounded above because of the possibility (free disposability) postulate. To prove the convexity, assume that $\alpha_1, \alpha_2 \in \Lambda$ and $\theta \in (0,1)$. Then, there exist $\beta_1, \beta_2 \in \mathbb{R}$ such that $(\alpha_i x_o, \beta_i y_o) \in T^{\mathrm{FDH_V}}$ for $i = 1, 2$.

We should show that $\alpha_3 := \theta\alpha_1 + (1-\theta)\alpha_2 \in \varLambda$. Without loss of generality, assume $\alpha_1 \geq \alpha_2$. Then $\alpha_3 x_o \geq \alpha_2 x_o$ which leads to $(\alpha_3 x_o, \beta_2 y_o) \in T^{\mathrm{FDH_V}}$ because of the possibility (free disposability) axiom. This implies $\alpha_3 \in \varLambda$.

To prove the closedness of $\varLambda$, let $\{\alpha_k\}$ be a sequence in $\varLambda$ such that $\alpha_k \longrightarrow \alpha_0$ as $k \longrightarrow +\infty$. We should show that $\alpha_0 \in \varLambda$. As $\alpha_k \in \varLambda$, there are two sequences $\lambda^k \subseteq (\{0,1\})^n$ and $\{\beta_k\} \subseteq [0,+\infty)$ such that, for each k,

$$\sum_{j\in J} \lambda_j^k = 1, \tag{3.3}$$

$$\sum_{j\in J} \lambda_j^k x_j \leq \alpha_k x_o, \tag{3.4}$$

$$\sum_{j\in J} \lambda_j^k y_j \geq \beta_k y_o \geq 0. \tag{3.5}$$

By (3.3) and (3.5), the sequence $\{\beta_k\}$ is bounded, and hence it has a convergent subsequence. For simplicity we denote this subsequence by $\{\beta_k\}$ again, and hence $\beta_k \longrightarrow \beta_0$ for some $\beta_0 \in [0,+\infty)$. Thus, $(\alpha_k x_o, \beta_k y_o) \in T^{\mathrm{FDH_V}}$ and $(\alpha_k, \beta_k) \longrightarrow (\alpha_0, \beta_0)$. This leads to $(\alpha_0 x_o, \beta_0 y_o) \in T^{\mathrm{FDH_V}}$ due to the closedness of $T^{\mathrm{FDH_V}}$. Therefore, $\alpha_0 \in \varLambda$, and the proof is complete. □

According to Lemma 3.1, $\varLambda = [\underline{\alpha}, +\infty)$ for some $\underline{\alpha} \in \mathbb{R}$. Since $(x_o, y_o) \in T^{\mathrm{FDH_V}}$, we have $\underline{\alpha} \leq 1$.

Now, we are going to obtain the closed form of the RF $\beta_o(.)$, *i.e.* the maximum proportion of the output vector y_o, possible in technology $T^{\mathrm{FDH_V}}$, for all feasible input vectors αx_o. Toward this end, we have to solve the following mixed-integer nonlinear programming problem for each $\alpha \in \varLambda$:

$$\begin{aligned} \beta_o(\alpha) = \max\ & \beta \\ s.t.\ & \sum_{j\in J} \lambda_j x_j \leq \alpha x_o, \\ & \sum_{j\in J} \lambda_j y_j \geq \beta y_o, \\ & \sum_{j\in J} \lambda_j = 1, \\ & \lambda_j \in \{0,1\};\ j \in J. \end{aligned} \tag{3.6}$$

Solving Problem (3.6) for all α values in $\varLambda$ is impossible in practice. Hence, we provide an algorithm to obtain the formula of $\beta_o(.)$ function. However, we use the structure of Problems (2.2) and (3.6) in sketching our algorithm, it works without solving any mathematical programming problem.

Suppose that $\mathrm{DMU}_o = (x_o, y_o)$ is the unit under consideration, it has G-IRS status, and it is $\mathrm{FDH_V}$-efficient. Although we consider only G-IRS case, the remain cases can be analyzed analogously. Since G-IRS prevails, increasing the input vector x_o to αx_o, $(\alpha > 1)$, leads to a greater proportional increase in all outputs at some moment for some α.

It is obvious that $\beta_o(\alpha)$ is a non-decreasing function of α. Although Podinovski [15] proved that RF is a concave continuous function under convex PPSs, it will be shown that it is a stepwise noncontinuous function under FDH technologies.

Since, by possibility axiom $(\alpha x_o, y_o) \in T^{\mathrm{FDH_V}}$ for each $\alpha \geq 1$, we have $\beta_o(\alpha) \geq 1$ for each $\alpha \geq 1$. On the other hand, since DMU_o is an $\mathrm{FDH_V}$-efficient unit, $\beta_o(1) = 1$. At the beginning, we are going to find the interval of α values on which $\beta_o(\alpha) = 1$. To this end, define

$$\alpha_{jo} := \max_i \frac{x_{ij}}{x_{io}}, \quad \beta_{jo} := \min_r \frac{y_{rj}}{y_{ro}}, \tag{3.7}$$

$$S^1 := \{j \in J : \beta_{jo} > 1\}, \tag{3.8}$$

$$\alpha^1 := \min_{j\in S^1} \alpha_{jo}, \tag{3.9}$$

$$F^1 := \{j \in S^1 :\ x_j \leq \alpha^1 x_o\}. \tag{3.10}$$

Theorem 3.2. *Assume that $DMU_o = (x_o, y_o)$ is the unit under consideration, it is FDH_V-efficient and G-IRS prevails at this unit. Then*
(i) $S^1 \neq \emptyset$.
(ii) $\beta_o(\alpha) = 1$ for each $\alpha \in [1, \alpha^1)$.
(iii) $\beta_o(\alpha^1) > 1$ and $\beta_o(\alpha^1) = \max_{j \in F^1} \beta_{jo}$.

Proof. (i): Since G-IRS prevails at DMU_o, we have $\theta_o^{\mathrm{FDH_C}} < 1$ and $\theta_o^{\mathrm{FDH_{ND}}} > \theta_o^{\mathrm{FDH_C}} = \theta_o^{\mathrm{FDH_{NI}}}$. We claim $\varphi_o^{\mathrm{FDH_{NI}}} = \varphi_o^{\mathrm{FDH_C}}$. Let $\sigma^* := (\lambda_k^*,\ \lambda_j^* = 0;\ j \neq k,\ \varphi = \varphi_o^{\mathrm{FDH_C}})$ be a part of an optimal solution to (2.2) with $\Delta = \mathrm{FDH_C}$. Then $\lambda_k^* > 0,\ \lambda_k^* x_k \leq x_o$, and $\lambda_k^* y_k \geq \varphi_o^{\mathrm{FDH_C}} y_o$. If $\lambda_k^* \leq 1$, then σ^* is a part of a feasible solution to (2.2) with $\Delta = \mathrm{FDH_{NI}}$, and so, $\varphi_o^{\mathrm{FDH_C}} \leq \varphi_o^{\mathrm{FDH_{NI}}} \leq \varphi_o^{\mathrm{FDH_C}}$. The desired equality is derived in this case. If $\lambda_k^* > 1$ and $\frac{\lambda_k^*}{\varphi_o^{\mathrm{FDH_C}}} \geq 1$, then the vector

$$\left(\lambda_k = \frac{\lambda_k^*}{\varphi_o^{\mathrm{FDH_C}}},\ \lambda_j = 0;\ j \neq k,\ \theta = \frac{1}{\varphi_o^{\mathrm{FDH_C}}}\right)$$

is a part of a feasible solution to (2.1) with $\Delta = \mathrm{FDH_{ND}}$. This leads to $\theta_o^{\mathrm{FDH_{ND}}} \leq \frac{1}{\varphi_o^{\mathrm{FDH_C}}} = \theta_o^{\mathrm{FDH_C}}$. This contradicts the assumption. The last case is $\lambda_k^* > 1$ and $\frac{\lambda_k^*}{\varphi_o^{\mathrm{FDH_C}}} < 1$. This implies $x_k \leq x_o$ and $y_k \geq \frac{\varphi_o^{\mathrm{FDH_C}}}{\lambda_k^*} y_o > y_o$, and contradicts $\mathrm{FDH_V}$-efficiency of DMU_o.

So far, we proved $\varphi_o^{\mathrm{FDH_{NI}}} = \varphi_o^{\mathrm{FDH_C}} > 1$. Hence, there exists some $j \in J$ and some $\lambda_j \in (0, 1]$ satisfying

$$\frac{y_{rj}}{y_{ro}} \geq \frac{\lambda_j y_{rj}}{y_{ro}} \geq \varphi_o^{\mathrm{FDH_{NI}}} > 1, \quad \forall r.$$

This implies $j \in S^1$.

(ii): First notice that $\alpha^1 > 1$ due to the $\mathrm{FDH_V}$-efficiency of DMU_o. By contradiction, assume that $\beta_o(\hat{\alpha}) > 1$ for some $\hat{\alpha} \in [1, \alpha^1)$. Then there exists some $k \in J$ such that $x_k \leq \hat{\alpha} x_o$ and $y_k \geq \beta_o(\hat{\alpha}) y_o$. Therefore, $k \in S^1$, and

$$\alpha^1 = \min_{j \in S^1} \alpha_{jo} \leq \alpha_{ko} = \max_i \frac{x_{ik}}{x_{io}} \leq \hat{\alpha} < \alpha^1.$$

This is an obvious contradiction, and the proof of this part is complete.

(iii): Assume that, the minimum in defining α^1 happens at index t. Then $t \in S^1$ and $x_t \leq \alpha^1 x_o$. Therefore, F^1 is a nonempty set.

Due to (3.7), we have $y_j \geq \beta_{jo} y_o$ for each j. Therefore, for each $j \in F^1$ we derive $(\alpha^1 x_o, \beta_{jo} y_o) \in T^{\mathrm{FDH_V}}$. This implies

$$\beta_o(\alpha^1) \geq \max_{j \in F^1} \beta_{jo}. \tag{3.11}$$

From the above inequality, it is clear that $\beta_o(\alpha^1) > 1$. On the other hand, according to Problem (3.6), there exists some $p \in J$ such that

$$x_p \leq \alpha^1 x_o, \quad y_p \geq \beta_o(\alpha^1) y_o. \tag{3.12}$$

Therefore, we have

$$\beta_{po} = \min_r \frac{y_{rp}}{y_{ro}} \geq \beta_o(\alpha^1) > 1,$$

which implies $p \in F^1$. So,

$$\beta_o(\alpha^1) \underbrace{\leq}_{\text{by (3.12)}} \min_r \frac{y_{rp}}{y_{ro}} = \beta_{po} \leq \max_{j \in F^1} \beta_{jo} \underbrace{\leq}_{\text{by (3.11)}} \beta_o(\alpha^1).$$

This completes the proof. $\square$

By Theorem 3.2, if the input vector x_o changes to αx_o for $\alpha \in [1, \alpha^1)$, then the proportional maximum output vector y_o does not change in response to this input-change, *i.e.*, $\beta_o(\alpha) = 1$. At $\alpha = \alpha^1$, the matter is different. If input vector x_o changes to $\alpha^1 x_o$, then the proportional maximum output vector y_o changes to $\big(\max_{j\in F^1} \beta_{jo}\big) y_o$.

In summary, so far, we showed

$$\beta_o(\alpha) = \begin{cases} 1, & \text{for } \alpha \in [1, \alpha^1) \\ \max\limits_{j\in F^1} \beta_{jo}, & \text{for } \alpha = \alpha^1. \end{cases}$$

If $(\alpha^1 x_o, \beta_o(\alpha^1) y_o)$ is an MPSS, then the procedure is terminated; otherwise, we are going to obtain $\beta_o(\alpha)$ formula for $\alpha > \alpha^1$. To this end, define

$$S^2 := \{j \in S^1 : \beta_{j_o} > \beta_o(\alpha^1)\}, \tag{3.13}$$

$$\alpha^2 := \min_{j\in S^2} \alpha_{jo}, \tag{3.14}$$

$$F^2 := \{j \in S^2 :\ x_j \leq \alpha^2 x_o\}. \tag{3.15}$$

Theorem 3.3 provides the α-interval on which $\beta_o(\alpha) = \beta_o(\alpha^1)$.

Theorem 3.3. *Assume that $DMU_o = (x_o, y_o)$ is the unit under consideration, it is FDH$_V$-efficient and G-IRS prevails at this unit. If $(\alpha^1 x_o, \beta_o(\alpha^1) y_o)$ is not an MPSS, then*
(i) $S^2 \neq \emptyset$.
(ii) $\beta_o(\alpha) = \beta_o(\alpha^1)$ for each $\alpha \in [\alpha^1, \alpha^2)$.
(iii) $\beta_o(\alpha^2) > \beta_o(\alpha^1)$ and $\beta_o(\alpha^2) = \max_{j\in F^2} \beta_{jo}$.

Proof. (i): Set $(\hat{x}, \hat{y}) := (\alpha^1 x_o, \beta_o(\alpha^1) y_o)$. Assume that $\hat{\varphi}^{\Delta}$ denotes the output-oriented efficiency score of $(\hat{x}, \hat{y})$. It can be shown that $\hat{\varphi}^{\mathrm{FDH_C}} = \frac{\alpha^1}{\beta_o(\alpha^1)} \varphi_o^{\mathrm{FDH_C}}$ and $\hat{\varphi}^{\mathrm{FDH_{NI}}} = \frac{\alpha^1}{\beta_o(\alpha^1)} \varphi_o^{\mathrm{FDH_{NI}}}$. The first equality is easy to prove, while in the proof of the second equality it should be noticed that $\lambda_k^* \alpha^1 \leq 1$ when $(\lambda_k^*,\ \lambda_j^* = 0;\ j \neq k)$ is a part of an optimal solution to (2.2) with $\Delta = \mathrm{FDH_{NI}}$.

Since G-IRS prevails at DMU$_o$, by a manner similar to the proof of Theorem 3.2, we have $\varphi_o^{\mathrm{FDH_{NI}}} = \varphi_o^{\mathrm{FDH_C}} > 1$. Hence, $\hat{\varphi}^{\mathrm{FDH_{NI}}} = \hat{\varphi}^{\mathrm{FDH_C}}$. Furthermore, $\hat{\varphi}^{\mathrm{FDH_C}} > 1$ because $(\hat{x}, \hat{y})$ is not an MPSS. Therefore, there exists some $j \in J$ and some $\lambda_j \in (0, 1]$ satisfying

$$\frac{y_{rj}}{y_{ro}} \geq \frac{\lambda_j y_{rj}}{y_{ro}} \geq \hat{\varphi}^{\mathrm{FDH_{NI}}} \beta_o(\alpha^1) > \beta_o(\alpha^1), \quad \forall r.$$

This implies $j \in S^2$.

The proof of parts (ii) and (iii) is similar to that of corresponding parts in Theorem 3.2, and is hence omitted. $\square$

Now, by Theorems 3.2 and 3.3, we have

$$\beta_o(\alpha) = \begin{cases} 1, & \text{for } \alpha \in [1, \alpha^1) \\ \max\limits_{j\in F^1} \beta_{jo}, & \text{for } \alpha \in [\alpha^1, \alpha^2), \\ \max\limits_{j\in F^2} \beta_{jo}, & \text{for } \alpha = \alpha^2. \end{cases}$$

If $(\alpha^2 x_o, \beta_o(\alpha^2) y_o)$ is an MPSS, then the procedure is terminated; otherwise, it will be continued. In the kth iteration of this procedure we have

TABLE 1. Data in Example 3.4.

DMUs	x	y
A	2	1
B	5	2
C	6	6
D	7	6.5
E	8	8
F	10.5	9
G	11	10

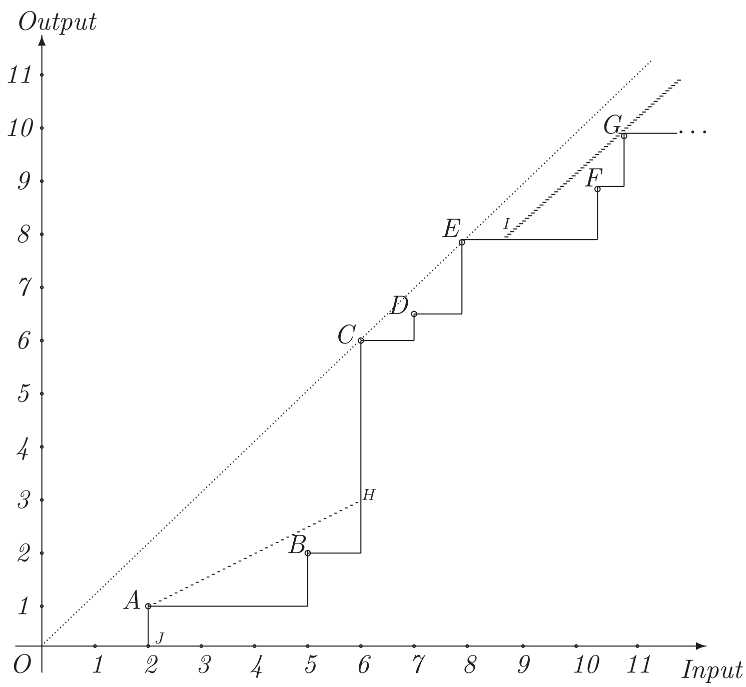

FIGURE 1. The FDH PPSs in Example 3.4.

$$\begin{aligned}
&S^k := \{j \in S^{k-1} : \beta_{j_o} > \beta_o(\alpha^{k-1})\},\\
&\alpha^k = \min_{j\in S^k} \alpha_{jo},\\
&F^k := \{j \in S^k : \; x_j \leq \alpha^k x_o\}.
\end{aligned}$$

By following this manner, until getting an MPSS, the RF $\beta_o(.)$ is fully obtained. Algorithm 1 and Example 3.4 clarify this procedure more.

Example 3.4. Consider a set of seven DMUs consisting of DMU$_A, \ldots,$ DMU$_G$, in which each DMU utilizes a single-input to produce a single-output. The data of these DMUs are listed in Table 1.

The frontier of the PPS with various RTS assumptions can be seen in Figure 1. The frontier in VRS case has been depicted by the dark line segments. The frontier under CRS assumption is the halfline starting from the origin passing through two points C, E. The frontier in NIRS case is the union of the halfline $\{G+\lambda(1,0)^t : \lambda \geq 0\}$

TABLE 2. α_{jA} and β_{jA} values in Example 3.4.

j	α_{jA}	β_{jA}
A	1	1
B	2.5	2
C	3	6
D	3.5	6.5
E	4	8
F	5.25	9
G	6.5	10

(recall that superscript "t" stands for transpose) with $\overline{OC} \cup \overline{CE} \cup \overline{EI} \cup \overline{IG}$. Moreover, the frontier under NDRS assumption is the union of the line segments $\overline{JA} \cup \overline{AH} \cup \overline{HC}$ and the halfline $\{C + \lambda(1,1)^t : \lambda \geq 0\}$.

Examining the GRS status of these DMUs by ([15], Thm. 3) shows that A, B have G-IRS status, C, E have G-CRS status, D has G-SCRS status, and F, G have G-DRS status.

We calculate the RF for DMU$_A$. The values of α_{jA} and β_{jA} have been listed in Table 2.

As mentioned above, G-IRS prevails at DMU$_A$. Therefore, a proportional increase in its input leads to a greater increase in its output at some moment. The RF (response of outputs to increase in inputs) for this DMU is calculated as follows. According to Table 2,

$$S^1 = \{B, C, D, E, F, G\}, \quad \alpha^1 = 2.5, \quad F^1 = \{B\},$$

and hence, by Theorem 3.2,

$$\beta_A(\alpha) = 1, \ \forall \alpha \in [1, 2.5), \quad \text{and} \quad \beta_A(2.5) = 2.$$

Setting $(\hat{x}, \hat{y}) := (\alpha^1 x_A, \beta_A(\alpha^1) y_A) = (5, 2)$, we have $\hat{\theta}^{\mathrm{FDH_C}} = 0.4$, and hence this point is not an MPSS. So, the procedure will be continued and we have

$$S^2 = \{C, D, E, F, G\}, \quad \alpha^2 = 3, \quad F^2 = \{C\}.$$

Thus, by Theorem 3.3,

$$\beta_A(\alpha) = 2, \ \forall \alpha \in [2.5, 3), \quad \text{and} \quad \beta_A(3) = 6.$$

Setting $(\hat{x}, \hat{y}) := (\alpha^2 x_A, \beta_A(\alpha^2) y_A) = (6, 6)$, we have $\hat{\theta}^{\mathrm{FDH_C}} = 1$, and hence this point is an MPSS. Therefore, the procedure will be terminated and we have

$$\beta_A(\alpha) = \begin{cases} 1, & \text{for } \alpha \in [1, 2.5) \\ 2, & \text{for } \alpha \in [2.5, 3) \\ 6, & \text{for } \alpha = 3 \end{cases}$$

Algorithm 1 summarizes the above procedure for calculating the RF, $\beta_o(.)$.

About Step 5 in Algorithm 1, this step is checking if $(\alpha^k x_o, \beta_o(\alpha^k) y_o)$ is an MPSS or not. To this end, one should check $\min_{j \in J}\{\max_i\{\frac{x_{ij}\lambda_*^{jo}}{\alpha^k x_{io}}\}\}$ is less than one or not, where $\lambda_*^{jo} = \max_r \frac{\beta_o(\alpha^k) y_{ro}}{y_{rj}}$. It is not difficult to see that $\lambda_*^{jo} = \frac{\beta_o(\alpha^k)}{\beta_{jo}}$, and hence

$$\max_i\{\frac{x_{ij}\lambda_*^{jo}}{\alpha^k x_{io}}\} = \frac{\beta_o(\alpha^k)}{\beta_{jo}} \frac{1}{\alpha^k} \alpha_{jo}.$$

Algorithm 1:

Step 0.	Let FDH_{V}-efficient $DMU_o = (x_o, y_o)$ with G-IRS status be given.
Step 1.	For $j = 1$ to n set $\alpha_{jo} = \max_i \dfrac{x_{ij}}{x_{io}}$ and $\beta_{jo} = \min_r \dfrac{y_{rj}}{y_{ro}}$.
Step 2.	Set $\alpha^0 = 1,\ \beta_o(\alpha^0) = 1,\ S^0 = J$, and $k = 1$.
Step 3.	Set $S^k = \{j \in S^{k-1} :\ \beta_{jo} > \beta_o(\alpha^{k-1})\}$, $\alpha^k = \min_{j\in S^k} \alpha_{jo}$, and $F^k = \{j \in S^k :\ x_j \le \alpha^k x_o\}$. Write "$\beta_o(\alpha) = \beta_o(\alpha^{k-1})$ for $\alpha \in [\alpha^{k-1}, \alpha^k)$."
Step 4.	Set $\beta_o(\alpha^k) = \max_{j\in F^k} \beta_{jo}$.
Step 5.	For $j = 1$ to n set $\theta^{jo} = \frac{\beta_o(\alpha^k)}{\alpha^k}\frac{\alpha_{jo}}{\beta_{jo}}$. If $\min_{j\in J}\{\theta^{jo}\} = 1$, then STOP; else set $k = k + 1$ and go to Step 3.

Example 3.5 below clarifies Step 1 of Algorithm 1.

Example 3.5. Consider a set of three DMUs consisting of DMU_A, DMU_B, and DMU_C, in which each DMU consumes two inputs to produce a single constant output.

Let $(x_{1A}, x_{2A}, y_{1A}) = (0.5, 3, 1)$, $(x_{1B}, x_{2B}, y_{1B}) = (1, 2, 1)$, and $(x_{1C}, x_{2C}, y_{1C}) = (2, 1, 1)$.

The frontier of the PPS with VRS assumption at level $y = 1$ has been depicted in Figure 2. Let $o = B$, *i.e.*, DMU_B be under consideration. By running Step 1 of Algorithm 1 for this unit, we have

$$\alpha_{AB} = \max\{\tfrac{0.5}{1}, \tfrac{3}{2}\} = \tfrac{3}{2},$$
$$\alpha_{CB} = \max\{\tfrac{2}{1}, \tfrac{1}{2}\} = \tfrac{2}{1}.$$

In fact, $\alpha_{CB} = 2$ is the minimum value of α such that $\alpha x_B \ge x_C$. Precisely speaking, due to Algorithm 1, we have $\alpha_{jo} = \max_i \dfrac{x_{ij}}{x_{io}}$, and so, $\alpha_{jo} x_o \ge x_j$. Furthermore, $\alpha_{jo} = \min\{\alpha : \alpha x_o \ge x_j\}$.

In Figure 2, consider $o = B$ and $j = C$. The shaded area denotes the vectors x (in input space) satisfying $x \ge x_C$. According to the above explanation, for obtaining α_{CB} we depict the ray starting from the origin in direction B. The first point at which this ray intersects the shaded area is corresponding to α_{CB}. In Figure 2, this point is $(2, 4)$, and so $\alpha_{CB} = 2$.

A similar figure and discussion can be provided for β_{jo} in output space.

Theorem 3.6 shows that Algorithm 1 terminates in finite many iterations and it is polynomial-time from complexity standpoint.

Theorem 3.6. *(Convergence and complexity) Algorithm 1 terminates in finite many iterations and its complexity is $O(n^3)$ (it is polynomial-time).*

Proof. Convergence: It is clear that, in Algorithm 1, $S^1 \subseteq J$ and $S^{k+1} \subseteq S^k$ for each k. Assume that max in Step 4 happens at $t \in F^k \subseteq S^k$. Then $t \notin S^{k+1}$. Hence, $S^{k+1} \subseteq S^k$ and $S^{k+1} \ne S^k$ for each k. Therefore, k index will be at most equal to n. If $(\alpha^k x_o, \beta_o(\alpha^k) y_o)$ is an MPSS for some $k \in \{1, 2, \dots, n-1\}$, then due to Step 5, algorithm terminates at iteration k. Otherwise, $(\alpha^k x_o, \beta_o(\alpha^k) y_o)$ is not an MPSS for each $k \in \{1, 2, \dots, n-1\}$; and we get $S^n = \emptyset$ which implies that $(\alpha^{n-1} x_o, \beta_o(\alpha^{n-1}) y_o)$ is an MPSS (see the proof of part (i) in Theorem 3.3). This makes a contradiction and proves the finite convergence of the algorithm.

Complexity: The number of basic arithmetic operations in Step 1 is at most equal to $(n-1)(2m + 2s - 2)$, and hence, taking into account that in performance analysis $n \gg m + s$, the complexity of this step is $O(n^2)$. Similarly, it is seen that the complexity of Step 3 is $O(n^2)$ as well. The complexity of Steps 4 and 5 is $O(n)$. Therefore, due to the maximum number of iterations (n), the complexity of Algorithm 1 is $O(n^3)$. $\square$

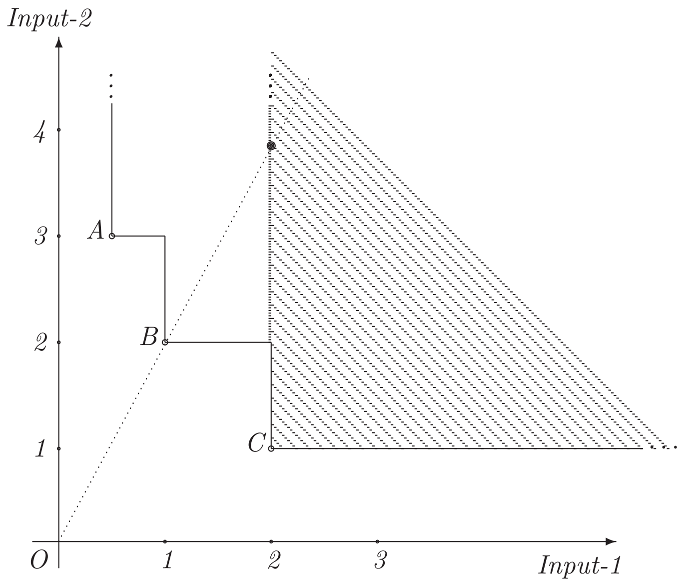

FIGURE 2. The $\mathrm{FDH_V}$ PPS (in input space) in Example 3.5.

4. A Real Application and Computational Discussions

The current section reports a real application showing the usefulness of the proposed algorithm. Beside the theoretical validity, this experimental study establishes the reasonability of the proposed algorithm from both practical and computational points of view.

We use the data set of 80 Iranian secondary schools participating in the Third International Mathematics and Science Study (TIMSS). The data is extracted from a project offered by Educational Organization of Tehran and done by Soleimani-damaneh and Sangari [23]. We have analyzed this data set in [12] as well. Each DMU has four input factors,

Input-1: the income level of the parents of the students,
Input-2: educational facilities of the students at home,
Input-3: contribution of the parents towards the school programs, and
Input-4: the education level of the teachers.

The single output is

Output: the school's GPA based on individual students' grades of each school in the TIMSS study.

The data have been listed in Table 3. See [23] for the manner of calculating the aforementioned input-output factors. We wrote the codes of determining FDH efficiency and GRS status, based on the enumeration methods investigated in [11, 19, 22, 24], under the MATLAB2015 environment. As mentioned in the preceding section, the current work focuses only on G-IRS case. Among the considered 80 DMUs (80 Iranian secondary schools participated in TIMSS study), five DMUs #20, #40, #58, #72, and #79 are $\mathrm{FHD_V}$-efficient and belong to the G-IRS class.

First we implemented Algorithm 1 for aforementioned five units in MATLAB2015 environment, but we found some errors (incompatibilities with theory) in the results. The reason is due to the precision of the numeric calculations in MATLAB. Notice that the precision of the numeric calculation plays a crucial role in Steps 3 and 5 of Algorithm 1. By default, MATLAB uses 16 digits of precision. We used "*vpa*" function to increase

TABLE 3. Data of the real-world illustration.

j	x_{1j}	x_{2j}	x_{3j}	x_{4j}	y_{ij}	j	x_{1j}	x_{2j}	x_{3j}	x_{4j}	y_{1j}
1	13.7	16.5	7	7	353.8	41	5.3	11	7	5	440.2
2	10.5	15.6	7	8	353.3	42	14.4	14.7	5	8	439
3	7.2	14.2	6	12	388.5	43	5.7	15.4	8	8	419.3
4	15.6	15.3	10	8	368.7	44	13	16.7	5	9	468.6
5	7.2	11.5	3	6	446.5	45	9.7	15.9	5	6	433
6	7.6	15	7	8	394.6	46	4.7	14	9	8	430.7
7	8.1	16.1	8	4	337.7	47	13.6	15.4	5	8	469.2
8	12.3	14.8	10	6	332.2	48	10.3	16.7	8	6	444.8
9	4.2	14	4	4	392.4	49	6.7	15.7	9	6	402.7
10	7.7	13.3	7	6	357.8	50	9.1	17.3	4	8	418.7
11	6.7	16.1	6	6	409.3	51	5.1	15.7	9	6	422.9
12	5.9	11.4	7	4	419.1	52	5.5	15.2	7	8	333.9
13	5.7	15.6	9	6	419.3	53	8.1	15.5	7	10	386.3
14	5.7	14.2	9	8	415.7	54	6.9	14.7	6	6	396.8
15	3.7	14.5	7	10	383	55	8.7	14.7	8	8	358.8
16	7.7	14.5	6	8	348.3	56	9.9	16.1	5	3	448.8
17	7.1	15	6	6	421	57	11.3	11.7	5	4	439.2
18	12.4	13.6	7	9	400.1	58	11.3	15.4	5	6	447.9
19	4.2	14	7	8	362.4	59	10.6	14.8	7	8	434.1
20	5.1	13.8	8	8	356.7	60	5	16.3	4	6	411.6
21	10.7	16	4	10	421.5	61	12.5	16.2	6	5	441.6
22	11.7	14.8	4	9	435.5	62	16.2	16.2	18	4	471.5
23	11.5	14.8	4	6	371.9	63	10.3	17	6	6	461.4
24	9.7	16.4	6	6	437.5	64	5.8	13.9	9	6	418.8
25	7.9	15.1	5	6	413.9	65	15.1	15	9	8	374.8
26	12.3	15.7	3	5	482.2	66	8	14.5	8	8	347.7
27	6	13.8	10	6	381.6	67	8	14.9	10	6	438.7
28	15	14.9	7	8	402.8	68	9.2	15.6	7	8	422.1
29	6.6	13.8	6	4	440.8	69	7.9	15.2	5	6	412.7
30	6.5	15.6	10	4	411.1	70	6.8	15	9	6	427.2
31	15.6	15.1	5	8	395.7	71	13.2	14.4	6	6	430.7
32	6.3	14.9	6	6	374.4	72	8.2	15.2	6	6	450.8
33	13.6	16.6	7	6	381.5	73	6.4	14.3	7	6	363.2
34	19.2	16.5	5	6	469.8	74	6.8	15.6	9	6	364
35	6.3	14.7	6	6	441.5	75	13.3	17.5	4	8	494.8
36	9.9	15.3	6	6	466.6	76	12.6	18.5	3	6	529.6
37	10.5	15.9	8	8	449.3	77	9.9	17.5	5	8	610.3
38	7.1	16.7	6	6	398.2	78	12.5	13.8	3	8	542.3
39	15.1	14.5	6	8	413	79	14.6	17.7	3	7	506
40	11.3	14.8	5	6	447.3	80	17.3	17.1	4	6	577.7

the precision. Although adding *vpa* function with a high degree of precision generated more proper results (compatible with theory), it increased the running time of the implementations considerably. To avoid a big running time, we implemented Algorithm 1 for five units #20, #40, #58, #72, and #79 in EXCEL, and derived the results listed in Tables 4–8. The numbers reported in these tables are rounded by four digits. We derived more precise outcomes in a reasonable running time (it takes about one minute for each DMU in our project, on a PC with Intel Core i3-3220 processor (3.30-GHz) and 4GB of installed memory).

In Tables 4–8, the last column from left denotes the FDH_C-efficiency score of $(\alpha^k x_o, \beta_o(\alpha^k) y_o)$ derived from Step 5 of the algorithm. In fact, the algorithm stops when this score equals to one, *i.e.*, the production point $(\alpha^k x_o, \beta_o(\alpha^k) y_o)$ is an MPSS.

TABLE 4. Results of Algorithm 1 for DMU_{20} in real-world illustration.

Iter. (k)	α^k	$\beta_{20}(\alpha^k)$	score
0	1.000	1.0000	0.8421
1	1.0145	1.1001	0.9131
2	1.0392	1.2341	1.0000

TABLE 5. Results of Algorithm 1 for DMU_{40} in real-world illustration.

Iter. (k)	α^k	$\beta_{40}(\alpha^k)$	score
0	1.000	1.0000	0.9772
1	1.0405	1.0013	0.9404
2	1.0878	1.0034	0.9013
3	1.0885	1.0780	0.9678
4	1.2500	1.1840	0.9256
5	1.3333	1.3644	1.0000

TABLE 6. Results of Algorithm 1 for DMU_{58} in real-world illustration.

Iter. (k)	α^k	$\beta_{58}(\alpha^k)$	score
0	1.000	1.0000	0.9785
1	1.0455	1.0020	0.9379
2	1.0885	1.0766	0.9678
3	1.2013	1.1824	0.9631
4	1.3333	1.3626	1.0000

Due to Algorithm 1 and Table 4, the closed form of the response function associated with DMU_{20} is

$$\beta_{20}(\alpha) = \begin{cases} 1, & \alpha \in [1, 1.0145) \\ 1.1001, & \alpha \in [1.0145, 1.0392) \\ 1.2341, & \alpha = 1.0392. \end{cases}$$

Recall that the numbers have been rounded by four digits. This function for DMU_{40} is as follows (according to Algorithm 1 and Tab. 5).

$$\beta_{40}(\alpha) = \begin{cases} 1, & \alpha \in [1, 1.0405) \\ 1.0013, & \alpha \in [1.0405, 1.0878) \\ 1.0034 & \alpha \in [1.0878, 1.0885) \\ 1.0780, & \alpha \in [1.0885, 1.2500) \\ 1.1840, & \alpha \in [1.2500, 1.3333) \\ 1.3644, & \alpha = 1.3333. \end{cases}$$

The closed form of the RFs of remaining three DMUs can be written analogously (invoking Tabs. 6–8). These five RFs have been illustrated in Figures 3 and 4.

From Figure 3a, it is seen that if the input of DMU_{20} increases from x_{20} to αx_{20} for $\alpha \in [1, 1.0145)$, then the maximum proportion of feasible output, in T^V, will be 1. Furthermore, if the input of DMU_{20} increases from x_{20} to αx_{20} for $\alpha \in [1.0145, 1.0392)$, then the maximum proportion of feasible output, in T^V, will be 1.1001. Moreover, if the input of DMU_{20} increases from x_{20} to $1.0392 x_{20}$, then the maximum proportion of feasible

TABLE 7. Results of Algorithm 1 for DMU_{72} in real-world illustration.

Iter. (k)	α^k	$\beta_{72}(\alpha^k)$	score
0	1.000	1.0000	0.9849
1	1.2073	1.0350	0.8443
2	1.3333	1.3538	1.0000

TABLE 8. Results of Algorithm 1 for DMU_{79} in real-world illustration.

Iter. (k)	α^k	$\beta_{79}(\alpha^k)$	score
0	1.000	1.0000	0.9986
1	1.0452	1.0466	1.0000

output, in T^V, will be 1.2341. Radial increasing the inputs of DMU_{20} as αx_{20} for $\alpha > 1.0392$ is not reasonable because $(1.0392x_{20}, \beta_{20}(1.0392)y_{20})$ is an MPSS (see the last column of Tab. 4). A similar interpretation can be provided for other investigated units.

In depicting Figure 3b corresponding to DMU_{40}, we have ignored the second α-interval, because α^2 is very close to α^3. It is seen that $\beta_{40}(\alpha^0), \beta_{40}(\alpha^1)$, and $\beta_{40}(\alpha^2)$ are very close to each other. A similar situation is observed for DMU_{58} and Figure 3c as well. In addition to this point, it is seen that the number of the line segments constructing the RFs is different among investigated units. In fact, taking the topmost single point (isolated black point in top of each figure) into account, the number of the line segments of each RF is corresponding to the number of the iterations of the algorithm and so related to the running time. Although this quantity is bounded above by n (the number of DMUs), we cannot say anything about its exact value. Indeed, this quantity is related to the number of the observed units which will be in the way of DMU_o in getting MPSS position. When these units are so close to each other or so close to the unit under consideration, the line segments generating RF might be close to each other (see Figs. 3b and 3c).

One of the interesting points is that the RF of DMU_{40} is very close to that of DMU_{58}. Notice that the input-output factors of these two units are very close to each other.

The information derived from the RF of the unit under investigation can be used for various practical purposes, including resource allocation, budget planning, and deciding about contracting/expanding the activity of the unit.

Although Algorithm 1 is polynomial-time, our computational experiments show that its running time strongly depends on the number of the line segments which construct RF (the number of the units which are observed in the way of DMU_o to get MPSS position) and the precision of numeric calculation that DM is going to has. The latter can be controlled by running the algorithm in MATLAB environment utilizing *vpa* function or EXCEL.

Although Podinovski [15] proved that RF is a concave continuous function under convex PPSs, from both theoretical and applied parts of the paper, it is seen that RF is a stepwise discontinuous function under FDH technologies. Scale Elasticity (SE) is one of the most important measures for performance analysis [17,18]. Under convex technologies, SE of a given VRS-efficient DMU is defined with respect to the derivative of RF; see [17,18]. The outcomes of the present paper show that, in contrast to convex technologies, in FDH ones the derivative of RF at a given $\mathrm{FDH_V}$-efficient DMU is either zero or infinity, and so it does not provide useful information about the SE measure at the corresponding unit. This important point shows that in FDH production technologies the SE measure should be redefined independent to the RF structure. It is worth studying from both theoretical and applied points of view.

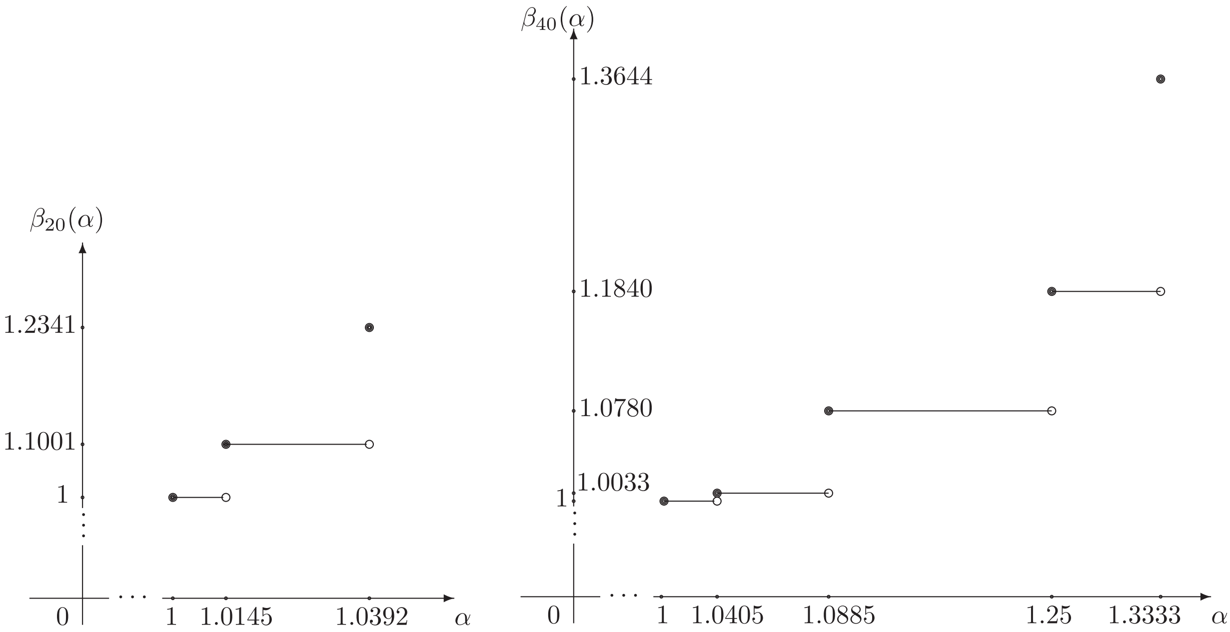

Figure 3(a): RF of DMU_{20}

Figure 3(b): RF of DMU_{40}

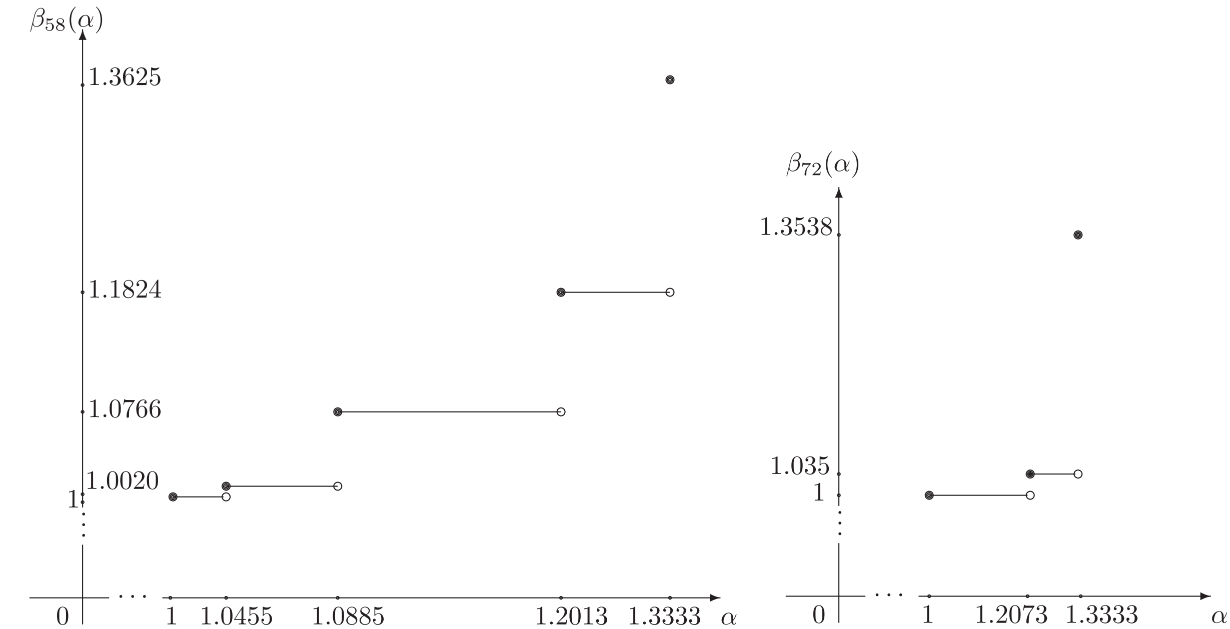

Figure 3(c): RF of DMU_{58}

Figure 3(d): RF of DMU_{72}

FIGURE 3. RFs of DMUs #20, #40, #58, and #72 in real-world illustration.

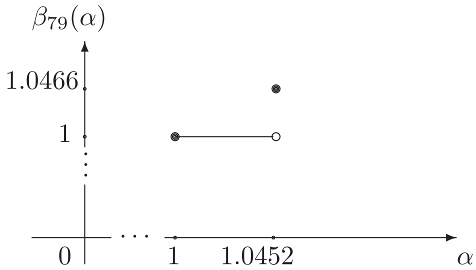

FIGURE 4. RF of DMU #79 in real-world illustration.

5. CONCLUSIONS

In this paper, a response function, which gives the radial maximum feasible output resulting from changes in inputs, has been investigated. Some theorems have been established to calculate the response function. Resulting from theoretical materials, a polynomial-time finitely convergent algorithm has been sketched which enables us to provide the full formula of the response function. In addition to the theoretical results, we have illustrated the applied and computational abilities of our approach.

REFERENCES

[1] O. Abdelsalam, M.D. Fethi, J.C. Matallínc, E. Tortosa-Ausina, On the comparative performance of socially responsible and Islamic mutual funds. *J. Econ. Behav. Organ.* **103** (2014) S108–S128.

[2] W. Briec, K. Kerstens, Input, output and graph technical efficiency measures on non-convex FDH models with various scaling laws: An integrated approach based upon implicit enumeration algorithms. *TOP* **14** (2006) 135–166.

[3] G. Cesaroni, D. Giovannola, Average-cost efficiency and optimal scale sizes in nonparametric analysis. *Eur. J. Oper. Res.* **242** (2015) 121–133.

[4] G. Cesaroni, K. Kerstens, I. Van de Woestyne, Global and local scale characteristics in convex and nonconvex nonparametric technologies: A first empirical exploration. *Eur. J. Oper. Res.* **259** (2017) 576-586.

[5] L. Cherchye, T. Kuosmanen, T. Post, FDH directional distance functions: With an application to European Commercial Banks. *J. Prod. Anal.* **15** (2001) 201–215.

[6] D. Deprins, L. Simar, H. Tulkens, Measuring labor efficiency in post offices. In: The Performance of Public Enterprises: Concepts and Measurements, edited by M. Marchand, P. Pestieau, and H. Tulkens. North Holland, Amsterdam (1984).

[7] W.E. Diewert, K.J. Fox, Reference technology sets, Free Disposal Hulls and productivity decompositions. *Econ. Lett.* **122** (2014) 238–242.

[8] A. Emrouznejad, G. Yang, A survey and analysis of the first 40 years of scholarly literature in DEA: 1978–2016. *Socio-Econ. Plan. Sci.* **61** (2018) 4–8.

[9] P. Hadjicostas, A.C. Soteriou, One-sided elasticities and technical efficiency in multi-output production: A theoretical framework. *Eur. J. Oper. Res.* **168** (2006) 425–449.

[10] K. Kerstens, P. Vanden Eeckaut, Estimating returns to scale using non- parametric deterministic technologies: a new method based on goodness-of-fit. *Eur. J. Oper. Res.* **113** (1999) 206–214.

[11] K. Kerstens, I. Van de Woestyne, Solution methods for nonconvex free disposal hull models: A review and some critical comments. *Asia Pacific J. Oper. Res.* **31** (2014) 1–13.

[12] P.J. Korhonen, M. Soleimani-damaneh, J. Wallenius, Ratio-based RTS determination in weight-restricted DEA models. *Eur. J. Oper. Res.* **215** (2011) 431–438.

[13] H. Leleu, A linear programming framework for free disposal hull technologies and cost functions: primal and dual models. *Eur. J. Oper. Res.* **168** (2006) 340–344.

[14] H. Leleu, Mixing DEA and FDH models together. *J. Oper. Res. Soc.* **60** (2009) 1730–1737.

[15] V.V. Podinovski, Efficiency and global scale characteristics on the “No free lunch” assumption only. *J. Prod. Anal.* **22** (2004) 227–257.

[16] V.V. Podinovski, Global and local returns to scale in performance measurement. *J. Oper. Res. Soc.* **55** (2004) 170–178.

[17] V.V. Podinovski, F.R. Førsund, Differential characteristics of efficient frontiers in data envelopment analysis. *Oper. Res.* **58** (2010) 1743–1754.

[18] V.V. Podinovski, F.R. Førsund, V.E. Krivonozhko, A simple derivation of scale elasticity in data envelopment analysis. *Eur. J. Oper. Res.* **197** (2009) 149–153.

[19] M. Soleimani-damaneh, G.R. Jahanshahloo, M. Reshadi, On the estimation of returns to scale in FDH models. *Eur. J. Oper. Res.* **174** (2006) 1055–1059.

[20] M. Soleimani-damaneh, A. Mostafaee, Stability of the classification of returns to scale in FDH models. *Eur. J. Oper. Res.* **196** (2009) 1223–1228.

[21] M. Soleimani-damaneh, A. Mostafaee, Identification of the anchor points in FDH models. *Eur. J. Oper. Res.* **246** (2015) 936–943.

[22] M. Soleimani-damaneh, M. Reshadi, A polynomial-time algorithm to estimate returns to scale in FDH models. *Comput. Oper. Res.* **34** (2007) 2168–2176.

[23] M. Soleimani-damaneh, A. Sangari, Evaluation of mathematical performance of the secondary schools participating in TIMSS study using Data Envelopment Analysis (DEA) technique. Working paper, Educational Organization of Tehran (in Persian) (2006).

[24] H. Tulkens, On FDH efficiency analysis: Some methodological issues and applications to retail banking, courts and urban transit. *J. Prod. Anal.* **4** (1993) 183–210.