



## New Advances on Fuzzy Linear Programming Problem by Semi-Infinite Programming Approach

Hadi ZAVIEH<sup>1,\*</sup> , Hadi NASSERI<sup>1</sup> , Chefi TRIKI<sup>2</sup> 

<sup>1</sup> Department of Applied Mathematics, University of Mazandaran, 47415, Babolsar, Iran.

<sup>2</sup> Department of Industrial Engineering, Sultan Qaboos University, 123, Muscat, Oman.

### Highlights

- This paper focuses on LSIP problem with fuzzy coefficient in objective and constraints.
- A multi-objective programming problem is proposed for solving linear semi-infinite Problem.
- A data envelopment analysis model is suggested as an application.

### Article Info

Received: 29 Jan 2021

Accepted: 18 Sep 2021

### Keywords

Fuzzy linear programming  
Multi-objective linear programming  
Linear semi-infinite programming  
Data envelopment analysis

### Abstract

As we are faced with more uncertainty problems in the real world, it is necessary to provide models that can provide appropriate solutions for dealing with these issues. In this study, we proposed a new approach to solving linear programming problem in the fuzzy environment based on solving a related multi-objective model. This kind of problem can be reduced to a fuzzy linear semi-infinite programming problem. In this way, we present a new mixed Multi-Objective Linear Semi-Infinite Programming (MOLSIP) model to solve the main problem, furthermore, as a practical case, we consider a fuzzy Data Envelopment Analysis (DEA) model which is a concern to an evaluation of the performance of Decision-Making Units (DMUs) in uncertainty environment. The new models show the advantage of our method over the previous ones in terms of certainty. Finally, numerical examples are included to illustrate the suggested solution procedure.

## 1. INTRODUCTION

The main idea of fuzzy sets theory was first proposed by Zadeh [1] that is widely used in optimization problems. The true observation we have found the Fuzzy Linear Programming Problem (FLP) is more practical in the real-world [2-6]. Semi-Infinite Programs (SIP) are optimization problems with either infinitely many constraints or infinitely many variables (but not both). Recently semi-infinite programming was used in different subjects of operation research and other branches of science [7-11]. Fang et al. [12] proposed a Fuzzy Linear Semi-Infinite Programming (FLSIP) problem with uncertain data on constraints coefficients. Furthermore, Wu and et al. have used the cutting plane method to solve LSIP problems [13]. Moreover, Nasseri et al. [14] extended their model to the general type, where the coefficients of the objective function in the LSIP problem are also considered by a kind of fuzzy numbers. In particular, they used linear ranking functions to reduce the mentioned problems to the common LSIP problem. Next, Nasseri and Zavieh presented a multi-objective model with fuzzy data in the constraints and objective function [15]. Especially, they used a weighted method for solving their FLSIP model. Until now, to solve a FLSIP problem with fuzzy cost coefficients, in the first phase, its target function to be converted to a definite function by ranking function, and then the achieved model is solved. But, in this study, we will transform the fuzzy objective function into the Multi-Objective problem and we will present a new model for solving the FLSIP model (MOLSIP).

Semi-infinite programming problem have a lot of applications in different field of science, Uciński [16] extended a kind of branch and bound algorithm for sensor selection problem to achieve the maximum amount for objective function in traversing the branch-and-bound tree. Geng et al. [17] formulated a linear

\*Corresponding author, e-mail: hadi.zavieh@yahoo.com

semi-infinite programming (LSIP) for the selective body biasing problem and then an algorithm based on the novel concept of Incremental Hyper-cubic Sampling (IHCS) was proposed. Hale et al. [18] formulated fault detection and isolation (FDI) problem as semi-infinite program with implicit functions embedded to deal the maximize FDI effectiveness at the worst uncertainty condition. He et al. [19] suggested a bivariate interval semi-infinite linear programming (BV-ISIP) model with various uncertainties in functional relations and parameters. The model can be reduced the risk of system-failure because each of the constraints is satisfied under all possible levels of independent variables by BV-ISIP. Simić et al. [20] proposed a semi-infinite programming model with interval-parameter for tire management and planning, also uncertainty conditions are supported by the model. The model used for tire retreading industry. A summary of non-convex semi-infinite programming with applications can be found in [21]. Fuzzy sets have found applications in several contexts. One of the successful applications is related to the Data Envelopment Analysis (DEA) as highlighted, for example, in [22-25]. In this paper, we will investigate the envelopment form of the DEA model. Hence, we can solve the envelopment form of the DEA model in fuzzy conditions by using the FLSIP problem.

The goal of paper is propose MOLSIP model to solve the fuzzy linear programming problem. In this study for first time a fuzzy LP problem is changed to multi-objective problem and then solved by semi-infinite approach, in the real world MOLSIP can be help to solve multi-objective problems in uncertainty environment, Also the practical case gives us the efficiency of decision Making Unites (DMUs) under semi-infinite constraints. For this intent, in this paper, section 2 outlines the basic fuzzy concepts required for the next sections. In section 3, we propose a new Multi-Objective Linear Semi-Infinite Programming (MOLSIP) problem, and then we will show that such a problem can be reduced to a Linear Semi-Infinite Programming (LSIP) problem with three objective functions. The optimality conditions of solutions to FLSIP are investigated, and a numerical algorithm to solve the main problem is given. In Section 4, we propose a practical case, and a new method for solving the envelopment form of DEA in fuzzy conditions is considered. In section 5 numerical examples are presented. In section 6, the suggested model is compared with some other existing models and section 7 is the conclusion of the paper.

## 2. PRELIMINARIES AND THE NECESSARY CONCEPT

This section contains some basic concepts of fuzzy sets and DEA, which are directly related to our discussion in this paper and taken from [26-29].

### 2.1. Fuzzy Sets and Fuzzy Linear Programming

**Definition 1.** Let  $\mathbb{R}$  be the real line. A fuzzy set  $\tilde{A}$  in  $\mathbb{R}$  is defined to be a set of ordered pairs  $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) : x \in \mathbb{R}\}$ , where  $\mu_{\tilde{A}}(x)$  is called the membership function for the fuzzy set. The membership function maps each element of  $\mathbb{R}$  to a membership value between 0 and 1.

**Definition 2.** Assume  $\tilde{A}$  is a fuzzy set and  $\alpha \in (0,1]$ , then an  $\alpha$ -cut of  $\tilde{A}$  is defined as  $\{x | x \in \mathbb{R}, \mu_{\tilde{A}}(x) \geq \alpha\}$ , and we briefly denote it as  $\tilde{A}_\alpha$ .

**Definition 3.** A fuzzy set  $\tilde{A}$  is *convex*, if

$$\mu_{\tilde{A}}(\lambda x_1 + (1 - \lambda)x_2) \geq \min\{\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)\}, \quad x_1, x_2 \in X, \quad \lambda \in [0,1].$$

Alternatively, a fuzzy set is convex, if and only if all  $\alpha$ -cuts are convex, where  $X$  is a collection of objects which is  $X \subseteq \mathbb{R}$ .

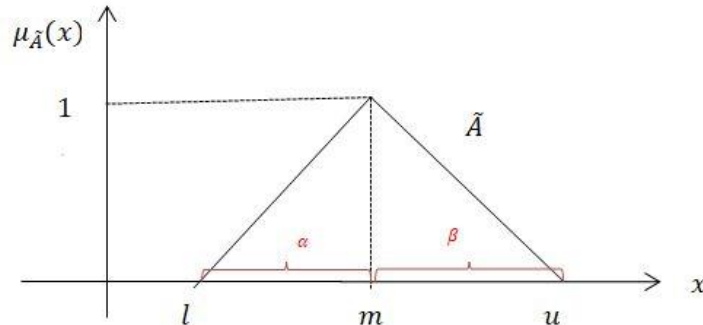
**Definition 4.** A fuzzy set  $\tilde{A}$  is *normal*, if there exists at least one  $x_0 \in \mathbb{R}$  such that  $\mu_{\tilde{A}}(x_0) = 1$ .

**Definition 5.** A fuzzy set  $\tilde{A}$  is a *fuzzy number*, if it has both properties normality and convexity.

**Definition 6.** The number  $\tilde{A}$  is a triangular fuzzy number, if its membership function is as follows:

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, & x \leq l, \\ \frac{x-l}{m-l}, & x \leq m, \alpha > 0, \\ \frac{u-x}{u-m}, & x \geq m, \beta > 0, \\ 0, & x \geq u. \end{cases}$$

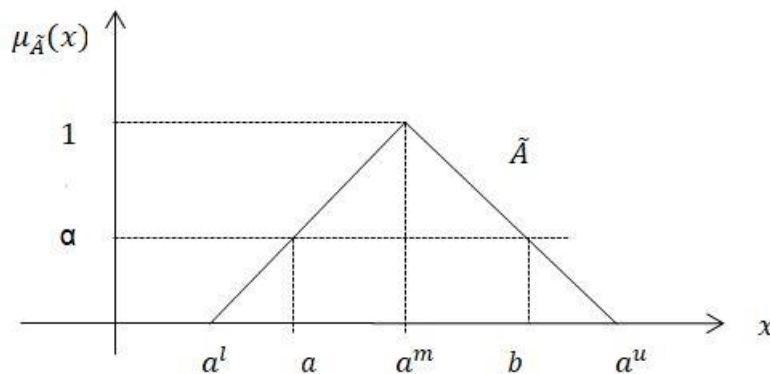
It is so simple to see that a triangular fuzzy number  $(l, m, u)$  can be rewritten by  $(m, \alpha, \beta)$ , where  $m - \alpha = l$  and  $m + \beta = u$ . The following figure is given to explain this notation.



**Figure 1.** A fuzzy number  $\tilde{A}$

These kinds of fuzzy numbers will be named a *fuzzy triangular number* (Figure 1).

In the above definition, when  $a = \inf \tilde{A}_\alpha$  and  $b = \sup \tilde{A}_\alpha$ , then  $\tilde{A}_\alpha = [a, b]$ . Figure 2 shows the  $\alpha$ -cut of  $\tilde{A}_\alpha$ , as introduced in the above definition.



**Figure 2.** The membership function and  $\alpha$ -cut of a fuzzy number  $\tilde{A}$

A triangular fuzzy number is shown as  $\tilde{A} = (a^l, a^m, a^u)$ , where  $a^m, a^l$  and  $a^u$  are respectively, the core, the lower, and the upper limits of support of the triangular fuzzy number  $\tilde{A}$ .

**Definition 7.** A fuzzy number  $\tilde{A}$  shall be called a (fuzzy triangular) zero number, symbolized by  $\tilde{0}$ , if it is written in the form  $\tilde{A} = \tilde{0} = (0,0,0)$ .

We denote the set of all fuzzy triangular numbers on  $\mathbb{R}$  by  $F(\mathbb{R})$ .

**Remark 1.** (i) If  $\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_n \in F(\mathbb{R})$ , then  $\tilde{M} = \tilde{A}_1 + \tilde{A}_2 + \dots + \tilde{A}_n \in F(\mathbb{R})$  and

$$\begin{aligned} m^- &= a_1 + a_2 + \dots + a_n, \\ m^+ &= b_1 + b_2 + \dots + b_n, \end{aligned}$$

where the interval  $[a_i, b_i]$  is the  $\alpha$ -cut of fuzzy number  $\tilde{A}_i$ , for  $i = 1, \dots, n$ , and the interval  $[m^-, m^+]$  is the  $\alpha$ -cut of fuzzy number  $\tilde{M}$ .

(ii) If  $\tilde{A} \in F(\mathbb{R})$  and  $k$  is a positive real number, then  $\tilde{M} = k \cdot \tilde{A} \in F(\mathbb{R})$  and

$$\begin{aligned} m^- &= k \cdot a, \\ m^+ &= k \cdot b, \end{aligned}$$

where the interval  $[a, b]$  is the  $\alpha$ -cut of fuzzy number  $\tilde{A}$ . In fact,  $\tilde{M}_\alpha = k \cdot \tilde{A}_\alpha$ .

(iii) If  $\tilde{A} \in F(\mathbb{R})$  and  $k$  is a negative real number, then  $\tilde{M} = k \cdot \tilde{A} \in F(\mathbb{R})$  and

$$\begin{aligned} m^- &= k \cdot b, \\ m^+ &= k \cdot a, \end{aligned}$$

where the interval  $[a, b]$  is the  $\alpha$ -cut of fuzzy number  $\tilde{A}$ .

(iv) If  $\tilde{A} \in F(\mathbb{R})$  and  $k = 0$ , then  $k \cdot \tilde{A} = \tilde{0}$ .

Here, we adopt the commonly used concept of  $\alpha$ -preference as defined in [6,7], and provide the following method for fuzzy ordering.

Since fuzzy ordering has a key role in solving fuzzy mathematical models, hence this study use follows the suggested approach by Fang et al. in [9].

**Definition 8.** If  $\tilde{A}_1, \tilde{A}_2 \in F(\mathbb{R})$  and  $\alpha \in (0,1]$ , then  $\tilde{A}_1 \geq_\alpha \tilde{A}_2$  if and only if

$$\begin{aligned} L_{\tilde{A}_1}(t) &\geq_\alpha L_{\tilde{A}_2}(t), \\ R_{\tilde{A}_1}(t) &\geq_\alpha R_{\tilde{A}_2}(t), \quad \forall t \in (\alpha, 1], \end{aligned}$$

where  $L$  is the lower bound of  $\alpha$ -cut, and  $R$  is the upper bound of  $\alpha$ -cut for fuzzy numbers.

Under the fuzzy ranking method mentioned above, given  $\alpha \in (0,1]$ , the FLP problem can be defined as follows:

$$\begin{aligned} \text{Min } \sum_{j=1}^n \tilde{c}_j x_j, & \tag{1} \\ \text{s.t: } \sum_{j=1}^n \tilde{a}_{ij} x_j &\geq_\alpha \tilde{b}_i, \quad i = 1, \dots, q \\ x_j &\geq 0, \quad j = 1, \dots, n, \end{aligned}$$

where  $\tilde{c}_j, \tilde{a}_{ij}, \tilde{b}_i \in F(\mathbb{R}), i = 1, \dots, q, j = 1, \dots, n$ .

**Lemma 1.** Constraints  $\sum_{j=1}^n \tilde{a}_{ij} x_j \geq_\alpha \tilde{b}_i$ , where  $x_j \geq 0$ , for all  $i = 1, \dots, q$  and  $j = 1, \dots, n$ , is equivalent to the following constraints:

$$\begin{aligned} \sum_{j=1}^n L_{\tilde{a}_{ij}}(t) x_j &\geq_\alpha L_{\tilde{b}_i}(t), \\ \sum_{j=1}^n R_{\tilde{a}_{ij}}(t) x_j &\geq_\alpha R_{\tilde{b}_i}(t), \quad \forall t \in [\alpha, 1]. \end{aligned}$$

**Proof.** The proof is straightforward by applying Definition 2.6. Since  $x_j \geq 0$ , for all  $j = 1, \dots, n$ , it is enough to use the ordering role which is given in Definition 2.6 for every constraint of Problem (1). ■

Now using the min operator action, we are going to present a new view for achieving an equivalent problem. For this aim, we see that an objective function  $\text{Min } \sum_{j=1}^n \tilde{c}_j x_j$ , where  $\tilde{c}_j = (c_j^l, c_j^m, c_j^u)$ ,  $j = 1, \dots, n$ , can be rewritten as follows:

$$\begin{aligned} \text{Max } Z_1 &= \sum_{j=1}^n (c_j^m - c_j^l) x_j \\ \text{Min } Z_2 &= \sum_{j=1}^n c_j^m x_j \\ \text{Min } Z_3 &= \sum_{j=1}^n (c_j^u - c_j^m) x_j. \end{aligned}$$

In fact since the left and right spread and the core of the triangular fuzzy number  $\tilde{c}_j$  are  $(c_j^m - c_j^l)$ ,  $(c_j^u - c_j^m)$  and  $c_j^m$  respectively. So, it is clear that the minimization of fuzzy function,  $\tilde{Z}_j = \sum_{j=1}^n \tilde{c}_j x_j$ , is

equal to minimizing the crisp functions  $\sum_{j=1}^n c_j^m x_j$  and  $\sum_{j=1}^n (c_j^u - c_j^m) x_j$  and also maximization of crisp function  $\sum_{j=1}^n (c_j^m - c_j^l) x_j$ . So, the proof is completed.

**Definition 9.** Semi-Infinite Programming (SIP) is an optimization model with finite variables and infinite constraints, or infinite variables and finite constraints. In the first case, The mathematical formulation is as follows:

$$\begin{aligned} & \text{Min}_{x \in X} f(x) \\ & \text{S. t: } g(x, y) \leq 0, \quad \forall y \in Y, \\ & \text{where} \\ & \quad f: \mathbb{R}^n \rightarrow \mathbb{R}, \\ & \quad g: \mathbb{R}^n * \mathbb{R}^m \rightarrow \mathbb{R}, \\ & \quad X \subseteq \mathbb{R}^n, \\ & \quad Y \subseteq \mathbb{R}^m. \end{aligned}$$

## 2.2. Data Envelopment Analysis

Data envelopment analysis is a non-parametric method for evaluating decision units (DMU) that suggested by Charnes et al. [28] and named the CCR model, the multiplier form of the CCR model is introduced as follows:

$$\begin{aligned} & \text{Max } \sum_{j=1}^n u_r y_{rj} \\ & \text{S. t: } \sum_{j=1}^n v_i x_{ij} = 1, \quad i = 1, \dots, q, \\ & \quad \sum_{j=1}^n u_r y_{rj} - \sum_{j=1}^n v_i x_{ij} \leq 0, \quad r = 1, \dots, z, \\ & \quad u_r, v_i \geq 0, \quad i = 1, \dots, q, \quad r = 1, \dots, z, \end{aligned} \quad (2)$$

where  $x_{ij}$  and  $y_{rj}$  are inputs and outputs of  $DMU_j$  and  $v_i, u_r$  are weight vectors of inputs and outputs, respectively. The dual form of model (2) is as follows:

$$\begin{aligned} & \text{Min } \theta_o \\ & \text{S. t: } \sum_{j=1}^n \lambda_j x_{ij} \leq \theta_o x_{io}, \quad i = 1, \dots, q, \\ & \quad \sum_{j=1}^n \lambda_j y_{rj} \geq y_{ro}, \quad r = 1, \dots, z, \\ & \quad \lambda_j \geq 0, \quad j = 1, \dots, n, \end{aligned} \quad (3)$$

where  $X_j = (x_{1j}, x_{2j}, \dots, x_{mj})^t$  and  $Y_j = (y_{1j}, y_{2j}, \dots, y_{sj})^t$  are the input vector and the output vector of the  $DMU_j$ , respectively.

## 3. MULTI-OBJECTIVE LINEAR SEMI-INFINITE PROGRAMMING (MOLSIP)

In this part, we propose a new approach for solving FLP problems. In the solution process, we need to solve the LSIP problem with several objectives. We denote this problem as MOLSIP in its abbreviated form. Hence, we will explain in the sequel how the FLP problem can be reduced to an LSIP model with several objectives. In this way, the appropriate technique and the solving approach to these well-known problems are presented. The above discussion can lead us to the following model

$$\begin{aligned} & \text{Max } Z_1 = \sum_{j=1}^n (c_j^m - c_j^l) x_j \\ & \text{Min } Z_2 = \sum_{j=1}^n c_j^m x_j \\ & \text{Min } Z_3 = \sum_{j=1}^n (c_j^u - c_j^m) x_j \\ & \text{s.t: } \sum_{j=1}^n L_{\tilde{a}_{ij}}(t) x_j \geq \alpha L_{\tilde{b}_i}(t), \quad \forall t \in [\hat{a}, 1], \quad i = 1, \dots, q, \\ & \quad \sum_{j=1}^n R_{\tilde{a}_{ij}}(t) x_j \geq \alpha R_{\tilde{b}_i}(t), \quad \forall t \in [\alpha, 1], \quad i = 1, \dots, q, \\ & \quad x_j \geq 0, \quad j = 1, \dots, n. \end{aligned} \quad (4)$$

**Collorary 3.1.** Problem (4) and the following problem are equivalent.

$$\begin{aligned}
 \text{Max } Z_1 &= \sum_{j=1}^n (c_j^m - c_j^l) x_j \\
 \text{Min } Z_2 &= \sum_{j=1}^n c_j^m x_j \\
 \text{Min } Z_3 &= \sum_{j=1}^n (c_j^u - c_j^m) x_j \\
 \text{s.t: } &\begin{pmatrix} f_{11}(t_1) & \cdots & f_{1n}(t_1) \\ \vdots & \ddots & \vdots \\ f_{w1}(t_w) & \cdots & f_{wn}(t_w) \end{pmatrix} (x_1, \dots, x_n)^t \leq \begin{pmatrix} b_1(t_1) \\ \vdots \\ b_w(t_w) \end{pmatrix}, \quad \forall t_i \in T, i = 1, \dots, w, \\
 &x_j \geq 0, \quad j = 1, \dots, n,
 \end{aligned} \tag{5}$$

where  $T$  is a compact metric space,  $f_{pj}(t)$ ,  $b_p(t)$ ,  $p = 1, \dots, w$ ,  $j = 1, \dots, n$  are real-valued continuous functions on  $T$ .

**Proof.** We just need to show that the constraints of both problems are the same. For this aim, we set:

- $f_{ij}(t) = L_{\tilde{a}_{ij}}(t)$ ,  $i = 1, \dots, q$ ,  $f_{ij}(t) = R_{\tilde{a}_{i-qj}}(t)$ ,  $i = q + 1, \dots, 2q$  and  $j = 1, \dots, n$ ,
- $b_i(t) = L_{\tilde{b}_i}(t)$ ,  $i = 1, \dots, q$ ,  $b_i(t) = R_{\tilde{b}_{i-q}}(t)$ ,  $i = q + 1, \dots, 2q$  and  $j = 1, \dots, n$ ,
- $w = 2q$ ,  $t \in [\alpha, 1]$ .

where  $\alpha \in (0,1]$ . Therefore, the result is obvious. ■

Problem (5) is a Multi-Objective Linear Semi-Infinite Programming (MOLSIP) problem with  $n$  variables and an infinite number of constraints. In the solution process, we will use the ‘‘cutting plane approach’’ for solving the LSIP problem. Problem (5) can be rewritten as follows:

$$\begin{aligned}
 \text{Max } Z_1 &= \sum_{j=1}^n (c_j^m - c_j^l) x_j \\
 \text{Min } Z_2 &= \sum_{j=1}^n c_j^m x_j \\
 \text{Min } Z_3 &= \sum_{j=1}^n (c_j^u - c_j^m) x_j \\
 \text{s.t: } &\begin{pmatrix} f_{11}(t_1^1) & \cdots & f_{1n}(t_1^1) \\ \vdots & \ddots & \vdots \\ f_{w1}(t_w^1) & \cdots & f_{wn}(t_w^1) \\ \text{---} & & \text{---} \\ \vdots & & \vdots \\ \text{---} & & \text{---} \\ f_{11}(t_1^k) & \cdots & f_{1n}(t_1^k) \\ \vdots & \ddots & \vdots \\ f_{w1}(t_w^k) & \cdots & f_{wn}(t_w^k) \end{pmatrix} (x_1, \dots, x_n)^t \geq \begin{pmatrix} b_1(t_1^1) \\ \vdots \\ b_w(t_w^1) \\ \text{---} \\ \vdots \\ \text{---} \\ b_1(t_1^k) \\ \vdots \\ b_w(t_w^k) \end{pmatrix}, \quad \forall t_i \in T, i = 1, \dots, w, \\
 &x_j \geq 0, \quad j = 1, \dots, n.
 \end{aligned} \tag{6}$$

**Definition.** *Constraint violation functions:* we consider each of constraint in model (6) as a function:

$$\sum_{j=1}^n f_{pj}(t) x_j^k - b_p(t), \quad \forall t \in T, \quad p = 1, \dots, w.$$

We will show this constraint with  $v_p^{k+1}(t)$ .

Let  $R^k$  be the feasible region of *MOLP* problem, which is defined in (3.3). If  $x^k = (x_1^k, x_2^k, \dots, x_n^k)$  is a Pareto optimal solution for Problem (3.3), then consider the ‘‘constraint violation functions’’ as follows:

$$v_p^{k+1}(t) = \sum_{j=1}^n f_{pj}(t) x_j^k - b_p(t), \quad \forall t \in T, \quad p = 1, \dots, w.$$

In the below, we present the algorithm for the suggested method

**Algorithm 3.1.**

**Assumption:** Assume that Problem (1) is given to be solved. Then, we consider its equivalent problem, that is, Problem (5).

For each of  $Z_1, Z_2$  and  $Z_3$ , do the following steps:

**Step 1:** Set  $k = 1$ ; choose any  $t_p^1 \in T$ ; set  $T_1 = \{t^1\} = (t_1^1, \dots, t_w^1)$ .

**Step 2:** Solve  $(LP)^k$  and obtain an optimal solution  $x^k$ .

**Step 3:** Find a minimizer  $t_p^{k+1}$  for  $v_p^{k+1}(t)$  over  $T$ , for  $p = 1, \dots, w$ .

**Step 4:** If  $v_p^{k+1}(t_p^{k+1}) \geq 0$ , for  $p = 1, \dots, w$ , then stop, with  $x^k$  is an optimal solution of LSIP. Otherwise, set  $T_{k+1} = T_k \cup \{t^{k+1}\}$  and  $k \leftarrow k + 1$ ; go to Step 1.

Using three times, the above algorithm will give us three optimal values for the three above mentioned objective functions. Now, using these values and solving the following problem, we obtain the optimal value of the MOLSIP problem

$$\begin{aligned} & \text{Max } \lambda & (7) \\ & \text{s.t. } \lambda \leq \mu_{Z_h}(x), \quad h = 1, 2, 3 \\ & \begin{pmatrix} f_{11}(t_1) & \cdots & f_{1n}(t_1) \\ \vdots & \ddots & \vdots \\ f_{w1}(t_w) & \cdots & f_{wn}(t_w) \end{pmatrix} (x_1, \dots, x_n)^t \leq \begin{pmatrix} b_1(t_1) \\ \vdots \\ b_w(t_w) \end{pmatrix}, \quad \forall t_i \in T, i = 1, \dots, w, \\ & x_j \geq 0, \quad j = 1, \dots, n. \end{aligned}$$

In the above model based on the achieve optimal values of the objective functions, we defined

$$Z_h^* = \max Z_h(x), \quad h = 1, 2, 3 \quad (8)$$

which are obtained by Algorithm 3.1, where  $Z_h(x)$ , is the  $h^{\text{th}}$  objective function ( $h = 1, 2, 3$ ), and

$$Z_h^- = \min_{h \neq j} \{Z_h(x_j^*)\}, \quad j = 1, 2, 3 \quad (9)$$

where  $x_j^*$  is the optimal value of each function obtained by Algorithm 4.1, for  $j = 1, 2, 3$ .

Now, based on relations (8) and (9), the membership function for each of the objective functions is defined as follows:

$$\mu_{Z_h}(x) = \frac{Z_h(x) - Z_h^-}{Z_h^* - Z_h^-}. \quad (10)$$

Finally, by solving model (7), we obtain the optimal fuzzy value for the model (6) with  $\lambda$  as a membership function value.

#### 4. A FUZZY DATA ENVELOPMENT ANALYSIS MODEL

In this section, by using Definition 2.8 we are presenting a model on the envelopment form of DEA in fuzzy condition. The input and output values of the model (3) are assumed to be in the form of fuzzy triangular numbers, and then the model can be rewritten as follows:

$$\begin{aligned} & \text{Min } \theta_o & (11) \\ & \text{S.t. } \theta_o \tilde{x}_{io} - \sum_{j=1}^n \lambda_j \tilde{x}_{ij} \geq \alpha \tilde{0}, \quad i = 1, \dots, q \\ & \quad \sum_{j=1}^n \lambda_j \tilde{y}_{rj} \geq \alpha \tilde{y}_{ro}, \quad r = 1, \dots, z \\ & \quad \lambda_j \geq 0, \quad j = 1, \dots, n. \end{aligned}$$

It is clear that using Definition 2.8 and Lemma 2.1, the above problem and the following problem are equivalent

$$\begin{aligned} & \text{Min } \theta_o & (12) \\ & \text{S.t. } \theta_o L_{\tilde{x}_{io}}(t) - \sum_{j=1}^n \lambda_j L_{\tilde{x}_{ij}}(t) \geq \alpha \tilde{0}, \quad i = 1, \dots, q \end{aligned}$$

$$\begin{aligned} \sum_{j=1}^n \lambda_j L_{\tilde{y}_{rj}}(t) &\geq_{\alpha} L_{\tilde{y}_{ro}}(t), \quad r = 1, \dots, z \\ \theta_0 R_{\tilde{x}_{i-q_0}}(t) - \sum_{j=1}^n \lambda_j R_{\tilde{x}_{i-q_j}}(t) &\geq_{\alpha} \tilde{0}, \quad i = 1, \dots, q \\ \sum_{j=1}^n \lambda_j R_{\tilde{x}_{r-z_j}}(t) &\geq_{\alpha} R_{\tilde{y}_{r-z_0}}(t), \quad r = 1, \dots, z \\ \lambda_j &\geq 0, \quad j = 1, \dots, n. \end{aligned}$$

By considering  $f_{pj}(t)$ ,  $f_{p_0}(t)$ ,  $p = 1, \dots, w$ ,  $j = 1, \dots, n$  and  $f_{hj}(t)$ ,  $f_{h_0}(t)$ ,  $h = 1, \dots, s$ ,  $j = 1, \dots, n$  and by using Collorary 3.1 we have:

$$\begin{aligned} f_{i_0}^1(t) &\triangleq L_{\tilde{x}_{i_0}}(t), & i = 1, \dots, q, & f_{i_0}^1(t) \triangleq R_{\tilde{x}_{i-q_0}}(t), & i = q + 1, \dots, 2q, \\ f_{ij}^2(t) &\triangleq L_{\tilde{x}_{ij}}(t), & i = 1, \dots, q, & f_{ij}^2(t) \triangleq R_{\tilde{x}_{i-q_j}}(t), & i = q + 1, \dots, 2q, \end{aligned}$$

$j = 1, \dots, n$ ,  $w \triangleq 2q$ ,  $t \in [\alpha, 1]$ .

$$\begin{aligned} f_{rj}^3(t) &\triangleq L_{\tilde{y}_{rj}}(t), & r = 1, \dots, z, & f_{rj}^3(t) \triangleq R_{\tilde{x}_{r-z_j}}(t), & r = z + 1, \dots, 2z, \\ f_{r_0}^4(t) &\triangleq L_{\tilde{y}_{r_0}}(t), & r = 1, \dots, z, & f_{r_0}^4(t) \triangleq R_{\tilde{y}_{r-z_0}}(t), & r = z + 1, \dots, 2z, \end{aligned}$$

$$j = 1, \dots, n, \quad s \triangleq 2z, \quad t \in [\alpha, 1].$$

where  $L_{\tilde{x}_{i_0}}$ ,  $L_{\tilde{x}_{ij}}$ ,  $L_{\tilde{y}_{rj}}$ ,  $L_{\tilde{y}_{r_0}}$  are the lower bound of  $\alpha$ -cut for fuzzy numbers  $\tilde{x}_{i_0}$ ,  $\tilde{x}_{ij}$ ,  $\tilde{y}_{rj}$  and  $\tilde{y}_{r_0}$  respectively, and  $R_{\tilde{x}_{i-q_0}}$ ,  $R_{\tilde{x}_{i-q_j}}$ ,  $R_{\tilde{x}_{r-z_j}}$ ,  $R_{\tilde{y}_{r-z_0}}$  are the upper bound of  $\alpha$ -cut for fuzzy numbers  $\tilde{x}_{i-q_0}$ ,  $\tilde{x}_{i-q_j}$ ,  $\tilde{x}_{r-z_j}$  and  $\tilde{y}_{r-z_0}$  respectively. The new DEA model is written as follows:

$$\begin{aligned} \text{Min } \theta_0 & \tag{13} \\ \text{S.t: } & \begin{pmatrix} f_{1_0}^1(t_1) & -f_{11}^2(t_1) & \dots & -f_{1n}^2(t_1) \\ f_{2_0}^1(t_2) & -f_{21}^2(t_2) & \dots & -f_{2n}^2(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_{w_0}^1(t_w) & -f_{w1}^2(t_w) & \dots & -f_{wn}^2(t_w) \\ \hline 0 & f_{11}^3(t_1) & \dots & f_{1n}^3(t_1) \\ 0 & f_{21}^3(t_2) & \dots & f_{2n}^3(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & f_{s1}^3(t_w) & \dots & f_{sn}^3(t_w) \end{pmatrix} \begin{pmatrix} \theta \\ \lambda_1 \\ \lambda^2 \\ \vdots \\ \lambda_n \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \hline f_{1_0}^4(t_1) \\ f_{2_0}^4(t_2) \\ \vdots \\ f_{s_0}^4(t_w) \end{pmatrix}, \\ & \forall t_i \in T, i = 1, \dots, w, \lambda_j \geq 0, j = 1, \dots, n. \end{aligned}$$

By solving the above model, we obtain the efficiency level of Decision Making Unit  $O$  ( $DMU_0$ ).

### 5. NUMERICAL EXAMPLES

Here, we are going to explore solving procedure of the FLP problem based on the presented approaches.

#### Example 5.1. (Case Study)

In a production factory two machines  $M_1$  and  $M_2$  working on two products  $P_1$  and  $P_2$ . The uncertain daily time required and the capacity of each machine for the production of each product comes in Table 1:

**Table 1.** The time required per units and machine capacity

Machines	Time per unit (minute)		Machine capacity (hour/day)
	$P_1$	$P_2$	
$M_1$	(0.5, 1.5, 1.7)	(1.5, 2, 2.5)	(3, 5, 9)
$M_2$	(1.2, 2, 2.3)	(0.5, 1, 1.2)	(4, 5, 8)



Since the price of each product can vary because of market fluctuations, the factory considered the profit of  $P_1$  close to 8 and for  $P_2$  close to 4. The factory wants to determine the number of products it must produce per day to achieve maximum profits.

Since the profit and time are ambiguous, so the number of units will also be inaccurate. The problem can be written as an FLP problem. Fuzzy triangular numbers for each inaccurate value are considered.

Since we suppose the profit of  $P_1$  is close to 8 so it can be modeled as (6, 8, 9), and  $P_2$  can be modeled as (3,4,7). Therefore, we formulate the FLP problem as follows:

$$\begin{aligned} \text{Max} Z &= (6,8,9)x_1 + (3,4,7)x_2 \\ \text{s.t.} & \quad (0.5,1.5,1.7)x_1 + (1.5,2,2.5)x_2 \leq_{\alpha} (3,5,9), \\ & \quad (1.2,2,2.3)x_1 + (0.5,1,1.2)x_2 \leq_{\alpha} (4,5,8), \\ & \quad x_1, x_2 \geq 0. \end{aligned}$$

Now, by Theorem 3.1 and Corollary 3.1, the model can be reduced as follows:

$$\begin{aligned} \text{Min } Z_1 &= 2x_1 + x_2 \\ & \quad \text{Max } Z_2 = 8x_1 + 4x_2 \\ & \quad \text{Max } Z_3 = x_1 + 3x_2 \\ \text{S.t.} & \quad \begin{pmatrix} t_1 + 0.5 & 0.5t_1 + 1.5 \\ 0.8t_2 + 1.2 & 0.5t_2 + 0.5 \\ -0.2t_3 + 1.7 & -0.5t_3 + 2.5 \\ -0.3t_4 + 2.3 & -0.2t_4 + 1.2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 2t_1+3 \\ t_2+4 \\ -4t_3+9 \\ -3t_4+8 \end{pmatrix}, \quad \forall t_i \in [\alpha, 1], \\ & \quad x_1, x_2 \geq 0. \end{aligned}$$

In this example, we assume that  $\alpha = 0.6$  and  $t^1 = (t_1^1, t_2^1, t_3^1, t_4^1) = (0.7, 0.8, 0.7, 0.8)$  as an optional point, then we have the regular multi-objective linear program,

$$\begin{aligned} \text{Min } Z_1 &= 2x_1 + x_2 \\ & \quad \text{Max } Z_2 = 8x_1 + 4x_2 \\ & \quad \text{Max } Z_3 = x_1 + 3x_2 \\ \text{S.t.} & \quad \begin{pmatrix} t_1^1 + 0.5 & 0.5t_1^1 + 1.5 \\ 0.8t_2^1 + 1.2 & 0.5t_2^1 + 0.5 \\ -0.2t_3^1 + 1.7 & -0.5t_3^1 + 2.5 \\ -0.3t_4^1 + 2.3 & -0.2t_4^1 + 1.2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 2t_1^1+3 \\ t_2^1+4 \\ -4t_3^1+9 \\ -3t_4^1+8 \end{pmatrix}, \quad \forall t_i \in [\alpha, 1], \\ & \quad x_1, x_2 \geq 0. \end{aligned}$$

By solving the current MOLP problem, for every objective function, we have:

$$\begin{aligned} Z_1^* &= 0 \text{ and } x^1 = (x_1^1, x_2^1) = (0,0) \\ Z_2^* &= 20.957 \text{ and } x^1 = (x_1^1, x_2^1) = (2.117, 1.005) \\ Z_3^* &= 7.135 \text{ and } x^1 = (x_1^1, x_2^1) = (0, 2.378). \end{aligned}$$

Define

$$\begin{aligned} v_1^2(t_1) &= (-t_1 - 0.5)x_1^1 + (-0.5t_1 - 1.5)x_2^1 - (-2t_1 - 3) \\ v_2^2(t_2) &= (-0.8t_2 - 1.2)x_1^1 + (-0.5t_2 - 0.5)x_2^1 - (-t_2 - 4) \\ v_3^2(t_3) &= (0.2t_3 - 1.7)x_1^1 + (0.5t_3 - 2.5)x_2^1 - (4t_3 - 9) \\ v_4^2(t_4) &= (0.3t_4 - 2.3)x_1^1 + (0.2t_4 - 1.2)x_2^1 - (3t_4 - 8) \end{aligned}$$

- For problem  $Z_1$  with  $x^1 = (x_1^1, x_2^1) = (0, 0)$  :

$$v_1^2(t_1) = 2t_1 + 3, v_2^2(t_2) = t_2 + 4, v_3^2(t_3) = -4t_3 + 9 \text{ and } v_4^2(t_4) = -3t_4 + 8.$$

The minimizers of  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4)$  over  $[\alpha, 1]$  are  $(0.6, 0.6, 1, 1)$ , respectively.

Hence, we choose  $t^2 = (t_1^2, t_2^2, t_3^2, t_4^2) = (0.6, 0.6, 1, 1)$ . Since  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4) \geq 0$ , the algorithm stops, and it concludes that the optimal solution is  $x^* = x^1 = (0, 0)$  with  $\alpha = 0.6$ .

- For problem  $Z_2$  with  $x^1 = (x_1^1, x_2^1) = (2.117, 1.005)$  :

$$v_1^2(t_1) = -0.619t_1 + 0.434, v_2^2(t_2) = -1.119t_2 + 0.958, v_3^2(t_3) = -3.075t_3 + 2.888 \text{ and } v_4^2(t_4) = -2.164t_4 + 1.924.$$

The minimizers of  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4)$  over  $[\alpha, 1]$  are  $(1, 1, 1, 1)$ , respectively.

Hence, we choose  $t^2 = (t_1^2, t_2^2, t_3^2, t_4^2) = (1, 1, 1, 1)$ . Since  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4) \geq 0$ , the algorithm iterates with a new linear program,

$$\begin{aligned} & \text{Max } Z_2 = 8x_1 + 4x_2 \\ & \text{S.t: } \begin{pmatrix} t_1^1 + 0.5 & 0.5t_1^1 + 1.5 \\ 0.8t_2^1 + 1.2 & 0.5t_2^1 + 0.5 \\ -0.2t_3^1 + 1.7 & -0.5t_3^1 + 2.5 \\ -0.3t_4^1 + 2.3 & -0.2t_4^1 + 1.2 \\ \dots & \dots \\ t_1^2 + 0.5 & 0.5t_1^2 + 1.5 \\ 0.8t_2^2 + 1.2 & 0.5t_2^2 + 0.5 \\ -0.2t_3^2 + 1.7 & -0.5t_3^2 + 2.5 \\ -0.3t_4^2 + 2.3 & -0.2t_4^2 + 1.2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 2t_1^1 + 3 \\ t_2^1 + 4 \\ -4t_3^1 + 9 \\ -3t_4^1 + 8 \\ \dots \\ 2t_1^2 + 3 \\ t_2^2 + 4 \\ -4t_3^2 + 9 \\ -3t_4^2 + 8 \end{pmatrix}, \\ & x_1, x_2 \geq 0. \end{aligned}$$

Solving  $(LP^2)$  in the optimal solution

$$Z_2^* = 20 \text{ and } x^2 = (x_1^2, x_2^2) = (2.5, 0).$$

Define

$$\begin{aligned} v_1^3(t_1) &= (-t_1 - 0.5)x_1^2 + (-0.5t_1 - 1.5)x_2^2 - (-2t_1 - 3) = -0.5t_1 + 1.75 \\ v_2^3(t_2) &= (-0.8t_2 - 1.2)x_1^2 + (-0.5t_2 - 0.5)x_2^2 - (-t_2 - 4) = -t_2 + 1 \\ v_3^3(t_3) &= (0.2t_3 - 1.7)x_1^2 + (0.5t_3 - 2.5)x_2^2 - (4t_3 - 9) = -3.5t_3 + 4.75 \\ v_4^3(t_4) &= (0.3t_4 - 2.3)x_1^2 + (0.2t_4 - 1.2)x_2^2 - (3t_4 - 8) = -2.25t_4 + 2.25. \end{aligned}$$

The minimizers of  $v_1^3(t_1), v_2^3(t_2), v_3^3(t_3), v_4^3(t_4)$  over  $[\alpha, 1]$  are  $(1, 1, 1, 1)$ , respectively.

Hence, we choose  $t = (t_1^3, t_2^3, t_3^3, t_4^3) = (1, 1, 1, 1)$ . Since  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4) \geq 0$ , the algorithm stops and  $x^* = x^2 = (2.5, 0)$  is the optimal solution with  $\alpha = 0.6$ .

- For problem  $Z_3$  with  $x^1 = (x_1^1, x_2^1) = (0, 2.378)$  :

$$v_1^2(t_1) = 0.811t_1 - 0.567, v_2^2(t_2) = -0.189t_2 + 2.811, v_3^2(t_3) = -2.811t_3 + 3.055 \text{ and } v_4^2(t_4) = -2.524t_4 + 5.149.$$

The minimizers of  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4)$  over  $[\alpha, 1]$  are  $(0.6, 1, 1, 1)$ , respectively.

Hence, we choose  $t^2 = (t_1^2, t_2^2, t_3^2, t_4^2) = (0.6, 1, 1, 1)$ . Since  $v_1^2(t_1) \geq 0$  and  $v_2^2(t_2), v_3^2(t_3), v_4^2(t_4) \geq 0$ , the algorithm should iterate with a new linear program,

$$\text{Max } Z_3 = x_1 + 3x_2$$

$$\text{S.t: } \begin{pmatrix} t_1^1 + 0.5 & 0.5t_1^1 + 1.5 \\ 0.8t_2^1 + 1.2 & 0.5t_2^1 + 0.5 \\ -0.2t_3^1 + 1.7 & -0.5t_3^1 + 2.5 \\ -0.3t_4^1 + 2.3 & -0.2t_4^1 + 1.2 \\ \dots \dots \dots & \dots \dots \dots \\ t_1^2 + 0.5 & 0.5t_1^2 + 1.5 \\ 0.8t_2^2 + 1.2 & 0.5t_2^2 + 0.5 \\ -0.2t_3^2 + 1.7 & -0.5t_3^2 + 2.5 \\ -0.3t_4^2 + 2.3 & -0.2t_4^2 + 1.2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 2t_1^1+3 \\ t_2^1+4 \\ -4t_3^1+9 \\ -3t_4^1+8 \\ \dots \dots \dots \\ 2t_1^2+3 \\ t_2^2+4 \\ -4t_3^2+9 \\ -3t_4^2+8 \end{pmatrix},$$

$$x_1, x_2 \geq 0.$$

Solving ( $LP^2$ ) concludes the optimal solution as follows:

$$Z_3^* = 7 \text{ and } x^2 = (x_1^2, x_2^2) = (0, 2.333).$$

Define,

$$\begin{aligned} v_1^3(t_1) &= (-t_1 - 0.5)x_1^2 + (-0.5t_1 - 1.5)x_2^2 - (-2t_1 - 3) = -0.834t_1 - 0.5 \\ v_2^3(t_2) &= (-0.8t_2 - 1.2)x_1^2 + (-0.5t_2 - 0.5)x_2^2 - (-t_2 - 4) = -0.166t_2 + 2.834 \\ v_3^3(t_3) &= (0.2t_3 - 1.7)x_1^2 + (0.5t_3 - 2.5)x_2^2 - (4t_3 - 9) = -2.834t_3 + 3.168 \\ v_4^3(t_4) &= (0.3t_4 - 2.3)x_1^2 + (0.2t_4 - 1.2)x_2^2 - (3t_4 - 8) = -2.533t_4 + 5.204. \end{aligned}$$

The minimizers of  $v_1^3(t_1), v_2^3(t_2), v_3^3(t_3), v_4^3(t_4)$  over  $[\alpha, 1]$  are  $(0.6, 1, 1, 1)$  respectively.

Hence, we choose  $t = (t_1^3, t_2^3, t_3^3, t_4^3) = (0.6, 1, 1, 1)$ . Since  $v_1^2(t_1), v_2^2(t_2), v_3^2(t_3), v_4^2(t_4) \geq 0$ , the algorithm stops, and we obtain the optimal solution as  $x^* = x^2 = (0, 2.333)$  with  $\alpha = 0.6$ .

Now, we have the optimal values of the three objective functions:

$$\begin{aligned} Z_1^* &= 0 \text{ and } x^* = (x_1^*, x_2^*) = (0, 0) \\ Z_2^* &= 20 \text{ and } x^* = (x_1^*, x_2^*) = (2.5, 0) \\ Z_3^* &= 7 \text{ and } x^* = (x_1^*, x_2^*) = (0, 2.333). \end{aligned}$$

And we can calculate the amount of each function while using the optimal value of other functions

$$Z_1(x_2^*) = 5, Z_1(x_3^*) = 2.333, Z_2(x_1^*) = 0, Z_2(x_3^*) = 9.332, Z_3(x_1^*) = 0, Z_3(x_2^*) = 2.5$$

Calculate  $Z_i^- = \text{Min}_{i \neq j} \{f_i(x_j^*)\}$ :

$$Z_1^- = 5, \quad Z_2^- = 0, \quad Z_3^- = 0.$$

Write a membership function for three functions as follow:

$$\mu_{Z_1}(x) = \frac{Z_1(x) - 5}{-5}, \quad \mu_{Z_2}(x) = \frac{Z_2(x)}{20}, \quad \mu_{Z_3}(x) = \frac{Z_3(x)}{7}.$$

Then, problem (7) is defined here as:

$$\begin{aligned} \text{Max } & \lambda \\ \text{s.t: } & \lambda \leq \frac{2x_1 + x_2 - 5}{-5}, \\ & \lambda \leq \frac{8x_1 + 4x_2}{20}, \end{aligned}$$

$$\begin{aligned} \lambda &\leq \frac{x_1 + 3x_2}{7}, \\ 1.2x_1 + 1.85x_2 &\leq 4.4, \\ 1.84x_1 + 0.9x_2 &\leq 4.8, \\ 1.56x_1 + 2.15x_2 &\leq 6.2, \\ 2.06x_1 + 1.04x_2 &\leq 5.6, \\ 1.5x_1 + 2x_2 &\leq 5, \\ 2x_1 + x_2 &\leq 5, \\ 1.1x_1 + 1.8x_2 &\leq 4.2, \\ x_1, x_2 &\geq 0. \end{aligned}$$

Finally, we obtain the following solution with  $\lambda$  membership function value for the original problem:

$$x_1^* = 0.8, \quad x_2^* = 0.9, \quad \lambda = 0.5 .$$

**Example 5.2.**

Assume that four production units use two inputs to produce an output. The fuzzy inputs and output of each production unit are listed in Table 2:

**Table 2.** Decision-Making Units with fuzzy inputs and output

inputs & output Units	$\tilde{X}_1$	$\tilde{X}_2$	$\tilde{y}$
$D_1$	(1,2,3)	(12,13,15)	(69,72,73)
$D_2$	(8,10,13)	(21,22,24)	(81,85,88)
$D_3$	(6,7,9)	(18,19,20)	(72,74,75)
$D_4$	(15,17,18)	(33,34,36)	(97,99,100)

We would like to achieve the efficiency level of each production unit. For this purpose, we define the following problem:

$$\begin{aligned} & \text{Min } \theta_0 \\ \text{S.t.} & \begin{pmatrix} t_1^1 + 1 & -t_1^1 - 1 & -2t_1^1 - 8 & -t_1^1 - 6 & -2t_1^1 - 15 \\ t_2^1 + 12 & -t_2^1 - 12 & -t_2^1 - 21 & -t_2^1 - 18 & -t_2^1 - 33 \\ -t_3^1 + 3 & t_3^1 - 3 & 3t_3^1 - 13 & 2t_3^1 - 9 & t_3^1 - 18 \\ -2t_4^1 + 15 & 2t_4^1 - 15 & 2t_4^1 - 24 & t_4^1 - 20 & 2t_4^1 - 36 \\ \hline 0 & 3t_5^1 + 69 & 4t_5^1 + 81 & 2t_5^1 + 72 & 2t_5^1 + 97 \\ 0 & -t_6^1 + 73 & -3t_6^1 + 88 & -t_6^1 + 75 & -t_6^1 + 100 \end{pmatrix} \begin{pmatrix} \theta \\ \lambda^1 \\ \lambda_2 \\ \lambda_3 \\ \lambda^4 \end{pmatrix} \geq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 3t + 69 \\ -t + 73 \end{pmatrix} \\ & \lambda_j \geq 0, \quad j = 1, \dots, n. \end{aligned}$$

Let  $t^1 = (t_1^1, t_2^1, t_3^1, t_4^1, t_5^1, t_6^1) = (0.7, 0.8, 0.7, 0.8, 0.7, 0.8)$  an arbitrary point, the results of the proposed method are as follows:

**Table 3.** The efficiency of Decision Making Units

Units	Efficiency	Lambdas
$D_1$	$\theta_1 = 1$	$\lambda_1 = 1$
$D_2$	$\theta_2 = 0.71$	$\lambda_1 = 1.19$
$D_3$	$\theta_3 = 0.72$	$\lambda_1 = 1.04$
$D_4$	$\theta_4 = 0.54$	$\lambda_1 = 1.38$

The results show that  $v_s^2(t_s) \geq 0, s = 1, \dots, 6$ , based on the proposed method, the first decision-making unit ( $DMU_1$ ) is the efficient unit ( $\theta_1 = 1$ ) between the decision units. The performance of units is as follows:

$DMU_1 > DMU_3 > DMU_2 > DMU_4$ . (Table 3)

## 6. COMPARISONS ANALYSIS

By comparing our model with Fang et al in [12] we are noted that:

1. Our proposed model considers the LSIP with fuzzy data in the objective function and constraints, and a solution is proposed for it. In contrast, the model presented by Fang et al. has not proposed a solution to the problem with fuzzy cast objective.
2. The model presented by Fang et al. for the semi-infinite problem with one-objective is presented, while our proposed model can handle multi-objective linear semi-infinite programming problem which is more realistic and complete.

Also by comparing our suggested model with Nasseri et al. [14] we founded that:

Nasseri et al. have used the ranking functions method to solve a semi-infinite problem with a fuzzy objective function, while the proposed method converts a fuzzy problem into a multi-objective linear semi-infinite programming problem, which is more practical, and then solve it.

## 7. CONCLUSION

In this study, we considered an LP problem with fuzzy data in objective and constraints. We presented a new model denoted as the "Multi-objective linear Semi-Infinite Programming (MOLSIP)" problem. By using  $\alpha$ -preference, we have shown that such a problem can be reduced to three linear semi-infinite programming problems. Then, we used the cutting plane algorithm for solving these three LP problems. Finally, from the MOLSIP problem, we obtained the values of the three objective functions that have been sub sequentially used to determine the optimal solution of the original problem. Furthermore, we emphasize that the proposed model and the described approach can be useful to solve some real applications such as the performance evaluation of the decision-making units within the DEA approach; this method can be extended for other models of DEA for future works.

## CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

## REFERENCES

- [1] Bellman, R.E., Zadeh, L. A., "Decision making in a fuzzy environment", Management Science, 17: 141-164, (1970).

- [2] Sharif Uddin, M., Miah, M., Al-Amin Khan, M., AlArjani, A., “A solving approach for fuzzy multi-objective linear fractional programming and application to an agricultural planting structure optimization problem”, *Chaos, Solitons & Fractals*, 141: 110352, (2020).
- [3] Efe, B., Efe, Ö. F., Kurt, M., “An integrated intuitionistic fuzzy set and mathematical programming approach for an occupational health and safety policy”, *Gazi University Journal of Science*, 30(2): 73 – 95, (2017).
- [4] Nafei, A., Yuan, W., Nasser, H., “A New Method for Solving Interval Neutrosophic Linear Programming Problems”, *Gazi University Journal of Science*, 33(4): 796 – 808, (2020).
- [5] Yang, G., Li, X., Huo, L., Liu, Q., “Goal programming tactic for uncertain multi-objective transportation problem using fuzzy linear membership function”, *Alexandria Engineering Journal*, 60(2): 2525-2533, (2021).
- [6] Zimmermann, H.J., “Fuzzy programming and linear programming with several objective functions”, *Fuzzy Sets and Systems*, 1: 45-55, (1978).
- [7] Gadhi, N.A., “Necessary optimality conditions for a nonsmooth semi-infinite programming problem”, *Journal of Global Optimization*, 74(1): 161-168, (2019).
- [8] Goberna, M.A., Gómez, S., Guerra, F., Todorov, M.I., “Sensitivity analysis in linear semi-infinite programming: perturbing cost and right-hand-side coefficients”, *European Journal of Operational Research*, 181: 1069-1085, (2007).
- [9] Jian, P.P., Li, L., XuZe, H. X., “An infeasible bundle method for nonconvex constrained optimization with application to semi-infinite programming problems”, *Numerical Algorithms*, 80(2): 397-427, (2019).
- [10] Joshi, B.C., Mishra, Sh.K., Kumar, P., “On semi-infinite mathematical programming problems with equilibrium constraints using generalized convexity”, *Journal of the Operations Research Society of China*, 29: 1-18, (2019).
- [11] Papp, D., “Semi-Infinite Programming”, *Wiley StatsRef: Statistics Reference Online*, 1-14, (2019).
- [12] Fang, S.C., Hu, C.F., Wang, H.F., Wu, S.Y., “Linear programming with fuzzy coefficients in constraints”, *Computers and Mathematics with Applications*, 37: 63-76, (1999).
- [13] Wu, S.Y., Fang, S.Ch., Lin, Ch.J., “Analytic center based cutting plane method for linear semi-infinite programming”, In López, M.A. *Semi-Infinite Programming: Recent Advances*, Kluwer, Dordrecht, 221–233, (2001).
- [14] Nasser, S.H., Behmanesh, E., Faraji, P., Fallahzadeh Shahabi N., “Semi-infinite programming to solve linear programming with triangular fuzzy coefficients”, *Annals of Fuzzy Mathematics and Informatics*, 1: 213-226, (2013).
- [15] Nasser, S.H., Zavieh, H., “A multi-objective method for solving fuzzy linear programming based on semi-infinite models”, *Fuzzy Information and Engineering*, 10: 95-102, (2018).
- [16] Uciński, D., “Sensor Selection with Non-smooth Design Criteria Based on Semi-Infinite Programming”, *IFAC-PapersOnLine*, 53(2): 75-39-7544, (2020).
- [17] Geng, H., Liu, J., Wen-Luo, P., Cheng, L., Grant, S., Shi, Y., “Selective body biasing for post-silicon tuning of sub-threshold designs: A semi-infinite programming approach with Incremental Hyper-cubic Sampling”, *Integration*, 55: 465-473, (2016).

- [18] Hale, W. T., Wlihelm, M. E., Palmer, K. A., Stuber, M. D., Bollas, G. M., “Semi-infinite programming for global guarantees of robust fault detection and isolation in safety-critical systems”, *Computers & Chemical Engineering*, 126: 218-230, (2019).
- [19] He, L., Huang, G., Lu, H., “Bivariate interval semi-infinite programming with an application to environmental decision-making analysis”, *European Journal of Operational Research*, 211(3): 452-465, (2011).
- [20] Simić, V., Dabić-Ostojić, S., Bojović, N., “Interval-parameter semi-infinite programming model for used tire management and planning under uncertainty”, *Computers & Industrial Engineering*, 113: 487-501, (2017).
- [21] Djelassi, H., Mitsos, A., Stein, O., “Recent advances in nonconvex semi-infinite programming: Applications and algorithms”, *EURO Journal on Computational Optimization*, 9: 100006, (2021).
- [22] Agarwal, Sh., “Efficiency Measure by fuzzy Data Envelopment analysis Model”, *Fuzzy Information and Engineering*, 6: 59-70, (2014).
- [23] Gölcükcü, A., “Fuzzy DEA, Banking, Customer, Optimist and Pessimist Approaches”, *Gazi University Journal of Science*, 28 (4): 561 – 569, (2015).
- [24] Jess, A., Jongen, H.Th., Neralic, L. Stein, O., “A semi-infinite programming model in data envelopment analysis”, *Optimization*, 49: 369-385, (2001).
- [25] Zhanxin, M. Ma., Wei, C., “Generalized fuzzy data envelopment analysis methods”, *Applied Soft Computing*, 19: 215-225, (2014).
- [26] Buckley, J.J., “A fast method of ranking alternatives using fuzzy numbers”, *Fuzzy Sets and Systems*, 30:337-348, (1989).
- [27] Cao, B.Y., “Optimal Models and Methods with Fuzzy Quantities”, *Studies in Fuzziness and Soft Computing*, volume 248, Springer-verlag, Berlin, Heidelberg, (2010).
- [28] Charnes, A., Cooper, W. W., Rhodes, E., “Measuring the efficiency of decision-making units”, *European Journal of Operational Research*, 2: 29-44, (1978).
- [29] Hettich, R., Kortanek, K.O., “Semi-infinite programming: Theory, methods and applications”, *SIAM Review*, 35: 380-429, (1993).