International Journal of Operations Research

International Journal of Operations Research Vol. 4, No. 2, 105−109 (2007)

Inverse Linear Programming in DEA

Gholam R. Amin[1,*] and Ali Emrouznejad[2]

[1]Department of Computer Sciences, Postgraduate Engineering Centre, Islamic Azad University of South Tehran Branch, Tehran-Iran

[2]Operations & Information Management Group, Aston Business School, Aston University, Birmingham B4 7ET, UK

Received April 2005; *Revised January 2007*; *Accepted February 2007*

Abstract**—**Despite the large uses of inverse DEA models, there is not any single application of inverse linear programming in DEA when the definition of inverse linear programming is taken under account. Thus the goal of this paper is applying the inverse linear programming into DEA field, and to provide a streamlined approach to DEA and Additive model. Having the entire efficient DMUs in DEA models is an important rule. To speed up the computations of the Additive DEA model this paper uses the inverse linear programming as an alternative procedure. It proposes an alternative inverse notion-based method which is capable to determine all the efficient DMUs of the model.

Keywords**—**Data envelopment analysis (DEA), Inverse linear programming

1. INTRODUCTION

Data Envelopment Analysis (DEA), as reported in Charnes et al. (1978) and extended in Banker et al. (1984), is a recognized tool to the assessment of performance of organizations. DEA has gained a wide range of successful applications measuring comparative efficiency of multiple inputs and multiple outputs of a homogeneous set of decision making units (DMUs) (Cooper et al. (2006)). New applications with more variables and more complicated models are being introduced in Cooper et al. (2006) and Emrouznejad et al. (2007). There are several instances in the literature of DEA applications involving large data sets. One of these studies involves a large data set containing 8000 DMUs (Barr and Durchholz (1997)). As more analysts apply the DEA methodology and as new applications introduced, data sets become larger and more effort is needed to spend on extracting information efficiently (Dulá (2006)). Several attempts have recently been made to improve the speed of the DEA calculations. Amin and Toloo (2004) proposed an efficient polynomial time algorithm to determine an assurance value in DEA models. Also Amin and Emrouznejad (2007) proposed an efficient form for the maximum non-Archimedean in the technology selection model with ordinal outputs. Despite the large uses of inverse DEA models (Wei et al. (2000), Pendharkar (2002)), there is not any single application of inverse linear programming in DEA when the definition of inverse linear programming is taken under account. An inverse programming problem consists of inferring the values of the model parameters such as cost coefficient, right hand side vector and the constraint matrix given the values of observable parameters, Ahuja and Orlin (2001).

The goal of this study is to integrate the inverse linear programming into DEA field, and providing a streamlined approach to DEA and Additive model. Under the L_1 norm of inverse linear programming, this paper proposes an alternative model and an algorithm for determining the Additive-based efficient DMUs. A numerical example has been used to demonstrate the use of inverse linear programming developed in this study.

2. INVERSE LINEAR PROGRAMMING

The inverse linear programming problem (ILPP) has first been investigated by Zhang and Liu (1996) and Huang and Liu (1999). They formulate the ILPP as a new linear program and they show how the solution to a new problem (which is similar to the original problem and the associated dual solutions) can be used to solve the inverse problem. We sketch their approach in the following.

Let S denotes the set of feasible solutions for a linear programming problem, say P. Assume that the relevant specified cost vector is $\boldsymbol{c}$ and $\boldsymbol{x}^0$ is a given feasible solution. The inverse linear programming problem is to perturb the cost vector $\boldsymbol{c}$ to $\boldsymbol{d}$ so that $\boldsymbol{x}^0$ be the optimal solution of P with respect to $\boldsymbol{d}$ and $\|\boldsymbol{d}-\boldsymbol{c}\|_p$ is minimized, where $\|\,.\,\|_p$ is some selected L_p norm (Ahuja and Orlin (2001)). Consider the following linear programming:

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} c_j x_j \\ s.t. \quad & \\ & \sum_{j=1}^{n} a_{ij} x_j = b_i \quad i = 1,\ldots,m \end{aligned}$$

[*] Corresponding author's email: G_Amin@azad.ac.ir

$$x_j \geq 0 \quad j=1,\ldots,n \tag{1}$$

Let $\mathbf{x}^0$ be a feasible solution of the model. The corresponding inverse problem under L_1 norm becomes as, (Zhang and Liu (1996)),

$$\begin{aligned} &\min \sum_{j=1}^{n} (\alpha_j + \beta_j) \\ &s.t. \\ &\quad \sum_{i=1}^{m} a_{ij}\pi_i - \alpha_j + \beta_j + \gamma_j = c_j \qquad \forall j \in L \\ &\quad \sum_{i=1}^{m} a_{ij}\pi_i - \alpha_j + \beta_j = c_j \qquad \forall j \in F \\ &\quad \alpha_j \geq 0,\ \beta_j \geq 0,\ j=1,\ldots,n, \\ &\quad \gamma_j \geq 0, \forall j \in L \end{aligned} \tag{2}$$

where $L = \{j: x_j^0 = 0\}$ and $F = \{j: x_j^0 > 0\}$.

Next we use ILPP as an alternative approach for Additive model.

3. INVERSE ADDITIVE MODEL

The Additive model for DMU$_k$ which using $\mathbf{x}_k = (x_{1k}, \ldots, x_{mk})$ as inputs and producing $\mathbf{y}_k = (y_{1k}, \ldots, y_{sk})$ as outputs, can be written by the following linear program:

$$\begin{aligned} &z_k^* = \min\ z_k = -\sum_{i=1}^{m} s_i^{in} - \sum_{r=1}^{s} s_r^{out} \\ &\quad s.t. \\ &\qquad -\sum_{j=1}^{n} x_{ij}\lambda_j - s_i^{in} = -x_{ik} \quad i=1,\ldots,m \\ &\qquad \sum_{j=1}^{n} y_{rj}\lambda_j - s_r^{out} = y_{rk} \quad r=1,\ldots,s \\ &\qquad \sum_{j=1}^{n} \lambda_j = 1 \\ &\qquad s_i^{in} \geq 0,\ i=1,\ldots,m,\ s_r^{out} \geq 0,\ r=1,\ldots,s \\ &\qquad \lambda_j \geq 0,\ \ j=1,\ldots,n \end{aligned} \tag{3}$$

The under evaluation DMU, DMU$_k$, is efficient if and only if $z_k^* = 0$. Assume that $\mathbf{x} = (\boldsymbol{\lambda}, \mathbf{s}^{in}, \mathbf{s}^{out}) \in \Re_+^{n+m+s}$ denotes the vector of variables of model (3). Now consider the feasible solution $\mathbf{x}^0 = (\mathbf{e}_k, \mathbf{0}_m, \mathbf{0}_s)$, where $\mathbf{e}_k$ is the n-vector with the kth position equals to one and zero elsewhere and $\mathbf{0}_p$ is the p-vector with all components equal to zero. That is $\lambda_k = 1$ and all the other variables equal to zero. The corresponding inverse linear program to $\mathbf{x}^0$ becomes as

$$\begin{aligned} &ILP(\mathbf{x}^0): \\ &\min \sum_{j=1}^{n+m+s} \left(\alpha_j + \beta_j\right) \\ &s.t. \\ &-\sum_{i=1}^{m} x_{ij}\pi_i + \sum_{r=1}^{s} y_{rj}\pi_{m+r} + \pi_{m+s+1} - \alpha_j + \beta_j + \gamma_j \\ &= 0 \quad \forall j \in I_n(k) \\ &-\pi_{j-n} - \alpha_j + \beta_j + \gamma_j = -1 \quad j = n+1,\ldots,n+m+s \\ &-\sum_{i=1}^{m} x_{ik}\pi_i + \sum_{r=1}^{s} y_{rk}\pi_{m+r} + \pi_{m+s+1} - \alpha_k + \beta_k = 0 \\ &\alpha_j \geq 0,\ \beta_j \geq 0, \\ &j = 1,\ldots,n+m+s,\ \gamma_j \geq 0 \ \ \forall j \in I_{n+m+s}(k) \end{aligned} \tag{4}$$

where $I_p(q) = \{j: 1, \ldots, p, j \neq q\}$. The definition of the inverse linear programming implies that DMU$_k$ is efficient if and only if $ILP(\mathbf{x}^0)$ has a zero optimal value.

4. AN ALTERNATIVE MODEL

Theorem 1. DMU$_k$ is efficient if and only if the following linear system has a solution.

$$\begin{aligned} &-\sum_{i=1}^{m} x_{ij}\pi_i + \sum_{r=1}^{s} y_{rj}\pi_{m+r} + \pi_{m+s+1} + \gamma_j = 0 \quad \forall j \in I_n(k) \\ &-\pi_{j-n} + \gamma_j = -1 \qquad j = n+1,\ldots,n+m+s \\ &-\sum_{i=1}^{m} x_{ik}\pi_i + \sum_{r=1}^{s} y_{rk}\pi_{m+r} + \pi_{m+s+1} = 0 \\ &\gamma_j \geq 0 \quad \forall j \in I_{n+m+s}(k) \end{aligned} \tag{5}$$

Proof. Assume the DMU$_k$ is efficient, then the corresponding inverse problem has the zero optimal value and therefore there is a solution say, $(\boldsymbol{\pi}^0, \boldsymbol{\gamma}^0)$ such that:

$$\begin{aligned} &-\sum_{i=1}^{m} x_{ij}\pi_i^0 + \sum_{r=1}^{s} y_{rj}\pi_{m+r}^0 + \pi_{m+s+1}^0 + \gamma_j^0 = 0 \qquad \forall j \in I_n(k) \\ &-\pi_{j-n}^0 + \gamma_j^0 = -1 \quad j = n+1,\ldots,n+m+s \\ &-\sum_{i=1}^{m} x_{ik}\pi_j^0 + \sum_{r=1}^{s} y_{rk}\pi_{m+r}^0 + \pi_{m+s+1}^0 = 0 \\ &\gamma_j^0 \geq 0 \qquad \forall j \in I_{n+m+s}(k) \end{aligned}$$

So the necessary condition holds. Conversely suppose that linear system (3) has a solution, say $(\bar{\boldsymbol{\pi}}, \bar{\boldsymbol{\gamma}})$. Taking $\bar{\boldsymbol{\alpha}} = \bar{\boldsymbol{\beta}} = \mathbf{0}_n$ makes the vector $(\bar{\boldsymbol{\pi}}, \bar{\boldsymbol{\gamma}}, \bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}})$ as a feasible solution of model (4). Since it has the zero optimal value therefore it is also the optimal solution of the model and therefore the corresponding Additive model has $z_k^* = 0$. This terminates the proof.

Corollary 1. DMU$_k$ is efficient if and only if the following system has a solution.

$$\begin{aligned} &-\sum_{i=1}^{m} x_{ij}\pi_i + \sum_{r=1}^{s} y_{rj}\pi_{m+r} + \pi_{m+s+1} + \gamma_j = \sum_{i=1}^{m} x_{ij} - \sum_{r=1}^{s} y_{rj} \\ &\forall j \in I_n(k) \\ &-\sum_{i=1}^{m} x_{ik}\pi_i + \sum_{r=1}^{s} y_{rk}\pi_{m+r} + \pi_{m+s+1} = \sum_{i=1}^{m} x_{ik} - \sum_{r=1}^{s} y_{rk} \end{aligned}$$

$$\pi_i \geq 0,\ i = 1,\ldots,m+s,\ \gamma_j \geq 0,\ \forall j \in I_k \quad (6)$$

Proof. It is easy to see that taking $\pi_i - 1 \geq 0$ as the new variables (i = 1, ..., m + s) in (3), an equivalent system (6) will be obtained.

Corollary 2. DMU$_k$ is efficient if and only if the following system has a solution.

$$-\sum_{i=1}^{m}(x_{ij} - x_{ik})\pi_i + \sum_{r=1}^{s}(y_{rj} - y_{rk})\pi_{m+r} + \gamma_j$$
$$= \sum_{i=1}^{m}(x_{ij} - x_{ik}) + \sum_{r=1}^{s}(y_{rk} - y_{rj}),\ \forall j \in I_n(k) \quad (7)$$
$$\pi_i \geq 0,\ i = 1,\ldots,m+s,\ \gamma_j \geq 0,\ \forall j \in I_n(k)$$

Proof. Since π_{m+s+1} in (6) is a free variable therefore using the last equation it can be removed from the other equations.

Hereafter we use the following equivalent notation for linear system (7).

$$(\mathbf{x}_k - \mathbf{x}_j)\boldsymbol{\pi}_1 + (\mathbf{y}_j - \mathbf{y}_k)\boldsymbol{\pi}_2 + \boldsymbol{\gamma}_j$$
$$= (\mathbf{1}_m\mathbf{x}_j - \mathbf{1}_m\mathbf{x}_k) + (\mathbf{1}_s\mathbf{y}_k - \mathbf{1}_s\mathbf{y}_j) \quad \forall j \in I_n(k) \quad (8)$$
$$\boldsymbol{\pi}_1 \geq 0_m,\ \boldsymbol{\pi}_2 \geq 0_s,\ \boldsymbol{\gamma}_j \geq 0,\ \forall j \in I_n(k)$$

where $\mathbf{x}_j = (x_{1j}, \ldots, x_{mj})$, $\mathbf{y}_j = (y_{1j}, \ldots, y_{sj})$ for j = 1, ..., n, $\boldsymbol{\pi}_1 = (\pi_1, \ldots, \pi_m)^t$ and $\boldsymbol{\pi}_2 = (\pi_{m+1}, \ldots, \pi_{m+s})^t$ is used for simplicity.

Theorem 2. If

$$\text{Min}\{\mathbf{1}_m\mathbf{x}_j - \mathbf{1}_s\mathbf{y}_j : j = 1, \ldots, n\} = \mathbf{1}_m\mathbf{x}_k - \mathbf{1}_s\mathbf{y}_k$$

then DMU$_k$ is efficient.

Proof. The hypothesis implies

$$\mathbf{1}_m\mathbf{x}_j - \mathbf{1}_s\mathbf{y}_j \geq \mathbf{1}_m\mathbf{x}_k - \mathbf{1}_s\mathbf{y}_k$$

So $\boldsymbol{\theta}_j = (\mathbf{1}_m\mathbf{x}_j - \mathbf{1}_m\mathbf{x}_k) + (\mathbf{1}_s\mathbf{y}_k - \mathbf{1}_s\mathbf{y}_j) \geq 0$. Clearly $(\bar{\boldsymbol{\pi}}_1, \bar{\boldsymbol{\pi}}_2, \bar{\boldsymbol{\gamma}}) = (\mathbf{0}_m, \mathbf{0}_s, \bar{\boldsymbol{\gamma}})$ is a feasible solution of linear system (8), where $\bar{\gamma}_j = \theta_j$. This completes the proof.

Now define $e_k(j) = (\mathbf{1}_m\mathbf{x}_j - \mathbf{1}_m\mathbf{x}_k) + (\mathbf{1}_s\mathbf{y}_k - \mathbf{1}_s\mathbf{y}_j)$ for each $j \in I_n(k)$ and consider the following set

$$J_k = \{j \in I_n(k) : e_k(j) < 0\}$$

If $J_k = \phi$ then according to Theorem 2 DMU$_k$ is efficient. Otherwise for each $j \in J_k$ introduce the artificial variables δ_j to the jth equation in (8) and consider the following linear program:

$$\delta_k^* = \min \sum_{j \in J_k} \delta_j$$
$$s.t.$$
$$(\mathbf{x}_k - \mathbf{x}_j)\boldsymbol{\pi}_1 + (\mathbf{y}_j - \mathbf{y}_k)\boldsymbol{\pi}_2 + \boldsymbol{\gamma}_j - \delta_j = e_k(j) \quad j \in J_k$$
$$(\mathbf{x}_k - \mathbf{x}_j)\boldsymbol{\pi}_1 + (\mathbf{y}_j - \mathbf{y}_k)\boldsymbol{\pi}_2 + \boldsymbol{\gamma}_j = e_k(j) \quad j \notin J_k \quad (9)$$
$$\boldsymbol{\pi}_1 \geq 0_m,\ \boldsymbol{\pi}_2 \geq 0_s,\ \boldsymbol{\gamma}_j \geq 0, j \in I_k(n), \delta_j \geq 0, j \in J_k$$

Notice that if $\delta_k^* > 0$, then system (8) is infeasible and consequently DMU$_k$ is inefficient. Otherwise $\delta_k^* = 0$ and DMU$_k$ will be efficient.

Theorem 3. There is at least one index k for which $J_k = \phi$.

Proof. Let the minimum stated in Theorem 2 occurs at index k. So for each $j \in J_k$, $e_k(j) \geq 0$.

Based on the theoretical results of this paper and in order to apply the alternative model efficiently we design a simple algorithm in the following section.

5. THE ALGORITHM

Figure 1 shows the algorithm of the alternative methodology for the DEA observed data set containing n DMUs. We use E and N as the sets of efficient and inefficient DMUs, respectively.

In the coming section we give some computational advantages of the proposed alternative model (9).

6. COMPUTATIONAL ILLUSTRATION

We propose model (9) and its corresponding algorithm given in Figure 1 as an alternative for the Additive DEA model. Besides of introducing the inverse definition based model into the DEA literature it has also some computational advantages comparing with the conventional Additive model (3). Due to the linear programming we know that solving the Additive model (3) requires a two phase method. The first phase minimizes the sum of the artificial variables which are considered to the m + s constraints and because of the feasibility of Additive model, model (3), the second phase continues solving the original model following the basic feasible solution (BFS) obtained from the first phase. Meanwhile the proposed alternative model (9) exactly is a first phase type model. In order to demonstrate the proposed methodology, we use the following data consisting of the 12 general hospitals with two inputs, Doctors and Nurses, and two outputs, Outpatients and Inpatients used in Cooper et al. (2006). Table 1 shows the data.

The initial computations of the algorithm implemented in Microsoft Excel. These results are shown in Table 2. Table 2 gives the following information:

$$J_A = \{I, J, K, L\},\quad J_B = I_{12}(B),\quad J_C = I_{12}(C) - \{B\},$$
$$J_D = I_{12}(C) - \{B, C, E\},\quad J_E = I_{12}(E) - \{B, C\},$$
$$J_F = \{I, J, K, L\},\quad J_G = \{A, F, I, J, K, L\},$$
$$J_H = \{A, F, G, I, J, K, L\},\quad J_I = \{K, L\},\quad J_J = \{K, L\},$$

$J_K = \phi$, $J_L = \{K\}$.

Table 1. Data for 12 hospitals

DMU	Doctors	Nurses	Outpatients	Inpatients
A	20	151	100	90
B	19	131	150	50
C	25	160	160	55
D	27	168	180	72
E	22	158	94	66
F	55	255	230	90
G	33	235	220	88
H	31	206	152	80
I	30	244	190	100
J	50	268	250	100
K	53	306	260	147
L	38	284	250	120

According to Theorem 3 $J_K = \phi$ which means that DMU_K is efficient. The above results also show that how many of artificial variables are needed for solving model (9) for each DMU_j. For example consider DMU_L. Since $|J_L| = 1$ the relevant model (9) contains only one artificial variable called δ_L and the objective function minimizes δ_L. Now we solve the main step of the algorithm which is solving model (9). We use WinQSB software. For the comparing purposes we also solved the Additive model by the same software in a Pentium 4 PC (Dual core, 2.8 MGHz). Table 3 gives the optimal values of the Additive and the alternative models and their CPU times, respectively.

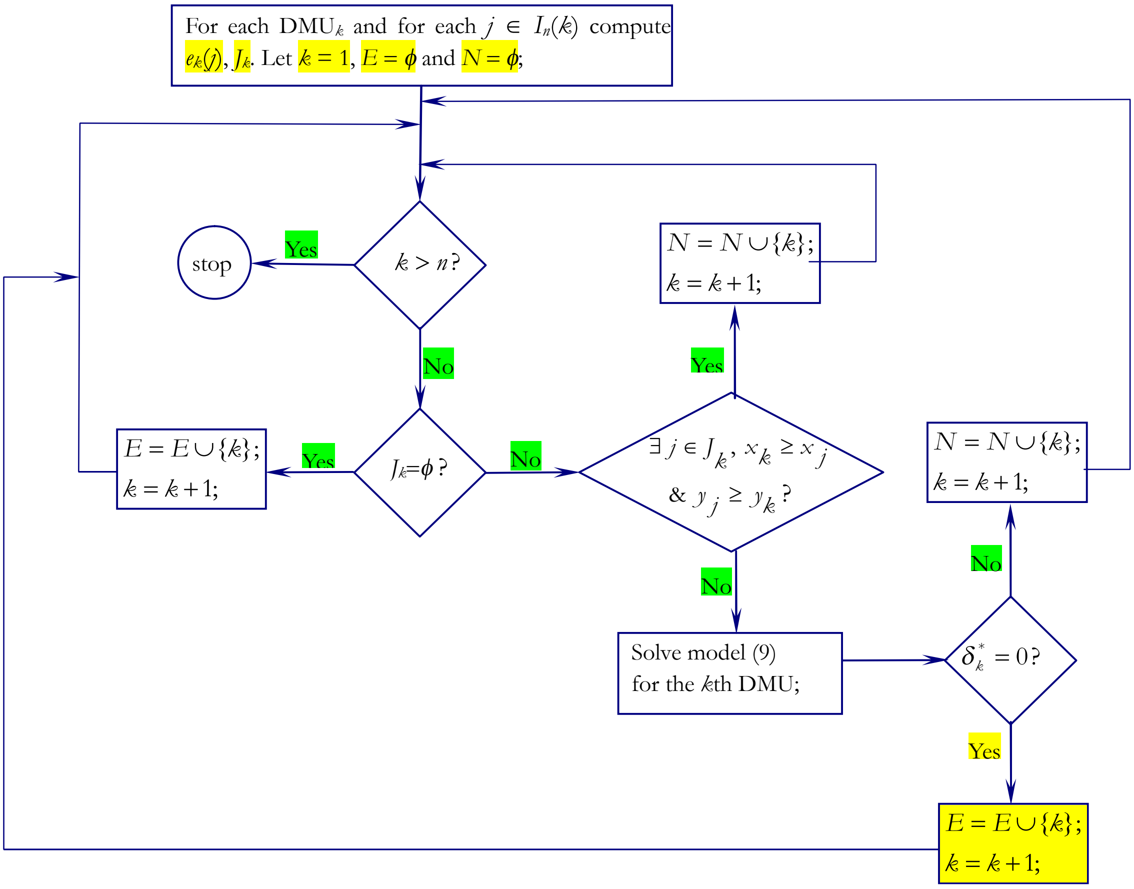

Figure 1. The algorithm.

Table 2. Computational results for 12 hospitals

j	$e_A(j)$	$e_B(j)$	$e_C(j)$	$e_D(j)$	$e_E(j)$	$e_F(j)$	$e_G(j)$	$e_H(j)$	$e_I(j)$	$e_J(j)$	$e_K(j)$	$e_L(j)$
A	0	-40	-35	-18	-24	0	-2	-10	10	10	57	30
B	40	0	5	22	16	40	38	30	50	50	97	70
C	35	-5	0	17	11	35	33	25	45	45	92	65
D	18	-22	-17	0	-6	18	16	8	28	28	75	48
E	24	-16	-11	6	0	24	22	14	34	34	81	54
F	0	-40	-35	-18	-24	0	-2	-10	10	10	57	30
G	2	-38	-33	-16	-22	2	0	-8	12	12	59	32
H	10	-30	-25	-8	-14	10	8	0	20	20	67	40
I	-10	-50	-45	-28	-34	-10	-12	-20	0	0	47	20
J	-10	-50	-45	-28	-34	-10	-12	-20	0	0	47	20
K	-57	-97	-92	-75	-81	-57	-59	-67	-47	-47	0	-27
L	-30	-70	-65	-48	-54	-30	-32	-40	-20	-20	27	0

Table 3. The optimal values and CPU times

j	z_j^*	CPU time (seconds)	δ_j^*	CPU time (seconds)
A	0	0.032	0	0.031
B	0	0.046	0	0.032
C	-25.25	0.046	286.62	0.031
D	0	0.033	0	0.032
E	-65.993	0.032	Inefficient	0
F	-41.375	0.047	95.26	0.03
G	0	0.033	0	0.031
H	-61.04	0.048	105.55	0.031
I	-25.002	0.033	4.067	0.03
J	0	0.032	0	0.012
K	0	0.016	0	0.012
L	0	0.016	Efficient	0

Note that the first two columns show the Additive optimal values and their CPU times and the last two columns give the proposed algorithm results. As it is shown the CPU times of the new algorithm is dominated by the CPU times of the Additive model. Also we have two zero CPU times in the alternative model. For efficient DMU_L, because $J_L = \phi$ and for inefficient DMU_E, because $j = A \in J_E$, $x_E \geq x_A$ & $y_A \geq y_E$. In fact the corresponding linear system (8) for DMU_E contains the following inconsistent equation

$$2\pi_1 + 7\pi_2 + 6\pi_3 + 24\pi_4 + \gamma_A = e_E(A) = -24$$

7. CONCLUSION REMARKS

Despite the large uses of inverse DEA models, there is not any single application of definition based of the inverse linear programming in DEA. Thus the contribution of this work is applying the inverse linear programming into DEA field to provide an alternative model to speed up the computations of the Additive model. The computational demonstration show that the CPU times of the alternative model is less than the existing Additive model.

REFERENCES

1. Ahuja, R.K. and Orlin J.B. (2001). Inverse optimization. *Operations Research*, 49: 771-783.
2. Amin, G.R. and Toloo, M. (2004). A polynomial-time algorithm for finding epsilon in DEA models. *Computers and Operations Research*, 31: 803-805.
3. Amin, G.R. and Emrouznejad, A. (2007). A note on DEA models in technology selection: An improvement for the Karsak and Ahiska's approach. *International Journal of Production Research*, 45(10): 2313-2316.
4. Banker, R.D., Charnes, A., and Cooper, W.W. (1984). Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management Science*, 9: 1078-1092.
5. Barr, R.S. and Durchholz, M.L. (1997). Parallel and hierarchical decomposition approaches for solving large-scale data envelopment analysis models. *Annals of Operations Research*, 73: 339-72.
6. Charnes, A., Cooper, W.W., and Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2: 429-444.
7. Cooper, W.W., Seiford, L.M., and Tone, K. (2006). *Introduction to Data Envelopment Analysis and its Uses with DEA-Solver Software and References*, Springer, New York.
8. Dulá, J.H. (2006). A computational study of DEA with massive data sets. *Computers and Operations Research*, (accepted).
9. Emrouznejad, A., Tavares, G., and Parker, B. (2007). A bibliography of data envelopment analysis (1978-2003). *Socio-Economic Planning Sciences*, (to appear).
10. Huang, S. and Liu, Z. (1999). On the inverse problem of linear programming and its application to minimum weight perfect *k*-matching. *European Journal of Operational Research*, 112: 421-426.
11. Pendharkar, P.C. (2002). A potential use of data envelopment analysis for the inverse classification problem. *Omega*, 30: 243-248.
12. Wei, Q., Zhang, J., and Zhang, X. (2000). An inverse DEA model for inputs/outputs estimate. *European Journal of Operational Research*, 121: 151-163.
13. Zhang J. and Liu, Z. (1996). Calculating some inverse linear programming problems. *Journal of Computational and Applied Mathematics*, 72: 261-273.
14. Zhu, J. (1999). Profitability and marketability of the top 55 U.S. commercial banks. *Management Science*, 45: 1270-1288.