Imprecise DEA via Standard Linear DEA Models with a Revisit to a Korean Mobile Telecommunication Company

Joe Zhu
Department of Management, Worcester Polytechnic Institute, Worcester, Massachusetts 01609, jzhu@wpi.edu

Data Envelopment Analysis (DEA) requires that the data for all inputs and outputs are known exactly. When some outputs and inputs are unknown decision variables, such as bounded and ordinal data, the DEA model becomes a nonlinear programming problem and is called imprecise DEA (IDEA). The nonlinear IDEA program can be converted into a linear program by an algorithm based upon scale transformations and variable alterations. Such an algorithm requires a set of special computational codes for each evaluation, because a different objective function and a different constraint with a set of new variables are present for each unit under evaluation. The current paper revisits a published Korean telecommunication analysis, and, by so doing, presents a new and simple approach to execute the IDEA through the standard linear DEA models. This greatly enhances the applicability of IDEA in applications, and the IDEA analysis is no longer limited to obtaining the efficiency scores. The key to the new approach lies in the finding that imprecise data can be easily converted into exact data. Based upon the exact data, models can be developed to determine all possible multiple optimal solutions in imprecise data, and to perform efficiency sensitivity analysis in IDEA.

Subject classifications: organization: employee performance evaluation; telecommunications: performance evaluation; effectiveness/performance: DEA.

Area of review: Decision Analysis.

History: Received November 2000; revisions received November 2001, November 2002; accepted December 2002.

1. Introduction

Data Envelopment Analysis (DEA) is a mathematical programming approach to performance evaluation (Cooper et al. 2000, Thanassoulis 2001, Zhu 2002). DEA requires that the data for all inputs and outputs are known exactly. When some outputs and inputs are imprecise data such as bounded or ordinal data, the DEA model becomes a nonlinear programming problem and is called imprecise DEA (IDEA). Kim et al. (1999) and Cooper et al. (1999) provide an IDEA algorithm based upon scale transformations and variable alterations to convert the nonlinear IDEA model into a linear problem when a mix of bounded, ordinal, and exact data are present. In the DEA literature, although ordinal relations have been studied (e.g., Dyson and Thanassoulis 1988; Cook et al. 1993, 1996; Golany 1988), some of the studies have focused on the output and input multipliers (weights).

We observe that when the IDEA algorithm of Kim et al. (1999) and Cooper et al. (1999) is used, the number of newly defined variables increases substantially, and a different objective function and a new constraint with a set of new variables are present for each decision-making unit (DMU) under evaluation. Thus, a set of special computational codes is required for each evaluation.

Cooper et al. (2001) demonstrate the use of their IDEA algorithm with an application to a Korean mobile telecommunication company. With a revisit to this company, the current paper shows that (i) the scale transformations and variable alterations in Kim et al. (1999) and Cooper et al. (1999, 2001) are unnecessary, (ii) imprecise data can be converted into exact data while preserving the efficiency ratings based upon the algorithm of Kim et al. (1999), Cooper et al. (1999, 2001), and (iii) the IDEA efficiency can be obtained via the standard linear DEA models. This paper investigates the discussions related to the procedure, computational results, and conclusions in Cooper et al. (2001). Finally, the paper develops new models which retrieve all possible alternative optima on imprecise data, and which improve on the Cooper et al. (2001) algorithm. Such models can also be used to perform sensitivity analysis for IDEA efficiency.

As a result of the current paper, a theoretical basis is established for some of the analyses and observations in Cooper et al. (2001), and a new and more practicable approach to Cooper et al. (2001) is available. Unlike the algorithm in Kim et al. (1999) and Cooper et al. (1999, 2001), the new approach can provide other efficiency results (e.g., returns to scale estimation, benchmarks, and paths for efficiency improvement) in addition to the efficiency scores.

2. Imprecise DEA

We first introduce the IDEA algorithm used in Cooper et al. (2001). Suppose we have a set of $n$ DMUs,
\{DMU_j : j = 1, 2, \ldots, n\}

where data can be expressed in the form of bounded data and ordinal data. The bounded multiplier model can be evaluated by the following CCR ratio model (Charnes, Cooper, Rhodes 1978)

\[
\max h_{\rho} = \frac{\sum_{i=1}^{s} u_i y_{\rho i}}{\sum_{j=1}^{m} v_j x_{\rho j}}
\]

\[
\text{s.t. } \sum_{i=1}^{s} u_i y_{ij} \leq 1 \quad \forall j
\]

\[
u_r, u_i \geq 0 \quad \forall r, i
\]

where, \(x_{io}\) and \(y_{ro}\) are the \(i\)th input and \(r\)th output for \(DMU_o\), respectively.

Model (1) can be transformed into the following CCR multiplier model

\[
\max \pi_o = \sum_{i=1}^{s} \pi_i y_{\rho i}
\]

\[
\text{s.t. } \sum_{i=1}^{s} \mu_i y_{ij} - \sum_{i=1}^{m} \omega_i x_{ij} = 0 \quad \forall j
\]

\[
\sum_{i=1}^{m} \omega_i x_{io} = 1
\]

\[
\mu_i, \omega_i \geq 0 \quad \forall r, i
\]

Model (2) becomes nonlinear if some outputs and inputs are unknown decision variables:

\[
\begin{align*}
\{ & x_{ij} \in D^{+}_i \\
& y_{ij} \in D^{-}_j \}
\end{align*}
\]

where \(D^+_i\) and \(D^-_j\) are data sets containing imprecise data in forms of bounded data and ordinal data. The bounded data can be expressed as

\[
\tilde{y}_{ij} \leq y_{ij} \leq \bar{y}_{ij}\quad \text{and}\quad \tilde{x}_{ij} \leq x_{ij} \leq \bar{x}_{ij}\quad \text{for } r \in BO, i \in BI
\]

where \(\tilde{y}_{ij}\) and \(\bar{y}_{ij}\) are the lower bounds and \(\tilde{x}_{ij}\) and \(\bar{x}_{ij}\) are the upper bounds, and \(BO\) and \(BI\) represent the associated sets containing bounded outputs and inputs, respectively.

The ordinal data can be expressed as

\[
\tilde{y}_{ij} \leq y_{ij} \leq \bar{y}_{ij}\quad \text{and}\quad \tilde{x}_{ij} \leq x_{ij} \leq \bar{x}_{ij}\quad \text{for } r \in DO, i \in DI
\]

or to simplify the presentation,

\[
\begin{align*}
y_{r1} \leq y_{r2} \leq \cdots \leq y_{rk} \leq \cdots \leq y_{rn} \quad (r \in DO) \\
x_{i1} \leq x_{i2} \leq \cdots \leq x_{ik} \leq \cdots \leq x_{in} \quad (i \in DI)
\end{align*}
\]

where \(DO\) and \(DI\) represent the associated sets containing ordinal outputs and inputs, respectively.

Based on scale transformations and variable alterations, the nonlinear DEA model (2) with (4), (5), and (6) can be transformed into a linear programming problem

\[
\begin{align*}
\max & \sum_{r} Y_{ro} \\
\text{s.t. } \sum_{r} Y_{rj} - \sum_{i} X_{ij} & \leq 0 \quad \forall j \\
\sum_{i} X_{io} & = 1 \\
X_{r} = (X_{gr}) & \in B^{+} \quad \forall r \quad Y_{r} = (Y_{gr}) \in B^{-} \quad \forall r \quad X_{io}^{o}, Y_{ro}^{o} \geq 0 \quad \forall i, r
\end{align*}
\]

where \(X_{ij} = \tilde{x}_{ij}\hat{\omega}_i, Y_{ij} = \tilde{y}_{ij}\mu_r, \hat{\omega}_i \geq \omega_i \cdot \max\{\tilde{x}_{ij}\}, \hat{\mu}_r \geq \mu_r \cdot \max\{\tilde{y}_{ij}\}, \tilde{x}_{ij} = \hat{x}_{ij}/\mu_r, \hat{\mu}_r = \mu_r/\max\{\tilde{y}_{ij}\}, X_{ij} = \tilde{x}_{ij}^{o}\hat{\omega}_i, Y_{ij} = \tilde{y}_{ij}^{o}\mu_r, \tilde{x}_{ij}^{o} = \max\{\tilde{x}_{ij}\}, \mu_r^{o} = \mu_r/\max\{\tilde{y}_{ij}\}, \) and \(D^{+}_i\) and \(D^{-}_j\) are transformed into \(B^{+}\) and \(B^{-}\), respectively.

It is obvious from model (7) that the number of newly defined variables \((Y_{ij} \text{ and } X_{ij}) \) increases substantially as the number of \(DMU\)s increases. The standard linear DEA models cannot be used, and a set of special computational codes is needed for each evaluation. Note that a different objective function \((\sum_{r} Y_{ro})\) and a new constraint \((\sum_{i} X_{io})\) with a set of new variables are present in model (7) for every \(DMU\) under evaluation.

### 3. New IDEA Procedure

The following theorem reveals IDEA's evaluation mechanism, and in turn provides the foundation for our new and simple procedure in executing IDEA without the need to apply the scale transformations and variable alternations and model (7).

**Theorem 1.** Suppose for \(DMU_o\), \(h_{\rho}^o\) is the optimal value to (1) with (3), and we have a set of optimal values of \(y_{ij}^{*} \in D^{+}_j\) and \(x_{ij}^{*} \in D^{-}_i\) and \(u_{ir}^{*}, v_{ij}^{*}\), then \(h_{\rho}^o\) remains the same:

(i) if \(\tilde{x}_{io} < x_{io}^{*}\) and \(\tilde{x}_{io} \in D^{-}_io\); (ii) if \(\tilde{y}_{ro} > y_{ro}^{*}\) and \(\tilde{y}_{ro} \in D^{+}_ro\); (iii) if \(\tilde{x}_{ij} > x_{ij}^{*}\) \((j \neq o)\) and \(\tilde{x}_{ij} \in D^{-}_ij\); (iv) if \(\tilde{y}_{ij} < y_{ij}^{*}\) \((j \neq o)\) and \(\tilde{y}_{ij} \in D^{+}_ij\).

Theorem 1 allows one to convert imprecise data into exact data. For example, when bounded data in (4) are present, we set \(y_{ro} = \bar{y}_{ro}\) and \(x_{io} = \tilde{x}_{io}\) for \(DMU_o\), and \(y_{ij} = Y_{rj}\) and \(x_{ij} = \bar{x}_{ij}\) for \(DMU_j\) \((j \neq o)\). Consequently, the IDEA algorithm and the related model (7) can be replaced by the linear CCR multiplier model (2) or the dual program.
to (2) with (obtained) exact data:
\[
\theta^*_o = \min \theta_o \quad \text{s.t.} \quad \sum_{j \neq o} \lambda_j \tilde{x}_{ij} + \lambda_o \tilde{x}_{io} \leq \theta_o \tilde{x}_{io} \quad i \in BI; \\
\sum_{j=1}^n \lambda_j x_{ij} \leq \theta_o x_{io} \quad i \notin BI; \\
\sum_{j \neq o} \lambda_j \tilde{y}_{rj} + \lambda_o \tilde{y}_r \geq \tilde{y}_r \quad r \in BO; \\
\sum_{j=1}^n \lambda_j y_{rj} \geq y_{ro} \quad r \notin BO; \\
\lambda_j \geq 0 \quad j = 1, 2, \ldots, n
\]

where \(y_{rj} (r \notin BO)\), and \(x_{ij} (i \notin BI)\) are exact data.

From Theorem 1, we have

**Corollary 1.** Suppose for \(DMU_o\), \(h^*_o\) is the optimal value to (1) when (some) outputs and inputs are only known to be within specific bounds given by (4). This \(h^*_o\) remains the same:

(i) if \(y_{ro}\) is decreased for \(DMU_o\);
(ii) if \(\tilde{x}_{io}\) is increased for \(DMU_o\);
(iii) if \(\tilde{y}_{rj}\) is increased for \(DMU_j (j \neq o)\);
(iv) if \(\tilde{x}_{ij}\) is increased for \(DMU_j (j \neq o)\).

Next, we develop a procedure to convert weak ordinal data into exact data through bounded data. Consider \(DMU_k\). Suppose we solve model (1) with (5) and (6) and obtain a set of optimal solutions with \(h^*_k\) such that

\[
y^*_1 \leq y^*_2 \leq \cdots \leq y^*_{r,k-1} \leq y^*_r \leq \cdots \leq y^*_n \\
(\text{in } DO) \quad (9)
\]

\[
x^*_1 \leq x^*_2 \leq \cdots \leq x^*_{i,k-1} \leq x^*_i \leq \cdots \leq x^*_n \\
(\text{in } DI). \quad (10)
\]

Note that \(\rho y^*_r (r \in DO)\) and \(\rho x^*_i (i \in DI)\) are also optimal for \(DMU_k\) where \(\rho\) is a positive constant. Therefore, we can arbitrarily set \(y^*_r = x^*_k = 1\). Note also that the ordinal relations usually can be expressed within a range, e.g., \([0, M]\), where \(M\) is the number of DMUs. If we do not have such a range, we can always set \(M\) sufficiently large.

Then, we can have a set of optimal solutions on ordinal outputs and inputs such that (9) and (10) can be expressed as

\[
0 \leq y^*_1 \leq y^*_2 \leq \cdots \leq y^*_{r,k-1} \leq y^*_r (= 1) \\
\leq \cdots \leq y^*_n \leq M \quad (r \in DO) \quad (11)
\]

\[
0 \leq x^*_1 \leq x^*_2 \leq \cdots \leq x^*_{i,k-1} \leq x^*_i (= 1) \\
\leq \cdots \leq x^*_n \leq M \quad (i \in DI). \quad (12)
\]

Now, for the ordinal outputs and inputs in \(DMU_j (j \neq k)\), we set up the following bounded data with respect to (11) and (12).

\[
y_{rj} \in [0, 1] \quad \text{and} \quad x_{ij} \in [0, 1]
\]

for \(DMU_j (j = 1, \ldots, k - 1) \quad (13)

\[
y_{rj} \in [1, M] \quad \text{and} \quad x_{ij} \in [1, M]
\]

for \(DMU_j (j = k + 1, \ldots, n) \quad (14)

Based upon Theorem 1, we know that for \(r \in DO\) and \(i \in DI\), \(h^*_o\) remains the same, and (11) and (12) are satisfied if \(y_{rj} = x_{ij} = 1\) for \(DMU_k\) and \(y_{rj} = 0, x_{ij} = 1\) for \(DMU_j (j = 1, \ldots, k - 1)\), and \(y_{rj} = 1, x_{ij} = M\) for \(DMU_j (j = k + 1, \ldots, n)\). In this manner, we have successfully converted the ordinal data into a set of exact data.

We now apply the newly developed IDEA procedure to the eight branch offices of the Korean mobile telecommunications company in Cooper et al. (2001). Table 1 presents the data where the third input (\(x_3\)) is an ordinal input and the third output (\(y_3\)) is a bounded output.

For the ordinal input \(x_3\), we first set \(M\) equal to 8 which is the number of DMUs (branch offices), and then use (13) and (14) to generate a set of exact data for each DMU under evaluation. Table 2 reports the exact data where the DMUs in the first row represent DMUs under evaluation.

<table>
<thead>
<tr>
<th>DMU (j)</th>
<th>(x_1): Manpower (num.)</th>
<th>(x_2): Operating cost (mill. $)</th>
<th>(x_3): Management level (rank)</th>
<th>(y_1): Revenue (mill. $)</th>
<th>(y_2): Facility success rate (%)</th>
<th>(y_3): (ratio %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seoul (1)</td>
<td>124</td>
<td>18.22</td>
<td>4</td>
<td>25.53</td>
<td>89.8</td>
<td>[80, 85]</td>
</tr>
<tr>
<td>Pusan (2)</td>
<td>95</td>
<td>9.23</td>
<td>2</td>
<td>18.43</td>
<td>99.6</td>
<td>[85, 90]</td>
</tr>
<tr>
<td>Taegu (3)</td>
<td>92</td>
<td>8.07</td>
<td>6</td>
<td>10.29</td>
<td>87.0</td>
<td>[75, 80]</td>
</tr>
<tr>
<td>Kwangju (4)</td>
<td>61</td>
<td>5.62</td>
<td>8</td>
<td>8.32</td>
<td>99.4</td>
<td>100</td>
</tr>
<tr>
<td>Taejun (5)</td>
<td>63</td>
<td>5.33</td>
<td>7</td>
<td>7.04</td>
<td>96.4</td>
<td>[70, 75]</td>
</tr>
<tr>
<td>Jeonju (6)</td>
<td>50</td>
<td>3.53</td>
<td>3</td>
<td>6.42</td>
<td>86.0</td>
<td>[90, 95]</td>
</tr>
<tr>
<td>Kangnung (7)</td>
<td>40</td>
<td>3.50</td>
<td>5</td>
<td>2.20</td>
<td>71.0</td>
<td>[80, 85]</td>
</tr>
<tr>
<td>Jeju (8)</td>
<td>16</td>
<td>1.17</td>
<td>1</td>
<td>2.87</td>
<td>98.0</td>
<td>[95, 100]</td>
</tr>
</tbody>
</table>

Note. The data in this column only reflect the relation \(x_{34} \geq x_{35} \geq \cdots \geq x_{32} \geq x_{38}\).

For example, the data under “DMU1” are the exact data for all 8 DMUs when DMU1 is under evaluation.

Table 3 reports the results obtained from model (8). The second column presents the efficiency scores which are the same efficiency ratings based upon Cooper et al. (2001) with \( \epsilon = 0 \). The nonzero slacks in DMUs 3, 5, and 7 indicate that the efficiency ratings for these DMUs would decrease if a positive \( \epsilon \) is used in Cooper et al. (2001). This is verified in DMUs 5 and 7 when \( \epsilon = 10^{-3} \).

Finally, note that unlike the Cooper et al. (2001) algorithm, the current IDEA approach can provide efficiency information, such as returns to scale estimation and benchmarks (referent DMUs) (see Table 3), in addition to the efficiency scores.

4. Multiple Optima and Data Retrieval

Note that optimal imprecise output/input values can be difficult to obtain from model (7), because of the variable alternations. Cooper et al. (2001) are able to retrieve one set of optimal solutions on the bounded output (\( y_3 \)) for each DMU under evaluation, and perform a simulation type sensitivity analysis to study the influence of upper and lower bounds on efficiency ratings. By using Theorem 1, we can develop models to determine all possible multiple optimal solutions on the bounded data so that all solutions can be retrieved for model (7). Furthermore, we can provide a theoretical basis for the observations and conclusions regarding the relationship between bounds and efficiency ratings in Cooper et al. (2001).

Recall that Corollary 1 indicates that if one increases the upper input bound(s) or decreases the lower output bound(s) for a DMU under evaluation, the efficiency score remains the same. However, if one changes the lower input bound(s) or the upper output bound(s) for an inefficient DMU, the efficiency rating is likely to change.

In fact, we can establish the following model to study the efficiency sensitivity with respect to upper output bound changes in inefficient DMUs,

\[
\max \psi_o
\]

\[\text{s.t. } \sum_{j=1}^{n} \lambda_j^o y_{ij} + \lambda_{o} y_{ro} \geq \psi_o \bar{y}_{ro} \quad r \in BO\]

\[\sum_{j=1}^{n} \lambda_j y_{ij} \geq y_{ro} \quad r \notin BO\]

\[\sum_{j=1}^{n} \lambda_j \hat{x}_{ij} \leq \bar{x}_{io} \quad i = 1, \ldots, m\]

\[\lambda_j \geq 0 \quad j = 1, \ldots, n\]

where \( \hat{x}_{ij} \) represents the exact data.

Applying model (15) to DMU7, we have \( \psi_o = 1.11765 \), indicating that if the upper bound of \( y_3 \) is increased to \( \psi_o \bar{y}_{37} \approx 95 \), DMU7 will have a unity efficiency score. This

### Table 2. Exact data for the imprecise output and input.

<table>
<thead>
<tr>
<th>DMU</th>
<th>Name</th>
<th>( x_1 )</th>
<th>( y_1 )</th>
<th>( x_2 )</th>
<th>( y_2 )</th>
<th>( x_3 )</th>
<th>( y_3 )</th>
<th>( x_4 )</th>
<th>( y_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Seoul</td>
<td>1</td>
<td>85</td>
<td>8</td>
<td>80</td>
<td>1</td>
<td>80</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>Pusan</td>
<td>1</td>
<td>85</td>
<td>1</td>
<td>90</td>
<td>1</td>
<td>85</td>
<td>1</td>
<td>85</td>
</tr>
<tr>
<td>3</td>
<td>Taegu</td>
<td>8</td>
<td>75</td>
<td>8</td>
<td>75</td>
<td>1</td>
<td>80</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>4</td>
<td>Kwangju</td>
<td>8</td>
<td>100</td>
<td>8</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>Taejun</td>
<td>8</td>
<td>80</td>
<td>8</td>
<td>70</td>
<td>1</td>
<td>70</td>
<td>1</td>
<td>70</td>
</tr>
<tr>
<td>6</td>
<td>Jeonju</td>
<td>1</td>
<td>90</td>
<td>8</td>
<td>90</td>
<td>1</td>
<td>90</td>
<td>1</td>
<td>90</td>
</tr>
<tr>
<td>7</td>
<td>Kangnung</td>
<td>8</td>
<td>80</td>
<td>8</td>
<td>80</td>
<td>1</td>
<td>80</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>8</td>
<td>Jeju</td>
<td>1</td>
<td>95</td>
<td>1</td>
<td>95</td>
<td>1</td>
<td>95</td>
<td>1</td>
<td>95</td>
</tr>
</tbody>
</table>

*The DMU under evaluation.

### Table 3. Efficiency results for the telecommunication branch offices.

<table>
<thead>
<tr>
<th>DMU (j)</th>
<th>Efficiency</th>
<th>Referent DMUs</th>
<th>Returns to Scale</th>
<th>Slacks**</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seoul (1)</td>
<td>1</td>
<td>( \Lambda_1^o = 1 )</td>
<td>constant</td>
<td>( x_1 )</td>
</tr>
<tr>
<td>Pusan (2)</td>
<td>1</td>
<td>( \Lambda_2^o = 1 )</td>
<td>constant</td>
<td>0</td>
</tr>
<tr>
<td>Taegu (3)</td>
<td>0.89435</td>
<td>( \Lambda_3^o = 0.496, \Lambda_4^o = 0.398 )</td>
<td>increasing</td>
<td>28.76</td>
</tr>
<tr>
<td>Kwangju (4)</td>
<td>1</td>
<td>( \Lambda_5^o = 1 )</td>
<td>constant</td>
<td>0</td>
</tr>
<tr>
<td>Taejun (5)</td>
<td>0.97545</td>
<td>( \Lambda_6^o = 0.503, \Lambda_7^o = 0.472 )</td>
<td>increasing</td>
<td>6.07</td>
</tr>
<tr>
<td>Jeonju (6)</td>
<td>1</td>
<td>( \Lambda_8^o = 1 )</td>
<td>constant</td>
<td>0</td>
</tr>
<tr>
<td>Kangnung (7)</td>
<td>0.89474</td>
<td>( \Lambda_9^o = 0.895 )</td>
<td>increasing</td>
<td>21.47</td>
</tr>
<tr>
<td>Jeju (8)</td>
<td>1</td>
<td>( \Lambda_1^o = 1 )</td>
<td>constant</td>
<td>0</td>
</tr>
</tbody>
</table>

Note. *Model (8) is used.

**No slacks are present on the third input (\( x_3 \)).
result is consistent with the sensitivity analysis performed by Cooper et al. (2001). The second and third columns of the Table 4 report the sensitivity analysis results for the three inefficient DMUs 3, 5, and 7 with the third column indicating the new upper output bound such that an inefficient DMU becomes efficient.

The following model can be used to determine the allowable changes in bounded data across all DMUs so that an efficient \(DMU_o\) under evaluation remains efficient.

\[
\begin{align*}
\text{max} \quad & \phi_o \\
\text{s.t.} \quad & \sum_{j=1}^{n} \lambda_j y_{ij} \geq \phi_o y_{io}, \quad r \in BO \\
& \sum_{j=1}^{n} \lambda_j y_{ij} \geq y_{io}, \quad r \notin BO \tag{16} \\
& \sum_{j=1}^{n} \lambda_j \hat{x}_{ij} \leq \hat{x}_{io}, \quad i = 1, \ldots, m \\
& \lambda_j \geq 0 \quad (j \neq o)
\end{align*}
\]

where \(\hat{x}_{ij}\) represents the exact data.

Note that for an efficient \(DMU_o\) under evaluation, increasing its upper output bounds or decreasing the lower output bounds for the remaining DMUs will not deteriorate the efficiency of \(DMU_o\). Thus, based upon Corollary 1, we should only consider possible decreases on upper output bound(s) for \(DMU_o\) and possible increases on lower output bound(s) for the remaining DMUs. That is, we consider \(y_{ij} = \pi y_{ij} \quad (j \neq o)\) and \(y_{io} = \pi y_{io}\), where \(\pi \geq 1\) and \(0 < \pi \leq 1\). We next determine all possible values on \(\pi\) and \(\pi\) that maintain the efficiency status of \(DMU_o\).

Based upon Seiford and Zhu (1998a), either the optimal value to (16) \(\phi^*_o < 1\) or model (16) is infeasible. Furthermore, based upon Seiford and Zhu (1998b) and Zhu (2001), if model (16) is infeasible, then for any \(\pi \geq 1\) and \(\pi \leq 1\), \(DMU_o\) remains efficient. That is, when infeasibility occurs, \(DMU_o\)'s efficiency is not affected by any changes in the output bounds across all DMUs.

If model (16) is feasible, then we have

**Theorem 2.** Suppose \(\phi^*_o\) is the optimal value to (16), then for \(r \in BO\)

(i) if \(\pi^* / \pi > \phi^*_o\), then \(DMU_o\) remains extreme-efficient;

(ii) if equality holds for \(\pi^* / \pi = \phi^*_o\), i.e., \(\phi^*_o \leq \pi^* / \pi \leq 1\), then \(DMU_o\) remains on the frontier (\(DMU_o\) has a unity efficiency score);

(iii) if \(\pi^* / \pi < \phi^*_o\) then \(DMU_o\) will not be extreme-efficient.

Theorem 2 indicates that \(DMU_o\) maintains a unity efficiency score as long as \(\pi^* / \pi \geq \phi^*_o\).

Since DMU4 has the exact value on \(y_i\), this DMU is not considered in the current study. We now consider the bounded output \(y_i\) for the remaining four efficient DMUs 1, 2, 6, and 8. The fourth and fifth columns of Table 4 summarize the results from model (16). Infeasibility of model (16) is associated with \(\phi^*_i\) (DMU1), \(\phi^*_2\) (DMU2) and \(\phi^*_6\) (DMU8), indicating that any change occurred in this bounded output across all eight DMUs does not change the efficiency rating of each of these DMUs. That is, any value within the bounds is optimal when DMUs 1, 2, and 8 are under evaluation by model (7) in Cooper et al. (2001).

For DMU6, we have \(\phi^*_6 = 0.97598\), i.e., DMU6 maintains a unity efficiency score if \(\pi^* / \pi \geq 0.97598\). Suppose \(\pi = 1\), i.e., the lower output bounds in other DMUs are fixed, then \(\pi^*\) must be greater than or equal to 0.97598 in order for DMU6 to maintain a unity efficiency score. Note that \(\bar{y}_{36}/\bar{y}_{36} = 90/95 \approx 0.94737 < \phi^*_6\). Thus, if the \(y_i\) of DMU6 is set equal to the lower bound of 90, DMU6 would be inefficient. In fact, the upper bound of DMU6 can only be decreased to \(95 \phi^*_6 \approx 93\). Through this type of sensitivity analysis, we are able to determine all possible alternative optimal solutions in imprecise data when the Cooper et al. (2001) algorithm is used. Furthermore, we can have sound evidence on the influence of upper and lower bounds on efficiency ratings.

### 5. Conclusions

The Korean telecommunication analysis in Cooper et al. (2001) is revisited with a new approach employing standard linear DEA models in dealing with imprecise data. The current paper demonstrates that bounded data and weak ordinal data can be converted into exact data. As a result, existing DEA methods and techniques can be employed. This increases the IDEA power in applications.

Cooper et al. (2001) point out that their optimization and algorithm tend to let the variables accumulate at lower and upper bounds. This is justified by Theorem 1. Some of the observations and conclusions in Cooper et al. (2001) are also verified by models and theorems developed for determining all possible optimal solutions in bounded data.

Theorem 1 can also be used to convert strong ordinal data and ratio bounded data as described in Kim et al. (1999) into exact data. Nonlinearity caused by any forms
of imprecise data can be easily resolved by the current new IDEA procedure based upon Theorem 1.

We do not discuss here the application of assurance regions (ARs) (Thompson et al. 1996), since incorporating ARs into the IDEA becomes a standard DEA process based upon the CCR multiplier model. Also, as indicated in Zhu (2003), AR constraints can become redundant when data are in ordinal relations.

Appendix. Proofs of Theorems 1 and 2

Proof of Theorem 1. (i) With the given set of optimal solutions, we have

\[ h_o^* = \frac{\sum_{r \in d^+} u^*_r y^*_r + \sum_{r \in d^-} u^*_r y^*_r}{\sum_{i \in d^+} x^*_i v^*_i + \sum_{i \in d^-} x^*_i v^*_i} \quad \text{and} \]
\[ \frac{\sum_{i \in d^+} v^*_i x^*_i + \sum_{i \in d^-} v^*_i x^*_i}{\sum_{r \in d^+} y^*_r + \sum_{r \in d^-} y^*_r} \leq 1 \quad \forall j \]

where, \( d^+ \) and \( d^- \) are index sets corresponding to \( D^+ \) and \( D^- \), respectively.

Now, on the basis of this set of optimal solutions, we define

\[ \tilde{u}_i = u^*_i, \quad \tilde{v}_j = v^*_j x^*_i \quad (i \in d^+) \quad \text{and} \quad \tilde{v}_i = v^*_i \quad (i \notin d^+) \]

Since \( \tilde{x}_{i_0} < x^*_i \), \( \tilde{v}_j > v^*_j \) \( (i \in d^-) \). Thus,

\[ \frac{\sum_{r \in d^+} \tilde{u}_r y^*_r + \sum_{r \in d^-} \tilde{u}_r y^*_r}{\sum_{i \in d^+} \tilde{v}_i x^*_i + \sum_{i \in d^-} \tilde{v}_i x^*_i} \leq \frac{\sum_{i \in d^+} v^*_i x^*_i + \sum_{i \in d^-} v^*_i x^*_i}{\sum_{r \in d^+} y^*_r + \sum_{r \in d^-} y^*_r} \quad \text{for} \quad j \notin o \]

and

\[ \frac{\sum_{r \in d^+} \tilde{u}_r y^*_r + \sum_{r \in d^-} \tilde{u}_r y^*_r}{\sum_{i \in d^+} \tilde{v}_i x^*_i + \sum_{i \in d^-} \tilde{v}_i x^*_i} = \frac{\sum_{i \in d^+} v^*_i x^*_i + \sum_{i \in d^-} v^*_i x^*_i}{\sum_{r \in d^+} y^*_r + \sum_{r \in d^-} y^*_r} = h_o^*. \]

Therefore, \( \tilde{u}_r, \tilde{v}_j, \tilde{v}_i, \tilde{x}_{i_0} \) are also optimal. This completes the proof of (i).

The proofs of (ii), (iii), and (iii) are similar to that of (i) and therefore are omitted.

Proof of Theorem 2 (Proof). (i) Suppose \( \pi' / \pi > \phi^*_o \), and \( DMU_o \) is not extreme-efficient when \( y_{rj} = \pi y_{rj} \) \( (j \neq o) \) and \( y_{ro} = \pi y_{ro} \) \( r \in BO \). Then, there exist \( \lambda_j \) \( (j \neq o) \) \( \geq 0 \) and \( \Phi^* \geq 1 \) in the following linear program (Thrall 1996):

\[ \Phi^* = \max \Phi \]
\[ \text{s.t.} \sum_{j \neq o}^{n} \lambda_j \hat{x}_{ij} \leq \hat{x}_{i_0} \quad i = 1, \ldots, m \quad (A1) \]
\[ \sum_{j \neq o}^{n} \lambda_j \hat{y}_{rj} \geq \Phi \hat{y}_{ro} \quad r = 1, \ldots, s \]
\[ \Phi, \lambda_j \quad (j \neq o) \geq 0 \]

where (*) represents the exact data obtained from imprecise data or the original exact data.

We have

\[ \sum_{j \neq o}^{n} \lambda_j (\pi y_{rj}) \geq \Phi \pi \hat{y}_{ro} \quad r \in BO \]
\[ \sum_{j \neq o}^{n} \lambda_j y_{rj} \geq \Phi y_{ro} \quad r \notin BO \]
\[ \sum_{j \neq o}^{n} \lambda_j x_{i_0} \leq y_{i_0} \quad 1, 2, \ldots, s. \]

This means that \( \lambda_j \) \( (j \neq o) \) \( \geq 0 \) and \( \phi^*_o = \Phi^*(\pi' / \pi) \) is a feasible solution to (16). But \( \Phi^*(\pi' / \pi) > \Phi^* \phi^*_o \), violating the optimality of \( \phi^*_o \). Thus, if \( \pi' / \pi > \phi^*_o \), then \( DMU_o \) remains extreme-efficient.

(ii) If \( \pi' / \pi = \phi^*_o \), then we assume \( DMU_o \) is not a frontier DMU when \( y_{rj} = \pi y_{rj} \) \( (j \neq o) \) and \( y_{ro} = \pi \hat{y}_{ro} \) \( r \in BO \) Thus, we have \( \Phi^* > 1 \) in (A1). Now we have \( \Phi^*(\pi' / \pi) > \Phi^* \phi^*_o \), violating the optimality of \( \phi^*_o \). Thus, if \( \phi^*_o < \pi' / \pi \leq 1 \), then \( DMU_o \) remains on the frontier.

(iii) The proof is similar to that of (i) and (ii) and therefore is omitted.

Endnotes

1. Theorem 1 is true regardless of the forms of imprecise data.

2. Note that in order to use model (7), one DMU must have the (maximum) exact data on \( y_j \) if one uses the algorithm in Cooper et al. (2001). The newly proposed IDEA approach does not have such a requirement.

3. The efficiency scores obtained from the Cooper et al. (2001) algorithm depend on the selection of \( e \). See Ali and Seiford (1993) for problems associated with treating \( e \) as a real number.


5. Similar results can be obtained if bounded inputs are considered.

6. \( y_{3j} = \pi y_{3j} \) \( (j \neq o) \) and \( y_{3o} = \pi \hat{y}_{3o} \).

Acknowledgments

The author is grateful to the comments and suggestions made by four anonymous referees, an associate editor, and the area editor, James Dyer.

References


