Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 807843, 14 pages
http://dx.doi.org/10.1155/2014/807843

Research Article

A Multiobjective Programming Method for Ranking All Units Based on Compensatory DEA Model

Haifang Cheng,[1] Yishi Zhang,[1] Jianhu Cai,[1,2] and Weilai Huang[1]

[1] *School of Management, Huazhong University of Science and Technology, Wuhan 430074, China*
[2] *College of Economics and Management, Zhejiang University of Technology, Hangzhou 310023, China*

Correspondence should be addressed to Jianhu Cai; hzdcjh@yahoo.com

Received 16 April 2014; Accepted 1 September 2014; Published 5 November 2014

Academic Editor: Lionel Amodeo

In order to rank all decision making units (DMUs) on the same basis, this paper proposes a multiobjective programming (MOP) model based on a compensatory data envelopment analysis (DEA) model to derive a common set of weights that can be used for the full ranking of all DMUs. We first revisit a compensatory DEA model for ranking all units, point out the existing problem for solving the model, and present an improved algorithm for which an approximate global optimal solution of the model can be obtained by solving a sequence of linear programming. Then, we applied the key idea of the compensatory DEA model to develop the MOP model in which the objectives are to simultaneously maximize all common weights under constraints that the sum of efficiency values of all DMUs is equal to unity and the sum of all common weights is also equal to unity. In order to solve the MOP model, we transform it into a single objective programming (SOP) model using a fuzzy programming method and solve the SOP model using the proposed approximation algorithm. To illustrate the ranking method using the proposed method, two numerical examples are solved.

1. Introduction

As a nonparametric method, data envelopment analysis (DEA) is proposed by [1] to evaluate the relative efficiency of a set of decision making units (DMUs) with the multiple inputs and outputs. For each DMU, the optimal weights for calculating efficiency value are obtained by solving a linear programming (LP) problem. By using DEA, DMUs can be divided into two categories: efficient DMUs and inefficient DMUs [2]. However, there are always more than one DMU to be evaluated as efficient, which would cause the problem that all DMUs cannot be fully discriminated [3]. Moreover, the efficiencies of different DMUs obtained by different sets of weights may be unable to be compared and ranked on the same basis [2–5].

To deal with the above-mentioned problems, many methods have been developed to rank all DMUs under the framework of DEA. The common weights DEA introduced by [6–8] is known as one of the popular methods in which all DMUs can be evaluated by a common set of weights (CSW). Since then, DEA models with CSW have been rapidly applied in many researches. For example, Sinuany-Stern and Friedman [9] developed a DR/DEA to provide the best CSW for given inputs and outputs in order to rank all DMUs on the same scale. Jahanshahloo et al. [10] presented a method to obtain the CSW of DMUs by solving only one problem, in order to measure the efficiency and to rank the efficient DMUs. Kao and Hung [11] proposed a compromise solution approach for generating a CSW which produces the vector of efficiency scores for the DMUs. Their approach is able to not only differentiate efficient DMUs but also detect abnormal efficiency scores on a common base. Amin and Toloo [12] presented an improved integrated DEA model in order to detect the most efficient DMUs. The method in their model is able to determine a CSW for all DMUs by solving a LP problem. Liu and Peng [13] introduced a DEA method to determine a CSW for the ranking of only DEA efficient DMUs. For the decision maker (DM), this ranking is based on the optimization of the group's efficiency. Jahanshahloo et al. [2] proposed two methods to rank DMUs concerning an ideal line or a special DMU. Their method needs to determine a CSW for efficient DMUs. Davoodi and Rezai [14] suggested a method to compute the efficiency scores of all DMUs and then rank them using a CSW determined by solving a LP problem. Ramón et al. [15] proposed a DEA approach for deriving a CSW to be used for the ranking of all DMUs. The idea of this approach is to minimize the deviations of the CSW from the DEA profiles of weights without zeros of the efficient DMUs. Lotfi et al. [16] developed the common weights DEA method to deal with total weights flexibility in order to allocate fixed resources using DEA. To see the other ranking approaches under the framework of DEA, we refer the readers to the review papers of [17, 18].

Recently, Khodabakhshi and Aryavash [19] proposed a very interesting variant of the basic DEA model for ranking all DMUs. In their DEA model, the sum of efficiency values of all DMUs is supposed to be equal to unity. This assumption implies that the efficiencies of all DMUs have compensatory features, so we also call it a compensatory DEA model in this paper. In order to rank all DMUs, the minimum and maximum efficiency values of each DMU are computed. Then the rank of each DMU is determined in proportion to a combination of its minimum and maximum efficiency values. To solve the compensatory DEA model easily, they transform it into a new LP model. However, the two models are not completely equivalent; thus the optimal solution of compensatory DEA model may not be obtained by solving the transformed LP model. Moreover, their ranking method does not provide more information about the weights used for calculating the efficiency scores of each DMU. To deal with these two problems, in this paper, we first improve their solution method and propose an approximation algorithm to solve the compensatory DEA model. Then, we apply the key idea of compensatory DEA model to develop a multiobjective programming (MOP) model for determining a CSW used for calculating the efficiency scores of each DMU. In the proposed MOP model, the objectives are to simultaneously maximize all common weights assigned to each input and output under constraints that the sum of efficiency values of all DMUs is equal to unity and the sum of all weights assigned to each input and output is also equal to unity. After the CSW has been determined by solving the MOP model, all DMUs can be ranked according to the efficiency scores weighted by it.

The rest of the paper is organized as follows. In Section 2, we first revisit compensatory DEA model for ranking all units and present an improved algorithm to solve the model. The proposed MOP model and solution approach are presented in Section 3. A numerical example is examined in Section 4 to illustrate the ranking method using the proposed model. Conclusions are offered in Section 5.

2. The Improved Method for Solving Compensatory DEA Model

In this section, we first revisit the compensatory DEA model proposed by [19] for ranking all DMUs. After pointing out the existing problem of their solution method, we will propose an approximation algorithm by improving it so that the model can be solved correctly. To demonstrate the improved algorithm, a numerical example is also presented in this section.

2.1. A Revisit to Compensatory DEA Model. Consider n DMUs that use m inputs to produce s outputs. Let x_{ij} ($i = 1, 2, \ldots, m$) and y_{rj} ($r = 1, 2, \ldots, s$) represent the input and output values of DMU_j ($j = 1, 2, \ldots, n$), respectively. Suppose that all input and output elements are nonnegative deterministic numbers. For a given DMU_j, then the efficiency scores are given as follows:

$$\theta_j = \frac{\sum_{r=1}^{s} y_{rj} u_r}{\sum_{i=1}^{m} x_{ij} v_i}, \quad j = 1, 2, \ldots, n, \tag{1}$$

where v_i ($i = 1, 2, \ldots, m$) and u_r ($r = 1, 2, \ldots, s$) are the input and output weights assigned to ith input and rth output, respectively.

Let DMU_o be a DMU under evaluation; then the following model is used to determine the minimum and maximum efficiency values of DMU_o:

$$\begin{aligned} &\text{(M1)} \quad \min \text{ and } \max \theta_o \\ &\text{s.t.} \quad \theta_j = \frac{\sum_{r=1}^{s} y_{rj} u_r}{\sum_{i=1}^{m} x_{ij} v_i}, \quad j = 1, 2, \ldots, n \\ &\qquad \sum_{j=1}^{n} \theta_j = 1 \\ &\qquad u_r, v_i, \theta_j \geq 0, \quad \forall i, j, r. \end{aligned} \tag{2}$$

Model (M1) must be run twice. The minimum/maximum value of θ_o is determined by minimizing/maximizing the objective function of model (M1). Model (M1) is a nonlinear programming. Using the transformation $w_{ij} := v_i \theta_j$, we can replace model (M1) by the following LP problem:

$$\begin{aligned} &\text{(M2)} \quad \min \text{ and } \max \theta_o = \sum_{r=1}^{s} y_{ro} u_r \\ &\text{s.t.} \quad \sum_{i=1}^{m} x_{io} v_i = 1 \\ &\qquad \sum_{i=1}^{m} x_{ij} w_{ij} - \sum_{r=1}^{s} y_{rj} u_r = 0, \\ &\qquad\qquad j = 1, 2, \ldots, n \\ &\qquad \sum_{j=1}^{n} w_{ij} = v_i, \quad i = 1, 2, \ldots, m \\ &\qquad u_r, v_i, w_{ij} \geq 0, \quad \forall i, j, r. \end{aligned} \tag{3}$$

Let $\theta_j^{\min}$ and $\theta_j^{\max}$ be the minimum and maximum values of θ_j, respectively, which can be obtained by solving model (M2). Then the following convex combinations are used to determine the efficiency values for each DMU:

$$\begin{aligned} &\theta_j = \lambda \theta_j^{\min} + (1 - \lambda) \theta_j^{\max}, \\ &0 \leq \lambda \leq 1, \quad j = 1, 2, \ldots, n, \end{aligned} \tag{4}$$

where

$$\lambda = \frac{1 - \sum_{j=1}^{n} \theta_j^{\max}}{\sum_{j=1}^{n} \left(\theta_j^{\min} - \theta_j^{\max}\right)}. \tag{5}$$

Considering (4) and the value of λ obtained in (5) together, the values of θ_j ($j = 1, 2, \ldots, n$) are determined. Now, the DMUs are fully ranked with respect to their efficiency scores. In other words, a DMU has a better rank if it has a greater efficiency score.

2.2. The Existing Problem for Solving Compensatory DEA Model. In compensatory DEA model, the optimal solutions of model (M1) can be found by solving model (M2). Since constraint $w_{ij} = v_i \theta_j$ is not considered in model (M2), model (M2) is a relaxation problem of model (M1), and two models are not completely equivalent. Hence, we may not obtain the optimal solutions of model (M1) by solving model (M2).

Let $\underline{u}_r'$ and $\underline{v}_i'$ be the optimal solutions of model (M2) with a minimum form. If they are also the optimal solutions of model (M1), then they must satisfy the constraints of model (M1); that is $\sum_{j=1}^{n} \underline{\theta}_j' = 1$, where the efficiency score $\underline{\theta}_j'$ is determined by (1). However the following example shows that the equation $\sum_{j=1}^{n} \underline{\theta}_j' = 1$ does not hold.

For example, suppose we want to evaluate DMU_1 in the numerical example presented in Section 2.4. For a minimum problem, we solve model (M2) to find the optimal solutions: $\underline{u}_1' = 0.000678$, $\underline{u}_2' = 0$, $\underline{v}_1' = 0.000130$, $\underline{v}_2' = 0.012707$, and $\underline{v}_3' = 0.050998$. Using (1) we can obtain that the efficiency scores $\underline{\theta}_j'$ of twelve DMUs are 0.0454, 0.0637, 0.0631, 0.0680, 0.0928, 0.0439, 0.0604, 0.0707, 0.0828, 0.0427, 0.0276, and 0.0599, respectively. Computing the sum of those efficiency scores we have $\sum_{j=1}^{12} \underline{\theta}_j' = 0.7210 < 1$, which means that $\underline{u}_r'$ and $\underline{v}_i'$ are not the optimal solutions of model (M1). Therefore, we cannot obtain the optimal solutions of model (M1) by solving model (M2).

2.3. The Proposed Approximation Algorithm for Solving Model (M1). In this section, the minimum problems as an example are used to illustrate the approximation algorithm. However, the procedure can be easily extended to solve the maximum problems.

Unlike the standard DEA, model (M1) involves the sum of linear-fractional functions in constraints. So model (M1) may not be considered a convex optimization problem. To prove it we first rewrite these constraints of linear-fractional functions as follows:

$$F_j = \theta_j \sum_{i=1}^{m} x_{ij} v_i - \sum_{r=1}^{s} y_{rj} u_r = 0, \quad j = 1, 2, \ldots, n. \tag{6}$$

The Hessian matrix of function F_j, with the order of variables being (θ, v, u), can be derived as

$$H\left(F_j\right) = \begin{pmatrix} 0 & x_j & 0 \\ x_j^T & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \in \mathfrak{R}^{(1+m+s)\times(1+m+s)}, \tag{7}$$

where $x_j = (x_{1j}, x_{2j}, \ldots, x_{mj})$. Zeros in the matrix are of appropriate dimension. Since principal minors of order 2 are $\left|H(F_j)_2\right| = -x_1^2 < 0$, that is, there exists one 2×2 principal minor in $H(F_j)$ which is negative, it can be concluded that these constraints function is nondefinite. Therefore, model (M1) is not a convex optimization, which implies that an efficient algorithm may be required to determine a globally optimal solution.

Although the optimal solutions of model (M1) cannot be found by solving model (M2), the following shows that the optimal solutions of model (M1) can be approximated by solving a sequence of LP models on the base of the optimal solution of model (M2).

Since the optimal solutions of model (M2) may not be unique, a secondary goal must be introduced in it. For the minimum problem, the secondary goal is to maximize the sum of efficiency scores of the rest DMUs; that is,

$$\max \sum_{\substack{j=1 \\ j\neq o}}^{n} \left(\frac{\sum_{r=1}^{s} y_{rj} u_r}{\sum_{i=1}^{m} x_{ij} v_i} \right). \tag{8}$$

The fractional form of the secondary objective function can be transformed to the following linear form:

$$\max \sum_{\substack{j=1 \\ j\neq o}}^{n} \left((n-1) \sum_{r=1}^{s} y_{rj} u_r - \sum_{i=1}^{m} x_{ij} v_i \right). \tag{9}$$

Therefore, introducing a sufficiently small positive number δ, model (M2) can also be rewritten as follows:

$$\begin{aligned} \text{(M3)} \quad \min \theta_o = {} & \sum_{r=1}^{s} y_{ro} u_r \\ & - \delta \sum_{\substack{j=1 \\ j\neq o}}^{n} \left((n-1) \sum_{r=1}^{s} y_{rj} u_r - \sum_{i=1}^{m} x_{ij} v_i \right) \\ \text{s.t.} \quad & \left(u_r, v_i\right) \in \aleph_2 \left(u, v\right), \end{aligned} \tag{10}$$

where $\aleph_2(u, v)$ denotes the set of the feasible solution to model (M2). Since model (M2) or (M3) is a relaxation problem of model (M1), we have the following lemma.

Lemma 1. *Let $\underline{u}_r$ and $\underline{v}_i$ be the optimal solutions of model (M3); then one has*

$$\sum_{j=1}^{n} \underline{\theta}_j \leq 1, \tag{11}$$

where $\underline{\theta}_j$ is determined by using (1).

Proposition 2. *If $\sum_{j=1}^{n} \underline{\theta}_j = 1$, then the optimal solutions $\underline{u}_r$ and $\underline{v}_i$ of model (M3) are also the optimal solutions of model (M1).*

Proof. If $\sum_{j=1}^{n} \underline{\theta}_j = 1$, then the optimal solutions $\underline{u}_r$ and $\underline{v}_i$ of model (M3) are the feasible solutions of model (M1). Since the objectives of two models are equivalent, the optimal solutions $\underline{u}_r$ and $\underline{v}_i$ of model (M3) are also the optimal solutions of model (M1). □

If $\sum_{j=1}^{n} \underline{\theta}_j < 1$, then we have to proceed further to find the optimal solutions on the basis of it. Let constraint $w_{ij} = v_i\theta_j$ be replaced by $w_{ij} \geq v_i\underline{\theta}_j$, then add it into model (M3) to formulate the following LP problem:

$$\begin{aligned} \text{(M4)}\quad \min \theta_o = &\sum_{r=1}^{s} y_{ro}u_r \\ &- \delta \sum_{\substack{j=1\\ j\neq o}}^{n} \left((n-1) \sum_{r=1}^{s} y_{rj}u_r - \sum_{i=1}^{m} x_{ij}v_i \right) \\ \text{s.t.}\quad & w_{ij} \geq v_i\underline{\theta}_j, \quad i = 1,2,\ldots,m,\ j = 1,2,\ldots,n, \\ & (u_r, v_i) \in \aleph_2(u, v). \end{aligned} \tag{12}$$

It is obvious that model (M4) is a relaxation problem of model (M1) and model (M3) is a relaxation problem of model (M4). Therefore, we have the following lemma.

Lemma 3. *Let $\underline{u}_r^{(1)}$ and $\underline{v}_i^{(1)}$ be the optimal solutions of model (M4); then one has*

$$\sum_{j=1}^{n} \underline{\theta}_j \leq \sum_{j=1}^{n} \underline{\theta}_j^{(1)} \leq 1, \tag{13}$$

where $\underline{\theta}_j^{(1)}$ is determined by (1).

According to Proposition 2, if $\sum_{j=1}^{n} \underline{\theta}_j^{(1)} = 1$, then $\underline{u}_r^{(1)}$ and $\underline{v}_i^{(1)}$ are also the optimal solutions of model (M1); otherwise, we must also continue to find the optimal solutions of model (M1).

Let $\underline{\theta}_j^{(1)}$ replace $\underline{\theta}_j$ in model (M4); then solve it to obtain the optimal solutions $\underline{u}_r^{(2)}$ and $\underline{v}_i^{(2)}$. Using (1) we will get the efficiency score $\underline{\theta}_j^{(2)}$ of each DMU and then judge whether $\sum_{j=1}^{n} \underline{\theta}_j^{(2)}$ is equal to 1 or not. The process is repeated until the sum of efficiency scores of all DMUs is approximately equal to 1 within given error.

According to the above analysis, the procedure of the proposed approximation algorithm to solve model (M1) can be described as follows.

Algorithm 4. Consider the following.

Step 1. Give permissible error $\varepsilon > 0$.

Step 2. Solve model (M3) to find the optimal solutions $\underline{u}_r$ and $\underline{v}_i$, and then calculate the efficiency score $\underline{\theta}_j$ of each DMU using (1).

Step 3. Compute the sum of the efficiency scores of all DMUs. If $\left|\sum_{j=1}^{n} \underline{\theta}_j - 1\right| \leq \varepsilon$, then stop with the optimal solutions $\underline{u}_r$ and $\underline{v}_i$ to model (M1); otherwise, $k := 1$; go to Step 4.

Step 4. Solve model (M4) to find optimal solutions $\underline{u}_r^{(k)}$ and $\underline{v}_i^{(k)}$, and then calculate the efficiency score $\underline{\theta}_j^{(k)}$ of each DMU using (1).

Step 5. Compute the sum of the efficiency scores of all DMUs. If $\left|\sum_{j=1}^{n} \underline{\theta}_j^{(k)} - 1\right| \leq \varepsilon$, then stop with the optimal solutions $\underline{u}_r^{(k)}$ and $\underline{v}_i^{(k)}$ to model (M1); otherwise go to Step 6.

Step 6. Let $\underline{\theta}_j^{(k)}$ replace $\underline{\theta}_j$ in model (M4); $k := k + 1$; go back to Step 4.

Proposition 5. *The optimal solutions of model (M4) will be close to the optimal solutions of model (M1) by using Algorithm 4.*

Proof. Let $\underline{u}_r^{(t)}$ and $\underline{v}_i^{(t)}$, $t = 1, 2, \ldots, k$, be the optimal solutions to model (M4) for tth solution; using (1) we will get the efficiency score $\underline{\theta}_j^{(t)}$ of each DMU; then we have

$$\sum_{j=1}^{n} \underline{\theta}_j \leq \sum_{j=1}^{n} \underline{\theta}_j^{(1)} \leq \sum_{j=1}^{n} \underline{\theta}_j^{(2)} \leq \cdots \leq \sum_{j=1}^{n} \underline{\theta}_j^{(k)} \leq 1. \tag{14}$$

Since $\{\sum_{j=1}^{n} \underline{\theta}_j^{(t)}\}$ increases monotonically and exists upper bound 1, we have $\sum_{j=1}^{n} \overline{\theta}_j^{(t)} \rightarrow 1$, which means that the optimal solutions $\underline{u}_r^{(t)}$ and $\underline{v}_i^{(t)}$ of model (M4) are close to the optimal solutions of model (M1) by repeatedly solving model (M4). □

We have discussed the approximation algorithm with minimum problem. For a maximum problem, let constraint $w_{ij} \geq v_i\underline{\theta}_j$ be replaced by constraint $w_{ij} \leq v_i\underline{\theta}_j$ in model (M4). Using the above approximation algorithm, we can also obtain the optimal solutions to model (M1) with maximum form.

2.4. A Numerical Example. In order to compare the proposed method with the method in [19], a numerical example used by them is presented in this subsection. There are twelve DMUs with three inputs (X_1, X_2, and X_3) and two outputs (Y_1, Y_2) as shown in Table 1. The minimum and maximum efficiency scores, the integrated score, and the rank of DMUs obtained by the method in [19] are exhibited in the seventh, the eighth, and the ninth column of Table 1, respectively.

In order to rank all DMUs by the proposed method, we first use the proposed approximation Algorithm 4 to calculate the minimum and maximum efficiency scores of each DMU and then use (4) and (5) to compute the integrated scores. Finally, the DMUs were ranked according to their integrated scores.

Using Algorithm 4, the procedure to calculate the minimum efficiency score of DMU_1 is described as follows.

Step 1. Give permissible error $\varepsilon = 0.001$.

Step 2. Let $\delta = 0.0001$. Solve model (M3) to find the optimal solutions: $\underline{u}_1 = 0.000678$, $\underline{u}_2 = 0$, $\underline{v}_1 = 0$, $\underline{v}_2 = 0.013872$, and $\underline{v}_3 = 0.050998$. Using (1) we can obtain that the efficiency scores $\underline{\theta}_j$ of twelve DMUs are 0.0454, 0.0644, 0.0646, 0.0697,

Table 1: The input-output data and the rankings of DMUs.

DMU	Inputs and outputs					The method in [19]			Proposed improved method		
	X_1	X_2	X_3	Y_1	Y_2	$[\theta_j^{\min}, \theta_j^{\max}]$	θ_j	Rank	$[\theta_j^{\min}, \theta_j^{\max}]$	θ_j	Rank
1	350	39	9	67	751	[0.0454, 0.0989]	0.0675	11	[0.0573, 0.0862]	0.0702	11
2	298	26	8	73	611	[0.0526, 0.1199]	0.0804	7	[0.0605, 0.0993]	0.0778	7
3	422	31	7	75	584	[0.0483, 0.1076]	0.0728	9	[0.0556, 0.0893]	0.0706	10
4	281	16	9	70	665	[0.0564, 0.1468]	0.0937	4	[0.0577, 0.1371]	0.0930	4
5	301	16	6	75	445	[0.0499, 0.1679]	0.0986	2	[0.0591, 0.1435]	0.0967	3
6	360	29	17	83	1070	[0.0393, 0.1402]	0.0810	6	[0.0393, 0.1249]	0.0774	8
7	540	18	10	72	457	[0.0304, 0.1262]	0.0700	10	[0.0348, 0.1248]	0.0749	9
8	276	33	5	74	590	[0.0549, 0.1358]	0.0883	5	[0.0589, 0.1190]	0.0857	5
9	323	25	5	75	1074	[0.0800, 0.2043]	0.1313	1	[0.0975, 0.1724]	0.1308	1
10	444	64	6	74	1072	[0.0363, 0.1394]	0.0789	8	[0.0363, 0.1394]	0.0822	6
11	323	25	5	25	350	[0.0267, 0.0666]	0.0432	12	[0.0325, 0.0562]	0.0430	12
12	444	64	6	104	1199	[0.0510, 0.1560]	0.0944	3	[0.0510, 0.1560]	0.0977	2

0.0964, 0.0444, 0.0643, 0.0704, 0.0845, 0.0420, 0.0282, and 0.0591, respectively.

Step 3. Computing the sum of the efficiency scores of all DMUs, we have $\sum_{j=1}^{12} \underline{\theta}_j = 0.7335$. Since $\left|\sum_{j=1}^{12} \underline{\theta}_j - 1\right| = 0.2665 > 0.001$, we do not obtain the optimal solutions to model (M1), and we have to proceed further to find the optimal solutions. Consider $k := 1$; go to Step 4.

Steps 4–6. Solve model (M4) repeatedly to find the optimal solutions $\underline{u}_r^{(k)}$ and $\underline{v}_i^{(k)}$, and then calculate the efficiency score $\underline{\theta}_j^{(k)}$ of each DMU using (1), $k = 1, 2, \ldots$. Computing the sum of the efficiency scores of all DMUs, we have $\{\sum_{j=1}^{12} \underline{\theta}_j^{(k)}\} = \{0.9592, 0.9945, 0.9993, \ldots\}$. Since $\left|\sum_{j=1}^{12} \underline{\theta}_j^{(3)} - 1\right| = 0.0007 < 0.001$, the optimal solutions of model (M1) have been found: $u_1^* = 0.000855, u_2^* = 0, v_1^* = 0, v_2^* = 0.022075, v_3^* = 0.015451$, and the objective function valve is $\theta_1^* = 0.0573$.

Hence, the minimum efficiency value of DMU_1 is $\theta_1^{\min} = 0.0573$. Similarly, using Algorithm 4, we can also obtain the minimum efficiency values of the rest DMUs. Let constraint $w_{ij} \geq v_i \underline{\theta}_j$ be replaced by constraint $w_{ij} \leq v_i \underline{\theta}_j$ in model (M4). Using Algorithm 4 we can also obtain the maximum efficiency values of all DMUs. The minimum and maximum efficiency scores of each DMU are shown in the tenth column of Table 1.

Using (4) and (5), the integrated score θ_j is determined and the results are exhibited in the eleventh column of Table 1. The twelfth column of Table 1 shows the results of rankings according to their integrated scores.

Table 1 shows that the rankings of DMUs obtained by proposed method are not exactly the same as that obtained by the method in [19]. For instance, DMU_6 is ranked as eighth in our method and sixth in the method of [19], whereas DMU_{10} is placed sixth in our method and eighth in the method of [19].

3. Proposed Multiobjective Programming Model Based on Compensatory DEA

3.1. The Multiobjective Programming Model for Ranking All Units. Khodabakhshi and Aryavash [19] contribute to a very interesting variant of the basic DEA model by supposing that the sum of efficiency values of all DMUs is equal to unity. Using their method all DMUs can be fully ranked. However, their method does not provide more information about the weight used for calculating the efficiency scores of each DMU. In this section, we aim to develop a new model to find a CSW, which can be used for the full ranking of all DMUs. To do it, here we propose a MOP model as follows:

$$\begin{aligned}
\text{(M5)}\quad & \max \{u_1, u_2, \ldots, u_s, v_1, v_2, \ldots, v_m\} \\
\text{s.t.}\quad & \frac{\sum_{r=1}^{s} \widehat{y}_{r1} u_r}{\sum_{i=1}^{m} \widehat{x}_{i1} v_i} + \frac{\sum_{r=1}^{s} \widehat{y}_{r2} u_r}{\sum_{i=1}^{m} \widehat{x}_{i2} v_i} + \cdots + \frac{\sum_{r=1}^{s} \widehat{y}_{rn} u_r}{\sum_{i=1}^{m} \widehat{x}_{in} v_i} = 1, \\
& \sum_{r=1}^{s} u_r + \sum_{i=1}^{m} v_i = 1 \\
& u_r, v_i \geq 0, \quad \forall r, i,
\end{aligned} \tag{15}$$

where v_i $(i = 1, 2 \ldots, m)$ and u_r $(r = 1, 2, \ldots, s)$ are the common weights assigned to ith input and rth output, respectively. $\widehat{x}_{ij}$ $(i = 1, 2, \ldots, m)$ and $\widehat{y}_{rj}$ $(r = 1, 2, \ldots, s)$ represent normalized input and output values of DMU_j $(j = 1, 2, \ldots, n)$, respectively. In order to eliminate the impacts of measurement units on a CSW, we normalize all inputs and outputs values by using the following equations [23]:

$$\begin{aligned}
\widehat{x}_{ij} &= \frac{x_{ij}}{\sum_{j=1}^{n} x_{ij}}, \quad i = 1, 2, \ldots, m, \; j = 1, 2, \ldots, n, \\
\widehat{y}_{rj} &= \frac{y_{rj}}{\sum_{j=1}^{n} y_{rj}}, \quad r = 1, 2, \ldots, s, \; j = 1, 2, \ldots, n.
\end{aligned} \tag{16}$$

In model (M5), in order to avoid a choice of zero common weights and get them as big as possible, the objectives are to simultaneously maximize the common weights assigned to each input and output. The first constraint is supposing that the sum of efficiency values of all DMUs is equal to unity, which can avoid the problem that more than one DMU is evaluated as efficient in DEA. The second constraint $\sum_{r=1}^{s} u_r + \sum_{i=1}^{m} v_i = 1$ is used to avoid arbitrariness in determining common weights. Without it, there will be an infinite number of common weights that can meet the model (M5).

3.2. Fuzzy Programming Method for Solving the MOP Model (M5). Many methods can be used to solve the MOP model, such as compromise programming method, goal programming method, and fuzzy programming method. In this paper, we use fuzzy programming methods to solve the MOP model (M5).

In the MOP model (M5), it is unlikely that all objectives simultaneously achieve their optimal solutions subject to the given constraints. So in practice the DM usually chooses a satisfying solution according to the aspiration level fixed for each objective [24, 25]. Let u_{ra}, $r = 1, 2, \ldots, s$, v_{ia}, $i = 1, 2, \ldots, m$, be the aspiration level of objectives; then model (M5) can be expressed as follows [24, 26, 27]:

$$\begin{aligned} &\text{(M6)} \quad \text{Find } u, v \\ &\text{To satisfy } u_r \tilde{\geq} u_{ra}, \quad r = 1, 2, \ldots, s, \\ &\qquad v_i \tilde{\geq} v_{ia}, \quad i = 1, 2, \ldots, m, \\ &\qquad (u_r, v_i) \in \aleph_5 (u, v) . \end{aligned} \tag{17}$$

The expressions $u_r \tilde{\geq} u_{ra}$ and $v_i \tilde{\geq} v_{ia}$ are represented by a fuzzy set called fuzzy goal, which means that the DM would be satisfied even for objective value slightly less than u_{ra} or v_{ia} within the value of allowed deviations. The symbols "$\tilde{\geq}$" denote the fuzzified versions of "$\geqslant$" and they can be read as "approximately greater than or equal to." The expression $(u_r, v_i) \in \aleph_5(u, v)$ is system constraint in model (M5), and $\aleph_5(u, v)$ denotes the set of the feasible solution of model (M5).

Model (M6) is a fuzzy multiobjective programming (FMOP) model. The procedure for finding the fuzzy efficient solution can be summarized as follows.

Algorithm 6. Consider the following.

Step 1 (determine the fuzzy aspiration level of each objective). Since the DM usually knows little about the objectives, this paper uses the ideal solutions of model (M5) to determine the value of the fuzzy aspiration level of each objective. The ideal solutions are just the optimal solutions of the individual objective subject to the system constraint. That is,

$$\begin{aligned} &\text{(M7)} \quad \max u_r \quad \text{or} \quad \max v_i \\ &\qquad (u_r, v_i) \in \aleph_5 (u, v) . \end{aligned} \tag{18}$$

Model (M7) may not be also considered as a convex optimization problem like model (M1), and it usually has multiple local optimal solutions. Thus model (M7) is more difficult to be solved. In the following, a similar approximation algorithm with Algorithm 4 is used.

Let

$$\theta_j = \frac{\sum_{r=1}^{s} \hat{y}_{rj} u_r}{\sum_{i=1}^{m} \hat{x}_{ij} v_i}, \quad j = 1, 2, \ldots, n; \tag{19}$$

then model (M7) can be rewritten as follows:

$$\begin{aligned} &\text{(M8)} \quad \max u_r \quad \text{or} \quad \max v_i \\ &\text{s.t.} \quad \theta_j \sum_{i=1}^{m} \hat{x}_{ij} v_i - \sum_{r=1}^{s} \hat{y}_{rj} u_r = 0, \quad j = 1, 2, \ldots, n, \\ &\qquad \sum_{j=1}^{n} \theta_j = 1, \\ &\qquad \sum_{r=1}^{s} u_r + \sum_{i=1}^{m} v_i = 1, \\ &\qquad u_r, v_i, \theta_j \geq 0, \quad \forall i, j, r. \end{aligned} \tag{20}$$

Using the transformation $w_{ij} = v_i \theta_j$ [19], model (M8) can be rewritten by the following LP problem:

$$\begin{aligned} &\text{(M9)} \quad \max u_r \quad \text{or} \quad \max v_i \\ &\text{s.t.} \quad \sum_{i=1}^{m} \hat{x}_{ij} w_{ij} - \sum_{r=1}^{s} \hat{y}_{rj} u_r = 0, \quad j = 1, 2, \ldots, n, \\ &\qquad \sum_{j=1}^{n} w_{ij} = v_i, \quad i = 1, 2, \ldots, m, \\ &\qquad \sum_{r=1}^{s} u_r + \sum_{i=1}^{m} v_i = 1, \\ &\qquad u_r, v_i, w_{ij} \geq 0, \quad \forall i, j, r. \end{aligned} \tag{21}$$

Let $\overline{u}_r$ and $\overline{v}_i$ be the optimal solutions of model (M9). Using (19), it is easy to compute the efficiency score $\overline{\theta}_j$ of each DMU. If $\sum_{j=1}^{n} \overline{\theta}_j = 1$, then $\overline{u}_r$ and $\overline{v}_i$ are also the optimal solutions of model (M7); otherwise, that is, $\sum_{j=1}^{n} \overline{\theta}_j \neq 1$, we have to proceed further to find the optimal solutions on the basis of the optimal solution of model (M9). In the following, we only discuss the case that $\sum_{j=1}^{n} \overline{\theta}_j > 1$ for the sake of convenient description. However, the procedure can be easily extended to the case that $\sum_{j=1}^{n} \overline{\theta}_j < 1$.

In order to find further the optimal solutions of model (M7) when $\sum_{j=1}^{n} \overline{\theta}_j > 1$, let constraints $w_{ij} = v_i \theta_j$ be replaced by constraints $w_{ij} \le v_i\overline{\theta}_j$, then add it into model (M9) to formulate the following LP problem:

$$\begin{aligned}(\text{M10})\ \ &\max u_r \quad \text{or} \quad \max v_i \\ &\text{s.t.}\quad w_{ij} \le v_i\overline{\theta}_j, \quad i = 1,2,\ldots,m,\ j = 1,2,\ldots,n, \\ &\qquad (u_r, v_i) \in \aleph_9(u,v),\end{aligned} \tag{22}$$

where $\aleph_9(u,v)$ denotes the set of the feasible solutions to model (M9).

Let $\overline{u}_r^{(1)}$ and $\overline{v}_i^{(1)}$ be the optimal solutions of model (M10). Using (19), we will get the efficiency score $\overline{\theta}_j^{(1)}$ of each DMU, if $\sum_{j=1}^{n}\overline{\theta}_j^{(1)} = 1$. Then $\overline{u}_r^{(1)}$ and $\overline{v}_i^{(1)}$ are the optimal solutions of model (M7); otherwise, we also have to proceed further to find the optimal solutions on the basis of it.

Let $\overline{\theta}_j^{(1)}$ replace $\overline{\theta}_j$ in model (M10), and then solve it to find the optimal solutions $\overline{u}_r^{(2)}$ and $\overline{v}_i^{(2)}$. Using (19) we will get the efficiency score $\overline{\theta}_j^{(2)}$ of each DMU and then judge whether $\sum_{j=1}^{n}\overline{\theta}_j^{(2)}$ is equal to 1 or not. The process is repeated until that the sum of efficiency scores of all DMUs is approximately equal to 1 within given error. According to Algorithm 4, the optimal solutions of model (M10) will be close to the optimal solutions of model (M7) by repeatedly solving model (M10).

We have discussed the approximation algorithm in the case that $\sum_{j=1}^{n}\overline{\theta}_j > 1$. If $\sum_{j=1}^{n}\overline{\theta}_j < 1$, then let constraint $w_{ij} \le v_i\overline{\theta}_j$ be replaced by constraint $w_{ij} \ge v_i\overline{\theta}_j$ in model (M10). According to the above analysis and Algorithm 4, we can also obtain the optimal solutions of model (M7).

Step 2 (construct membership functions). In the model (M6), each fuzzy goal is represented by fuzzy sets and defined by membership functions $\mu_r(u_r)$ and $\mu_i(v_i)$ $(r = 1,2,\ldots,s, i = 1,2,\ldots,m)$. Since linear membership functions are used more than other types of membership functions in the literature [28], we choose the linear membership functions. For each objective, the lower bound is zero, and the upper bound is the fuzzy aspiration level determined by solving model (M7). Hence, the membership functions for each objective may be defined as follows:

$$\begin{aligned}\mu_r(u_r) &= \begin{cases} 0 & u_r \le 0 \\ \dfrac{u_r}{u_{ra}} & 0 \le u_r \le u_{ra} \\ 1 & u_r \ge u_{ra}, \end{cases} \\ r &= 1,2,\ldots,s \\ \mu_i(v_i) &= \begin{cases} 0 & v_i \le 0 \\ \dfrac{v_i}{v_{ia}} & 0 \le v_i \le v_{ia} \\ 1 & v_i \ge v_{ia}, \end{cases} \\ i &= 1,2,\ldots,m.\end{aligned} \tag{23}$$

Step 3 (build an auxiliary crisp model). Introducing the auxiliary variable λ, the FMOP model (M6) is transformed into the following auxiliary crisp model [27, 29, 30]:

$$\begin{aligned}(\text{M11})\ \ &\text{Max}\,\lambda + \delta\left(\sum_{r=1}^{s}\frac{u_r}{u_{ra}} + \sum_{i=1}^{m}\frac{v_i}{v_{ia}}\right) \\ &\text{s.t.}\quad u_r \ge u_{ra}\lambda, \quad r = 1,2,\ldots,s, \\ &\qquad v_i \ge v_{ia}\lambda, \quad i = 1,2,\ldots,m, \\ &\qquad (u_r, v_i) \in \aleph_5(u,v),\end{aligned} \tag{24}$$

where δ is a sufficiently small positive number. The optimal solution to model (M11) is a fuzzy efficient solution to the FMOP model (M6) and the Pareto optimal solution of MOP model (M5) [24, 29, 30].

Step 4 (solve model (M11) to find the optimal CSW). The method for solving model (M11) is similar to that for solving model (M7). Using the approximation algorithm, we solve model (M11) to find the optimal solutions u_r^* $(r = 1,2,\ldots,s)$ and v_i^* $(i = 1,2,\ldots,m)$, which is also the optimal CSW. Then, using (19) we compute the optimal efficiency scores θ_j^* $(j = 1,2,\ldots,n)$ of each DMU, which can be used to fully rank all DMUs. A DMU has a better rank if it has a greater efficiency score.

4. Numerical Examples

In this section, we provide two numerical examples to illustrate our method and then compare the results with the results of the existing models to show the potential usage of the proposed method in the full ranking of DMUs.

Example 1. Consider the numerical example used in Section 2.4. There are twelve DMUs with three inputs (X_1, X_2, and X_3) and two outputs (Y_1, Y_2) as shown in Table 1. In order to rank all DMUs by using the proposed method, we first normalize all inputs and outputs values by using (16); the normalized values are shown in Table 2. The integrated score and the rank of DMUs obtained by the method in [19] are exhibited again in the seventh and the eighth column of Table 2, respectively.

Using the normalized values we can build a MOP model (M5), which is transformed into FMOP model (M6). We solve model (M6) to find the fuzzy efficient solution, which is also the Pareto optimal solution of MOP model (M5), using Algorithm 6.

The procedure to solve the model (M6) can be described as follows.

Step 1 (determine the fuzzy aspiration level of each objective by solving model (M7)). The algorithm for solving model (M7) is similar to the approximation algorithm, Algorithm 4.

Give permissible error $\varepsilon = 0.001$. For the first objective, $\max u_1$, solve model (M9) to find optimal solutions: $\overline{u}_1 = 0.09333$, $\overline{u}_2 = 0$, $\overline{v}_1 = 0.40844$, $\overline{v}_2 = 0.28013$, and

TABLE 2: The normalized input-output data and the rankings of DMUs.

DMU	The normalized values of inputs and outputs					The method in [19]		Proposed method	
	X_1	X_2	X_3	Y_1	Y_2	θ_j	Rank	θ_j	Rank
1	0.0802	0.1010	0.0968	0.0773	0.0847	0.0675	11	0.0703	10
2	0.0683	0.0674	0.0860	0.0842	0.0689	0.0804	7	0.0874	6
3	0.0967	0.0803	0.0753	0.0865	0.0659	0.0728	9	0.0774	7
4	0.0644	0.0415	0.0968	0.0807	0.0750	0.0937	4	0.0954	4
5	0.0690	0.0415	0.0645	0.0865	0.0502	0.0986	2	0.1042	2
6	0.0825	0.0751	0.1828	0.0957	0.1207	0.0810	6	0.0750	8
7	0.1238	0.0466	0.1075	0.0830	0.0515	0.0700	10	0.0640	11
8	0.0633	0.0855	0.0538	0.0854	0.0665	0.0883	5	0.0956	3
9	0.0740	0.0648	0.0538	0.0865	0.1211	0.1313	1	0.1250	1
10	0.1018	0.1658	0.0645	0.0854	0.1209	0.0789	8	0.0718	9
11	0.0740	0.0648	0.0538	0.0288	0.0395	0.0432	12	0.0413	12
12	0.1018	0.1658	0.0645	0.1200	0.1352	0.0944	3	0.0923	5

$\overline{v}_3 = 0.21810$. Calculate the efficiencies score $\overline{\theta}_j$ of each DMU using (19) and their sum; we have $\sum_{j=1}^{12} \overline{\theta}_j = 1.2678$.

Since $\left|\sum_{j=1}^{12} \overline{\theta}_j - 1\right| = 0.2678 > \varepsilon$, we do not obtain the optimal solutions to model (M7). Thus we have to proceed calculating further to find the optimal solutions. We solve model (M10) repeatedly, to find optimal solutions $\overline{u}_r^{(k)}$ and $\overline{v}_i^{(k)}$, $k = 1, 2, \ldots$, until $\left|\sum_{j=1}^{n} \overline{\theta}_j^{(k)} - 1\right| \le \varepsilon$. After repeating 4 times for solving model (M10), we have $\left|\sum_{j=1}^{n} \overline{\theta}_j^{(4)} - 1\right| = 0.00079 < \varepsilon$, which means that the optimal solutions have been found: $u_1^* = 0.07516$, $u_2^* = 0$, $v_1^* = 0.32684$, $v_2^* = 0.27280$, and $v_3^* = 0.32521$. And the objective function valve is $u_1^* = 0.07516$. Therefore the fuzzy aspiration level of the first objective is $u_{1a} = 0.07516$.

Similarly, we can also solve model (M7) to obtain the fuzzy aspiration levels of the rest of the objectives: $u_{2a} = 0.07673$, $v_{1a} = 0.92510$, $v_{2a} = 0.93421$, and $v_{3a} = 0.93073$.

Step 2 (construct membership functions). The mathematical expression of membership functions is given in (23). For each objective, the lower bound is zero, and the upper bound is the fuzzy aspiration level determined by solving model (M7). According to Step 1, we have $u_{1a} = 0.07516$, $u_{2a} = 0.07673$, $v_{1a} = 0.92510$, $v_{2a} = 0.93421$, and $v_{3a} = 0.93073$.

Step 3 (build an auxiliary crisp model (M11)). Having determining the fuzzy aspiration level of each objective and the form of membership functions, we can get the auxiliary crisp model (M11) by introducing the auxiliary variable λ.

Step 4 (solve model (M11) to obtain the optimal CSW). The method for solving model (M11) is similar to that of model (M7). Set $\delta = 0.01$; we solve model (M11) to obtain the optimal solutions $\lambda^* = 0.33133$, $u_1^* = 0.05014$, $u_2^* = 0.02542$, $v_1^* = 0.30652$, $v_2^* = 0.30954$, and $v_3^* = 0.30838$. They are the fuzzy efficient solutions to model (M6) and the Pareto optimal solutions to model (M5). Therefore we have obtained the optimal CSW: $u_1^* = 0.05014$, $u_2^* = 0.02542$, $v_1^* = 0.30652$, $v_2^* = 0.30954$, and $v_3^* = 0.30838$.

Using (19) we can compute the optimal efficiency score θ_j^* of each DMU, and the results are shown in the ninth column of Table 2. The tenth column of Table 2 shows the results of ranking by our method. It shows that all DMUs are fully ranked with their optimal efficiency scores θ_j^*.

Table 2 shows that the rankings of DMUs using the proposed method are not entirely consistent with that of the method in [19]. The main advantage of our method compared to the method in [19] is that a CSW for fully ranking DMUs is derived and all DMUs may be able to be compared and ranked on the same basis. Moreover, we improve their solution method and propose an approximation algorithm to solve their DEA model.

Example 2. Measure the performance of nations participating in the Olympic Games. Consider the example studied by Zhang et al. [20] and Azizi and Wang [21], in which 73 countries or areas are evaluated in terms of two inputs and three outputs defined as follows:

X_1: gross domestic product (GDP),

X_2: total population of the country or area,

Y_1: number of gold medals won by the country,

Y_2: number of silver medals won by the country,

Y_3: number of bronze medals won by the country.

The input and output data of the Athens 2004 Summer Olympic Games [20, 21] are presented in Table 3. We use this example to compare the proposed method with Azizi and Wang's method [21] and Ramezani-Tarkhorani et al.'s method [22].

In Azizi and Wang's method [21], a pair of bounded DEA models was proposed to measure the interval efficiencies of DMUs. The lower bound of the interval efficiency is called the worst relative efficiency or pessimistic efficiency, and its value is determined using improved pessimistic DEA model [21]. The upper bound of the interval efficiency is called the best relative efficiency or the optimistic efficiency, and its value is determined using the conventional DEA model. Using the

Table 3: Data of the Athens 2004 Summer Olympic Games [20, 21].

Country or area (DMU)	Inputs		Outputs		
	GDP (billion $)	Population (thousands)	Gold	Silver	Bronze
Argentina	151.94	38372	2	0	4
Australia	617.61	19942	17	16	16
Austria	289.72	8171	2	4	1
Azerbaijan	8.54	8355	1	0	4
Bahamas	5.5	319	1	0	1
Belarus	22.75	9811	2	6	7
Belgium	352	10400	1	0	2
Brazil	599.73	183913	5	2	3
Britain	2125.51	59479	9	9	12
Bulgaria	23.91	7780	2	1	9
Cameroon	14.43	16038	1	0	0
Canada	995.83	31958	3	6	3
Chile	93.65	16124	2	0	1
China	1649.39	1307989	32	17	14
China, Hong Kong	164.55	6963	0	1	0
Chinese Taipei	305.2	22689	2	2	1
Colombia	95.19	44915	0	0	2
Croatia	33.2	4540	1	2	2
Cuba	44.54	11245	9	7	11
Czech Republic	107.05	10229	1	3	4
Denmark	242.34	5414	2	0	6
Dominican Republic	19.44	8768	1	0	0
Egypt	77.03	72642	1	1	3
Eritrea	0.62	4232	0	0	1
Estonia	11.2	1335	0	1	2
Ethiopia	8.21	75600	2	3	2
Finland	186.18	5235	0	2	0
France	2018.08	60257	11	9	13
Georgia	4.45	4518	2	2	0
Germany	2706.67	82645	13	16	20
Greece	205.49	11098	6	6	4
Hungary	99.35	10124	8	6	3
India	661.05	1087124	0	1	0
Indonesia	257.87	220077	1	1	2
Iran	168.97	68803	2	2	2
Israel	116.34	6601	1	0	1
Italy	1680.69	58033	10	11	11
Jamaica	8.71	2639	2	1	2
Japan	4668.42	127924	16	9	12
Kazakhstan	40.75	14839	1	4	3
Kenya	15.62	33467	1	4	2
Korea, Republic	681.47	47645	9	12	9
Latvia	13.66	2318	0	4	0
Lithuania	22.17	3443	1	2	0
Mexico	676.5	105699	0	3	1
Mongolia	1.29	2614	0	0	1
Morocco	49.82	31020	2	1	0
Netherlands	577.98	16226	4	9	9

Table 3: Continued.

Country or area (DMU)	Inputs		Outputs		
	GDP (billion $)	Population (thousands)	Gold	Silver	Bronze
New Zealand	96.97	3989	3	2	0
Nigeria	71.33	128709	0	0	2
Norway	250.44	4598	5	0	1
Paraguay	7	6017	0	1	0
Poland	241.77	38559	3	2	5
Portugal	167.24	10441	0	2	1
Romania	71.32	21790	8	5	6
Russia	582.73	143899	27	27	38
Serbia and Montenegro	24.13	10510	0	2	0
Slovakia	41.09	5401	2	2	2
Slovenia	32.79	1967	0	1	3
South Africa	212.9	47208	1	3	2
Spain	992.99	42646	3	11	5
Sweden	346.53	9008	4	2	1
Switzerland	358	7240	1	1	3
Syria	23.74	18582	0	0	1
Thailand	163.49	63694	3	1	4
Trinidad and Tobago	12.54	1301	0	0	1
Turkey	300.09	72220	3	3	4
Ukraine	65.04	46989	9	5	9
United Arab Emirates	95.72	4284	1	0	0
United States	11733.47	295410	36	39	27
Uzbekistan	9.72	26209	2	1	2
Venezuela	107.49	26282	0	0	2
Zimbabwe	5.82	12936	1	1	1

traditional DEA model, the DEA efficiency or the optimistic efficiency of each participating country at the Athens 2004 Olympic Games is determined, and the results are shown in the second column of Table 4. The efficiency intervals of the 73 DMUs according to Azizi and Wang's method are presented in the third column of Table 4. The fourth column of Table 4 shows the rankings based on the efficiency intervals [21].

Liu and Peng [13] proposed a common weights analysis methodology to generate a CSW for the performance indices of only DEA efficient DMUs. All DMUs are then ranked according to the efficiency scores weighted by the CSW. Ramezani-Tarkhorani et al. [22] pointed out the problem and proposed a new approach to rank all DMUs with common weights. From the second column of Table 4, it can be seen that, among 73 DMUs, 18 DMUs are DEA efficient. Using the data of these efficient DMUs, the CSW is determined as $u_1 = 0.01$, $u_2 = 5.829$, $u_3 = 3.235$, $v_1 = 0.01$, and $v_2 = 0.01$. The efficiency scores of DMUs can then be calculated using the CSW and the results are shown in the fifth column of Table 4, and corresponding rankings are shown in the sixth column of Table 4.

In order to rank all DMUs by using the proposed method, we first normalize all inputs and outputs values by using (16) and then build a MOP model (M5), which is transformed into FMOP model (M6). We solve model (M6) to find the fuzzy efficient solution, which is also the Pareto optimal solution of MOP model (M5), using Algorithm 6. The procedure to solve model (M6) can be described as follows.

Step 1 (determine the fuzzy aspiration level of each objective by solving model (M7)). The algorithm for solving model (M7) is similar to the approximation algorithm, Algorithm 4. Give permissible error $\varepsilon = 0.01$, for each objective, $\max u_r$ or $\max v_i$, solve model (M7), respectively, to obtain the fuzzy aspiration levels of all objectives: $u_{1a} = 0.00474$, $u_{2a} = 0.00442$, $u_{3a} = 0.00349$, $v_{1a} = 0.99867$, and $v_{2a} = 0.99688$.

Step 2 (construct membership functions). The mathematical expression of membership functions are given in (23). For each objective, the lower bound is zero, and the upper bound is the fuzzy aspiration level determined by solving model (M7) in Step 1.

Step 3 (build an auxiliary crisp model (M11)). Having determining the fuzzy aspiration level of each objective and the form of membership functions, we can get the auxiliary crisp model (M11) by introducing the auxiliary variable λ.

Step 4 (solve model (M11) to obtain the optimal CSW). The method for solving model (M11) is similar to that of model (M7). Set $\delta = 0.01$, we solve model (M11) to obtain the

Table 4: Efficiencies of the countries participating in the Athens 2004 Olympic Games.

Country or area (DMU)	DEA efficiencies	The method in [21]		The method in [22]		Proposed method	
		Efficiency intervals	Rank	Efficiencies	Rank	Efficiencies	Rank
Argentina	0.08978	[0.00004, 0.08978]	52	0.03364	55	0.00429	44
Australia	1.00000	[0.00023, 1.00000]	26	0.70622	5	0.02271	15
Austria	0.42598	[0.00010, 0.42598]	34	0.31406	17	0.00746	33
Azerbaijan	1.00000	[0.00030, 1.00000]	23	0.15484	32	0.02075	16
Bahamas	1.00000	[0.00039, 1.00000]	22	1.00000	1	0.08595	1
Belarus	1.00000	[0.00212, 1.00000]	2	0.58614	7	0.05144	4
Belgium	0.11826	[0.00001, 0.11826]	67	0.06027	50	0.00222	63
Brazil	0.03770	[0.00004, 0.03770]	54	0.01161	66	0.00180	65
Britain	0.22016	[0.00004, 0.22016]	51	0.14832	34	0.00405	51
Bulgaria	1.00000	[0.00068, 1.00000]	12	0.44803	10	0.04399	6
Cameroon	0.15419	[0.00001, 0.15419]	65	0.00006	73	0.00295	59
Canada	0.17040	[0.00005, 0.17040]	47	0.13567	36	0.00351	55
Chile	0.10916	[0.00003, 0.10916]	58	0.02007	61	0.00501	41
China	1.00000	[0.00010, 1.00000]	39	0.01105	67	0.00200	64
China, Hong Kong	0.08323	[0.00004, 0.08323]	53	0.08178	43	0.00177	67
Chinese Taipei	0.10304	[0.00005, 0.10304]	48	0.06486	49	0.00388	53
Columbia	0.05115	[0.00001, 0.05115]	71	0.01437	65	0.00125	69
Croatia	0.47307	[0.00050, 0.47307]	17	0.39662	13	0.02538	12
Cuba	1.00000	[0.00155, 1.00000]	4	0.67743	6	0.07166	2
Czech Republic	0.38991	[0.00023, 0.38991]	27	0.29448	18	0.01436	22
Denmark	0.93665	[0.00003, 0.93665]	56	0.34351	14	0.00872	29
Dominican Republic	0.17635	[0.00002, 0.17635]	62	0.00011	72	0.00468	42
Egypt	0.09781	[0.00014, 0.09781]	31	0.02138	60	0.00255	61
Eritrea	1.00000	[0.00004, 1.00000]	49	0.07643	44	0.00832	30
Estonia	1.00000	[0.00069, 1.00000]	11	0.91362	3	0.04368	7
Ethiopia	1.00000	[0.00045, 1.00000]	19	0.03171	56	0.00427	45
Finland	0.22139	[0.00007, 0.22139]	43	0.21505	24	0.00340	56
France	0.24442	[0.00004, 0.24442]	50	0.15195	33	0.00465	43
Georgia	1.00000	[0.00340, 1.00000]	1	0.25823	20	0.04032	8
Germany	0.51400	[0.00005, 0.51400]	44	0.18523	28	0.00508	40
Greece	0.68020	[0.00025, 0.68020]	25	0.42443	12	0.02014	17
Hungary	0.95138	[0.00052, 0.95138]	16	0.43782	11	0.03607	9
India	0.00337	[0.00001, 0.00337]	73	0.00054	70	0.00004	73
Indonesia	0.02180	[0.00004, 0.02180]	55	0.00559	68	0.00069	71
Iran	0.05368	[0.00010, 0.05368]	35	0.02631	58	0.00306	58
Israel	0.04833	[0.00002, 0.04833]	63	0.04831	52	0.00409	49
Italy	0.23392	[0.00006, 0.23392]	46	0.16714	30	0.00534	38
Jamaica	1.00000	[0.00127, 1.00000]	6	0.46528	9	0.06032	3
Japan	0.14589	[0.00002, 0.14589]	66	0.06896	47	0.00235	62
Kazakhstan	0.35502	[0.00075, 0.35502]	10	0.22199	23	0.01790	20
Kenya	0.85474	[0.00134, 0.85474]	5	0.08899	42	0.00914	28
Korea, Republic	0.47165	[0.00015, 0.47165]	30	0.20518	25	0.01028	26
Latvia	1.00000	[0.00194, 1.00000]	3	1.00000	2	0.04885	5
Lithuania	0.49341	[0.00064, 0.49341]	14	0.33673	15	0.02417	14
Mexico	0.02075	[0.00003, 0.02075]	59	0.01948	62	0.00096	70
Mongolia	1.00000	[0.00007, 1.00000]	42	0.12370	38	0.01290	23
Morocco	0.12119	[0.00017, 0.12119]	29	0.01883	63	0.00415	47
Netherlands	0.67746	[0.00013, 0.67746]	32	0.48570	8	0.01079	25

Table 4: Continued.

Country or area (DMU)	DEA efficiencies	The method in [21]		The method in [22]		Proposed method	
		Efficiency intervals	Rank	Efficiencies	Rank	Efficiencies	Rank
New Zealand	0.71636	[0.00017, 0.71636]	28	0.28606	19	0.01560	21
Nigeria	0.04570	[0.00001, 0.04570]	72	0.00502	69	0.00052	72
Norway	1.00000	[0.00002, 1.00000]	60	0.06775	48	0.00809	32
Paraguay	0.33965	[0.00094, 0.33965]	7	0.09677	41	0.00721	35
Poland	0.11449	[0.00009, 0.11449]	36	0.07181	45	0.00626	37
Portugal	0.14606	[0.00008, 0.14606]	41	0.14039	35	0.00423	46
Romania	0.54421	[0.00068, 0.54421]	13	0.22247	22	0.02859	10
Russia	1.00000	[0.00043, 1.00000]	20	0.19420	27	0.01886	19
Serbia and Montenegro	0.24230	[0.00055, 0.24230]	15	0.11067	39	0.00732	34
Slovakia	0.44929	[0.00043, 0.44929]	21	0.33348	16	0.02585	11
Slovenia	1.00000	[0.00025, 1.00000]	25	0.77679	4	0.02535	13
South Africa	0.06503	[0.00011, 0.06503]	33	0.05054	51	0.00367	54
Spain	0.32958	[0.00008, 0.32958]	40	0.18407	29	0.00518	39
Sweden	0.42647	[0.00005, 0.42647]	45	0.15964	31	0.00643	36
Switzerland	0.29930	[0.00003, 0.29930]	57	0.20458	26	0.00392	52
Syria	0.08710	[0.00001, 0.08710]	68	0.01739	64	0.00166	68
Thailand	0.09048	[0.00009, 0.09048]	37	0.02944	57	0.00415	48
Trinidad and Tobago	0.37071	[0.00002, 0.37071]	61	0.24628	21	0.01227	24
Turkey	0.05986	[0.00009, 0.05986]	38	0.04200	54	0.00407	50
Ukraine	0.68481	[0.00082, 0.68481]	9	0.12401	37	0.01891	18
United Arabic Emirates	0.07446	[0.00001, 0.07446]	69	0.00023	71	0.00316	57
United States	1.00000	[0.00003, 1.00000]	64	0.10257	40	0.00265	60
Uzbekistan	0.82038	[0.00048, 0.82038]	18	0.04699	53	0.00827	31
Venezuela	0.05859	[0.00001, 0.05859]	70	0.02452	59	0.00179	66
Zimbabwe	0.59653	[0.00090, 0.59653]	8	0.07011	46	0.01012	27

optimal solutions: $\lambda^* = 0.33132$, $u_1^* = 0.00157$, $u_2^* = 0.00146$, $u_3^* = 0.00116$, $v_1^* = 0.49818$, and $v_2^* = 0.49763$. They are the fuzzy efficient solutions to model (M6) and the Pareto optimal solutions to model (M5). Therefore we have obtained the optimal CSW: $u_1^* = 0.00157$, $u_2^* = 0.00146$, $u_3^* = 0.00116$, $v_1^* = 0.49818$, and $v_2^* = 0.49763$. The efficiency scores of DMUs can then be calculated using the CSW and the results are shown in the seventh column of Table 4, and corresponding rankings are shown in the eighth column of Table 4.

Table 4 shows that the rankings of DMUs using three methods are not entirely consistent with each other. However, Spearman's rank correlation coefficients for the proposed model and Azizi and Wang's model [21] and Ramezani-Tarkhorani et al.'s model [22] are calculated as $r_s = 0.7929$ and 0.8344, respectively. For the $\alpha = 0.05$ significant level, all correlation coefficients are statistically significant. This result shows that the rank that is determined by the proposed model has the same direction as that of Azizi and Wang's model [21] and Ramezani-Tarkhorani et al.'s model [22].

The main features of the proposed method compared to Azizi and Wang's method [21] and Ramezani-Tarkhorani et al.'s method [22] are summarized as follows.

(1) In Azizi and Wang's method [21], a pair of efficiencies, the pessimistic efficiency and the optimistic efficiency, is used to rank all DMUs; in the proposed method, the neutral efficiency, not the pessimistic efficiency or the optimistic efficiency, is used to rank all DMUs.

(2) In Azizi and Wang's method [21], for each DMU, the optimal weights for calculating the pessimistic efficiency and the optimistic efficiency are obtained by solving a LP problem, respectively, and the efficiencies of DMUs may be unable to be ranked on the same basis; in the proposed method, a MOP model is proposed to derive a CSW, and then all DMUs can be ranked on the same basis.

(3) In Ramezani-Tarkhorani et al.' model [22], only the date set of DEA efficient DMUs is used to determine a CSW, and all DMUs are then ranked according to the efficiency scores weighted by the CSW; in the proposed method, the date set of all DMUs is used to determine a CSW, and then all DMUs can be ranked with complete information of all DMUs.

(4) In Ramezani-Tarkhorani et al.' model [22], more than one DMU is usually evaluated as efficient DMUs, and thus new rules may be needed to rank the DMUs; in the proposed method, the efficiencies of all DMU have compensatory features, and thus all DMUs can be fully discriminated.

5. Conclusions

In this paper, we have developed a MOP model based on compensatory DEA model to derive a CSW for fully ranking DMUs. There are four features of the proposed ranking method as follows.

(1) In this paper, an approximation algorithm based on LP model is proposed to solve nonlinear programming models (M1), (M7), and (M11), which are not the convex optimization problem. It is desirable for the reason that optimal solution for the LP problem is global in contrast to the nonlinear one.

(2) We suppose that the sum of efficiency values of all DMUs is equal to unity in the MOP model. It has compensatory feature and can avoid the problem that more than one DMU is evaluated as efficient. Hence, using the proposed approach one can get a full ranking of all DMUs.

(3) In order to avoid a choice of zero common weights, the objectives of the proposed model are to simultaneously maximize all common weights assigned to each input and output. So we can get a set of nonzero common weights as big as possible.

(4) In this paper a CSW is used for calculating the efficiency score of each DMU. Hence, all DMUs may be able to be compared and ranked on the same basis using the efficiency scores.

In future research, we will develop a new MOP model for fully ranking DMUs by considering new objectives or some weights constraints.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (71202140), the Fundamental Research for the Central Universities (HUST: 2013QN099), the Post-Doctorial Foundation of China (no. 2012T50651 and no. 2011M501210), and the Science & Technology Project of Zhejiang Province (no. 2012C25078).

References

[1] A. Charnes, W. W. Cooper, and E. Rhodes, "Measuring the efficiency of decision making units," *European Journal of Operational Research*, vol. 2, no. 6, pp. 429–444, 1978.

[2] G. R. Jahanshahloo, F. H. Lotfi, M. Khanmohammadi, M. Kazemimanesh, and V. Rezaie, "Ranking of units by positive ideal DMU with common weights," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7483–7488, 2010.

[3] Y.-M. Wang, Y. Luo, and Y.-X. Lan, "Common weights for fully ranking decision making units by regression analysis," *Expert Systems with Applications*, vol. 38, no. 8, pp. 9122–9128, 2011.

[4] C. Kao, "Malmquist productivity index based on common-weights DEA: the case of Taiwan forests after reorganization," *Omega*, vol. 38, no. 6, pp. 484–491, 2010.

[5] J. Sun, J. Wu, and D. Guo, "Performance ranking of units considering ideal and anti-ideal DMU with common weights," *Applied Mathematical Modelling*, vol. 37, no. 9, pp. 6301–6310, 2013.

[6] W. Cook, Y. Roll, and A. Kazakov, "A DEA model for measuring the relative efficiencies of highway maintenance patrols," *Information Systems and Operational Research*, vol. 28, no. 2, pp. 113–124, 1990.

[7] Y. Roll, W. D. Cook, and B. Golany, "Controlling factor weights in data envelopment analysis," *IIE Transactions*, vol. 23, no. 1, pp. 2–9, 1991.

[8] Y. Roll and B. Golany, "Alternate methods of treating factor weights in DEA," *Omega*, vol. 21, no. 1, pp. 99–109, 1993.

[9] Z. Sinuany-Stern and L. Friedman, "DEA and the discriminant analysis of ratios for ranking units," *European Journal of Operational Research*, vol. 111, no. 3, pp. 470–478, 1998.

[10] G. R. Jahanshahloo, A. Memariani, F. H. Lotfi, and H. Z. Rezai, "A note on some of DEA models and finding efficiency and complete ranking using common set of weights," *Applied Mathematics and Computation*, vol. 166, no. 2, pp. 265–281, 2005.

[11] C. Kao and H.-T. Hung, "Data envelopment analysis with common weights: the compromise solution approach," *Journal of the Operational Research Society*, vol. 56, no. 10, pp. 1196–1203, 2005.

[12] G. R. Amin and M. Toloo, "Finding the most efficient DMUs in DEA: an improved integrated model," *Computers and Industrial Engineering*, vol. 52, no. 1, pp. 71–77, 2007.

[13] F.-H. F. Liu and H. H. Peng, "Ranking of units on the DEA frontier with common weights," *Computers & Operations Research*, vol. 35, no. 5, pp. 1624–1637, 2008.

[14] A. Davoodi and H. Z. Rezai, "Common set of weights in data envelopment analysis: a linear programming problem," *Central European Journal of Operations Research*, vol. 20, no. 2, pp. 355–365, 2012.

[15] N. Ramón, J. L. Ruiz, and I. Sirvent, "Common sets of weights as summaries of DEA profiles of weights: with an application to the ranking of professional tennis players," *Expert Systems with Applications*, vol. 39, no. 5, pp. 4882–4889, 2012.

[16] F. H. Lotfi, A. Hatami-Marbini, P. J. Agrell, N. Aghayi, and K. Gholami, "Allocating fixed resources and setting targets using a common-weights DEA approach," *Computers & Industrial Engineering*, vol. 64, no. 2, pp. 631–640, 2013.

[17] N. Adler, L. Friedman, and Z. Sinuany-Stern, "Review of ranking methods in the data envelopment analysis context," *European Journal of Operational Research*, vol. 140, no. 2, pp. 249–265, 2002.

[18] F. Hosseinzadeh Lotfi, G. R. Jahanshahloo, M. Rostamy-Malkhlifeh, Z. Moghaddas, and M. Vaez-Ghasemi, "A review of ranking models in data envelopment analysis," *Journal of Applied Mathematics*, vol. 2013, Article ID 492421, 20 pages, 2013.

[19] M. Khodabakhshi and K. Aryavash, "Ranking all units in data envelopment analysis," *Applied Mathematics Letters*, vol. 25, no. 12, pp. 2066–2070, 2012.

[20] D. Zhang, X. Li, W. Meng, and W. Liu, "Measuring the performance of nations at the olympic games using DEA models with different preferences," *Journal of the Operational Research Society*, vol. 60, no. 7, pp. 983–990, 2009.

[21] H. Azizi and Y.-M. Wang, "Improved DEA models for measuring interval efficiencies of decision-making units," *Measurement*, vol. 46, no. 3, pp. 1325–1332, 2013.

[22] S. Ramezani-Tarkhorani, M. Khodabakhshi, S. Mehrabian, and F. Nuri-Bahmani, "Ranking decision-making units using common weights in DEA," *Applied Mathematical Modelling*, vol. 38, no. 15-16, pp. 3890–3896, 2014.

[23] Y. M. Wang, Y. Luo, and L. Liang, "Ranking decision making units by imposing a minimum weight restriction in the data envelopment analysis," *Journal of Computational and Applied Mathematics*, vol. 223, no. 1, pp. 469–484, 2009.

[24] M. Jiménez and A. Bilbao, "Pareto-optimal solutions in fuzzy multi-objective linear programming," *Fuzzy Sets and Systems*, vol. 160, no. 18, pp. 2714–2721, 2009.

[25] P. Vasant, A. Bhattacharya, B. Sarkar, and S. K. Mukherjee, "Detection of level of satisfaction and fuzziness patterns for MCDM model with modified flexible S-curve MF," *Applied Soft Computing Journal*, vol. 7, no. 3, pp. 1044–1054, 2007.

[26] A. Baykasoglu and T. Gocken, "Multi-objective aggregate production planning with fuzzy parameters," *Advances in Engineering Software*, vol. 41, no. 9, pp. 1124–1131, 2010.

[27] H. Zimmermann, "Fuzzy programming and linear programming with several objective functions," *Fuzzy Sets and Systems*, vol. 1, no. 1, pp. 45–55, 1978.

[28] M. Jiménez, "Ranking fuzzy numbers through the comparison of its expected intervals," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 4, no. 4, pp. 379–388, 1996.

[29] Y.-J. Lai and C.-L. Hwang, "Possibilistic linear programming for managing interest rate risk," *Fuzzy Sets and Systems*, vol. 54, no. 2, pp. 135–146, 1993.

[30] S. M. Guu and Y. K. Wu, "Two-phase approach for solving the fuzzy linear programming problems," *Fuzzy Sets and Systems*, vol. 107, no. 2, pp. 191–195, 1999.