Data envelopment analysis (DEA) in presence of both imprecise data and general weight restrictions; model and a heuristic solution approach

Bohlool Ebrahimi[1],
Ph.D Candidate, Technology Development Institute (ACECR), Sharif university branch,
Tehran, Iran,
b_ebrahimi@jdsharif.ac.ir

Morteza Rahmani[2]
Associate professor,
Technology Development Institute (ACECR), Sharif university branch,
Tehran, Iran,
rahmanimr@jdsharif.ac.ir

Mehdi Bashiri[3]
, Associate professor, Department of Industrial Engineering,
Shahed University,
Tehran, Iran,
bashiri@shahed.ac.ir

***Abstract*—In real-life application of DEA models, data maybe imprecise such as ordinal or bounded data. Also in this model considering weight restrictions are necessary to prevent overestimate or underestimate some inputs or outputs in performance evaluation process. Existing works in this area considered only assurance region type 1 with imprecise data. This paper considers DEA model in presence of both imprecise data and general weight restrictions includes assurance region type 1, type 2 and absolute weight restrictions. The model is nonlinear and non-convex, so in most cases the professional solvers are unable to find the global optimum solution. We developed a genetic algorithm (GA) for solving the model. To obtain the best configuration of GA parameters, we applied the Taguchi experimental design methodology. Two numerical examples are given to demonstrate the effectiveness of developed GA. Also, the sensitivity analysis of the model in presence of ordinal data is discussed.**

Keywords- Data envelopment analysis (DEA); Imprecise data; Weight restrictions; Genetic algorithm

I. Introduction

Data envelopment analysis (DEA), introduced by Charnes, Cooper, and Rhodes [1], is a mathematical programming to calculate the relative efficiency of a set of decision making units (DMUs). In the standard DEA models, DMUs are free in choosing the weights in order to maximize their relative efficiency. Consequently, some inputs and outputs can be overestimated or ignored in performance evaluation process. Also the viewpoint of decision maker about the importance of criteria is not considered. It should be noted that the complete flexibility in the selection of weights is significant in the identification of inefficient DMUs. In addition, the traditional DEA models make an assumption that input and output data are precise. But in some real-life applications the data maybe imprecise such as bounded, ordinal and so on.

In some applications of DEA, data are imprecise and we need to consider weight restrictions. Cooper et al. [2] were the first to consider simultaneously imprecise data and weight restrictions in DEA.

To the best of our knowledge, there is not any reference to consider DEA model in presence of both imprecise data and generalized form of weight restrictions includes assurance region type 1, type 2 and absolute weight restrictions. In this paper, we consider DEA model in presence of both imprecise data and general weight restrictions. The model is nonlinear and non-convex. Hence, the optimum solution may not be a global optimum solution of the model. So, to solve the problem we developed a genetic algorithm (GA).

The reminder of the paper is organized as follows: section 2 briefly reviews weight restrictions and imprecise data in DEA. In section 3, a genetic algorithm is developed to estimate the relative efficiency in presence of both generalized form of weight restrictions and imprecise data. Numerical examples and conclusions are given in section 4 and 5, respectively.

II. LITERATURE REVIEW

A. DEA models in presence of weight restrictions

The Maximin model to calculate the relative efficiency is as follows (DMUp is under evaluation) [3]:

$$\max_{u,v\geq 0}\left\{\frac{\sum_r u_r y_{rp}}{\sum_i v_i x_{ip}} \Big/ \max_j\left\{\frac{\sum_r u_r y_{rj}}{\sum_i v_i x_{ij}}\right\}\right\} \quad (1)$$

Model (1) can be converted to the following model (2):

$$\begin{aligned} Max\ & \sum_r u_r y_{rp} \\ s.t.\ & \sum_i v_i x_{ip} = 1 \\ & \sum_r u_r y_{rj} - \sum_i v_i x_{ij} \leq 0,\ \forall j \\ & u_r, v_i \geq 0,\ \forall i, r \end{aligned} \quad (2)$$

As mentioned in section 1, the weights in model (1) and (2) are free. Weight restrictions are proposed to prevent overestimate or underestimate some inputs or outputs as well as to consider the viewpoint of decision maker about the weights of criteria (inputs and outputs). The most popular type of weight restrictions is linear constraints, which can be categorized into assurance regions type 1 (ARI), type 2 (ARII) and absolute weight restrictions. These types of weight restrictions are given in table 1.

Table 1: Linear weight restrictions

Assurance region type 1 (ARI)	$\alpha_i \leq v_i/v_{i+1} \leq \beta_i\ ,\ \lambda_r \leq u_r/u_{r+1} \leq \theta_r$
Assurance region type 2 (ARII)	$\gamma_i v_i \geq u_r$
Absolute weight restrictions	$\delta_i \leq v_i \leq \tau_i\ ,\ \rho_r \leq u_r \leq \eta_r$

where $\theta_r, \alpha_i, \beta_i, \gamma_i, \delta_i, \tau_i, \rho_r, \eta_r, \lambda_r$ are scalars. Tracy and Chen [4] proposed a generalized expression that could represent linear weight restrictions as follows:

$$\Omega = \{(u,v) : \alpha \leq a^T u + b^T v \leq \beta\} \quad (3)$$

In (3), a^T and b^T are vectors of proper dimensions of α and β. By choosing appropriate values for the α, β, a^T and b^T, we could obtain ARI, ARII and the absolute weight restrictions.

Model (2) in presence of ARI is always feasible and can calculate the relative efficiency correctly. But, by using the absolute weight restrictions and ARII some problems such as infeasibility and underestimation of the efficiency may be occurs. The interest readers can refer to [5, 6] for more study about the weight restrictions in DEA.

In 2010, Khalili et al. [6] proposed the nonlinear and non-convex model (4) to calculate the relative efficiency in presence of generalized weight restrictions (3). The model eliminated the mentioned drawbacks.

$$\begin{aligned} Max\ & \sum_r u_r y_{rp} \\ s.t.\ & c\sum_i v_i x_{ip} = 1 \\ & \sum_r u_r y_{rj} - c\sum_i v_i x_{ij} \leq 0,\ \forall j \\ & \alpha t \leq a^T u + b^T v \leq \beta t \\ & c, t, u_r, v_i \geq 0,\ \forall i, r \end{aligned} \quad (4)$$

B. DEA models with imprecise data

As mentioned in section 1, in some real-life applications the data maybe imprecise such as bounded, weak ordinal and ratio bound data. Cooper et al. [2] considered weak ordinal and bounded data in DEA. The model was nonlinear and non-convex. They converted the model into an equivalent linear programming through scale transformations and variable alterations. Zhu [7] showed that the scale transformations in [2] are redundant. He proposed a simpler method to calculate the relative efficiency in presence of imprecise data. To deal with ordinal data, he used the unit invariant property of DEA model and convert ordinal data into interval data format. Wang et al. [8] proposed a new pair of interval DEA models to deal with bounded data. They convert ordinal data into bounded data. Kao [9] and Park [10] proposed other method to deal with imprecise data. The interest reader can find a brief review on imprecise data in [11, 12, 13].

C. DEA models in presence of both imprecise data and weight restrictions

To the best of our knowledge, the only papers that discussed DEA model in presence of both imprecise data and weight restrictions are [2, 14, 15, and 16]. Cooper et al. [2] showed that model (2) in presence of both imprecise data (ordinal data and bounded data) and ARI is nonlinear and non-convex. They used the following two steps to convert the model into an equivalent linear programming.

Step 1: Scale transformations (normalizations).

Step 2: Variable alterations.

In bounded data format the first step needs an exact data with the maximum value. Cooper et al. [15] introduced some dummy variables to remove the limitation. Park [16] showed that step 1 and using dummy variables are redundant. He converted the model into an equivalent linear programming by variable alterations.

D. Application of heuristics in solving DEA models

Heuristic methods are the best way to solving complex problems. Heuristics such as tabu search, simulated annealing, GA and particle swarm optimization (PSO) increase the chance of finding global optimal solution. Some researchers have been used these methods to estimate the relative efficiency scores in DEA models. Wen & Li [17] and Wen et al. [18] used credibility measure approach for using fuzzy numbers in DEA. They used GA in solving Fuzzy DEA (FDEA) model. Meng [19] proposed a new satisficing data envelopment analysis (DEA) model in presence of fuzzy inputs and outputs. He developed a PSO algorithm to solve the model.

Literature review shows that existing references only considered ARI with imprecise data. By using ARI we could only set weight restrictions among the weights of inputs or outputs and we cannot set an upper bound or lower bound for weights. Also, we cannot create any link between the weights of inputs and outputs. Based on this lack, in the next section we discuss DEA model in presence of both imprecise data and generalized form of weight restrictions to fill this gap.

III. DEA IN PRESENCE OF BOTH IMPRECISE DATA AND GENERALIZED WEIGHT RESTRICTIONS

Consider model (4) with imprecise data:

$$\begin{aligned} & Max \ \sum_r u_r y_{rp} \\ & s.t. \quad c\sum_i v_i x_{ip} = 1 \\ & \qquad \sum_r u_r y_{rj} - c\sum_i v_i x_{ij} \le 0, \ \forall j \\ & \qquad \alpha t \le a^T u + b^T v \le \beta t \\ & \qquad (x_{ij}) \in \varphi_i^-; \quad (y_{rj}) \in \varphi_r^+ \\ & \qquad c, t, \ u_r, v_i \ge 0, \ \forall i, r \end{aligned} \tag{5}$$

Where $(x_{ij}) \in \varphi_i^-$ and $(y_{rj}) \in \varphi_r^+$ represent any or all of the following imprecise data (Equations 6 – 8).

Bounded data

$$\begin{aligned} & \underline{x}_{ij} \le x_{ij} \le \bar{x}_{ij} \qquad (i \in BI) \\ & \underline{y}_{rj} \le y_{rj} \le \bar{y}_{rj} \qquad (r \in BO) \end{aligned} \tag{6}$$

where $\underline{y}_{rj}$ and $\bar{y}_{rj}$ are the lower and the upper bounds for outputs, $\underline{x}_{ij}$ and $\bar{x}_{ij}$ are the lower and the upper bounds for inputs, and BO and BI represent the associated sets containing bounded outputs and inputs, respectively.

Weak ordinal data

$$\begin{aligned} & x_{i1} \le x_{i2} \le ... \le x_{in} \qquad (i \in DI) \\ & y_{r1} \le y_{r2} \le ... \le y_{rn} \qquad (r \in DO) \end{aligned} \tag{7}$$

where DO and DI represent the associated sets containing weak ordinal outputs and inputs, respectively.

Ratio bounded data

$$\begin{aligned} & G_{ij} \le x_{ij} / x_{ij_o} \le H_{ij} \qquad (j \ne j_o) \ \ (r \in RO) \\ & L_{rj} \le y_{rj} / y_{rj_o} \le U_{rj} \qquad (j \ne j_o) \ \ (i \in RI) \end{aligned} \tag{8}$$

where L_{rj} and G_{ij} represent the lower bounds, U_{rj} and H_{ij} represent the upper bounds, and RO and RI represent the associated sets containing ratio bounded outputs and inputs, respectively.

As mentioned in previous section, model (4) successfully calculates the relative efficiency. So, model (5) can be used to calculate the relative efficiency in presence of both imprecise data and generalized form of weight restrictions. The next theorem shows a special property of the model in presence of ordinal data.

Theorem 1: in model (5), suppose the rth output of DMUs is in ordinal data format and DMUp has the best rank. In the other words, suppose $y_{rp} \ge y_{rj}, \forall j$. In this case, DMUp is efficient.

Proof: In calculation of the relative efficiency score of DMUp, model (5) could selects y_{rp} enough large positive number and set $y_{rj}, \forall j \ne p$ very small numbers such that DMUp dominates all of the other DMUs. It should be noted that, we cannot set an upper bound for ordinal data in model (5) (the basic idea has been used in many references such as: [2, 7, 8, 9, 14 and 15]). Since, DEA models in presence of generalized form of weight restrictions are not unit invariant.

Note that a similar theorem can be presented for ordinal data in inputs.

A. Developing a genetic algorithm to solve model (5)

Since GA has been used effectively in solving many complex problems (see [17, 18, 20]), here we use it to estimate the relative efficiency in model (5). In special cases, for example when data are crisp, nonlinear solvers (eg, PATHNLP or MINOS) existing in professional optimizations softwares such as GAMS can be used to solve the model [6]. It should be noted that the model is nonlinear and non-convex. Hence, the optimum solution may not be a global optimum solution of the model. Also, we believed that when some inputs and outputs are in bounded and ordinal data format, the mentioned nonlinear solvers are unable to obtain the global optimum of the model (see example 2). So, to estimate the efficiency scores we developed a genetic algorithm (GA).

Model (5) has a lot of variables and constraints, so it is hard to generate feasible solutions in GA. Here we use the basic form of DEA model (1) to estimate the relative efficiency in presence of both imprecise data and generalized form of weight restrictions. So, to calculate the relative efficiency of DMUp the following model is used:

$$Max\left\{\frac{\sum_r u_r y_{rp}}{\sum_i v_i x_{ip}} \Big/ \underset{j}{Max}\left\{\frac{\sum_r u_r y_{rj}}{\sum_i v_i x_{ij}}\right\}\right\} \quad (9)$$

$$s.t. \quad \alpha \le a^T u + b^T v \le \beta$$

$$(x_{ij}) \in \varphi_i^-; \quad (y_{rj}) \in \varphi_r^+$$

$$u_r, v_i \ge 0, \ \forall i, r$$

This model has less variables and constraint comparing with model (5). Consequently, model (9) is easier than model (5) to generate feasible solutions. It is not too hard to show that model (5) and model (9) are equivalent (a proof is given in [6] when data are crisp). The main steps of developed GA to solve model (9) is as follows.

Chromosome representation: We use a nonnegative vector $\lambda = (u, v, x, y)$ to say a solution. u, v, x and y are vector. u_r is the weight of rth output, v_i is the weight of ith input, $x_{ij}, \forall i, j$ are the inputs and $y_{rj}, \forall r, j$ are the outputs.

Initialization process: We randomly generate λ. The feasibility of λ can be checked by the constraints of model (9). If it is feasible, we accept it as a solution. If not, then we regenerate randomly another solution until a feasible one is obtained. Repeating the process Pop_size times, we could make Pop_size initial feasible solutions. Pop_size is one of GA parameters which is size of population.

Evaluation process: the objective function of model (9) is used to calculate the fitness for all chromosomes.

Selection and crossover operations: we use the ranking selection method for selecting two parents for producing a child. We generate a random variable $0 \le r \le 1$ and make a child as follows:

$$Child = r * parent1 + (1 - r) * parent2$$

The feasibility of any child can be checked by the constraints of model (9). We repeat this process until a feasible child obtained. This process are repeated Number_of_Children times.

Mutation operations: suppose $|\lambda| = k$, in this case we randomly generate an integer number $n_1 \in \{1, 2, ..., k\}$. Then we select the $n_1 th$ element of vector λ to make a mutation as follows:

Suppose λ_{n_1} be the $n_1 th$ element of vector λ .we randomly generate $0 \le r_{n_1} \le 1$ and set $\lambda_{n_1} = \lambda_{n_1} + (2r_{n_1} - 1)\varepsilon$ in which ε is a small positive number. The feasibility of new λ can be checked by using the constraints of model (9). We repeat this process until a feasible solution obtained. This process are repeated Number_of_Mutations times.

Population update: we add the best chromosomes from the past population to the population of children and mutations to achieve the new population.

The algorithm will be stopped after the appropriate predetermined number of generations.

IV. NUMERICAL EXAMPLES

In this section two numerical examples are presented to illustrate the content of the paper. First example is presented for testing the solution quality of developed GA. In the example we use the Taguchi methodology for tuning the parameter in the developed GA. Second example is given to demonstrate that in some cases the professional solvers are unable to solve model (5). Also, in this example a sensitivity analysis on ordinal data is performed.

Example 1: the data for this example are taken from [6]. The data and the corresponding absolute weight restrictions are shown in table 3 and 4, respectively. The exact efficiency scores are given in the seventh column of table 3. To estimate the efficiency scores by the developed GA, first, we use the design of experiments methodology to obtain the best configuration of GA parameters. We consider 4 levels for each parameter as follows:

Table 2: levels of GA parameters for experimental design

Levels	Crossover	Mutation	Pop_size
1	0.25	0.15	20
2	**0.3**	0.2	25
3	0.4	**0.25**	30
4	**0.5**	0.3	**35**

We used the Taguchi method and performed 16 experiments for DMU1. The result of MINITAB is shown in figure 1.

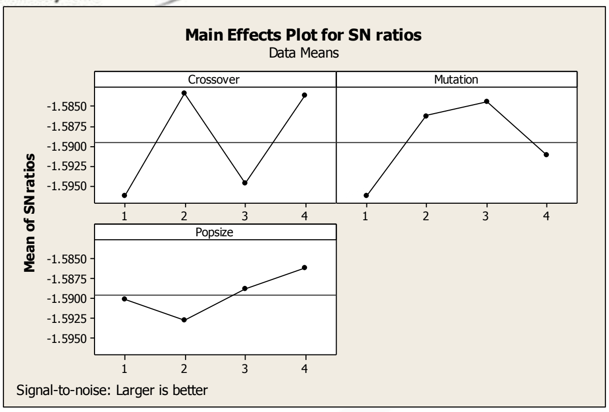

Figure 1: the result of MINITAB for parameter tuning of example 1

According to the figure 1, the best configurations of GA parameters are highlighted in table 2. So, we set the crossover rate to 0.5, the mutation rate to 0.25 and the population size to 35. We set the number of generations to 200 and run the developed GA 5 times for each DMU. The mean of objective functions are shown in last column of table 3. As it can be seen, the efficiency scores estimated by the developed GA are very near to the exact values. Figure 2, shows the convergence of developed GA for DMU3. The runtime is less than one second for each DMU.

Table 3: data and efficiency scores for 10 DMUs in example 1

DMUs	Inputs			Outputs		Model (4)	Efficiency score by GA
	1	2	3	1	2		
1	1	0.8	5.4	0.9	7	0.8358	0.8330
2	1.5	1	4.8	1	9.5	0.8748	0.8728
3	1.2	2.1	5.1	0.8	7.5	0.6398	0.6394
4	1	0.6	4.2	0.9	9	1.0000	1.0000
5	1.8	0.5	6	0.7	8	0.6819	0.6816
6	0.7	0.9	5.2	1	5	1.0000	0.9952
7	1	0.3	5	0.8	7	1.0000	0.9948
8	1.2	1.5	5.5	0.75	7.5	0.6176	0.6175
9	1.4	1.8	5.7	0.65	5.5	0.4699	0.4680
10	0.8	0.9	4.5	0.85	9	1.0000	1.0000

Table 4: weight bounds for example 1

	Inputs			Outputs	
	1	2	3	1	2
Upper bound	0.7	0.8	0.2	0.9	0.1
Lower bound	0.2	0.2	0.1	0.7	0.02

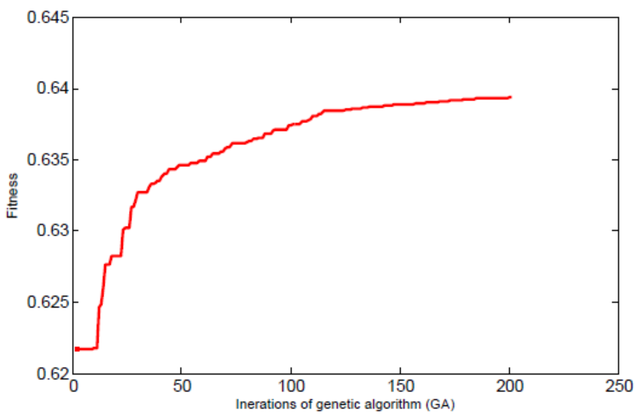

Figure 2: The trend of genetic algorithm convergence for DMU_3

Example 2: Consider 3 DMUs each uses two inputs to produce two outputs. The data are presented in table 5.

Table 5: Data for 3 DMUs

DMU No.	Input 1 (*)	Input 2	Output 1 (*)	Output 2
1	x_{11}	$[4,5]$	y_{11}	$[1,2]$
2	x_{12}	$[1,3]$	y_{12}	$[3,4]$
3	x_{13}	$[1,2]$	y_{13}	$[4,5]$

*Ranking such that: $x_{11} \geq x_{13} \geq x_{12}$ & $y_{12} \geq y_{13} \geq y_{11}$.

Input 2 and output 2 are in bounded data format and Input 1 and output 1 are in weak ordinal data format. Suppose we have the following weight restrictions:

$$u_2 \geq v_1 \ \& \ v_2 \geq u_1 \ \& 0 \leq v_1, u_1 \leq 100$$

We found that GAMS NLP solvers such as MINOS and PATHNLP are unable to solve model (5) for the data.

As explained in theorem 1, in this example we cannot set upper bounds for ordinal data and converted them into bounded data format. Indeed, as it will be shown, the efficiency scores are very sensitive to upper bound of ordinal data.

The efficiency scores are estimated by using the developed GA by considering the different values for the upper bounds to x_{11} and y_{12}. We set the generations to 2000 and run the developed GA five times for each DMU and selected the maximum fitness. The result showed that the efficiency scores of DMU2 and DMU3 are equal to one in any condition for upper bounds of x_{11} and y_{12}. In the other words, these DMUs are efficient. The result for DMU2 is completely consistent with theorem 1. Table 6 summarized the result of developed GA for DMU1.

Table 6: the efficiency score of DMU_1 with different upper bounds for ordinal data

Upper bounds		Eff. Scores	Upper bounds		Eff. Scores	Upper bounds		Eff. Scores
x_{11}	y_{12}		x_{11}	y_{12}		X_{11}	y_{12}	
1	1	0.4988	1	1	0.4988	1	1	0.4988
10	10	0.6886	10	1	0.5024	1	10	0.5372
20	20	0.8033	20	1	0.5488	1	20	0.5733
40	40	0.8991	40	1	0.6184	1	40	0.6335
80	80	0.9251	80	1	0.7	1	80	0.6422
100	100	0.949	100	1	0.7225	1	100	0.7376
200	200	0.9762	200	1	0.7974	1	200	0.7919

As it can be seen from table 6, the efficiency score of DMU1 increase when the upper bounds of x_{11} and y_{12} are increase. In the other words, the efficiency score of DMU1 has a positive direct relationship with the upper bounds of x_{11} and y_{12}. It is not hard to show that DMU1 will be efficient if we do not consider any upper bound for the ordinal data. The result is unacceptable, because DMU1 is dominated by DMU2 and DMU3. To prevent such results, in real applications, we need to ask the decision maker to specify appropriate upper bounds for ordinal data.

V. CONCLUSIONS

In this paper DEA model in presence of both imprecise data and generalized form of weight restrictions is discussed. We show that if a DMU had the best rank in an input or output in ordinal data format, then it will be efficient. Due to the complexity of the model a genetic algorithm (GA) developed to solve it. To illustrate the content of the paper two numerical examples are presented. Four levels are considered for each parameters of developed GA and the Taguchi method has been used for tuning the parameters. The results show that the best configuration of developed GA parameters is crossover = 03 or 0.5, mutation = 0.25 and Pop_size=35. The first example shows that the solution quality of developed GA is very good. The second example shows that model (5) cannot be solved in some cases with professional NLP solvers. To solve the problem the proposed GA is used. Our sensitivity analysis showed that the efficiency score in presence of ordinal data has a direct relationship with the upper bound of this data.

This study is initial stage of investigation DEA model in presence of both generalized weight restrictions and imprecise data. Other research can be done in presence of both generalized weight restrictions and fuzzy/stochastic data. Also, other heuristics methods can be developed to solve these models.

REFERENCES

[1] A. Charnes, W.W. Cooper, E. Rhodes, “Measuring the efficiency of decision making units”, *Eur. J. Oper. Res.*, 2, 429–44, 1978.

[2] W.W. Cooper, K.S. Park, G. Yu, “IIDEA and AR-IDEA: models for dealing with imprecise data in DEA’, *Manage. Sci.*, 45, 597– 607, 1999.

[3] R.G. Thompson, R.M. Thrall, “Polyhedral assurance region with linked constraints. New Direction in Computational Economics”, vols. 121–133. Kluwer Academic Pub., Dordrecht, The Netherlands, 1994.

[4] D.L. Tracy, B. Chen, “A generalized model for weight restrictions in data envelopment analysis”, *J. Oper. Res. Soc.*, 56, 390–396, 2005.

[5] R. Allen, A. Athanassopoulos, R.G. Dyson, E. Thanassoulis, “Weights restrictions and value judgments in data envelopment analysis: Evolution development and future directions”, *Ann. Oper. Res.*, 73, 13–34, 1997.

[6] M. Khalili, A.S. Camanho, M. Portela, M. Alirezaee, “An improvement on the Tracy and Chen model ‘A generalized model for weight restrictions in DEA’”, *J. Oper. Res. Soc.*, 61, 1789–1793, 2010.

[7] J. Zhu, “Imprecise data envelopment analysis (IDEA): a review and improvement with an application”, *Eur. J. Oper. Res.*,144, 513–529, 2003.

[8] Y.M. Wang, R. Greatbanks, J.B. Yang, “Interval efficiency assessment using data envelopment analysis”, *Fuzzy Sets and Sys.*, 153, 347–370, 2005.

[9] C. Kao, “Interval efficiency measures in data envelopment analysis with imprecise data”, *Eur. J. Oper. Res.*, 174, 1087–1099, 2006.

[10] K.S. Park, “Duality, efficiency computations and interpretations in imprecise DEA”, *Eur. J. Oper. Res.*, 200, 289–296, 2010.

[11] A. Marbini, A. Emrouznejad, P. Agrell, “Interval data without sign restrictions in DEA”, *Appl. Math. Model.*, 38, 2028–2036, 2014.

[12] E. Karsak, M. Dursun, “An integrated supplier selection methodology incorporating QFD and DEA with imprecise data”, *Expert Syst. Appl.*, 41, 6995–7004, 2014.

[13] K.K. Damghani, M. Tavana, E.H. Saami, “A data envelopment analysis model with interval data and undesirable output for combined cycle power plant performance assessment”, *Expert Syst. Appl.*, in press, 2014.

[14] W.W. Cooper, K.S. Park, G. Yu, “An illustrative application of IDEA (imprecise data envelopment analysis) to a Korean mobile telecommunication company”, *Oper. Res.*, 49, 807–820, 2001.

[15] W.W. Cooper, K.S. Park, G. Yu, “IDEA (imprecise data envelopment analysis) with CMDs (column maximum decision making units)”’ *J. Oper. Res. Soc.*, 52, 176–181, 2001.

[16] K.S. Park, “Simplification of the Transformations and Redundancy of Assurance Regions in IDEA (Imprecise DEA)”, *J. Oper. Res. Soc.*, 55, 1363-1366, 2004.

[17] M.L. Wen, H. Li, “Fuzzy data envelopment analysis (DEA): Model and ranking method”, *J. of Comput. Appl. Math.*, 223, 872–878, 2009.

[18] M. Wen, C. You, R., Kang, “A new ranking method to fuzzy data envelopment analysis”, *Comput. Math. Appl.*, 59, 3398-3404, 2010.

[19] M. Meng, “A hybrid particle swarm optimization algorithm for satisficing data envelopment analysis under fuzzy chance constraints”, *Expert. Syst. Appl.*, 41, 2074–2082, 2014.

[20] M.L. Wen, K. Iwamura, “Fuzzy facility location-allocation problem under the Hurwicz criterion”, *Eur. J. Oper. Res.*, 184, 627–635, 2008.