J. Soc. Korea Ind. Syst. Eng Vol. 38, No. 4 : 117-131, December 2015
http://dx.doi.org/10.11627/jkise.2015.38.4.117
ISSN : 2005-0461(print)
ISSN : 2287-7975(online)

Data Envelopment Analysis with Imprecise Data Based on Robust Optimization

Sungmook Lim†

Dongguk Business School, Dongguk University, Seoul

부정확한 데이터를 가지는 자료포락분석을 위한 로버스트 최적화 모형의 적용

임 성 묵†

동국대학교 서울캠퍼스 경영대학

Conventional data envelopment analysis (DEA) models require that inputs and outputs are given as crisp values. Very often, however, some of inputs and outputs are given as imprecise data where they are only known to lie within bounded intervals. While a typical approach to addressing this situation for optimization models such as DEA is to conduct sensitivity analysis, it provides only a limited ex-post measure against the data imprecision. Robust optimization provides a more effective ex-ante measure where the data imprecision is directly incorporated into the model. This study aims to apply robust optimization approach to DEA models with imprecise data. Based upon a recently developed robust optimization framework which allows a flexible adjustment of the level of conservatism, we propose two robust optimization DEA model formulations with imprecise data; multiplier and envelopment models. We demonstrate that the two models consider different risks regarding imprecise efficiency scores, and that the existing DEA models with imprecise data are special cases of the proposed models. We show that the robust optimization for the multiplier DEA model considers the risk that estimated efficiency scores exceed true values, while the one for the envelopment DEA model deals with the risk that estimated efficiency scores fall short of true values. We also show that efficiency scores stratified in terms of probabilistic bounds of constraint violations can be obtained from the proposed models. We finally illustrate the proposed approach using a sample data set and show how the results can be used for ranking DMUs.

Keywords : Data Envelopment Analysis, Robust Optimization, Imprecise Data, Linear Optimization

1. 서 론

자료포락분석(Data Envelopment Analysis, DEA)은 다수의 투입요소와 산출요소를 가지는 의사결정단위(Decision Making Unit, DMU)들의 상대적 효율성을 측정하기 위한 선형최적화 기반의 방법으로서, Charnes et al.[4]에 의해 개발된 이후로 수많은 이론적 연구 및 실무적 활용이 이루어져 왔다[5, 9].

DEA 모형이 가지는 특징 중의 하나는 DMU들에 관한 관찰 데이터에 기반하여 실증적으로 생산가능집합을 추정한다는 점이다[8]. DEA가 가지는 데이터 기반 실증적 모형이라는 특징은 생산함수에 대한 가정을 요구하지 않는다는 장점으로 작용하는 동시에 데이터의 불확실성, 측정오류 등에 의해 생기는 변동에 큰 영향을 받는다는 단점으로 나타나기도 한다[18]. DEA가 데이터의 경향성을 추정하는 회귀 모형과는 달리 효율적 경계선에 기반하여

Received 1 October 2015; Finally Revised 11 November 2015; Accepted 3 December 2015
† Corresponding Author : sungmook@dongguk.edu

효율성을 추정하는 모형이라는 점에서 데이터 변동에 따른 영향은 더욱 크게 나타난다.

기본적인 DEA 모형에서는 투입요소 및 산출요소의 정확한 값이 주어지는 것을 가정한다. 하지만 앞서 언급한 대로 DEA 모형의 효율성 계산결과는 데이터의 변동에 크게 영향을 받게 되므로, 투입요소 및 산출요소 값이 부정확한 경우 이에 따른 변동성을 고려하는 것이 모형의 실무적 활용도를 높이기 위해 필요하다. DEA와 같은 최적화 수리 모형에서 데이터의 부정확성을 고려하기 위해 일반적으로 취하는 방법은 민감도 분석(sensitivity analysis)을 수행하는 것이다[10, 15]). 즉, 고정적인 투입요소 및 산출요소의 값을 가정하여 DEA 모형을 풀어 효율성 점수를 계산한 후, 투입요소 및 산출요소의 변화에 따른 효율성 점수의 영향관계를 분석하는 방식으로 데이터의 부정확성을 감안하는 방법이다. 그러나 민감도 분석은 사후적인 분석 방법이라는 점에서 한계가 존재하며, 보다 직접적으로 사전적인 수단으로 데이터의 부정확성을 DEA 모형 내에 도입하여 분석하는 방법이 필요하다.

데이터의 부정확성을 직접적으로 DEA 모형 내에 포함시켜 분석하는 방법으로 가장 대표적인 것이 Cooper et al.[6]의 IDEA(Imprecise DEA) 모형이다. IDEA 모형에서는 투입요소 및 산출요소의 값이 상하한 값을 가지는 구간(interval)의 형태로 표현되거나 요소들 간 순위 관계의 형태로 표현되는 경우를 가정한다. 데이터 값들을 변수의 형태로 표현함으로써 IDEA 모형은 비선형 최적화 모형이 되고, Cooper et al.[6]는 척도변환(scale transformation)과 변수치환(variable alternation)을 통해 이를 선형최적화 모형으로 변환한다.

Despotis and Smirlis[7]는 복잡한 척도변환과 변수치환 대신에 간단한 변수치환만으로도 IDEA 모형을 선형화할 수 있음을 보였고, 이와 더불어 주어진 데이터 부정확성 하에서 효율성 점수의 상하한을 도출할 수 있음도 보였다. Zhu[19, 20]도 이와 유사하게 비선형 최적화 모형인 IDEA 모형이 사실상 간단한 변수치환을 통해 선형최적화 모형으로 변환할 수 있다는 사실을 보인 바 있다.

한편, 일반적인 최적화 문제에서 데이터의 부정확성(불확실성)을 처리하는 방법 중 하나로 최근 들어 많은 연구가 이루어지고 있는 분야는 로버스트 최적화이다. 로버스트 최적화는 주어진 문제를 구성하는 데이터의 불확실성이 최악의 경우로 발현되는 경우에 대해 가장 최선의 해를 찾는 방법으로서, 데이터 불확실성 내의 어떠한 경우에라도 제약조건을 만족시킬 수 있는 해 중에서 가장 목적함수의 값을 최적으로 만드는 해를 찾는 것을 기본 개념으로 한다. 선형최적화에서의 로버스트 최적화 모형을 처음 제안한 연구는 Soyster[17]인데, 제약식 각 열벡터의 불확실성이 상호 독립적인 볼록집합의 형태로 주어지는 경우를 고려하였으며 선형최적화의 형태로 모형화할 수 있음을 보였다. 그러나 이러한 로버스트 최적화 접근법은 지나친 보수성(conservatism)을 가진다는 단점이 있는데, 도출되는 해가 어떠한 경우에라도 가능해가 되는 강건성(robustness)은 가지지만 데이터의 명목값(nominal values)에 대한 해당 해의 최적성은 심각하게 저해되는 경향이 있다. 해의 강건성이 높은 수준으로 요구되는 제어시스템 등의 응용에서는 로버스트 최적화가 유용하지만, 대부분의 경영과학 관련 의사결정문제에서는 그렇지 않기 때문에 로버스트 최적화의 활용이 제한적이었다.

Ben-Tal and Nemirovski[2]는 상기와 같은 로버스트 최적화의 과도한 보수성을 일정 부분 완화하기 위해 선형최적화 문제에서의 데이터 불확실성을 타원체의 형태로 모형화하는 로버스트 최적화 모형을 제안하였고, 볼록최적화 문제의 일종인 원추 이차계획법 문제로 변환되어 효율적인 내부점 해법의 적용이 가능해 짐을 보인 바 있다. 이 모형에서는 데이터 불확실성을 나타내는 타원체의 크기를 조절함으로써 해의 강건성과 최적성 간의 균형을 탄력적으로 조정할 수 있도록 하였다.

Bertsimas and Sim[3]은 선형최적화 문제에 대한 새로운 형태의 로버스트 최적화 접근법을 제안하였는데, 데이터의 불확실성은 구간의 형태로 가정한 후 각 제약식 내에서 명목 값으로부터의 변동이 허용되는 계수의 개수를 제한하는 방식을 취함으로써 해의 강건성과 최적성 간의 균형을 조정하도록 하였다. 또한 변동 허용 계수의 개수에 따른 제약조건 위반 확률에 대한 상한을 제시하였다. 이들 모형의 장점은 Ben-Tal and Nemirovski[2]의 모형과는 달리 선형최적화의 형태로 로버스트 최적화 문제를 변환할 수 있다는 점이다.

본 연구에서는 Bertsimas and Sim[3]의 로버스트 최적화 접근법을 적용하여 부정확한 데이터를 가지는 DEA 모형에 대한 해법을 개발하고자 한다. DEA 모형은 어떤 관점에서 효율성을 측정하는지에 따라 승수모형(multiplier model)과 포락모형(envelopment model)으로 구분될 수 있으며, 그들 간에는 쌍대관계가 성립한다. 승수모형은 일정 제약조건 하에서 평가대상 DMU의 효율성 점수를 최대로 만드는 투입요소 및 산출요소 가중치를 결정하는 최대화 문제인 반면, 포락모형은 평가대상 DMU와 효율적 경계선 간의 거리를 측정하여 효율성 점수를 결정하는 최소화 문제의 형태를 가진다. 정확한 값의 데이터를 가지는 DEA 모형에서는 승수모형과 포락모형의 풀이 결과가 동일하지만, 부정확한 데이터를 가지는 경우에 대해 로버스트 최적화를 적용하면 전혀 다른 결과가 도출된다. 이에, 본 연구에서는 두 가지 모형 각각에 대한 로버스트 최적화 접근법 적용 결과를 제시하고, 각각의 모형에서 서로

다른 유형의 위험이 고려되고 있다는 점을 밝히고자 한다. 또한, 두 가지 모형을 통해 확률적 척도로 계층화되는 효율성 점수의 범위를 도출할 수 있음을 보이고자 한다. 이러한 효율성 점수의 범위는 DMU들 간의 순위를 결정하는 등의 용도로 유용하게 활용될 수 있다.

로버스트 최적화 접근법을 DEA 모형에 적용한 최근의 연구로는 Sadjadi and Omrani[12, 13], Sadjadi et al.[14], Shokouhi et al.[16]이 있다. Sadjadi and Omrani[13]는 로버스트 최적화 방법을 DEA에 적용한 최초의 연구로 볼 수 있고, Sadjadi and Omrani[12]는 이 연구에 부트스트래핑 기법을 추가적으로 적용하였다. 그들의 연구는 산출요소에서의 부정확성만을 반영한 모형을 제시함으로써 그 응용범위가 상당히 제한적이었다. 또한 승수모형에 대한 결과만을 제시하고 있어 데이터 부정확성이 DEA 모형에서 가지는 위험들 중 하나의 측면만을 고려한다는 한계가 있었다. 반면 본 연구에서는 포락모형에서의 데이터 부정확성도 동시에 고려할 필요가 있다는 점을 논의하도록 한다(제 4.3절). 이러한 문제점을 인식하여, Sadjadi et al. [14]은 투입요소 및 산출요소의 부정확성을 동시에 고려하면서 Ben-Tal and Nemirovski[2]의 로버스트 최적화 접근법을 포락모형에 적용한 결과를 제시한 바 있다. 그러나 그들의 모형은 선형계획법이 아닌 원추 이차계획법 모형으로 정형화되어 해법 측면의 실용성이 다소 떨어지며, 또한 포락모형만을 고려하고 있어 Sadjadi and Omrani [13]의 문제점과 유사하게 데이터 부정확성이 DEA 모형에서 가지는 위험들 중 다른 하나의 측면만을 고려한다는 한계점이 있다. 한편, Shokouhi et al.[16]은 구간 데이터를 가지는 DEA를 위한 Despotis and Smirlis[7]의 두 가지 모형 중 낙관적 모형에 대해 Bertsimas and Sim[3]의 로버스트 최적화 접근법을 적용하였고, 주어진 데이터 부정확성 하에서 효율성 점수 최대값부터 최소값까지 특정 파라미터의 조정을 통해 계산해 낼 수 있음을 보였다. 그러나 그들의 연구는 Bertsimas and Sim[3]가 제시한 로버스트 최적화의 기본적인 개념을 엄밀하게 적용했다고 볼 수 없다. 원칙적으로 로버스트 최적화는 명목값을 중심으로 데이터의 변동성을 고려하여야 하나, 그들의 모형에서는 데이터 구간의 한 극단값을 시작점으로 하여 변동성을 고려하고 있다. 이 경우 Bertsimas and Sim[3]가 제시하고 있는 제약식 위반 확률에 대한 논의가 어렵게 되고, 실제 그들의 연구에서 그러한 논의가 누락되어 있다. 또한, 승수모형 측면에서의 로버스트 최적화만을 제시하고 있어 데이터 부정확성이 DEA 모형에서 가지는 두 가지 측면의 위험을 포괄적으로 논의할 수 없게 된다.

본 연구는 상기와 같은 기존 연구의 한계를 극복하려고 하며, 다음과 같은 구성으로 논의를 진행한다. 제 2장에서는 데이터 불확실성이 구간의 형태로 주어지는 경우에 대한 기존의 IDEA 모형을 제시하고, 제 3장에서는 일반적인 선형최적화에 대한 Bertsimas and Sim[3]의 로버스트 최적화 접근법에 대해 논의한다. 이어서 제 4장에서는 투입지향 BCC 모형[1]의 승수모형 및 포락모형을 대상으로 투입요소 및 산출요소 데이터가 구간의 형태로 주어지는 경우에 대한 로버스트 최적화 접근법의 적용 결과를 제시하고, 도출되는 효율성 점수 및 고려되는 위험의 특성에 대해 논의한다. 제 5장에서는 수치 예제를 대상으로 개발한 모형에 대한 실증 분석을 수행하고, 제 6장에서는 본 연구의 결론을 제시하도록 한다.

2. 구간 데이터를 가지는 DEA 모형

본 장에서는 데이터의 불확실성이 구간의 형태로 주어지는 경우 기존 IDEA 모형에 대해 설명한다. 원래 IDEA 모형에서는 구간 데이터와 함께 순위 데이터도 함께 고려하고 있으나 본 연구에서는 구간 데이터에 한정하여 살펴보도록 한다.

효율성 평가의 대상이 되는 n개의 DMU가 있고 각각의 DMU는 m개의 투입요소와 s개의 산출요소를 가진다고 하자. 그리고 DMU $k(k=1,2,\cdots,n)$의 투입요소 벡터를 $\boldsymbol{x}_k=(x_{1k},\cdots,x_{mk})^{\mathrm{T}}\in R_+^m$, 산출요소 벡터를 $\boldsymbol{y}_k=(y_{1k},\cdots,y_{sk})^{\mathrm{T}}\in R_+^s$라고 하자. 고정규모수익성(constant returns to scale, CRS)을 가정할 때 DMU k의 투입지향 효율성은 다음을 풀어 측정한다.

$$\begin{aligned}
\max\quad & \sum_{r=1}^{s}u_r y_{rk} \\
\text{s.t.}\quad & \sum_{r=1}^{s}u_r y_{rj}-\sum_{i=1}^{m}v_i x_{ij}\le 0 \qquad \forall j, \\
& \sum_{i=1}^{m}v_i x_{ik}=1, \\
& u_r\ge 0,\ v_i\ge 0 \quad \forall r,i \\
& \boldsymbol{x}_{i.}=(x_{ij})\in D_i^- \quad \forall i, \\
& \boldsymbol{y}_{r.}=(y_{rj})\in D_r^+ \quad \forall r.
\end{aligned} \tag{1}$$

여기서 x_{ij}와 y_{rj}의 값은 상하한을 가지는 구간에 속하는 것으로 가정하고, 해당 구간 데이터 값들의 집합을 각각 $D_i^-\subset R^n$와 $D_r^+\subset R^n$라고 표시한다.

투입요소 및 산출요소의 값이 정확하게 주어진다면, 상기 모형은 표준적인 투입지향 BCC 모형이 되고 선형최적화 문제가 된다. 하지만 x_{ij}와 y_{rj}의 값이 D_i^-와 D_r^+로 속하는 부정확한 값이라고 하고, 이를 Cooper et al.[6]와

같이 변수로 취급한다면 상기 모형은 비선형 최적화 문제가 된다. Cooper et al.[6]은 척도변환 및 변수치환을 통해 이 비선형 최적화 문제를 다음과 같은 선형 최적화 문제로 변환이 가능함을 보였다.

$$\begin{aligned}
\max \quad & \sum_{r=1}^{s} Y_{rk} \\
\text{s.t.} \quad & \sum_{r=1}^{s} Y_{rj} - \sum_{i=1}^{m} X_{ij} \le 0 \qquad \forall j, \\
& \sum_{i=1}^{m} X_{ik} = 1, \\
& Y_{rj}^{0} \ge 0,\ X_{ij}^{0} \ge 0 \qquad \forall r, i \\
& \boldsymbol{X}_{i.} = (X_{ij}) \in B_i^{-} \quad \forall i, \\
& \boldsymbol{Y}_{r.} = (Y_{rj}) \in B_r^{+} \quad \forall r.
\end{aligned}$$

여기서 $X_{ij} = \bar{x}_{ij}\bar{v}_i$, $Y_{rj} = \bar{y}_{rj}\bar{u}_r$, $\bar{v}_i = v_i \cdot \max_j\{x_{ij}\}$, $\bar{u}_r = u_r \cdot \max_j\{y_{rj}\}$, $\bar{x}_{ij} = x_{ij}/\max_j\{x_{rj}\}$, $\bar{y}_{rj} = y_{rj}/\max_j\{y_{rj}\}$, $X_{ij}^0 = \bar{x}_{ij}^0\bar{v}_i$, $Y_{rj}^0 = \bar{y}_{rj}^0\bar{u}_{r,}$ $\bar{x}_{ij}^0 = \max_j\{\bar{x}_{ij}\}$, $\bar{y}_{rj}^0 = \max_j\{\bar{y}_{rj}\}$ 이고, B_i^-와 B_r^+는 상기 변수치환에 따라 D_i^-와 D_r^+를 변환한 것이다.

Despotis and Smirlis[7]와 Zhu[19, 20]는 상기 IDEA 모형에서 변수로 취급했던 부정확한 투입요소 및 산출요소의 값을 특정 상수 값으로 고정한 후 표준적인 DEA 모형을 풀면 IDEA 모형에서 도출되는 효율성 점수를 동일하게 계산할 수 있음을 보였다. 구체적으로, 평가대상인 DMU k의 투입요소 및 산출요소의 값은 각각 하한 및 상한으로 고정하고, 다른 DMU들의 투입요소 및 산출요소의 값은 각각 상한 및 하한으로 고정한 후 표준적인 DEA 모형을 풀면 IDEA 모형에 의한 DMU k의 효율성 점수를 얻을 수 있다. 즉, 다음의 선형최적화 문제를 풀어 효율성 점수를 계산한다.

$$\begin{aligned}
\max \quad & \sum_{r=1}^{s} u_r y_{rk}^{U} \\
\text{s.t.} \quad & \sum_{r=1}^{s} u_r y_{rj}^{L} - \sum_{i=1}^{m} v_i x_{ij}^{U} \le 0 \qquad \forall j \ne k, \\
& \sum_{r=1}^{s} u_r y_{rk}^{U} - \sum_{i=1}^{m} v_i x_{ik}^{L} \le 0, \qquad\qquad (2) \\
& \sum_{i=1}^{m} v_i x_{ik}^{L} = 1, \\
& u_r \ge 0, v_i \ge 0 \qquad \forall r, i
\end{aligned}$$

여기서 y_{rj}^L와 y_{rj}^U는 D_i^-와 D_r^+에 나타난 y_{rj}의 하한과 상한을 각각 나타내고, x_{ij}^L, x_{ij}^U는 x_{ij}의 하한과 상한을 각각 나타낸다.

사실상 모형 (1) 또는 모형 (2)로 정형화되는 IDEA 모형은 주어진 데이터 부정확성 하에서 평가대상 DMU가 가질 수 있는 최대의 효율성 점수를 계산한다. 즉, 평가대상 DMU에게 가장 유리한 방식으로 데이터의 부정확성이 발현되고 그와 동시에 가장 유리한 방식으로 가중치를 선택할 수 있도록 하였을 때 얻을 수 있는 최선의 효율성 점수를 계산한다. 이러한 점에서 IDEA 모형은 데이터 부정확성을 지나치게 낙관적으로 반영하고 있다고 볼 수 있다. 한편, Despotis and Smirlis[7]는 모형 (2)와 더불어 데이터의 부정확성이 평가대상 DMU에게 가장 불리한 방식으로 발현되는 경우에 대한 효율성 점수를 계산하는 모형도 제시하였는데 다음과 같다.

$$\begin{aligned}
\max \quad & \sum_{r=1}^{s} u_r y_{rk}^{L} \\
\text{s.t.} \quad & \sum_{r=1}^{s} u_r y_{rj}^{U} - \sum_{i=1}^{m} v_i x_{ij}^{L} \le 0 \qquad \forall j \ne k, \\
& \sum_{r=1}^{s} u_r y_{rk}^{U} - \sum_{i=1}^{m} v_i x_{ik}^{L} \le 0, \qquad\qquad (3) \\
& \sum_{i=1}^{m} v_i x_{ik}^{U} = 1, \\
& u_r \ge 0, v_i \ge 0 \qquad \forall r, i
\end{aligned}$$

로버스트 최적화가 주어진 문제를 구성하는 데이터의 불확실성이 최악의 경우로 발현되는 경우에 대해 가장 최선의 해를 찾는 방법이라는 점에서, 상기 모형 (3)은 로버스트 최적화의 개념을 가장 기본적인 방식으로 구현한 것이라고 볼 수 있다. 다만, 로버스트 최적화의 지나친 보수성을 그대로 함께 가지게 된다는 점을 한계점으로 볼 수 있다.

3. 로버스트 선형최적화

본 장에서는 로버스트 선형최적화 방법에 대해 살펴본다. 우선 다음의 일반적인 선형계획법 문제를 생각해보자.

$$\begin{aligned}
\max \quad & \sum_{j} c_j x_j \\
\text{s.t.} \quad & \sum_{j} a_{ij} x_j \le b_i \quad \forall i, \qquad\qquad (4) \\
& l_j \le x_j \le u_j \quad \forall j.
\end{aligned}$$

상기 모형에서 데이터의 불확실성은 제약식 행렬 $A = (a_{ij})$에서만 발생한다고 가정한다. 행렬 A의 행 i에서 불확실성을 가지는 계수의 열 지수 집합을 J_i라고 할 때, 행

i에 속하는 계수 $a_{ij}(j \in J_i)$는 확률변수 $\tilde{a}_{ij}(j \in J_i)$로 모형화 된다. 여기서 확률변수 $\tilde{a}_{ij}$의 정확한 확률분포는 알지 못하지만 명목 값(nominal value)인 a_{ij}을 평균으로 하는 대칭분포를 따르며 유한한 상한 및 하한을 가진다는 것, 즉 그 값의 범위가 $[a_{ij}-\hat{a}_{ij}, a_{ij}+\hat{a}_{ij}]$이라는 것만 알려져 있다고 가정한다. 새로운 확률변수 $n_{ij}=(\tilde{a}_{ij}-a_{ij})/\hat{a}_{ij}$를 정의하면, n_{ij}는 [-1, 1]의 값을 취하는 대칭인 확률분포를 따른다고 할 수 있다.

로버스트 최적화의 기본 개념은 데이터 불확실성이 최악의 상태로 발현될 때에도 모든 제약식을 만족시킬 수 있는 해들 중에서 목적함수를 가장 최대로 만드는 해를 찾는 것이다. 이러한 개념에 따라 Soyster[17]에 의해 제시된 로버스트 선형최적화 모형은 다음과 같다.

$$\begin{aligned} \max \quad & \sum_j c_j x_j \\ \text{s.t.} \quad & \sum_j a_{ij} x_j + \sum_{j \in J_i} \hat{a}_{ij} y_j \le b_i \quad \forall i, \\ & -y_j \le x_j \le y_j \quad \forall j, \\ & l_j \le x_j \le u_j \quad \forall j, \\ & y_j \ge 0 \quad \forall j. \end{aligned}$$

이러한 Soyster[17]의 모형은 지나치게 보수적인 해가 도출된다는 단점이 있다. 주어진 데이터 불확실성 하에서 어떠한 경우에도 모든 제약식을 만족하는 해가 도출되어 해의 강건성은 극대화되지만, 그에 반해 목적함수 값으로 표현되는 해의 성능은 크게 저하되는 경향이 있다. 이러한 단점은 특히 경영과학 분야에서의 로버스트 최적화 응용을 어렵게 만든다.

Bertsimas and Sim[3]은 이러한 로버스트 최적화의 지나친 보수성을 완화하기 하기 위해 각 행 내의 변동 가능한 계수의 개수를 제한하는 방식을 제안하였다. 즉, Soyster[17]의 모형에서는 각 행 내 변동 가능한 계수는 모두 변동할 수 있다고 가정하여 최악의 경우를 대비하는 방식인 반면, Bertsimas and Sim[3]은 변동 가능한 계수의 개수를 제한한 상황 하에서 최악의 경우를 대비하는 방식을 취한다. 그 방식은 구체적으로 다음과 같다.

행 i에 대해 폐구간 [0, $|J_i|$] 내의 값을 취하는 매개변수 Γ_i를 정의하여 J_i에 속하는 행 i의 계수들 중 최대 Γ_i개의 값만 명목 값으로부터의 변화가 허용되도록 한다. 만일 Γ_i가 정수가 아니라면 최대 $\lfloor \Gamma_i \rfloor$개의 계수는 최대폭만큼 변화가 허용되고 그 이외 하나의 계수 $a_{it}(t \in J_i)$에 대해서는 $(\Gamma_i - \lfloor \Gamma_i \rfloor)\hat{a}_{it}$만큼의 변화가 허용된다. 이러한 Γ_i의 역할은 로버스트 최적화가 가지는 보수성으로 인한 해의 성능 저하와 해의 강건성 간의 상충관계를 조절하는 것이다. 즉, Γ_i를 $|J_i|$로 두면 전통적인 Soyster[17]의 로버스트 최적화와 같이 최대의 보수성, 최대의 강건성을 가지게 되는 반면 해의 성능은 가장 나빠지게 된다. 한편 Γ_i의 값을 줄일수록 보수성을 줄여 해의 강건성은 저하되지만 해의 성능은 반대급부로 개선된다. 이러한 Γ_i의 도입은 행 i의 $|J_i|$개 계수 모두가 동시에 해의 성능에 나쁜 방향으로 변화할 확률은 대단히 작고, 실제로는 단지 몇 개의 계수만이 그러한 방향으로 변화할 개연성이 크다는 사실의 관찰에 기초한다. 즉, Γ_i의 값을 작게 설정하더라도 해의 성능 개선폭에 비해 강건성의 저하는 상대적으로 작은 특성을 가진다.

각 행 i에 대해 Γ_i가 상기와 같이 정의될 때 모형 (4)에 대한 로버스트 최적화 모형은 다음과 같다.

$$\begin{aligned} \max \quad & \sum_j c_j x_j \\ \text{s.t.} \quad & \sum_j a_{ij} x_j \\ & + \max_{\{S_i \cup \{t_i\} | S_i \subseteq j_i, |S_i| = \lfloor \Gamma_i \rfloor, t_i \in J_i \setminus S_i\}} \sum_{j \in S_i} \hat{a}_{ij} y_j \\ & + (\Gamma_i^- - \lfloor \Gamma_i^- \rfloor) \hat{a}_{it_i} y_{t_i} \le b_i \quad \forall i \\ & \quad -y_j \le x_j \le y_j \quad \forall j, \\ & \quad l_j \le x_j \le u_j \quad \forall j, \\ & \quad y_j \ge 0 \quad \forall j. \end{aligned} \tag{5}$$

Bertsimas and Sim[3]은 상기 비선형 최적화 모형이 다음과 같은 선형 최적화 모형으로 동치 변환될 수 있음을 보였다.

$$\begin{aligned} \max \quad & \sum_j c_j x_j \\ \text{s.t.} \quad & \sum_j a_{ij} x_j + z_i \Gamma_i + \sum_{j \in J_i} p_{ij} \le b_i \quad \forall i, \\ & z_i + p_{ij} \ge \hat{a}_{ij} y_j, \quad \forall i, \ j \in J_i, \\ & -y_j \le x_j \le y_j \quad \forall j, \\ & l_j \le x_j \le u_j \quad \forall j, \\ & p_{ij} \ge 0 \quad \forall i, j \in J_i, \\ & y_j \ge 0 \quad \forall j, \ z_i \ge 0 \quad \forall i. \end{aligned} \tag{6}$$

상기 모형의 특성은 다음과 같다. 각 행 i에서 Γ_i개의 계수만이 변한다고 가정할 때 상기 모형으로부터 얻어지는 해는 어떠한 상황에서도 가능해가 된다. 더불어 Γ_i개보다 더 많은 계수가 변한다고 하더라도 그 해는 높은 확률로 가능해가 된다는 바람직한 특성이 있다. Bertsimas and Sim[3]은 모형 (6)을 풀어 얻은 로버스트 최적해가 i번째 제약식을 위반할 확률에 대한 몇 가지 상한을 제시하고 있는데, 계산 용이성과 정확성을 고려할 때 가장 우수한 상한으로 다음을 제안하고 있다.

$$(1-\mu)\,C(n,\lfloor \nu \rfloor) + \sum_{i=\lfloor \nu \rfloor + 1}^{n} C(n,l) \tag{7}$$

여기서

$$C(n,l)=\begin{cases}\frac{1}{2^n}, & \text{if} \quad l=0 \text{ or } l=n,\\ \frac{1}{\sqrt{2\pi}}\sqrt{\frac{n}{(n-1)l}}\,e^{\left(n\log\left(\frac{n}{2(n-l)}\right)+l\log\left(\frac{n-l}{l}\right)\right)}, & o/w\end{cases}$$

이고, $n=|J_i|$, $\nu=(\Gamma_i+n)/2$, $\mu=\nu-\lfloor \nu \rfloor$이다. 여러 가지 $|J_i|$ 값에 대해, 제약식 위반 확률이 1%, 5%, 10% 미만이 되게 하는 Γ_i값은 각각 <Table 1>과 같다.

<Table 1> Choice of Γ_i as a function of $|J_i|$

$\lvert J_i \rvert$	Γ_i		
	Prob. Violation < 1%	Prob. Violation < 5%	Prob. Violation < 10%
5	5	4.8	4.2
10	8.3	6.4	5.3
50	17.5	12.8	10.1
100	24.3	17.6	13.9
200	33.9	24.3	19.2
2,000	105.1	74.6	58.4

예를 들어, $|J_i|=2{,}000$인 경우, 즉 행 i에 불확실한 값을 가지는 계수의 개수가 2,000개인 경우를 생각해 보자. 만일 Γ_i의 값을 2,000개로 설정한다면 모든 불확실한 계수가 명목값으로부터 변화될 수 있도록 허용하는 것이고, 그렇게 로버스트 최적화 모형을 구성하여 얻어진 해는 어떠한 경우에도 해당 제약식을 실제 위반하는 일은 발생하지 않는다. 하지만, Γ_i의 값을 2,000보다 훨씬 작은 105.1로 설정하더라도, 즉 2,000개의 계수 중 105.1개만이 명목 값으로부터의 변화가 허용되도록 제한하더라도 모형으로부터 얻어진 해가 해당 제약식을 실제 위반할 확률은 1% 미만이 된다. 만일 Γ_i를 74.6 또는 58.4로 두더라도 제약식 위반 확률은 각각 5%, 10% 미만이 된다. 이러한 결과는 주어진 $|J_i|$에 비해 Γ_i를 크게 줄이더라도 제약식 위반 확률은 그다지 커지지 않음을 보여준다. 이에 반해 Γ_i를 줄이면 그에 비례하여 도출되는 해의 성능은 크게 향상된다. Γ_i의 값에 따른 로버스트 최적화 모형의 강건성과 해의 성능 간의 상기와 같은 상쇄관계 특성은 Bertsimas and Sim[3]의 로버스트 최적화 방법의 유효성을 잘 보여준다.

4. 로버스트 DEA 모형

이제 앞서 살펴본 로버스트 선형최적화 방법을 부정확한 데이터를 가지는 DEA 모형에 적용한다. 본 장에서는 투입지향 BCC 모형을 기준으로 두 가지 형태의 DEA 모형, 즉 승수모형과 포락모형에 대해 로버스트 선형최적화 방법을 적용한 결과를 제시하고, IDEA 모형과의 관계 등 그 특성에 대해 논의하고자 한다.

모형 전개 과정에서 사용할 기호는 제 2장에서 정의된 바를 따르되, 투입요소 및 산출요소의 값은 명목 값 x_{ij}와 y_{rj}를 중심으로 편차가 $\hat{x}_{ij}$와 $\hat{y}_{rj}$인 구간의 형태로 주어진다고 가정한다. 즉, 투입요소 값의 구간은 $[x_{ij}-\hat{x}_{ij}, x_{ij}+\hat{x}_{ij}]$이고, 산출요소 값의 구간은 $[y_{rj}-\hat{y}_{rj}, y_{rj}+\hat{y}_{rj}]$로 주어진다.

4.1 승수모형에 대한 로버스트 최적화

투입요소 및 산출요소의 값이 명목값 x_{ij}와 y_{rj}인 경우에 대해, DMU k의 효율성 점수를 계산하는 투입지향 BCC 승수모형은 다음과 같다.

$$\begin{aligned} \max \quad & e_k \\ \text{s.t.} \quad & e_k \le \sum_{r=1}^{s} u_r y_{rk} + \xi \\ & \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} + \xi \le 0 \quad \forall j, \\ & \sum_{i=1}^{m} v_i x_{ik} \le 1, \\ & u_r \ge 0,\ v_i \ge 0 \quad \forall r, i. \end{aligned} \tag{8}$$

상기 모형은 일반적인 투입지향 BCC 승수모형과 조금 차이가 있지만 상호 동치임을 쉽게 알 수 있다. 즉 상기 모형에서는 목적함수에 부정확한 계수가 포함되지 않도록 하기 위해 변수 e_k를 도입하여 목적함수를 제약식으로 처리하였고, 최적해에서는 항상 마지막 제약식이 등식으로 성립함을 이용하여 $\sum_{i=1}^{m} v_i x_{ik}=1$를 부등식으로 대체하였다.

모형 (8)에 Bertsimas and Sim[3]의 로버스트 최적화 접근법을 적용하기 위해, DMU j의 투입요소 및 산출요소 값들 중 부정확한 값들의 지수(index) 집합을 각각 J_j^-와 J_j^+로 정의한다. 또한, $|J_j^-|$개의 투입요소 값들과 $|J_j^+|$개의 산출요소 값들 중 변동이 허용되는 값들의 개수를 각각 Γ_j^-와 Γ_j^+라고 하자. 그러면 모형 (8)에 대한 로버스트 최적화 모형은 다음과 같다.

$$
\begin{aligned}
\max \quad & e_k \\
\text{s.t.} \quad & e_k \le \sum_{r=1}^{s} u_r y_{rk} + \xi \\
& - \max_{\{S_k^+ \cup \{t_k^+\} | S_k^+ \subseteq J_k^+,\, |S_k^+| = \lfloor \Gamma_k^+ \rfloor,\, t_k^+ \in J_k^+ \setminus S_k^+\}} \left\{ \sum_{r \in S_k^+} p_r \hat{y}_{rk} + (\Gamma_k^+ - \lfloor \Gamma_k^+ \rfloor) p_{t_k^+} \hat{y}_{t_k^+ k} \right\}, \\
& \sum_{r=1}^{s} u_r y_{rj} + \max_{\{S_j^+ \cup \{t_j^+\} | S_j^+ \subseteq J_j^+,\, |S_j^+| = \lfloor \Gamma_j^+ \rfloor,\, t_j^+ \in J_j^+ \setminus S_j^+\}} \left\{ \sum_{r \in S_j^+} p_r \hat{y}_{rj} + (\Gamma_j^+ - \lfloor \Gamma_j^+ \rfloor) p_{t_j^+} \hat{y}_{t_j^+ j} \right\} + \xi \le \\
& \sum_{i=1}^{m} v_i x_{ij} - \max_{\{S_j^- \cup \{t_j^-\} | S_j^- \subseteq J_j^-,\, |S_j^-| = \lfloor \Gamma_j^- \rfloor,\, t_j^- \in J_j^- \setminus S_j^-\}} \left\{ \sum_{i \in S_j^-} q_i \hat{x}_{ij} + (\Gamma_j^- - \lfloor \Gamma_j^- \rfloor) q_{t_j^-} \hat{x}_{t_j^- j} \right\} \quad \forall j, \\
& \sum_{i=1}^{m} v_i x_{ik} - \max_{\{S_k^- \cup \{t_k^-\} | S_k^- \subseteq J_k^-,\, |S_k^-| = \lfloor \Gamma_k^- \rfloor,\, t_k^- \in J_k^- \setminus S_k^-\}} \left\{ \sum_{i \in S_k^-} q_i \hat{x}_{ik} + (\Gamma_k^- - \lfloor \Gamma_k^- \rfloor) q_{t_k^-} \hat{x}_{t_k^- k} \right\} \le 1, \\
& -p_r \le u_r \le p_r \quad \forall r, \\
& -q_i \le v_i \le q_i \quad \forall i, \\
& p_r, q_i \ge 0 \quad \forall r, i, \quad u_r, v_i \ge 0 \quad \forall r, i.
\end{aligned} \tag{9}
$$

Γ_j^-와 Γ_j^+를 어떻게 설정하는지에 따라 모형 (9)로부터 얻어진 해의 모형 (8)에서의 성능과 강건성이 달라지게 된다. 예를 들어 $\Gamma_j^- = |J_j^-|$, $\Gamma_j^+ = |J_j^+|$로 하면 모형 (9)의 해는 항상 모형 (8)의 가능해가 되어 강건성은 최대가 되는 반면, 그 성능(모든 데이터가 명목 값을 취할 때 모형 (8)의 목적함수 값)은 가장 나빠지게 된다. 반대로 $\Gamma_j^- = 0$, $\Gamma_j^+ = 0$라고 하면 모형 (9)의 해는 모든 데이터가 명목 값을 취할 때의 모형 (8)의 최적해와 같아지게 되어 성능은 최대가 되지만, 데이터 부정확성을 전혀 감안하지 못하게 되어 강건성은 가장 나빠진다.

이러한 관계를 모형 (9)를 통해 산출되는 효율성 점수의 관점에서 설명하면 다음과 같다. Γ_j^-와 Γ_j^+의 값을 크게 할수록 모형 (9)로 계산되는 효율성 점수가 실제 효율성 점수보다 크게 될 위험이 줄어들게 되는 반면, 좀 더 보수적인 효율성 점수 추정이 이루어지게 된다. 반대로 Γ_j^-와 Γ_j^+의 값을 작게 할수록 좀 더 낙관적인 효율성 점수 추정이 이루어지는 반면, 모형 (9)로 계산되는 효율성 점수가 실제 효율성 점수를 초과하게 될 위험이 커지게 된다. 극단적인 경우로 $\Gamma_j^- = |J_j^-|$, $\Gamma_j^+ = |J_j^+|$로 둔다면 모형 (9)는 Despotis and Smirlis[7]가 제시한 모형 (3)과 동일한 결과를 산출한다는 점은 쉽게 알 수 있다. 즉, 주어진 데이터 부정확성 하에서 DMU k가 가질 수 있는 최악의 효율성 점수가 산출된다. 하지만, 모든 변동 가능한 데이터(투입요소 및 산출요소의 값)가 DMU k의 효율성 점수가 최악이 되는 방향으로 동시에 변동할 확률은 대단히 희박하므로 모형 (3)으로부터 산출되는 효율성 점수는 현실적이지 못하다. 모형 (9)는 Γ_j^-와 Γ_j^+의 값을 조정할 수 있게 허용함으로써 강건성은 적정 수준으로 유지한 채 보다 개연성 높은 효율성 점수의 산출을 가능케 해준다.

이제 모형 (9)가 선형최적화 문제로 변환될 수 있음을 보이고자 하는데, Bertsimas and Sim[3]이 제시한 절차를 따르도록 한다. 우선, 모형 (9)을 풀어 얻어진 최적해 $\boldsymbol{p}^* = (p_r^*)$, $\boldsymbol{q}^* = (q_i^*)$, $\boldsymbol{u}^* = (u_r^*)$, $\boldsymbol{v}^* = (v_i^*)$에 대해 $p_r^* = |u_r^*|$, $q_i^* = |v_i^*|$가 성립함은 자명하다. 그리고,

$$
\beta_j^+(\boldsymbol{u}^*, \Gamma_j^+) = \max_{\{S_j^+ \cup \{t_j^+\} | S_j^+ \subseteq J_j^+,\, |S_j^+| = \lfloor \Gamma_j^+ \rfloor,\, t_j^+ \in J_j^+ \setminus S_j^+\}} \left\{ \sum_{r \in S_j^+} |u_r^*| \hat{y}_{rj} + (\Gamma_j^+ - \lfloor \Gamma_j^+ \rfloor) |u_{t_j^+}^*| \hat{y}_{t_j^+ j} \right\}
$$

은 다음의 선형최적화 문제의 최적값과 같다.

$$
\begin{aligned}
\beta_j^+(\boldsymbol{u}^*, \Gamma_j^+) = \max \quad & \sum_{r \in J_j^+} |u_r^*| \hat{y}_{rj} h_{rj}^+ \\
\text{s.t.} \quad & \sum_{r \in J_j^+} h_{rj}^+ \le \Gamma_j^+, \\
& 0 \le h_{rj}^+ \le 1 \quad \forall r \in J_j^+
\end{aligned}
$$

이 문제의 쌍대문제를 구하면 다음과 같다.

$$
\begin{aligned}
\min \quad & z_j^+ \Gamma_j^+ + \sum_{r \in J_j^+} w_{rj}^+ \\
\text{s.t.} \quad & z_j^+ + w_{rj}^+ \ge |u_r^*| \hat{y}_{rj} \quad \forall r \in J_j^+, \\
& w_{rj}^+ \ge 0 \quad \forall r \in J_j^+, \quad z_j^+ \ge 0
\end{aligned}
$$

이와 유사하게

$$
\beta_j^-(\boldsymbol{v}^*, \Gamma_j^-) = \max_{\{S_j^- \subseteq J_j^-,\, |S_j^-| = \lfloor \Gamma_j^- \rfloor,\, t_j^- \in J_j^- \setminus S_j^-\}} \left\{ \sum_{i \in S_j^-} |v_i^*| \hat{x}_{ij} + (\Gamma_j^- - \lfloor \Gamma_j^- \rfloor) |v_{t_j^-}^*| \hat{x}_{t_j^- j} \right\}
$$

의 쌍대문제를 구성할 수 있으며 이 두 가지 쌍대문제를 모형 (9)에 대입하면 다음과 같은 선형최적화 문제가 얻어진다.

$$
\begin{aligned}
\max \quad & e_0 \\
\text{s.t.} \quad & e_0 \le \sum_{r=1}^{s} u_r y_{r0} - z_0^+ \Gamma_0^+ - \sum_{r \in J_0^+} w_{r0}^+ + \xi, \\
& \sum_{r=1}^{s} u_r y_{rj} + z_j^+ \Gamma_j^+ - \sum_{r \in J_j^+} w_{rj}^+ + \xi \le \\
& \sum_{i=1}^{m} v_i x_{ij} - z_j^- \Gamma_j^- - \sum_{i \in J_j^-} w_{ij}^- \quad \forall j, \\
& \sum_{i=1}^{m} v_i x_{i0} + z_0^- \Gamma_0^- + \sum_{i \in J_0^-} w_{i0}^- \le 1, \\
& z_j^+ + w_{rj}^+ \ge p_r \hat{y}_{rj} \quad \forall j, r \in J_j^+, \\
& z_j^- + w_{ij}^- \ge q_i \hat{x}_{ij} \quad \forall j, i \in J_j^-, \\
& w_{rj}^+ \ge 0 \quad \forall j, r \in J_j^+, \\
& w_{ij}^- \ge 0 \quad \forall j, i \in J_j^-, \\
& z_j^+ \ge 0 \ \ \forall j, \quad z_j^- \ge 0 \ \ \forall j, \\
& -p_r \le u_r \le p_r \ \ \forall r, \ -q_i \le v_i \le q_i \ \ \forall i, \\
& p_r, q_i \ge 0 \ \ \forall r, i, \ \ u_r, v_i \ge 0 \ \ \forall r, i.
\end{aligned} \tag{10}
$$

모형 (10)의 목적함수 형태가 최대화이고 제약식의 부등호 방향에 따라 두 가지 쌍대문제의 목적함수 최소화가 암시적으로 이루어짐을 알 수 있다.

4.2 포락모형에 대한 로버스트 최적화

투입요소 및 산출요소의 값이 명목 값 x_{ij}와 y_{rj}인 경우에 대해, DMU k의 효율성 점수를 계산하는 투입지향 BCC 포락모형은 다음과 같다.

$$
\begin{aligned}
\min \quad & \theta \\
\text{s.t} \quad & \sum_{j=1}^{n} x_{ij} \lambda_j \le \theta x_{ik} \quad \forall i, \\
& \sum_{j=1}^{n} y_{rj} \lambda_j \ge y_{rk} \quad \forall r, \\
& \sum_{j=1}^{n} \lambda_j = 1, \quad \lambda_j \ge 0 \ \ \forall j.
\end{aligned}
$$

제약식 좌변과 우변에 동일하게 나타나는 데이터(x_{ik}와 y_{rk})를 이항시켜 하나의 변수에 대한 계수로 합치면 다음과 같다.

$$
\begin{aligned}
\max \quad & -\theta \\
\text{s.t.} \quad & \sum_{j=1, j \ne k}^{n} x_{ij} \lambda_j + x_{ik}(\lambda_k - \theta) \le 0 \quad \forall i, \\
& -\sum_{j=1, j \ne k}^{n} y_{rj} \lambda_j - y_{rk}(\lambda_k - 1) \le 0 \quad \forall r, \\
& \sum_{j=1}^{n} \lambda_j = 1, \quad \lambda_j \ge 0 \quad \forall i.
\end{aligned}
$$

새로운 변수 $\lambda_{k^-} = \lambda_k - \theta$와 $\lambda_{k^+} = \lambda_k - 1$를 도입하고 이 변수 치환을 DMU 지수 집합 $\{1, 2, \cdots, k, \cdots, n\}$에 반영하기 위해 지수 집합에서 k를 제거하고 새로운 두 지수 k^-와 k^+를 추가한다. 또한 $x_{ik^-} = x_{ik}$, $\hat{x}_{ik^-} = \hat{x}_{ik}$, $y_{rk^+} = y_{rk}$, $\hat{y}_{rk^+} = \hat{y}_{rk}$로 정의한다. 이를 통해 위 문제를 다시 쓰면 다음과 같다.

$$
\begin{aligned}
\max \quad & \lambda_{k^-} - \lambda_{k^+} - 1 \\
\text{s.t.} \quad & \sum_{j=1, j \ne k}^{n} x_{ij} \lambda_j + x_{ik^-} \lambda_{k^-} \le 0 \quad \forall i, \\
& -\sum_{j=1, j \ne k}^{n} y_{rj} \lambda_j - y_{rk^+} \lambda_{k^+} \le 0 \quad \forall r, \\
& \sum_{j=1, j \ne k}^{n} \lambda_j + \lambda_{k^+} = 0, \\
& \lambda_j \ge 0 \quad \forall j \ne k, \quad \lambda_{k^+} \ge -1.
\end{aligned} \tag{11}
$$

모형 (11)에 Bertsimas and Sim[3]의 로버스트 최적화 접근법을 적용하기 위해, 투입요소 i의 값이 부정확한 DMU들의 지수 집합을 J_i^-, 산출요소 r의 값이 부정확한 DMU들의 지수 집합을 J_r^+라고 정의한다. 또한, $|J_i^-|$개의 투입요소 값들 중 변동이 허용되는 값들의 개수를 Γ_i^-, $|J_r^+|$개의 산출요소 값들 중 변동이 허용되는 값들의 개수를 Γ_r^+라고 하자. 그러면 모형 (11)에 대한 로버스트 최적화 모형은 다음과 같다.

$$
\begin{aligned}
\max \quad & \lambda_{k^-} - \lambda_{k^+} - 1 \\
\text{s.t.} \quad & \sum_{j=1, j \ne k}^{n} x_{ij} \lambda_j + x_{ik^-} \lambda_{k^-} \\
& + \max_{\{S_i \cup \{t_i\} | S_i \subseteq J_i^-, |S_i| = \lfloor \Gamma_i^- \rfloor, t_i \in J_i^- \setminus S_i\}} \left\{ \sum_{j \in S_i} \hat{x}_{ij} w_j + (\Gamma_i^- - \lfloor \Gamma_i^- \rfloor) \hat{x}_{it_i} w_{t_i} \right\} \le 0 \quad \forall i, \\
& -\sum_{j=1, j \ne k}^{n} y_{rj} \lambda_j - y_{rk^+} \lambda_{k^+} \\
& + \max_{\{S_r \cup \{t_r\} | S_r \subseteq J_r^+, |S_r| = \lfloor \Gamma_r^+ \rfloor, t_r \in J_r^+ \setminus S_r\}} \left\{ \sum_{j \in S_r} \hat{y}_{rj} w_j + (\Gamma_r^+ - \lfloor \Gamma_r^+ \rfloor) \hat{y}_{rt_r} w_{t_r} \right\} \le 0 \quad \forall r, \\
& \sum_{j=1, j \ne k}^{n} \lambda_j + \lambda_{k^+} = 0, \\
& -w_j \le \lambda_j \le w_j \quad \forall j \ne k, \quad w_j \ge 0 \ \ \forall j, \\
& -w_{k^-} \le \lambda_{k^-} \le w_{k^-}, \\
& -w_{k^+} \le \lambda_{k^+} \le w_{k^+}, \\
& \lambda_j \ge 0 \quad \forall j \ne k, \quad \lambda_{k^+} \ge -1.
\end{aligned} \tag{12}
$$

승수모형의 경우와 마찬가지로, Γ_i^-와 Γ_r^+를 어떻게 설정하는지에 따라 모형 (12)로부터 얻어진 해의 모형 (11)에서의 성능과 강건성이 달라진다. 즉, Γ_i^-와 Γ_r^+의 값을 크게 할수록 모형 (12)의 해가 모형 (11)의 제약식을 위반할 위험은 낮아지지만, 그 성능(모든 데이터가 명목 값을 취할 때 모형 (11)의 목적함수 값)은 나빠진다. 반대로 Γ_i^-와 Γ_r^+의 값을 작게 할수록 해의 성능은 좋아지지만 모형 (11)의 제약식을 위반할 위험은 높아지게 된다.

하지만 이러한 관계가 모형 (12)를 통해 산출되는 효율성 점수의 관점에서는 승수모형의 경우와 정반대의 상황으로 나타나는데, 이는 승수모형과 포락모형의 목적함수 형태가 최대화와 최소화로 서로 다르기 때문이다. 승수모형에서는 평가대상 DMU의 효율성 점수를 최대화하는 모형이므로 이에 대한 로버스트 최적화 모형은 주어진 데이터 부정확성 내에서 최악의 효율성 점수를 최대화하는 문제로 나타나게 된다. 따라서 승수모형의 로버스트 최적화 모형을 통해 산출되는 효율성 점수는 데이터의 명목 값을 기준으로 산출된 효율성 점수보다 일반적으로 낮아진다. 반면, 포락모형에서는 평가대상 DMU의 효율성 점수를 최소화하는 모형이므로 이에 대한 로버스트 최적화 모형은 주어진 데이터 부정확성 내에서 최선의 효율성 점수를 최소화하는 문제로 나타나게 된다. 따라서 포락모형의 로버스트 최적화 모형을 통해 산출되는 효율성 점수는 데이터의 명목 값을 기준으로 산출된 효율성 점수보다 일반적으로 높아진다.

포락모형인 모형 (12)를 통해 산출되는 효율성 점수와 Γ_i^- 및 Γ_r^+ 간의 관계를 다시 정리하면 다음과 같다. Γ_i^-와 Γ_r^+의 값을 크게 할수록 모형 (12)로 계산되는 효율성 점수가 실제 효율성 점수보다 작게 될 위험이 줄어들게 되는 반면, 좀 더 낙관적인 효율성 점수 추정이 이루어지게 된다. 반대로 Γ_i^-와 Γ_r^+의 값을 작게 할수록 좀 더 비관적인 효율성 점수 추정이 이루어지는 반면, 모형 (12)로 계산되는 효율성 점수가 실제 효율성 점수보다 작게 될 위험은 커지게 된다. 극단적인 경우로 $\Gamma_i^- = |J_i^-|$, $\Gamma_r^+ = |J_r^+|$로 둔다면 모형 (12)는 Despotis and Smirlis[7]가 제시한 모형 (2), Cooper et al.[6]의 IDEA 모형과 동일한 결과를 산출한다는 점은 쉽게 알 수 있다. 즉, 주어진 데이터 부정확성 하에서 DMU k가 가질 수 있는 최선의 효율성 점수가 산출된다. 하지만, 모든 변동 가능한 데이터(투입요소 및 산출요소의 값)가 DMU k의 효율성 점수가 최선이 되는 방향으로 동시에 변동할 확률은 대단히 희박하므로 모형 (2)로부터 산출되는 효율성 점수는 현실적이지 못하다. 모형 (12)는 Γ_i^-와 Γ_r^+의 값을 조정할 수 있게 허용함으로써 강건성은 적정 수준으로 유지한 채 보다 개연성 높은 효율성 점수의 산출을 가능케 해준다.

승수모형의 경우와 유사한 방법으로 모형 (12)가 선형 최적화 문제로 변환될 수 있음을 보일 수 있다. 그 과정은 앞서 기술한 것과 크게 다르지 않으므로 여기서는 그 결과만을 아래의 모형 (13)과 같이 제시하도록 한다.

$$
\begin{aligned}
\max \quad & \lambda_{k^-} - \lambda_{k^+} - 1 \\
\text{s.t.} \quad & \sum_{j=1, j\neq k}^{n} x_{ij}\lambda_j + x_{ik^-}\lambda_{k^-} + z_i^- \Gamma_i^- + \sum_{j \in J_i^-} p_{ij}^- \le 0 \quad \forall i, \\
& -\sum_{j=1, j\neq k}^{n} y_{rj}\lambda_j - y_{rk^+}\lambda_{k^+} + z_r^+ \Gamma_r^+ + \sum_{j \in J_r^+} p_{rj}^+ \le 0 \quad \forall r, \\
& z_i^- + p_{ij}^- \ge \hat{x}_{ij} w_j \quad \forall i, j \in J_i^-, \\
& z_r^+ + p_{rj}^+ \ge \hat{y}_{rj} w_j \quad \forall r, j \in J_r^+, \\
& -w_j \le \lambda_j \le w_j \quad \forall j \neq k, \quad w_j \ge 0 \ \ \forall j, \\
& -w_{k^-} \le \lambda_{k^-} \le w_{k^-}, \\
& -w_{k^+} \le \lambda_{k^+} \le w_{k^+}, \\
& \sum_{j=1, j\neq k}^{n} \lambda_j + \lambda_{k^+} = 0, \\
& \lambda_j \ge 0 \ \ \forall j \neq k, \ \lambda_{k^+} \ge -1, \\
& z_i^-, z_r^+ \ge 0 \ \ \forall i, r, \\
& p_{ij}^- \ge 0 \quad \forall i, j \in J_i^-, \ p_{rj}^+ \ge 0 \ \ \forall r, j \in J_r^+.
\end{aligned}
\tag{13}
$$

4.3 DEA 모형에서 로버스트 최적화의 의미

지금까지 투입지향 BCC 모형을 기준으로 투입요소 및 산출요소 값의 부정확성이 구간의 형태로 주어질 때 승수모형 및 포락모형에 대해 Bertsimas and Sim[3]의 로버스트 최적화 방법을 적용한 결과를 제시하였다. 여기서는 두 가지 DEA 모형에서 로버스트 최적화가 가지는 의미에 관한 논의하기로 한다.

앞서 기술한 바 있듯이 로버스트 최적화는 제약식을 위반할 위험을 완화하면서 최선의 해를 찾기 위한 방법이라고 볼 수 있다. Soyster[17]은 제약식 위반 위험을 완전히 제거하는 가장 보수적인 방법이고, Ben-Tal and Nemirovski [2]는 타원체의 형태로 범위가 한정된 데이터 불확실성 하에서 제약식 위반 위험을 제거하는 방법을 취함으로써 지나친 보수성을 완화시킨 방법이다. Bertsimas and Sim[3]의 접근법 또한 변동이 가능한 데이터의 개수가 한정된 상태에서 제약식 위반 위험을 제거함으로써 로버스트 최적화가 가지는 지나친 보수성을 완화하는 방법이다. 그렇다면 DEA 모형에서 로버스트 최적화가 가지는 의미를 생각해 볼 필요가 있으며, 효율성 점수 측정의 관점에서 구체적으로 어떤 위험을 완화하고자 하는 것인지 고려하여야 한다.

두 가지 형태의 DEA 모형, 즉 승수모형과 포락모형 중 어떤 모형을 선택하여 로버스트 최적화를 적용하는지에 따라 전혀 다른 위험을 고려하게 된다. 승수모형에 로버스트 최적화를 적용하는 경우에는 해당 모형을 통해 산출되는 효율성 점수가 실제 효율성 점수를 초과할 위험을 완화하고자 하는 것이고, 포락모형에 로버스트 최적화를 적용하는 경우에는 산출되는 효율성 점수가 실제 효율성 점수에 미달할 위험을 완화하고자 하는 것이다. 따라서, 승수모형에 Soyster[17]의 로버스트 최적화 방법을 적용한다면 주어진 데이터 부정확성 내에서 평가대상 DMU가 가질 수 있는 최악의 효율성 점수를 계산하게 되고, 이는 Despotis and Smirlis[7]가 제시한 모형 (3)과 동일한 결과를 도출한다는 것을 알 수 있다. 이와 반대로 포락모형에 Soyster[17]의 로버스트 최적화 방법을 적용한다면 주어진 데이터 부정확성 내에서 평가대상 DMU가 가질 수 있는 가장 높은 효율성 점수를 계산하게 되고, 이는 Despotis and Smirlis[7]가 제시한 모형 (2) 또는 Cooper et al. [6]의 IDEA 모형과 동일한 결과를 도출하게 된다. 즉, 여기서 IDEA 모형은 사실상 포락모형에 Soyster [17]의 로버스트 최적화 방법이 적용된 것이라는 점을 알 수 있다.

Soyster[17]의 로버스트 최적화 방법이 가지는 지나친 보수성은 DEA 모형에 적용할 때에도 동일하게 발생한다. 즉, 승수모형에 Soyster[17]의 로버스트 최적화 방법이 적용된 결과인 모형 (2)는 과도하게 낮은 효율성 점수를 산출하게 되고, 포락모형에 Soyster[17]의 로버스트 최적화 방법이 적용된 결과로 볼 수 있는 모형 (3) 또는 IDEA 모형은 과도하게 높은 효율성 점수를 산출하게 된다. 이러한 문제점을 해결하기 위한 하나의 방법으로 본 연구에서 제시한 대로 Bertsimas and Sim[3]의 로버스트 최적화 방법을 두 가지의 DEA 모형에 적용하여 보수성을 조정하는 것이 유용한 현실적 대안이 될 수 있다.

4.4 효율성 점수의 확률적 계층화

본 연구에서 채택한 Bertsimas and Sim[3]의 로버스트 최적화 방법이 가지는 장점 중 하나는 변동 가능 계수의 개수를 제한하는 파라미터를 통해 해의 성능과 보수성 간의 균형을 조절할 수 있다는 것이다. 또한, 해당 파라미터의 값에 따른 제약식 위반 확률의 상한을 계산할 수도 있다.

이 방법론을 DEA 모형에 적용하는 경우 효율성 점수에 대한 확률적 계층화가 가능해 진다. 예를 들어, 승수모형에 대한 로버스트 최적화 모형에서는 제약식 위반 확률을 0%, 5%, 10%, ⋯, 50% 등으로 설정하고 이에 따른 Γ_j^-와 Γ_j^+의 값을 각각 찾은 후 모형 (10)을 풀면 각 경우에 대한 효율성 점수가 점차 증가하는 형태로 도출된다. 즉, 제약식 위반 확률을 0%로 설정하면 모형 (10)에서 Γ_j^-와 Γ_j^+의 값은 각각 $|J_j^-|$와 $|J_j^+|$이 될 것이고, 도출되는 효율성 점수는 평가대상 DMU가 주어진 데이터 부정확성 하에서 얻을 수 있는 가장 낮은 점수가 된다. 이 경우 얻어진 효율성 점수가 실제 효율성 점수보다 더 높을 확률은 0%가 된다. 제약식 위반 확률을 점점 높여갈수록 이러한 보수성은 완화되면서 도출되는 효율성 점수는 높아진다.

포락모형에 대한 로버스트 최적화 모형에서도 마찬가지로 제약식 위반 확률을 0%, 5%, 10%, ⋯, 50% 등으로 설정하고 이에 따른 Γ_i^-와 Γ_r^+의 값을 각각 찾은 후 모형 (13)을 풀면 각 경우에 대한 효율성 점수가 점차 감소하는 형태로 도출된다. 즉, 제약식 위반 확률을 0%로 설정하면 모형 (13)에서 Γ_i^-와 Γ_r^+의 값은 각각 $|J_i^-|$와 $|J_r^+|$이 될 것이고, 도출되는 효율성 점수는 평가대상 DMU가 주어진 데이터 부정확성 하에서 얻을 수 있는 가장 높은 점수가 된다. 이 경우 얻어진 효율성 점수가 실제 효율성 점수보다 더 낮을 확률은 0%가 된다. 제약식 위반 확률을 점점 높여갈수록 이러한 보수성은 완화되면서 도출되는 효율성 점수는 낮아진다.

상기와 같은 방식으로 효율성 점수를 확률적으로 계층화함으로써 부정확한 데이터를 가지는 DEA 모형에 대한 기존의 해법을 보다 현실적 유용성을 가지도록 개선할 수 있다. 기존의 IDEA 모형이나 Despotis and Smirlis[7]의 모형 (2)를 통해 도출되는 효율성 점수를 기반으로 DMU의 효율성을 분석하는 것은 과도한 보수성과 희박한 시나리오를 가정하는 것이므로, 본 연구에서 개발한 방법을 통해 보다 합리적인 보수성과 현실적인 시나리오 하에서의 효율성 점수 분석을 수행하는 것이 더 바람직하다. 예를 들어, 계산되는 효율성 점수가 실제 효율성 점수보다 높거나 낮을 확률을 현실적으로 허용 가능한 수치(예를 들어 20%)로 설정한 후, 모형 (10)이나 모형 (13)을 풀어 도출되는 효율성 점수를 기반으로 효율성 분석을 수행하는 것이 가능하다.

5. 수치 예제

본 장에서는 앞서 개발한 로버스트 최적화 DEA 모형을 실제 수치 예제에 적용한 결과를 검토하고, 이를 통해 모형의 특성 및 시사점을 논의하도록 한다.

실험에 사용할 데이터는 Oral et al.[11]에서 소개된 터키 철강산업 분야의 37개 R&D 프로젝트 평가 데이터를 사용한다. 데이터에 포함된 터키 철강산업 분야 37개의 R&D 프로젝트는 한정된 예산을 두고 상호 경쟁하며, 각

<Table 2> Data of 37 R&D Project Proposals(Oral et al.[10])

R&D Project	Indirect Economic Contribution	Direct Economic Contribution	Technological Contribution	Social Contribution	Scientific Contribution	Budget
1	67.53	70.82	62.64	44.91	46.28	84.20
2	58.94	62.86	57.47	42.84	45.64	90.00
3	22.27	9.68	6.73	10.99	5.92	50.20
4	47.32	47.05	21.75	20.82	19.64	67.50
5	48.96	48.48	34.90	32.73	26.21	75.40
6	58.88	77.16	35.42	29.11	26.08	90.00
7	50.10	58.20	36.12	32.46	18.90	87.40
8	47.46	49.54	46.89	24.54	36.35	88.80
9	55.26	61.09	38.93	47.71	29.47	95.90
10	52.40	55.09	53.45	19.52	46.57	77.50
11	55.13	55.54	55.13	23.36	46.31	76.50
12	32.09	34.04	33.57	10.60	29.36	47.50
13	27.49	39.00	34.51	21.25	25.74	58.50
14	77.17	83.35	60.01	41.37	51.91	95.00
15	72.00	68.32	25.84	36.64	25.84	83.80
16	39.74	34.54	38.01	15.79	33.06	35.40
17	38.50	28.65	51.18	59.59	48.82	32.10
18	41.23	47.18	40.01	10.18	38.86	46.70
19	53.02	51.34	42.48	17.42	46.30	78.60
20	19.91	18.98	25.49	8.66	27.04	54.10
21	50.96	53.56	55.47	30.23	54.72	74.40
22	53.36	46.47	49.72	36.53	50.44	82.10
23	61.60	66.59	64.54	39.10	51.12	75.60
24	52.56	55.11	57.58	39.69	56.49	92.30
25	31.22	29.84	33.08	13.27	36.75	68.50
26	54.64	58.05	60.03	31.16	46.71	69.30
27	50.40	53.58	53.06	26.68	48.85	57.10
28	30.76	32.45	36.63	25.45	34.79	80.00
29	48.97	54.97	51.52	23.02	45.75	72.00
30	59.68	63.78	54.80	15.94	44.04	82.90
31	48.28	55.58	53.30	7.61	36.74	44.60
32	39.78	51.69	35.10	5.30	29.57	54.50
33	24.93	29.72	28.72	8.38	23.45	52.70
34	22.32	33.12	18.94	4.03	9.58	28.00
35	48.83	53.41	40.82	10.45	33.72	36.00
36	61.45	70.22	58.26	19.53	49.33	64.10
37	57.78	72.10	43.83	16.14	31.32	66.40

각의 프로젝트는 <Table 2>와 같이 5개의 산출요소와 1개의 투입요소로 특징지어진다. 산출요소로는 간접적 경제 기여도(indirect economic contribution), 직접적 경제 기여도(direct economic contribution), 기술적 기여도(technological contribution), 사회적 기여도(social contribution), 과학적 기여도(scientific contribution)가 포함되고, 투입요소로는 비용, 즉 예산이 포함된다. 모든 요소의 값은 전문가 설문을 통해 얻은 0~100사이의 점수로 표현된다.

원래 데이터에서는 투입요소 및 산출요소의 값이 정확한 값으로 주어지는 경우를 가정하고 있는데, 본 연구에서는 이를 수정하여 모든 데이터에 10%의 부정확성이 있는 것으로 가정한다. 예를 들어, 1번 프로젝트의 경우 간접적 경제 기여도와 예산이 각각 67.53과 84.20으로 주어져 있는데, 본 연구에서는 해당 값들이 10%의 편차를 가진 구간 값, 즉 각각 [60.78, 74.28]과 [75.78, 92.62]로 주어진다고 가정한다.

<Table 3> Efficiency Scores from Robust Optimization Applied to the Multiplier Model(10)

R&D Project	Nominal	Probability of Violation						
		0%	5%	10%	20%	30%	40%	50%
1	1	0.5626	0.5755	0.5947	0.6395	0.6989	0.7749	0.8433
2	0.7629	0.4371	0.4472	0.4621	0.4827	0.5196	0.5708	0.6259
3	0.5680	0.4623	0.4729	0.4887	0.5056	0.5196	0.5326	0.5437
4	0.5384	0.3953	0.4044	0.4179	0.4324	0.4443	0.4598	0.4789
5	0.5667	0.3584	0.3666	0.3789	0.3920	0.4044	0.4227	0.4541
6	0.8866	0.5192	0.5312	0.5489	0.5679	0.5837	0.6150	0.6502
7	0.6368	0.3544	0.3625	0.3746	0.3876	0.4006	0.4318	0.4703
8	0.4468	0.3017	0.3086	0.3190	0.3330	0.3461	0.3604	0.3747
9	0.7169	0.4140	0.4235	0.4376	0.4528	0.4712	0.5122	0.5565
10	0.6452	0.3652	0.3736	0.3867	0.4315	0.4743	0.5193	0.5600
11	0.7100	0.3779	0.3866	0.4068	0.4571	0.5019	0.5494	0.5914
12	0.6489	0.5083	0.5199	0.5373	0.5559	0.5719	0.5896	0.6053
13	0.5453	0.4087	0.4181	0.4321	0.4482	0.4639	0.4796	0.4959
14	1	0.6028	0.6167	0.6373	0.6747	0.7207	0.7932	0.8609
15	1	0.5772	0.5905	0.6102	0.6321	0.6586	0.7017	0.7713
16	0.9238	0.7069	0.7231	0.7473	0.7731	0.7944	0.8159	0.8437
17	1	0.7978	0.8161	0.8434	0.8734	0.9025	0.9300	0.9535
18	0.7781	0.5646	0.5775	0.5969	0.6175	0.6392	0.6628	0.6849
19	0.6412	0.3541	0.3622	0.3744	0.3876	0.4121	0.4591	0.5049
20	0.5513	0.4433	0.4534	0.4686	0.4848	0.4982	0.5106	0.5214
21	1	0.4356	0.4456	0.4605	0.4859	0.5316	0.5802	0.6241
22	0.7091	0.3393	0.3471	0.3587	0.3765	0.4151	0.4564	0.5015
23	1	0.5451	0.5576	0.5827	0.6298	0.6974	0.7756	0.8466
24	1	0.3723	0.3808	0.3936	0.4186	0.4570	0.5016	0.5528
25	0.4502	0.3600	0.3683	0.3806	0.3937	0.4046	0.4147	0.4237
26	0.8955	0.4646	0.4757	0.5001	0.5399	0.5921	0.6464	0.6984
27	0.8935	0.5000	0.5115	0.5286	0.5766	0.6347	0.6955	0.7499
28	0.3895	0.3065	0.3136	0.3241	0.3353	0.3445	0.3533	0.3623
29	0.6809	0.3919	0.4008	0.4143	0.4347	0.4781	0.5238	0.5658
30	0.7232	0.3852	0.3941	0.4135	0.4618	0.5062	0.5537	0.5961
31	1	0.6359	0.6505	0.6723	0.6955	0.7195	0.7830	0.8483
32	0.6481	0.4820	0.4930	0.5095	0.5278	0.5462	0.5638	0.5818
33	0.5588	0.4503	0.4606	0.4760	0.4925	0.5060	0.5187	0.5308
34	1	0.8182	0.8369	0.8649	0.8948	0.9195	0.9425	0.9620
35	1	0.7542	0.7715	0.7973	0.8278	0.8575	0.8921	0.9273
36	1	0.5820	0.5954	0.6272	0.6935	0.7543	0.8183	0.8780
37	1	0.5579	0.5707	0.5898	0.6102	0.6315	0.6617	0.7316

상기 데이터를 바탕으로 모형 (10)과 모형 (13)을 각각의 DMU를 평가대상으로 하여 푼 결과는 <Table 3> 및 <Table 4>와 같다. 두 가지 표에서 Nominal 열에 있는 값들은 투입요소 및 산출요소가 명목 값을 가질 때 계산되는 효율성 점수로서 승수모형과 포락모형에서 동일한 값이 도출된다. 두 모형에 대한 로버스트 최적화 모형의 계산 결과는 7가지의 제약식 위반확률에 대해 각각 산출하였다.

승수모형에서 제약식 위반 확률 0%에 해당하는 효율성 점수(<Table 3>)는 Despotis and Smirlis[7]의 모형 (3)의 결과와 동일하며, 주어진 데이터 불확실성 하에서 각각의 DMU가 가질 수 있는 최악의 효율성 점수에 해당한다. 해당 효율성 점수가 실제 효율성 점수보다 높을 위험은 완전히 제거되지만 지나치게 비관적인 결과가 초래된다. 제약식 위반 확률을 높여갈수록 도출되는 효율성 점수는 비교적 민감하게 변동하며 점차 증가함을 알 수 있는데, 위험은 어느 정도 수용하면서 보다 현실성 있는 효율성 점수가 도출되는 것으로 볼 수 있다.

포락모형에서 제약식 위반 확률 0%에 해당하는 효율

<Table 4> Efficiency Scores from Robust Optimization Applied to the Envelopment Model(13)

R&D project	Nominal	Probability of Violation						
		0%	5%	10%	20%	30%	40%	50%
1	1	1	1	1	1	1	1	1
2	0.7629	1	1	1	1	1	1	1
3	0.5680	0.7119	0.7119	0.7119	0.7119	0.7119	0.7005	0.6359
4	0.5384	1	1	1	0.9881	0.9345	0.8792	0.7508
5	0.5667	1	1	1	1	0.9322	0.8654	0.7748
6	0.8866	1	1	1	1	1	1	1
7	0.6368	1	1	1	1	1	0.9361	0.8378
8	0.4468	0.8329	0.8329	0.8329	0.7986	0.7436	0.6891	0.6101
9	0.7169	1	1	1	1	1	1	1
10	0.6452	1	1	1	1	1	1	0.991374
11	0.7100	1	1	1	1	1	1	1
12	0.6489	0.8614	0.8614	0.8614	0.8614	0.8614	0.8284	0.7514
13	0.5453	0.7849	0.7849	0.7844	0.7766	0.7692	0.7398	0.6402
14	1	1	1	1	1	1	1	1
15	1	1	1	1	1	1	1	1
16	0.9238	1	1	1	1	1	1	1
17	1	1	1	1	1	1	1	1
18	0.7781	1	1	1	1	1	1	1
19	0.6412	1	1	1	1	1	1	0.9547
20	0.5513	0.6880	0.6880	0.6880	0.6880	0.6880	0.6782	0.6205
21	1	1	1	1	1	1	1	1
22	0.7091	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1
24	1	1	1	1	1	1	1	1
25	0.4502	0.6099	0.6099	0.6099	0.6099	0.6061	0.5863	0.5217
26	0.8955	1	1	1	1	1	1	1
27	0.8935	1	1	1	1	1	1	1
28	0.3895	0.5232	0.5232	0.5232	0.5232	0.5216	0.5046	0.4518
29	0.6809	1	1	1	1	1	1	0.964252
30	0.7232	1	1	1	1	1	1	1
31	1	1	1	1	1	1	1	1
32	0.6481	1	1	1	1	1	1	0.8594
33	0.5588	0.7253	0.7253	0.7253	0.7253	0.7253	0.6925	0.6358
34	1	1	1	1	1	1	1	1
35	1	1	1	1	1	1	1	1
36	1	1	1	1	1	1	1	1
37	1	1	1	1	1	1	1	1

성 점수(<Table 4>)는 Despotis and Smirlis[7]의 모형 (2) 또는 Cooper et al.[6]의 IDEA 모형의 결과와 동일하며, 주어진 데이터 불확실성 하에서 각각의 DMU가 가질 수 있는 최선의 효율성 점수에 해당한다. 해당 효율성 점수가 실제 효율성 점수보다 낮을 위험은 완전히 제거되지만 지나치게 낙관적인 결과가 나타난다. 제약식 위반 확률을 높여갈수록 도출되는 효율성 점수는 점차 감소함을 알 수 있는데, 위험은 일정 부분 수용하면서 보다 현실성 있는 효율성 점수를 얻는 것으로 볼 수 있다. 하나의 예를 통한 결과로 일반화하기는 어렵지만, 포락모형의 경우 제약식 위반 확률의 변화에 따른 효율성 점수의 변화 정도가 승수모형에 비해 상대적으로 크게 둔감한 특징이 나타난다. 따라서 효율성 점수의 확률적 계층화 측면에서는 승수모형에 로버스트 최적화 방법을 적용하는 것이 더욱 유용한 정보를 제공하는 것으로 판단된다.

한편, <Table 5>에는 앞서 도출된 효율성 점수들을 기준으로 R&D Project의 순위를 매긴 결과가 나타나 있다. 명목 데이터에 기초하여 산출된 효율성 점수를 기준으로

<Table 5> Ranks of Efficiency Scores from Different Models

R&D project	Nominal	Model(10)		Model(13)	
		Prob. Violation 0%	Prob. Violation < 20%	Prob. Violation 0%	Prob. Violation < 20%
1	1	10	8	1	1
2	18	21	22	1	1
3	29	18	18	34	34
4	34	25	28	1	29
5	30	32	32	1	1
6	16	13	14	1	1
7	28	33	34	1	1
8	36	37	37	31	31
9	20	23	25	1	1
10	26	30	29	1	1
11	21	28	24	1	1
12	24	14	15	30	30
13	33	24	26	32	32
14	1	6	7	1	1
15	1	8	9	1	1
16	13	4	4	1	1
17	1	2	2	1	1
18	17	9	11	1	1
19	27	34	33	1	1
20	32	20	21	35	35
21	1	22	20	1	1
22	22	35	35	1	1
23	1	12	10	1	1
24	1	29	30	1	1
25	35	31	31	36	36
26	14	17	16	1	1
27	15	15	13	1	1
28	37	36	36	37	37
29	23	26	27	1	1
30	19	27	23	1	1
31	1	5	5	1	1
32	25	16	17	1	1
33	31	19	19	33	33
34	1	1	1	1	1
35	1	3	3	1	1
36	1	7	6	1	1
37	1	11	12	1	1

한 순위와 모형 (10) 또는 모형 (13)을 통해 산출되는 효율성 점수를 기준으로 한 순위에 유의미한 차이가 발생함을 알 수 있다. 특히 승수모형에 로버스트 최적화를 적용한 모형 (10)의 경우 더 뚜렷한 차이를 보이고 있다. 어느 정도의 제약식 위반 위험을 수용할지를 의사결정자가 판단한 후 그에 따라 산출되는 효율성 점수를 기준으로 R&D 프로젝트를 평가한다면 지나친 보수성을 완화하여 보다 현실성 있는 결과를 얻을 수 있을 것으로 판단된다.

6. 결 론

본 연구에서는 부정확한 데이터를 가지는 DEA 모형에 대해 로버스트 최적화 방법을 적용한 결과와 그 특성에 대해 논의하였다. 두 가지 DEA 모형인 승수모형과 포락모형에 대해 로버스트 최적화 방법을 적용하였으며, 각각 경우가 서로 다른 위험을 고려하고 있음을 밝혔다. 또한, 부정확한 데이터를 가지는 DEA 모형의 해법에 대한 IDEA 모형

등 기존의 연구들은 승수모형 또는 포락모형에 대해 전통적인 로버스트 최적화 방법을 적용한 결과임을 보였고, 이는 과도하게 보수적인 결과를 초래함을 논의하였다. 전통적인 로버스트 최적화 방법을 DEA 모형에 적용했을 때 초래되는 과도한 보수성을 극복하기 위해 변동 가능한 데이터의 개수를 제한하는 방식의 로버스트 최적화 방법을 채택하여 승수모형 및 포락모형에 적용하였다. 그 적용 결과를 통해 효율성 점수를 확률적으로 계층화할 수 있음도 보였다. 수치예제를 통해 실제 계산 결과를 살펴본 바, 본 연구 모형을 통해 도출되는 효율성 점수가 기존 모형의 결과와 차별적이면서 보다 유용한 결과를 도출할 수 있음을 보였다.

Acknowledgement

This work was supported by the Dongguk University Research Fund of 2014.

References

[1] Banker, R.D., Charnes, A., and Cooper, W.W., Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Management Science*, 1984, Vol. 30, No. 9, pp. 1078-1092.

[2] Ben-Tal, A. and Nemirovski, A., Robust solutions of uncertain linear programs. *Operations Research Letters*, 1999, Vol. 25, No. 1, pp. 1-13.

[3] Bertsimas, D. and Sim, M., The price of robustness. *Operations Research*, 2004, Vol. 52, No. 1, pp. 35-53.

[4] Charnes, A., Cooper, W.W., and Rhodes, E., Measuring the efficiency of decision making units. *European Journal of Operational Research*, 1978, Vol. 2, No. 6, pp. 429-444.

[5] Cook, W.D. and Seiford, L.M., Data envelopment analysis(DEA)-Thirty years on. *European Journal of Operational Research*, 2009, Vol. 192, No. 1, pp. 1-17.

[6] Cooper, W.W., Park, K.S., and Yu, G., IDEA and AR-IDEA : Models for dealing with imprecise data in DEA. *Management Science*, 1999, Vol. 45, No. 4, pp. 597-607.

[7] Despotis, D.K. and Smirlis, Y.G., Data envelopment analysis with imprecise data. *European Journal of Operational Research*, 2002, Vol. 140, No. 1, pp. 24-36.

[8] Lim, S., A non-oriented DEA game cross efficiency model for supplier selection. *Journal of Society of Korea Industrial and Systems Engineering*, 2015, Vol. 38, No. 2, pp. 108-119.

[9] Liu, J.S., Lu, L.Y., Lu, W.M., and Lin, B.J., Data envelopment analysis 1978~2010 : A citation-based literature survey. *Omega*, 2013, Vol. 41, No. 1, pp. 3-15.

[10] Neralic, L., Preservation of efficiency and inefficiency classification in data envelopment analysis. *Mathematical Communications*, 2004, Vol. 9, No. 1, pp. 51-62.

[11] Oral, M., Kettani, O., and Lang, P., "A methodology for collective evaluation and selection of industrial R&D projects. *Management Science*, 1991, Vol. 37, No. 7, pp. 871-885.

[12] Sadjadi, S.J. and Omrani, H., A bootstrapped robust data envelopment analysis model for efficiency estimating of telecommunication companies in Iran. *Telecommunications Policy*, 2010, Vol. 34, No. 4, pp. 221-232.

[13] Sadjadi, S.J. and Omrani, H., Data envelopment analysis with uncertain data : An application for Iranian electricity distribution companies. *Energy Policy*, 2008, Vol. 36, No. 11, pp. 4247-4254.

[14] Sadjadi, S.J., Omrani, H., Abdollahzadeh, S., Alinaghian, M., and Mohammadi, H., A robust super-efficiency data envelopment analysis model for ranking of provincial gas companies in Iran. *Expert Systems with Applications*, 2011, Vol. 38, No. 9, pp. 10875-10881.

[15] Seiford, L.M. and Zhu, J., Sensitivity analysis of DEA models for simultaneous changes in all the data. *Journal of the Operational Research Society*, 1998, Vol. 49, No. 10, pp. 1060-1071.

[16] Shokouhi, A.H., Hatami-Marbini, A., Tavana, M., and Saati, S., A robust optimization approach for imprecise data envelopment analysis. *Computers and Industrial Engineering*, 2010, Vol. 59, No. 3, pp. 387-397.

[17] Soyster, A.L., Technical note-convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 1973, Vol. 21, No. 5, pp. 1154-1157.

[18] Yoo, S., Kim, Y., Kim, J., and Choi, J., The evaluation of administrative efficiency of the Korean university using DEA model. *Journal of the Korean Society for Quality Management*, 2014, Vol. 42, No. 4, pp. 647-664.

[19] Zhu, J., Imprecise data envelopment analysis (IDEA) : A review and improvement with an application. *European Journal of Operational Research*, 2003, Vol. 144, No. 3, pp. 513-529.

[20] Zhu, J., Imprecise DEA via standard linear DEA models with a revisit to a Korean mobile telecommunication company. *Operations Research*, 2004, Vol. 52, No. 2, pp. 323-329.

ORCID

Sungmook Lim | http://orcid.org/0000-0003-2690-1133