





Review

# Lexicographic Methods for Fuzzy Linear Programming

Boris Pérez-Cañedo <sup>1,\*</sup>, José Luis Verdegay <sup>2,\*</sup>, Eduardo René Concepción-Morales <sup>3</sup>  
and Alejandro Rosete <sup>4</sup>

<sup>1</sup> Department of Mathematics, University of Cienfuegos, Cienfuegos 55100, Cuba

<sup>2</sup> Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain

<sup>3</sup> Department of Informatics, University of Cienfuegos, Cienfuegos 55100, Cuba; econcep@ucf.edu.cu

<sup>4</sup> Facultad de Ingeniería Informática, Universidad Tecnológica de La Habana José Antonio Echeverría (Cujae), Marianao La Habana 19390, Cuba; rosete@ceis.cujae.edu.cu

\* Correspondence: bpcanedo@ucf.edu.cu (B.P.-C.); verdegay@decsai.ugr.es (J.L.V.)

Received: 22 August 2020; Accepted: 7 September 2020; Published: 9 September 2020



**Abstract:** Fuzzy Linear Programming (FLP) has addressed the increasing complexity of real-world decision-making problems that arise in uncertain and ever-changing environments since its introduction in the 1970s. Built upon the Fuzzy Sets theory and classical Linear Programming (LP) theory, FLP encompasses an extensive area of theoretical research and algorithmic development. Unlike classical LP, there is not a unique model for the FLP problem, since fuzziness can appear in the model components in different ways. Hence, despite fifty years of research, new formulations of FLP problems and solution methods are still being proposed. Among the existing formulations, those using fuzzy numbers (FNs) as parameters and/or decision variables for handling inexactness and vagueness in data have experienced a remarkable development in recent years. Here, a long-standing issue has been how to deal with FN-valued objective functions and with constraints whose left- and right-hand sides are FNs. The main objective of this paper is to present an updated review of advances in this particular area. Consequently, the paper briefly examines well-known models and methods for FLP, and expands on methods for fuzzy single- and multi-objective LP that use lexicographic criteria for ranking FNs. A lexicographic approach to the fuzzy linear assignment (FLA) problem is discussed in detail due to the theoretical and practical relevance. For this case, computer codes are provided that can be used to reproduce results presented in the paper and for practical applications. The paper demonstrates that FLP that is focused on lexicographic methods is an active area with promising research lines and practical implications.

**Keywords:** fuzzy linear programming; fully fuzzy linear programming; fully fuzzy multi-objective linear programming; fuzzy linear assignment problem; fuzzy number; fuzzy inequality constraint; lexicographic ranking criteria

## 1. Introduction

Bellman and Zadeh's [1] celebrated work on fuzzy decision-making gave birth to Fuzzy Mathematical Programming. Particularly, Fuzzy Linear Programming (FLP) became one of the most studied and developed fields in the Fuzzy Sets and Systems area. FLP combines classical Linear Programming (LP) theory (a fundamental tool for optimization) and Fuzzy Sets theory (a rigorous mathematical theory for modeling the imprecision of ill-defined concepts, such as those that arise from natural language). FLP makes it possible to take into account the inaccuracies of human thinking within optimization processes and, thus, expands the application of classical LP. The influence of the context in the decision-making process has been recently studied in [2] with special emphasis on FLP.

The concept of a fuzzy set plays a pivotal role in Bellman and Zadeh’s [1] theory. Zadeh [3] defined fuzzy sets as classes of objects with a continuum of grades of membership, and extended the classical notions of union, intersection, inclusion, etc. to these sets. Formally, a fuzzy set  $\tilde{A}$  in a space of points  $X$  is characterized by a membership function  $\mu_{\tilde{A}}$ , which associates with each point  $x \in X$  a real number in the interval  $[0, 1]$ ; the value of  $\mu_{\tilde{A}}(x)$  at  $x$  is termed grade of membership of  $x$  in  $\tilde{A}$ , and the nearer this value to unity, the higher the grade of membership of  $x$  in  $\tilde{A}$ .

Bellman and Zadeh [1] viewed decisions as the confluence of goals and constraints. How this confluence is modeled in a fuzzy sense is still a topic of research. However, the interpretation of confluence as an intersection of fuzzy sets representing goals and constraints by means of the min operator has become traditional (The intersection of fuzzy sets can be more generally defined via t-norms [4]). Thus, according to this interpretation, given a fuzzy goal  $\tilde{G}$  and a fuzzy constraint  $\tilde{C}$  in a space of alternatives  $X$ ,  $\tilde{G}$  and  $\tilde{C}$  are combined to form a fuzzy decision set  $\tilde{D} = \tilde{G} \cap \tilde{C}$  with membership function  $\mu_{\tilde{D}}(x) = \min \{ \mu_{\tilde{G}}(x), \mu_{\tilde{C}}(x) \}$  for  $x \in X$ . The optimal decision is taken as any alternative that maximizes  $\mu_{\tilde{D}}$ ; hence,  $\mu_{\tilde{D}}(x^*) = \max_{x \in X} \mu_{\tilde{D}}(x)$ . This idea was adopted by Zimmermann [5] and Tanaka et al. [6], who formally initiated and developed FLP.

Taking as a starting point the classical formulation of the LP problem (1), we are faced with a situation in which the goal of the decision-maker and the problem constraints are better expressed as fuzzy sets.

$$\begin{aligned} \max z(x) &= \sum_{j=1}^n c_j x_j \\ \text{s.t. } a_i(x) &= \sum_{j=1}^n a_{ij} x_j \leq b_i \text{ for } i \in \{1, \dots, m\}, \\ x_j &\geq 0 \text{ for } j \in \{1, \dots, n\}. \end{aligned} \tag{1}$$

Zimmermann [5] proposed problem (2) as a fuzzified version of (1), in which  $\lesssim$  denotes fuzzy inequality and has the linguistic interpretation “essentially smaller than or equal to” and  $z_0$  is an aspiration level that is chosen by the decision-maker.

$$\begin{aligned} \text{Find } x \\ \text{s.t. } z(x) &\gtrsim z_0, \\ a_i(x) &\lesssim b_i \text{ for } i \in \{1, \dots, m\}, \\ x_j &\geq 0 \text{ for } j \in \{1, \dots, n\}. \end{aligned} \tag{2}$$

Zimmermann [5] proposed using the linear membership function to handle the fuzzy inequalities

$$\mu_{\tilde{z}}(x) = \begin{cases} 1 & \text{for } z(x) > z_0, \\ 1 - \frac{z_0 - z(x)}{d_0} & \text{for } z_0 - d_0 \leq z(x) \leq z_0, \\ 0 & \text{for } z(x) < z_0 - d_0, \end{cases} \quad \mu_{\tilde{a}_i}(x) = \begin{cases} 1 & \text{for } a_i(x) < b_i, \\ 1 - \frac{a_i(x) - b_i}{d_i} & \text{for } b_i \leq a_i(x) \leq b_i + d_i, \\ 0 & \text{for } a_i(x) > b_i + d_i, \end{cases}$$

where  $d_i$  ( $i = 0, \dots, m$ ) are subjectively chosen constants of admissible violations. Thus, according to Bellman and Zadeh’s [1] model, an optimal solution is obtained by solving  $\max_{x_j \geq 0} \min \{ \mu_{\tilde{z}}(x), \mu_{\tilde{a}_1}(x), \dots, \mu_{\tilde{a}_m}(x) \}$ .

Zimmermann [5] introduced the auxiliary variable  $\alpha$  and obtained the equivalent problem  $\max \{ \alpha : \mu_{\tilde{z}}(x) \geq \alpha, \mu_{\tilde{a}_1}(x) \geq \alpha, \dots, \mu_{\tilde{a}_m}(x) \geq \alpha, x_j \geq 0 \text{ for } j = 1, \dots, n \}$ .

Tanaka et al. [6] proposed an algorithm to solve a fuzzified version of (1) given by  $\max \{ g(x) : a_1(x) \lesssim b_1, \dots, a_m(x) \lesssim b_m, x_j \geq 0 \text{ for } j = 1, \dots, n \}$ , where  $\lesssim$  has the same interpretation as in Zimmermann’s [5] method, and  $g(x)$  is a membership function obtained by normalizing the objective

function  $z(x)$ ; i.e.,  $g(x) = z(x)/v$  and  $v = \max \{z(x) : a_1(x) \leq b_1 + d_1, \dots, a_m(x) \leq b_m + d_m, x_j \geq 0 \text{ for } j = 1, \dots, n\}$ .

Verdegay [7] generalized the approaches of Tanaka et al. [6] and Zimmermann [5] by demonstrating that their solutions are particular values in the fuzzy solution set obtained through the parametric LP problem (3). Note that, for all values of  $\alpha \in [0, 1]$ , conventional LP problems are obtained, whose corresponding solutions can be integrated into a fuzzy solution via the representation theorem for fuzzy sets. A general view on the theoretical developments obtained from [7] has been presented in [8].

$$\begin{aligned} \max \quad & z(x) \\ \text{s.t.} \quad & a_i(x) \leq b_i + d_i(1 - \alpha) \text{ for } i \in \{1, \dots, m\}, \\ & \alpha \in [0, 1], x_j \geq 0 \text{ for } j \in \{1, \dots, n\}. \end{aligned} \quad (3)$$

In order to better address the inexactness and vagueness of data when modeling real-world problems, it has been suggested to use fuzzy numbers (FNs) as parameters in FLP problems [9]. This modeling approach raises the issue of comparing FNs that appear as objective function values and constraint right-hand/left-hand side values. In [9], the authors obtained an equivalent non-linear programming formulation and proposed a specific solution algorithm. The use of linear ranking functions to compare FNs and transform the FLP problems into crisp ones has become widespread due to the simplicity of the resulting crisp transformations. For example, in [10], the authors considered FLP problems with FNs in both sides of the constraints, and proposed several auxiliary crisp LP problems that were obtained as a result of applying known linear ranking functions. A fuzzy multi-objective LP problem was addressed in [11], in which each objective function may come from a different decision-maker. By using an appropriate linear ranking function for each decision-maker, it is possible to capture each decision-maker's individual ranking attitude and, thus, transform the problem into a crisp multi-objective one. Kaur and Kumar [12] defined the product of unrestricted LR-type FNs, and proposed a method for FLP with fuzzy parameters and fuzzy decision variables that applies Yager's ranking function to the objective function values and to both sides of the fuzzy constraints. On a similar line, Yang et al. [13] proposed two methods for solving FLP problems: one method solves FLP problems with fuzzy inequality constraints and the other FLP problems with flexible fuzzy inequality constraints. Both of the methods use the expected value of an FN as the linear ranking function for making the crisp transformations.

Despite being widely used, linear ranking functions have little discriminative capabilities, as they may map different FNs into the same real number, which, in turn, may yield unreasonable results [14]. Lexicographic ranking criteria can overcome this issue by defining multiple comparison indices arranged in a hierarchical fashion [15]. This fact has motivated the development of lexicographic methods for FLP. In this paper, we contribute, with an updated review, to this particular area of FLP. We do it by first establishing a general framework for lexicographic FLP that organizes existing models and solution methods in a way that they can be well studied, applied in real-world problems, and eventually extended from the perspectives of other theories in the field of Soft Computing. Consequently, in Section 2, we give an overview of FNs, linear ranking functions, lexicographic ranking criteria, and their use in FLP. Thereafter, we review some lexicographic methods for FLP. A lexicographic method for the Fuzzy Linear Assignment (FLA) problem has also been included due to the theoretical and practical relevance of Linear Assignment (LA) problems. Lexicographic methods for fuzzy multi-objective linear programming are reviewed in Section 3. Concluding remarks and ideas for possible extensions of the presented methods are provided in Section 4.

## 2. Fuzzy Linear Programming: Lexicographic Approaches

This section surveys several methods for solving FLP problems based on the use of lexicographic criteria for ranking FNs. We first introduce the concepts of FN, trapezoidal FN (TrFN), and LR-type FN. To avoid an unnecessary lengthening of the paper, we shall not present the arithmetic operations for FNs; the reader is referred to Dubois and Prade’s [16] work on LR-type FNs and the textbooks [17,18]. Note that addition and multiplication operations for FNs shall be denoted generically by  $\oplus$  and  $\otimes$ , respectively; however, their definitions may be different according to specific FLP problems, solution methods, and applications.

As ranking FNs is an important step in the solution of FLP problems, we shall introduce, only in a general sense, two ranking methodologies that are relevant to the subsequent discussion. A detailed analysis of several ranking methods and their rationality, as well as an introduction to this topic, can be found in [19].

### 2.1. Fuzzy Numbers

A real FN  $\tilde{a}$  is any fuzzy subset of the real line  $\mathbb{R}$ , whose membership function  $\mu_{\tilde{a}}$  is:

- A continuous mapping from  $\mathbb{R}$  to the closed interval  $[0, 1]$ ,
- constant on  $(-\infty, e]$ :  $\mu_{\tilde{a}}(x) = 0 \forall x \in (-\infty, e]$ ,
- strictly increasing on  $[e, f]$ ,
- constant on  $[f, g]$ :  $\mu_{\tilde{a}}(x) = 1 \forall x \in [f, g]$ ,
- strictly decreasing on  $[g, h]$ , and
- constant on  $[h, +\infty)$ :  $\mu_{\tilde{a}}(x) = 0 \forall x \in [h, +\infty)$ ;

$e \leq f \leq g \leq h$  are real numbers. In addition, we may have  $e = -\infty$ , or  $f = g$ , or  $e = f$ , or  $g = h$ , or  $h = +\infty$  [16]. We denote the set of all FNs by  $\mathcal{F}(\mathbb{R})$ .

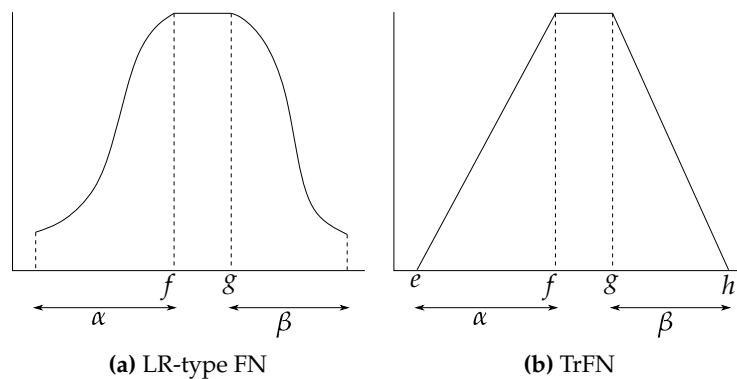
A special type of FN is the trapezoidal FN (TrFN), whose membership function is given by

$$\mu_{\tilde{a}}(x) = \begin{cases} (x - e)/(f - e) & \text{for } e \leq x \leq f, \\ 1 & \text{for } f \leq x \leq g, \\ (h - x)/(h - g) & \text{for } g \leq x \leq h, \\ 0 & \text{otherwise,} \end{cases}$$

and it is denoted by  $\tilde{a} = (e, f, g, h)$ . Clearly, when  $f = g$  a TrFN reduces to a triangular FN (TFN). Dubois and Prade [16] introduced the LR-type representation of FNs by considering the so-called left and right reference functions  $L$  and  $R$ , respectively.  $L$  (or  $R$ ) is a reference function if (1)  $L(x) = L(-x)$ , (2)  $L(0) = 1$ , and (3)  $L$  is nonincreasing on  $[0, +\infty)$ . Symbolically, an LR-type FN is written as  $\tilde{a} = (f, g, \alpha, \beta)_{LR}$  and its membership function takes the form

$$\mu_{\tilde{a}}(x) = \begin{cases} L((f - x)/\alpha) & \text{for } x \leq f, \\ R((x - g)/\beta) & \text{for } x \geq g, \\ 1 & \text{otherwise,} \end{cases}$$

where  $\alpha$  and  $\beta$  are the left and right spreads of  $\tilde{a}$ , respectively. If  $L(x) = R(x) = \max\{0, 1 - |x|\}$ , then the LR-type FN  $\tilde{a}$  reduces to a TrFN  $\tilde{a} = (e, f, g, h)$ , where  $e = f - \alpha$  and  $h = g + \beta$ . We may also write  $\tilde{a} = (e, f, h)$  when referring to a TFN, and  $\tilde{a} = (f, \alpha, \beta)_{LR}$  in the case of LR-type FNs without a flat. Figure 1 depicts the graphical representation of an LR-type FN and a TrFN.



**Figure 1.** Graphical representation of an LR-type fuzzy number (FN) and a trapezoidal FN (TrFN).

2.2. Ranking Functions and Lexicographic Ranking Criteria

At some point in solving fuzzy decision-making problems, two or more FNs representing evaluations of available alternatives need to be compared. As a result of such comparisons, a ranking of the alternatives is obtained, which allows for the selection of the best one. Several methodologies for ranking FNs have been proposed [19]. However, because different FNs may overlap each other, ranking them properly remains a challenging problem in Fuzzy Sets theory.

One way of ranking FNs consists in the use of a linear ranking function  $\mathcal{R} : \mathcal{F}(\mathbb{R}) \rightarrow \mathbb{R}$  to map FNs into real numbers and derive the ranking of the FNs from the comparison of the corresponding real numbers. However, different FNs may be mapped into the same real number and, therefore,  $\mathcal{R}$  may fail to distinguish between different FNs, which in turn could lead to deciding for worse alternatives. Lexicographic ranking criteria use multiple comparison indices and thus improve the discriminative capabilities of linear ranking functions. A rather general lexicographic ranking criterion, which needs to be particularized for the specific decision-making situation at hand, is given in Definition 1. Examples of such a criterion can be found in [20,21].

**Definition 1.** For an arbitrary FN  $\tilde{a}$  with  $p$  defining parameters, let  $f_k$  (for  $k = 1, \dots, p$ ) be  $p$  linear functions of the parameters of  $\tilde{a}$ . Assume that the coefficient matrix corresponding to the  $p$  linear functions is non-singular. Furthermore, let  $\leq_{lex}$  denote the lexicographic order relation on  $\mathbb{R}^p$ . Given any two FNs  $\tilde{a}$  and  $\tilde{b}$  (with the same type of membership functions) the strict inequality  $\tilde{a} < \tilde{b}$  holds if and only if  $(f_k(\tilde{a}))_{k=1, \dots, p} <_{lex} (f_k(\tilde{b}))_{k=1, \dots, p}$ . The weak inequality  $\tilde{a} \preceq \tilde{b}$  holds if and only if  $(f_k(\tilde{a}))_{k=1, \dots, p} <_{lex} (f_k(\tilde{b}))_{k=1, \dots, p}$  or  $(f_k(\tilde{a}))_{k=1, \dots, p} = (f_k(\tilde{b}))_{k=1, \dots, p}$ .

The lexicographic order relation  $\preceq$  satisfies the total order properties; that is,  $\preceq$  is reflexive, transitive, anti-symmetric and complete [22].

2.3. Lexicographic Methods for Solving Fuzzy Linear Programming Problems

A model for a FLP problem in which the parameters and/or the decision variables take on FNs is formulated in Equation (4). Such a type of FLP problem is termed Fully Fuzzy Linear Programming (FFLP) problem, and was initially introduced in [23].

$$\begin{aligned}
 & \min \text{ or } \max \sum_{j=1}^n \tilde{c}_j \otimes \tilde{x}_j \\
 & \text{s.t. } \sum_{j=1}^n \tilde{a}_{ij} \otimes \tilde{x}_j \{ \preceq, =, \succ \} \tilde{b}_i \text{ for } i \in \{1, 2, \dots, m\}.
 \end{aligned}
 \tag{4}$$

We shall consider that all of the parameters  $\tilde{c}_j$ ,  $\tilde{a}_{ij}$  and  $\tilde{b}_i$ , and decision variables  $\tilde{x}_j$  are FNs with the same type of membership functions. Variations of this model appear when, e.g., the decision variables are assumed crisp, in which case one deals with a fuzzy number linear programming problem, or when the decision variables and the right-hand side of the constraints are FNs and the remaining parameters are crisp, in which case one deals with a fuzzy variable linear programming problem. Other variations might appear as well; e.g., when only some parameters and/or decision variables are assumed as FNs.

Having presented the necessary concepts, we survey in this subsection several lexicographic methods for solving FLP problem (4). However we do it within a general framework, as the existing methods share many of their characteristics (see Table A1 in Appendix A).

To solve FLP problem (4), we must first clarify what we mean by minimizing FN-valued functions and how fuzzy inequality constraints are handled. Thus, in accordance with the ranking methodologies already discussed, we may say that either  $\min \{ \tilde{a}, \tilde{b} \} = \tilde{a}$  iff  $\mathcal{R}(\tilde{a}) \leq \mathcal{R}(\tilde{b})$ , or, by using Definition 1,  $\min \{ \tilde{a}, \tilde{b} \} = \tilde{a}$  iff  $(f_k(\tilde{a}))_{k=1,\dots,p} \leq_{\text{lex}} (f_k(\tilde{b}))_{k=1,\dots,p}$ . The latter approach overcomes the indistinguishability issues of the linear ranking functions, and it leads to lexicographic solution methods. Similarly,  $\tilde{a} \preceq \tilde{b}$  iff  $\mathcal{R}(\tilde{a}) \leq \mathcal{R}(\tilde{b})$ , or alternatively  $\tilde{a} \preceq \tilde{b}$  iff  $(f_k(\tilde{a}))_{k=1,\dots,p} \leq_{\text{lex}} (f_k(\tilde{b}))_{k=1,\dots,p}$ . A frequent approach for handling fuzzy inequality constraints has been however to define  $\tilde{a} \preceq \tilde{b}$  by means of a partial order, such as the use of the crisp element-wise inequality between the corresponding components of the parametric representation of the FNs in both sides of the fuzzy inequality constraints, or replacing  $\leq_{\text{lex}}$  with the usual element-wise inequality  $\leq$  between two vectors (see, e.g., [24–27]). Thus, leaving aside the cases that simply use linear ranking functions, the following crisp models can be obtained as possible transformations of FLP problem (4); we have assumed that  $\tilde{z} = \sum_{j=1}^n \tilde{c}_j \otimes \tilde{x}_j$ ,  $\tilde{a}_i = \sum_{j=1}^n \tilde{a}_{ij} \otimes \tilde{x}_j$ ,  $I_{\text{eq}}$  contains the indices of the fuzzy equality constraints and  $I_{\text{ineq}}$  the indices of the fuzzy inequality constraints.

$$\begin{aligned} & \text{lex min or lex max } (f_k(\tilde{z}))_{k=1,\dots,p} \\ & \text{s.t. } (g_k(\tilde{a}_i))_{k=1,\dots,p} \{ \leq_{\text{lex}}, \geq_{\text{lex}} \} (g_k(\tilde{b}_i))_{k=1,\dots,p} \text{ for } i \in I_{\text{ineq}}, \\ & \tilde{a}_i = \tilde{b}_i \text{ for } i \in I_{\text{eq}}, \end{aligned} \tag{Model 1}$$

$$\begin{aligned} & \text{lex min or lex max } (f_k(\tilde{z}))_{k=1,\dots,p} \\ & \text{s.t. } \tilde{a}_i \{ \leq_p, \geq_p \} \tilde{b}_i \text{ for } i \in I_{\text{ineq}}, \\ & \tilde{a}_i = \tilde{b}_i \text{ for } i \in I_{\text{eq}}, \end{aligned} \tag{Model 2}$$

where  $\leq_p$  is a partial order for FNs and each  $g_k$  is a linear function of the parameters of the FNs that may or may not be the same as  $f_k$ . Lastly, the fuzzy equality constraints are defined element-wise from the corresponding parametric representations of the FNs in both sides of the constraints. In order to solve either of the models, Algorithm 1 for classical lexicographic optimization is used [28].

---

**Algorithm 1:** Lexicographic optimization (minimization case).

---

**Data:** Feasible set  $X$  of (Model 1) or (Model 2) and objective functions  $f_k, k = 1, \dots, p$ .

Initialization: Define  $X_1 := X$  and let  $k := 1$ ;

Solve the single-objective optimization problem

$$\min_{\tilde{x} \in X_k} f_k(\tilde{z}); \tag{P_k}$$

**while**  $k \leq p$  **do**

**if** (P<sub>k</sub>) has a unique optimal solution **then**

        stop and return the solution;

**else if** (P<sub>k</sub>) is unbounded **then**

        stop, the lexicographic optimization problem is unbounded;

**else if**  $k=p$  **then**

        stop, the set of optimal solutions of the lexicographic optimization problem is

$$\left\{ \tilde{x} \in X_p : f_p(\tilde{z}) = \min_{\tilde{x} \in X_p} f_p(\tilde{z}) \right\}.$$

**else**

        Let  $X_{k+1} := \left\{ \tilde{x} \in X_k : f_k(\tilde{z}) = \min_{\tilde{x} \in X_k} f_k(\tilde{z}) \right\}$  and  $k := k + 1$ .

**end**

**end**

---

### 2.3.1. Hashemi et al.’s Method

Hashemi et al.’s [29] approach corresponds with (Model 1) for  $p = 2$ . They proposed a two-phase method to solve FFLP problems with inequality constraints in which the parameters and decision variables are symmetric LR-type FNs; i.e., all of the fuzzy parameters and decision variables have the form  $\tilde{a} = (f, \alpha)_L$ . They used a specific lexicographic ranking criterion based on the possibilistic mean  $\mathcal{M}(\tilde{a}) = f$  and standard deviation  $\mathcal{S}(\tilde{a}) = \alpha/\sqrt{6}$ , which is defined as  $\tilde{a} \preceq \tilde{b}$  iff  $(\mathcal{M}(\tilde{a}), -\mathcal{S}(\tilde{a})) \leq_{\text{lex}} (\mathcal{M}(\tilde{b}), -\mathcal{S}(\tilde{b}))$ . Assuming a maximization problem, in the first phase of their method, the set of all feasible solutions that maximize the mean value of the fuzzy objective function is obtained, and in the second phase, the standard deviation of the fuzzy objective function is minimized subject to the set obtained in the first phase. Hashemi et al.’s [29] method is however limited for real applications, as it relies too much on the form of the FNs and the proposed lexicographic criterion (see [25,29] for detailed illustrations of each phase).

### 2.3.2. Hosseinzadeh Lotfi et al.’s Method

Hosseinzadeh Lotfi et al.’s [30] approach corresponds with (Model 2) for  $p = 2$ . The authors proposed a method for solving FFLP problems having only equality constraints. They used the nearest symmetric TFN approximation of an arbitrary FN in parametric form  $\tilde{a} = (\underline{a}(r), \bar{a}(r))$  ( $r \in [0, 1]$ ), and a lexicographic criterion for ranking symmetric TFNs to find an approximate solution of the original FFLP problem. The nearest symmetric TFN of  $\tilde{a}$  is given by  $(a - \alpha, a, a + \alpha)$ , where  $a = \frac{1}{2} \int_0^1 (\underline{a}(r) + \bar{a}(r)) dr$  and  $\alpha = \frac{3}{2} \int_0^1 (\bar{a}(r) - \underline{a}(r))(1 - r) dr$ . For arbitrary symmetric TFNs  $\tilde{a} = (a - \alpha, a, a + \alpha)$  and  $\tilde{b} = (b - \beta, b, b + \beta)$ ,  $\tilde{a} \preceq \tilde{b}$  iff  $(a, -\alpha) \leq_{\text{lex}} (b, -\beta)$ . We see that this criterion is similar to that of Hashemi et al.’s [29] for ranking symmetric LR-type FNs.

### 2.3.3. Kaur and Kumar's Methods

Kaur and Kumar [31] presented a lexicographic method for solving a variation of FLP problem (4) in which only the coefficients of the objective function are considered FNs. The authors used Farhadinia's [21] lexicographic criterion for ranking LR-type FNs. Their method was extended to a fully fuzzy environment in [32] but considering only TrFNs, fuzzy equality constraints and a different lexicographic ranking criterion. A version of this latter method for FFLP with unrestricted LR-type FNs in the problem parameters and decision variables is given in [33]. Kaur and Kumar's approaches [31–33] correspond with (Model 2) for  $p = 4$ .

### 2.3.4. Ezzati et al.'s Method

Ezzati et al. [34] noticed that Hosseinzadeh Lotfi et al.'s [30] method can only yield approximate solutions of FFLP problems. They proposed a new lexicographic criterion for ranking TFNs, and a method for solving FFLP problems with equality constraints in which the problem parameters take on unrestricted TFNs and the decision variables taken on non-negative TFNs. According to their ranking criterion, given any two TFNs  $\tilde{a} = (a_1, a_2, a_3)$  and  $\tilde{b} = (b_1, b_2, b_3)$ ,  $\tilde{a} \preceq \tilde{b}$  iff  $(a_2, a_1 - a_3, a_1 + a_3) \leq_{\text{lex}} (b_2, b_1 - b_3, b_1 + b_3)$ . Their approach corresponds with (Model 2) for  $p = 3$ . A similar method, but with LR-type FNs, is given in [35]. A more efficient version of Ezzati et al.'s [34] method is given in [36] for solving FFLP problems having only equality constraints. To handle fuzzy inequality constraints, Ezzati et al. [34] introduced non-negative triangular fuzzy slack and surplus variables as needed. However, given any two TFNs  $\tilde{a}$  and  $\tilde{c}$  with  $\tilde{a} \preceq \tilde{c}$ , there does not always exist a TFN  $\tilde{b}$ , such that  $\tilde{a} \oplus \tilde{b} = \tilde{c}$ . Furthermore, Bhardwaj and Kumar [37] noticed that, in the case of Ezzati et al.'s [34] method, the introduction of non-negative fuzzy slack and surplus variables leads to a contradiction with Ezzati et al.'s [34] ranking criterion and therefore yields unfeasible solutions. A modified version of Ezzati et al.'s [34] method that handles fuzzy inequality constraints properly is given in [38]. Ezzati et al.'s [34] method has been applied to transportation, investment [34] and supply chain [39] problems in fuzzy environments.

### 2.3.5. Das et al.'s Method

Das et al. [40] proposed a method for solving FFLP problems with equality and inequality constraints having unrestricted trapezoidal fuzzy parameters and non-negative trapezoidal fuzzy decision variables. Assuming TrFNs in the parametric form  $\tilde{a} = (e, f, g, h)$ , Das et al.'s [40] lexicographic ranking criterion for the objective function values defines  $f_1(\tilde{a}) = e - f$ ,  $f_2(\tilde{a}) = f$ ,  $f_3(\tilde{a}) = (f + g)/2$  and  $f_4(\tilde{a}) = h - g$ . However, the handling of fuzzy inequality constraints was not consistent with this lexicographic ranking criterion. Ebrahimnejad and Verdegay [25] noticed this fact and modified Das et al.'s [40] method to handle fuzzy inequality constraints correctly. The modified approach [25] exactly corresponds with (Model 2) for  $p = 4$  when  $\tilde{a}_i \leq_p \tilde{b}_i$  if and only if  $f_k(\tilde{a}_i) \leq f_k(\tilde{b}_i)$  for  $k \in \{1, \dots, 4\}$ . Thus, a lexicographic criterion is used with the objective function values and a partial order with the fuzzy inequality constraints. Das et al. [40] illustrated their method by solving production planning and diet problems in fully fuzzy environments.

### 2.3.6. Lexicographic Method

Pérez-Cañedo and Concepción-Morales [14] proposed a method to solve FFLP problems with equality and inequality constraints in which the problem parameters and decision variables can take on unrestricted LR-type FNs. The approach followed in [14] corresponds exactly with (Model 1). Originally, in this method, the same lexicographic ranking criterion is used with the objective function values and the fuzzy inequality constraints. However, different lexicographic ranking criteria for the objective function and the inequality constraints can be easily incorporated by a proper redefinition of



the  $g_k$  functions. The authors proposed the following transformation of those constraints to handle the lexicographic constraints in (Model 1):

$$\begin{aligned}
 & -M \sum_{r=1}^{k-1} y_{ir} + \epsilon y_{ik} \leq g_k(\tilde{b}_i) - g_k(\tilde{a}_i) \leq M y_{ik} \text{ if the } i\text{th constraint is a } \leq_{\text{lex}}\text{-type constraint,} \\
 & -M \sum_{r=1}^{k-1} y_{ir} + \epsilon y_{ik} \leq g_k(\tilde{a}_i) - g_k(\tilde{b}_i) \leq M y_{ik} \text{ if the } i\text{th constraint is a } \geq_{\text{lex}}\text{-type constraint,} \\
 & y_{ik} \in \{0, 1\} \text{ for } i \in I_{\text{ineq}} \text{ and } k \in \{1, \dots, p\},
 \end{aligned}$$

for positive real values of  $\epsilon$  and  $M$  sufficiently small and large, respectively. The authors illustrated their method by means of simple fuzzy production planning problem. An application in fuzzy linear regression can be found in [41]. Recently, this method was extended to a fully intuitionistic fuzzy environment in [42].

### 2.3.7. Illustrative Example and Comparative Analysis

Consider the following FFLP problem taken from [14].

$$\begin{aligned}
 & \max (8, 8, 2, 2)_{LR} \otimes \tilde{x}_1 \oplus (12, 12, 2, 2)_{LR} \otimes \tilde{x}_2 \oplus (1, 1, 0.25, 0.25)_{LR} \otimes \tilde{x}_3 \\
 & \text{s.t. } (5, 5, 0.5, 0.5)_{LR} \otimes \tilde{x}_1 \oplus (5, 5, 0.5, 0.5)_{LR} \otimes \tilde{x}_2 \oplus \tilde{x}_3 = (150, 155, 45, 52)_{LR}, \\
 & (6, 6, 0.25, 0.25)_{LR} \otimes \tilde{x}_1 \oplus (2, 2, 0.25, 0.25)_{LR} \otimes \tilde{x}_2 \preceq (120, 125, 18, 22)_{LR}, \\
 & (1, 1, 0.5, 0.25)_{LR} \otimes \tilde{x}_1 \oplus (4, 4, 0.25, 0.5)_{LR} \otimes \tilde{x}_2 \preceq (100, 110, 42, 38)_{LR}, \\
 & \tilde{x}_1, \tilde{x}_2 \text{ are non-negative LR-type FNs, and } \tilde{x}_3 \text{ is an unrestricted LR-type FN,} \\
 & L(x) = R(x) = \max\{0, 1 - |x|\}.
 \end{aligned}$$

From its formulation, we notice that the methods in Table A1 (with the exception of [14]) cannot be used to solve this problem. On solving the above FFLP problem with method [14] and the lexicographic ranking criterion [21], we obtain  $\tilde{x}_1 = (12.72, 12.72, 1.62, 0)_{LR}$ ,  $\tilde{x}_2 = (21.81, 22.81, 0, 6.22)_{LR}$  and  $\tilde{x}_3 = (-22.72, -22.72, 20.39, 0)_{LR}$ , with fuzzy value  $(\tilde{v})_{\leq_{\text{lex}}} = (340.90, 352.90, 110.04, 163.89)_{LR}$ .

Table 1 shows two solutions obtained with alternative methods for handling fuzzy inequality constraints. Figure 2 depicts the fuzzy values of the solutions that are given in Table 1 and the one obtained with the lexicographic method [14]; considering that this is a maximization problem, it can be seen that the lexicographic method yielded an intuitively better solution.

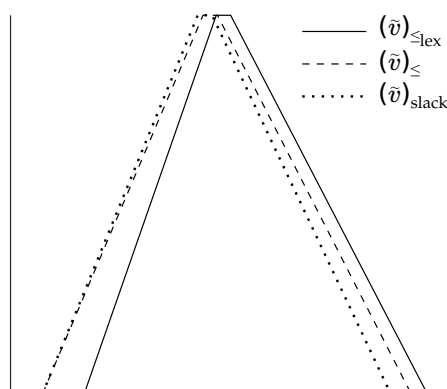


Figure 2. Fuzzy values of the solutions shown in Table 1 and the one obtained with the lexicographic method [14].

**Table 1.** The results obtained by using alternative methods for handling fuzzy inequality constraints.

Solution	Method	
	Adding Non-Negative Fuzzy Slack Variables	Using Order Relation $\leq$ Instead of $\leq_{lex}$
$\tilde{x}_1$	$(13.58, 13.58, 0, 0.23)_{LR}$	$(13.33, 13.33, 0, 0)_{LR}$
$\tilde{x}_2$	$(19.25, 20.25, 5.59, 5.37)_{LR}$	$(19.98, 20.98, 6.29, 6.33)_{LR}$
$\tilde{x}_3$	$(-14.16, -14.16, 3.40, 4.24)_{LR}$	$(-16.61, -16.61, 0, 0)_{LR}$
Fuzzy value	$(\tilde{v})_{slack} = (325.50, 337.50, 129.41, 148.78)_{LR}$	$(\tilde{v})_{\leq} = (329.91, 341.91, 133.77, 161.48)_{LR}$

### 2.4. Fuzzy Linear Assignment Problem

The LA problem is a combinatorial optimization problem, in which a decision-maker wishes to assign  $n$  resources (persons or machines) to  $n$  tasks, on a one-to-one basis, at minimum cost. The LA problem has  $n!$  feasible assignments, which makes complete enumeration procedures extremely inefficient. Kuhn [43] proposed a primal-dual algorithm, known as the Hungarian Method, which solves LA problems with worst-case time complexity of  $O(n^4)$ . The Hungarian Method made possible for the first time to tackle high-dimensional LA problems in a more efficient way.

In real-world decision-making situations, the coefficients of the LA problem are often obtained from decision-makers and experts as subjective evaluations of potential assignments whose costs are only approximately known. In the framework of Fuzzy Sets theory, these cost coefficients can be taken as FNs. Hence, it makes sense to formulate and solve a fuzzy version of the classical LA problem. Consequently, let  $S_n$  denote the set of all the permutations of  $I = \{1, 2, \dots, n\}$  and  $\tilde{C} = [\tilde{c}_{ij}]_{n \times n}$ ,  $\tilde{c}_{ij} \in \mathcal{F}(\mathbb{R})$ , a matrix comprising the FNs (costs) of the potential assignments. The objective is to find a permutation  $\hat{\varphi} \in S_n$  that minimizes  $\sum_{i=1}^n \tilde{c}_{i\varphi(i)} = \tilde{c}_{1\varphi(1)} \oplus \tilde{c}_{2\varphi(2)} \oplus \dots \oplus \tilde{c}_{n\varphi(n)}$ . Thus, the FLA problem can be formulated as in Equation (5).

$$\min_{\varphi \in S_n} \sum_{i=1}^n \tilde{c}_{i\varphi(i)} \tag{5}$$

The FLP formulation of the FLA problem is given by Equation (6).

$$\begin{aligned} \min \quad & \sum_{i=1}^n \sum_{j=1}^n \tilde{c}_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^n x_{ij} = 1 \text{ for } i \in I, \\ & \sum_{i=1}^n x_{ij} = 1 \text{ for } j \in I, \\ & x_{ij} \in \{0, 1\} \text{ for } i, j \in I. \end{aligned} \tag{6}$$

For simplicity, it shall be assumed that all of the cost coefficients are TrFNs; although, a more general approach considering different types of FNs simultaneously has been recently investigated in [44]. There are several methods for solving FLA problems; the reader is referred to [44] for a description of some of those methods. Here, we present three methods that can be regarded as representative of the existing ones.

#### 2.4.1. Kumar and Gupta’s Method

Kumar and Gupta [45] transformed the FLA problem into a crisp one by using a linear ranking function, which, for an arbitrary TrFN  $\tilde{a} = (e, f, g, h)$ , has the form  $\mathcal{R}(\tilde{a}) = (e + f + g + h)/4$ . Thus, according to their method, LA problem (7) is obtained.

$$\begin{aligned} \min \quad & \sum_{i=1}^n \sum_{j=1}^n \mathcal{R}(\tilde{c}_{ij})x_{ij} \\ \text{s.t.} \quad & \text{the same constraints of problem (6)}. \end{aligned} \tag{7}$$

The solution of LA problem (7) is obtained by any of the existing classical assignment methods.

#### 2.4.2. Baykasoğlu et al.’s Method

Although Baykasoğlu et al.’s [46] method was originally proposed to solve FLA problems with triangular fuzzy coefficients in the context of multiple-attribute group decision-making, an extension to the case of trapezoidal fuzzy coefficients is straightforward.

Let  $\tilde{C} = [\tilde{c}_{ij}]_{n \times n}$  denote the fuzzy cost matrix of the FLA problem, where each  $\tilde{c}_{ij} = (e_{ij}, f_{ij}, g_{ij}, h_{ij})$  is a TrFN for  $i, j \in I$ . By using Baykasoğlu et al.’s [46] method, the FLA problem is transformed into a four-objective crisp LA problem, as in Equation (8), with  $C_e = [e_{ij}]_{n \times n}$ ,  $C_f = [f_{ij}]_{n \times n}$ ,  $C_g = [g_{ij}]_{n \times n}$  and  $C_h = [h_{ij}]_{n \times n}$  being the coefficient matrices of each of the objective functions. Compromise programming [47] with a preferred metric is then used to solve the four-objective crisp LA problem.

$$\begin{aligned} \min \quad & \left( \sum_{i=1}^n \sum_{j=1}^n e_{ij}x_{ij}, \sum_{i=1}^n \sum_{j=1}^n f_{ij}x_{ij}, \sum_{i=1}^n \sum_{j=1}^n g_{ij}x_{ij}, \sum_{i=1}^n \sum_{j=1}^n h_{ij}x_{ij} \right) \\ \text{s.t.} \quad & \sum_{j=1}^n x_{ij} = 1 \text{ for } i \in I, \\ & \sum_{i=1}^n x_{ij} = 1 \text{ for } j \in I, \\ & x_{ij} \in \{0, 1\} \text{ for } i, j \in I. \end{aligned} \tag{8}$$

#### 2.4.3. Lexicographic Method

As noticed previously, the use of a linear ranking function for making crisp transformations does not guarantee optimal solutions for the FLP problem under consideration. Knowing that lexicographic ranking criteria have better discriminative capabilities than linear ranking functions, it seems natural to use Definition 1 in order to transform FLA problem (5) into lexicographic linear assignment (LLA) problem (9).

$$\text{lex min}_{\varphi \in S_n} \sum_{i=1}^n \left( f_k(\tilde{c}_{i\varphi(i)}) \right)_{k=1, \dots, 4} \tag{9}$$

**Theorem 1.** *If  $\hat{\varphi}$  is an optimal assignment for LLA problem (9), then  $\hat{\varphi}$  is also an optimal assignment for FLA problem (5).*

**Proof.** Similar to that of [44]. □

To solve LLA problem (9), the authors in [44] adhered to Burkard et al.’s [48] general algebraic framework. Burkard et al. [48] assumed that the cost coefficients of the LA problem are elements of a totally ordered commutative semigroup  $(H, *, \leq_H)$  with internal composition  $*$  and total order relation  $\leq_H$  fulfilling the following properties:

1.  $\forall x, y, z \in H, x \leq_H y \implies x * z \leq_H y * z$  ( $\leq_H$  is compatible with  $*$ );
2.  $\forall x, y \in H$  with  $x \leq_H y \exists z \in H$  such that  $x * z = y$ .

Burkard et al. [48] also stated and proved an optimality criterion, and proposed a general solution algorithm that can be seen as a generalization of the Hungarian Method.

It can be shown that  $(\mathbb{R}^p, +, \leq_{\text{lex}})$  is a totally ordered commutative semigroup that fulfills the above-mentioned properties. Hence, LLA problem (9) can be optimally solved within Burkard et al.'s [48] algebraic framework; Theorem 1 guarantees that an optimal solution of LLA problem (9) is optimal for FLA problem (5) as well.

2.4.4. Illustrative Example and Comparative Analysis

The example in this subsection is taken from [44]. The FLA problem consists of four tasks  $(T_i, i = 1, \dots, 4)$  and four resources  $(R_j, j = 1, \dots, 4)$ , and the assignment costs are the TrFNs shown in Table 2. The lexicographic method is implemented in Python codes A1–A3 in Listings A1–A3 from Appendix A and uses package *munkres* version 1.0.8 [49].

Table 2 shows the assignments made by Baykasoğlu et al.'s [46] method ( $\Delta$ ) with  $L_2$  and  $L_\infty$  metrics, Kumar and Gupta's [45] method ( $\circ$ ), and the lexicographic method [44] ( $\bullet$ ) with  $f_1(\tilde{a}) = (e + f + g + h)/4, f_2(\tilde{a}) = (f + g)/2, f_3(\tilde{a}) = h - e$  and  $f_4(\tilde{a}) = f - e$ .

Table 2. Fuzzy cost matrix of the fuzzy linear assignment (FLA) problem given in [44].

	R1	R2	R3	R4
T1	$\bullet\circ(0, 1, 4, 15)$	$(3, 5, 8, 10)$	$\Delta(1, 2, 7, 11)$	$(4, 6, 8, 9)$
T2	$\Delta(4, 5, 5, 8)$	$\circ(2, 5, 6, 7)$	$\bullet(1, 3, 5, 11)$	$(3, 6, 9, 11)$
T3	$(1, 2, 8, 9)$	$(1, 5, 6, 8)$	$(5, 6, 8, 10)$	$\bullet\circ\Delta(1, 4, 5, 10)$
T4	$(5, 7, 9, 12)$	$\bullet\Delta(2, 4, 6, 8)$	$\circ(2, 5, 6, 7)$	$(4, 6, 6, 7)$

Figure 3 depicts the membership functions of the total fuzzy assignment costs obtained using Kumar and Gupta's [45] method  $(\tilde{c})_\circ = (5, 15, 21, 39)$ , Baykasoğlu et al.'s [46] method with  $L_2$  and  $L_\infty$  metrics  $(\tilde{c})_\Delta = (8, 15, 23, 37)$  and the lexicographic method [44]  $(\tilde{c})_\bullet = (4, 12, 20, 44)$ . When considering that we seek a solution with minimum cost, from Figure 3, we see that the ranking  $(\tilde{c})_\bullet \prec (\tilde{c})_\circ \prec (\tilde{c})_\Delta$  seems reasonable. In fact, the ranking methods [20,21,50–52] support this conclusion as well. We note that the solution yielded by Baykasoğlu et al.'s [46] method with  $L_1$  metric coincides with that of the lexicographic method. However, using Baykasoğlu et al.'s [46] method requires solving four crisp LA problems to find the ideal points of each objective function, and a fifth problem, linear or non-linear, depending on the choice of the metric, to find a compromise solution.

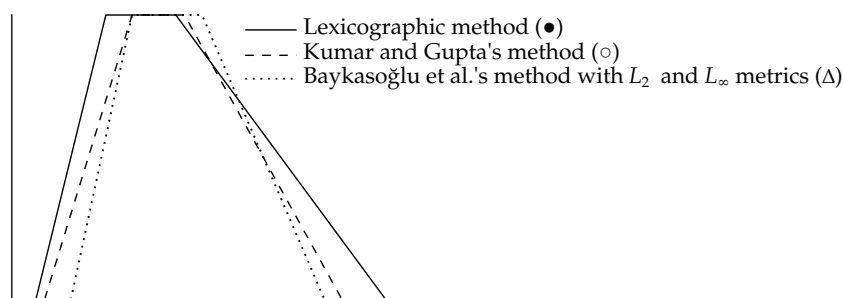


Figure 3. Graphs of the membership functions of the total fuzzy assignment costs.

3. Lexicographic Methods for Fuzzy Multi-Objective Linear Programming

The Fully Fuzzy Multi-Objective Linear Programming (FFMOLP) problem is given by Equation (10). The first reference to this topic is the work of Buckley et al. [53], where the authors used an evolutionary algorithm to find approximate solutions. Recent works on this topic mainly use linear ranking functions for making crisp transformations [54,55]. In what follows, we review two lexicographic methods.

$$\begin{aligned} & \min \text{ or } \max \left( \sum_{j=1}^n \tilde{c}_j^{(1)} \otimes \tilde{x}_j, \sum_{j=1}^n \tilde{c}_j^{(2)} \otimes \tilde{x}_j, \dots, \sum_{j=1}^n \tilde{c}_j^{(r)} \otimes \tilde{x}_j \right) \\ & \text{s.t. } \sum_{j=1}^n \tilde{a}_{ij} \otimes \tilde{x}_j \{ \preceq, =, \succeq \} \tilde{b}_i \text{ for } i \in \{1, 2, \dots, m\}. \end{aligned} \tag{10}$$

### 3.1. Yang et al.'s Method

Yang et al. [56] combined Ezzati et al.'s [34] lexicographic ranking criterion with Bellman and Zadeh's [1] decision-making model to solve FFMOLP problems having only equality constraints, in which the problem parameters are arbitrary TFNs and the decision variables are non-negative TFNs.

Let  $\tilde{z}^{(t)}(\tilde{x}) = \sum_{j=1}^n \tilde{c}_j^{(t)} \otimes \tilde{x}_j$  and  $X = \left\{ \tilde{x} : \sum_{j=1}^n \tilde{a}_{ij} \otimes \tilde{x}_j = b_i \text{ for } i \in \{1, \dots, m\}; \tilde{x}_j \geq 0 \text{ for } j \in \{1, \dots, n\} \right\}$ , their method (for minimization problems) is depicted in Algorithm 2. We note that Algorithm 2 is a generalization of Yang et al.'s [56] method, and it allows for a decision-maker to use a lexicographic ranking criterion of his/her choice through a proper definition of the  $f_k$  functions.

---

**Algorithm 2:** Yang et al.'s [56] method (minimization case).

---

**Data:** Objective functions  $\tilde{z}^{(t)}$  (for  $t = 1, \dots, r$ ), feasible set  $X$  and Ezzati et al.'s [34] lexicographic ranking criterion  $f_1(\tilde{a}) = a_2, f_2(\tilde{a}) = a_1 - a_3$ , and  $f_3(\tilde{a}) = a_1 + a_3$ .

**output:** Values of the decision variables  $\tilde{x}_j$ .

Initialization: Define  $X_1 := X$  and let  $k := 1$ ;

**while**  $k \leq 3$  **do**

Solve the single-objective optimization problems ( $F_k$ ) and denote the corresponding optimal solutions as  $\tilde{x}^{(t)}$  for  $t = 1, \dots, r$ .

$$\min_{\tilde{x} \in X_k} f_k(\tilde{z}^{(t)}(\tilde{x})) \text{ for } t \in \{1, \dots, r\}. \tag{F_k}$$

Let  $f_{kt}^{\min} := \min \{ f_k(\tilde{z}^{(t)}(\tilde{x}^{(1)})), \dots, f_k(\tilde{z}^{(t)}(\tilde{x}^{(r)})) \}$  and  $f_{kt}^{\max} := \max \{ f_k(\tilde{z}^{(t)}(\tilde{x}^{(1)})), \dots, f_k(\tilde{z}^{(t)}(\tilde{x}^{(r)})) \}$ , and construct the membership functions

$$\mu_{kt} \left( f_k(\tilde{z}^{(t)}(\tilde{x})) \right) = \begin{cases} 0 & \text{for } f_k(\tilde{z}^{(t)}(\tilde{x})) > f_{kt}^{\max}, \\ \frac{f_{kt}^{\max} - f_k(\tilde{z}^{(t)}(\tilde{x}))}{f_{kt}^{\max} - f_{kt}^{\min}} & \text{for } f_{kt}^{\min} \leq f_k(\tilde{z}^{(t)}(\tilde{x})) \leq f_{kt}^{\max}, \\ 1 & \text{otherwise.} \end{cases}$$

Set  $X_k^\alpha := \{ \tilde{x} \in X_k : \mu_{kt} (f_k(\tilde{z}^{(t)}(\tilde{x}))) \geq \alpha_k \text{ for } t \in \{1, \dots, r\} \}$  and solve problem ( $P_k$ ).

$$\alpha_k^* = \max_{\tilde{x} \in X_k^\alpha} \alpha_k. \tag{P_k}$$

**if**  $k < 3$  **then**

Let  $X_{k+1} := \{ \tilde{x} \in X_k : \mu_{kt} (f_k(\tilde{z}^{(t)}(\tilde{x}))) \geq \alpha_k^* \text{ for } t \in \{1, \dots, r\} \}$ ;

**end**

$k := k + 1$ ;

**end**

---

### 3.2. Fuzzy Epsilon-Constraint Method

Pérez-Cañedo et al. [15] showed that the transformation of FFMOLP problems into crisp multi-objective linear programming problems by means of linear ranking functions does not guarantee Pareto optimal fuzzy solutions. Thus, taking as a starting point the lexicographic method for fully fuzzy single-objective linear programming presented in Section 2.3.6, in [15], the authors extended the classical definitions of dominance and Pareto optimality to the fuzzy case using lexicographic ranking criteria, and proposed a fuzzy epsilon-constraint method that guarantees Pareto optimal fuzzy solutions. In what follows, we briefly present their approach only considering the minimization case.

**Definition 2** ([15]). Let  $\preceq$  be given as in Definition 1 and  $X$  denote the set of all feasible fuzzy solutions of FFMOLP problem (10). A decision vector  $\tilde{x} \in X$  is said to dominate  $\tilde{y} \in X$  if  $\sum_{j=1}^n \tilde{c}_j^{(t)} \otimes \tilde{x}_j \preceq \sum_{j=1}^n \tilde{c}_j^{(t)} \otimes \tilde{y}_j$  for  $t \in \{1, \dots, r\}$  with at least one strict inequality.

**Definition 3.** A decision vector  $\tilde{x} \in X$  is said to be Pareto optimal fuzzy solution of FFMOLP problem (10) if there does not exist another  $\tilde{y} \in X$ , such that  $\tilde{y}$  dominates  $\tilde{x}$ .

By introducing auxiliary variables  $\tilde{z}$ ,  $\tilde{s}_t^+$ ,  $\tilde{s}_t^-$ , real- and FN-valued parameters  $\lambda_t$  and  $\tilde{e}_t$  for  $t \in \{1, \dots, r\} \setminus \{q\}$ , respectively, FFMOLP problem (10) is transformed into fully fuzzy single-objective linear programming problem (11), which is solved using the lexicographic method from Section 2.3.6. Again, it is assumed that all of the fuzzy parameters and decision variables have the same type of membership functions. In this formulation,  $\tilde{M}$  is a constant FN, e.g.,  $\tilde{M} = (0, m, m, m)_{LR}$  with  $m$  sufficiently large, whose sole purpose is to guarantee the existence of  $\tilde{z}$  so that the first fuzzy equality constraint holds for optimal values of  $\tilde{x}$ ,  $\tilde{s}^+$  and  $\tilde{s}^-$  [15].

$$\begin{aligned}
 & \min \tilde{z} \\
 & \text{s.t. } \tilde{z} \oplus \sum_{t \neq q} \lambda_t \tilde{s}_t^+ = \sum_{j=1}^n \tilde{c}_j^{(q)} \otimes \tilde{x}_j \oplus \sum_{t \neq q} \lambda_t \tilde{s}_t^- \oplus \tilde{M}, \\
 & \sum_{j=1}^n \tilde{c}_j^{(t)} \otimes \tilde{x}_j \oplus \tilde{s}_t^+ = \tilde{e}_t \oplus \tilde{s}_t^- \text{ for } t \in \{1, \dots, r\} \setminus \{q\}, \\
 & \tilde{s}_t^- \preceq \tilde{s}_t^+ \text{ for } t \in \{1, \dots, r\} \setminus \{q\}, \\
 & \sum_{j=1}^n \tilde{a}_{ij} \otimes \tilde{x}_j \{ \preceq, =, \succ \} \tilde{b}_i \text{ for } i \in \{1, \dots, m\}, \\
 & \tilde{s}_t^-, \tilde{s}_t^+ \text{ for } t \in \{1, \dots, r\} \setminus \{q\} \text{ are non-negative FNs.}
 \end{aligned} \tag{11}$$

**Theorem 2** ([15]). Let  $(\tilde{x}, \tilde{s}^-, \tilde{s}^+, \tilde{z})$  be an optimal fuzzy solution of FFLP problem (11) with  $\lambda_t > 0 \forall t \in \{1, \dots, r\} \setminus \{q\}$ . Subsequently,  $\tilde{x}$  is a Pareto optimal fuzzy solution of FFMOLP problem (10).

In [15], the authors exemplified their method by solving a three-objective fully fuzzy project crashing problem. This method has been recently used in [57] to solve a fully fuzzy multi-objective berth allocation problem.

### 3.3. Illustrative Example and Comparative Analysis

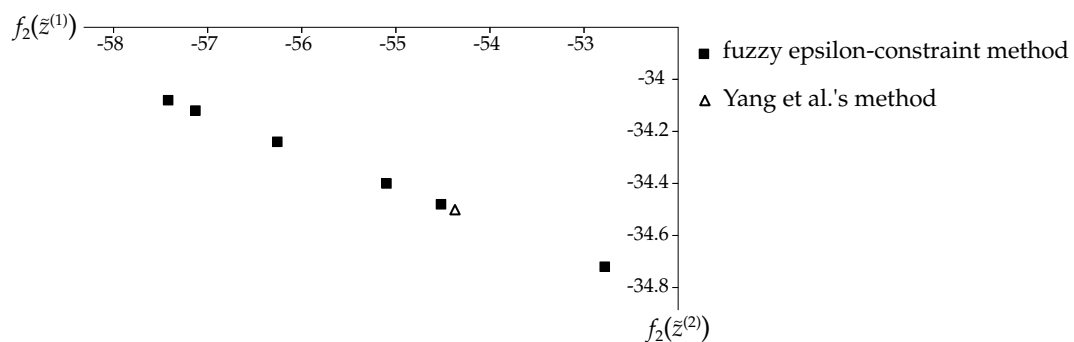
Yang et al. [56] illustrated their lexicographic method by solving the following FFMOLP problem, in which all of the parameters and decision variables are TFNs.

$$\begin{aligned} \min \bar{z}^{(1)} &= (7, 10, 11) \otimes \bar{x}_1 \oplus (8, 10, 13) \otimes \bar{x}_2, \\ \min \bar{z}^{(2)} &= (2, 3, 4) \otimes \bar{x}_1 \oplus (4, 7, 12) \otimes \bar{x}_2, \\ \text{s.t. } &(1, 2, 4) \otimes \bar{x}_1 \oplus (2, 8, 10) \otimes \bar{x}_2 = (3, 22, 36), \\ &(2, 3, 6) \otimes \bar{x}_1 \oplus (4, 10, 15) \otimes \bar{x}_2 = (6, 29, 54), \\ &\bar{x}_1, \bar{x}_2 \text{ are non-negative TFNs.} \end{aligned}$$

According to Yang et al. [56], their method yields  $\bar{x}_1 = (0, 3, 3.3743)$ ,  $\bar{x}_2 = (1.5, 2, 2.2503)$  with objective functions values  $\bar{z}^{(1)} = (12, 50, 66.3712)$  and  $\bar{z}^{(2)} = (6, 23, 40.5008)$ . However, the authors must have done some rounding or miscalculation, as this solution is not satisfying the problem constraints exactly. Yang et al.'s [56] method only yields one solution to the FFMOLP problem; hence, it lacks the necessary flexibility in real-world applications. On the other hand, the fuzzy epsilon-constraint method [15], equipped with Ezzati et al.'s [34] lexicographic ranking criterion, yields multiple Pareto optimal fuzzy solutions, as shown in Table 3. Figure 4 depicts a portion of the Pareto front of the FFMOLP problem corresponding to the solutions in Table 3 plus Yang et al.'s [56] solution. From the objective functions values in Table 3, the reader may notice that all of the solutions are tied with respect to Ezzati et al.'s [34] first ranking index; hence, to distinguish one solution from another, we must resort to the other indices and, in this case, the second index is enough to conclude that no solution dominates another.

**Table 3.** Solutions of the Fully Fuzzy Multi-Objective Linear Programming (FFMOLP) problem obtained by using the fuzzy epsilon-constraint method [15].

Pareto Optimal Fuzzy Solutions						
$\bar{x}_1$	(0, 3, 3.1)	(0, 3, 3.4)	(0, 3, 3.5)	(0, 3, 3.7)	(0, 3, 3.85)	(0, 3, 3.9)
$\bar{x}_2$	(1.5, 2, 2.36)	(1.5, 2, 2.24)	(1.5, 2, 2.2)	(1.5, 2, 2.12)	(1.5, 2, 2.06)	(1.5, 2, 2.04)
$\bar{z}^{(1)}$	(12, 50, 64.78)	(12, 50, 66.52)	(12, 50, 67.1)	(12, 50, 68.26)	(12, 50, 69.13)	(12, 50, 69.42)
$\bar{z}^{(2)}$	(6, 23, 40.72)	(6, 23, 40.48)	(6, 23, 40.4)	(6, 23, 40.24)	(6, 23, 40.12)	(6, 23, 40.08)



**Figure 4.** Portion of the Pareto front of the FFMOLP problem in terms of Ezzati et al.'s [34] second ranking index.

#### 4. Concluding Remarks

In this paper, we have presented a review of existing lexicographic methods for FLP. These methods can be grouped into two main models. Enclosed by (Model 1) are those methods that handle fuzzy inequality constraints by using lexicographic ranking criteria (we have only found two methods of this kind in the existing literature) and enclosed by (Model 2) are those that use partial orders or introduce fuzzy slack and surplus variables to handle the fuzzy inequality constraints. In the future, it can be expected that these methods will be extended to deal with FLP problems in which the parameters and/or decision variables take on generalizations of FNs, such as interval-valued FNs,

intuitionistic FNs, interval-valued intuitionistic FNs, and so on; an interesting topic in this regard is the hybridization with Rough Sets theory (see, e.g., [58] and references therein). Such generalizations allow for incorporating other levels of imprecision, e.g., by considering that membership grades may not be fixed real numbers but belong to some closed subintervals of  $[0, 1]$ , there may be hesitation with regard to the values of some parameters, and data may be inconsistent due to the existence conflicting values.

Currently, in the case of FFMOLP, there are only two lexicographic methods available. Hence, we believe that new lexicographic methods for FFMOLP will emerge as extensions of classical ones given the many applications of classical multi-objective optimization. Particularly, a fully fuzzy extension of classical compromise programming is an interesting research line to explore in connection with fuzzy-valued distance functions. Here too, other methods using generalizations of FNs as parameters and/or decision variables can be expected.

Lexicographic methods for FLP are quite recent. Although they have been used in some real-world decision-making situations, their usefulness and advantages in this regard over other established approaches have yet to be explored in a deeper way. Therefore, researchers that are interested in this topic may consider not only the generalizations commented here, but real-world applications as well. In this regard, a fundamental issue to address is the non-existence of suitable software for rapid development and implementation of real-world applications. A step in this direction has been recently taken in [59], where the authors presented an open-source R package for FLP. A similar package, or perhaps an extension, could incorporate the lexicographic methods presented in this review, thus making these methods available for use as well.

In this review, we have only considered lexicographic methods for FLP. The non-linear case has not been arbitrarily left out. In fact, to the best of our knowledge, no research has been conducted on the use of lexicographic methods for Fuzzy Non-linear Programming (FNLP). FNLP is a research area with many real-world applications (see, e.g., [60], where Fuzzy Sets theory and genetic algorithms are combined to solve a congestion management problem). We believe that lexicographic methods may also prove useful in this area.

**Author Contributions:** Conceptualization, B.P.-C. and J.L.V.; methodology, B.P.-C. and J.L.V.; software, B.P.-C.; validation, J.L.V., E.R.C.-M. and A.R.; formal analysis, B.P.-C., E.R.C.-M. and A.R.; investigation, B.P.-C., J.L.V., E.R.C.-M. and A.R.; resources, B.P.-C.; data curation, B.P.-C.; writing—original draft preparation, B.P.-C., J.L.V., E.R.C.-M. and A.R.; writing—review and editing, B.P.-C., J.L.V., E.R.C.-M. and A.R.; visualization, B.P.-C.; supervision, J.L.V., E.R.C.-M. and A.R.; project administration, J.L.V.; funding acquisition, J.L.V. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded in part by the Spanish Ministry of Economy and Competitiveness and FEDER funds from the European Union under project TIN2017-86647-P.

**Acknowledgments:** The authors are grateful to the editors and the anonymous reviewers for their constructive comments and suggestions.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

LP	Linear Programming
FLP	Fuzzy Linear Programming
FNLP	Fuzzy Non-linear Programming
FFLP	Fully Fuzzy Linear Programming
FFMOLP	Fully Fuzzy Multi-Objective Linear Programming
LA	Linear Assignment
FLA	Fuzzy Linear Assignment
LLA	Lexicographic Linear Assignment
FN	Fuzzy Number
TFN	Triangular Fuzzy Number
TrFN	Trapezoidal Fuzzy Number



Appendix A

Table A1. Characteristics of some lexicographic methods for solving FLP problems.

Method	Fuzzy Number	Fuzzified Elements	Unrestricted Parameters and Decision Variables	Order Relation for Fuzzy Inequality Constraints
[29]	symmetric LR-type FN	all	no	lexicographic (total order)
[30]	TFN	all	no	n/a <sup>1</sup>
[31]	LR-type FN	c <sub>j</sub> 's	no	n/a
[32]	TrFN	all	no	n/a <sup>1</sup>
[33]	LR-type FN	all	yes	n/a <sup>1</sup>
[34]	TFN	all	no	n/a <sup>2</sup>
[40]	TrFN	all	no	partial order
[14]	LR-type FN	all	yes	lexicographic (total order)

<sup>1</sup> Fuzzy equality constraints only. <sup>2</sup> Transformation into equalities by means of fuzzy slack and surplus variables.

```
import numbers
class TrFN:
    def __init__(self, a=0, b=0, c=0, d=0):
        self.a, self.b, self.c, self.d = a, b, c, d
    def __add__(self, other):
        if isinstance(other, numbers.Number):
            other = TrFN(other, other, other, other)
        return TrFN(self.a+other.a, self.b+other.b, self.c+other.c,
                    self.d+other.d)
    def __sub__(self, other):
        if isinstance(other, numbers.Number):
            other = TrFN(other, other, other, other)
        return TrFN(self.a-other.d, self.b-other.c, self.c-other.b,
                    self.d-other.a)
    def __neg__(self):
        return TrFN(-self.d, -self.c, -self.b, -self.a)
    def __str__(self):
        return 'TrFN(%s,%s,%s,%s)' % (self.a, self.b, self.c, self.d)
    def __repr__(self):
        return 'TrFN(%s,%s,%s,%s)' % (self.a, self.b, self.c, self.d)
```

Listing A1: Python code A1—Simple class for TrFNs (file name *trfn.py*).

```
import numbers
class Lex:
    def __init__(self, *args):
        self.l = list(args)
    def __lt__(self, other):
        if isinstance(other, numbers.Number):
            other = Lex(*(other,)*len(self.l))
        return self.l < other.l
    def __gt__(self, other):
        if isinstance(other, numbers.Number):
            other = Lex(*(other,)*len(self.l))
        return self.l > other.l
    def __getitem__(self, b):
        return self.l[b]
```

```

def __eq__(self, other):
    if isinstance(other, numbers.Number):
        other = Lex(*(other,)*len(self.l))
    return self.l == other.l
def __le__(self, other):
    return self < other or self == other
def __ge__(self, other):
    return self > other or self == other
def __add__(self, other):
    return Lex(*[self.l[i]+other.l[i] for i in range(len(self.l))])
def __sub__(self, other):
    return Lex(*[self.l[i]-other.l[i] for i in range(len(self.l))])
def __neg__(self):
    return Lex(*[-self.l[i] for i in range(len(self.l))])
def __str__(self):
    return 'Lex(%s)','%','.join(['%s']*len(self.l)) % tuple(self.l)
def __repr__(self):
    return 'Lex(%s)','%','.join(['%s']*len(self.l)) % tuple(self.l)

```

Listing A2: Python code A2–A class used for the lexicographic transformation (file name *lex.py*).

```

from munkres import Munkres
from lex import Lex
from trfn import TrFN
c11, c12, c13, c14 = (TrFN(0,1,4,15), TrFN(3,5,8,10), TrFN(1,2,7,11),
                    TrFN(4,6,8,9))
c21, c22, c23, c24 = (TrFN(4,5,5,8), TrFN(2,5,6,7), TrFN(1,3,5,11),
                    TrFN(3,6,9,11))
c31, c32, c33, c34 = (TrFN(1,2,8,9), TrFN(1,5,6,8), TrFN(5,6,8,10),
                    TrFN(1,4,5,10))
c41, c42, c43, c44 = (TrFN(5,7,9,12), TrFN(2,4,6,8), TrFN(2,5,6,7),
                    TrFN(4,6,6,7))
C = [[c11, c12, c13, c14],
      [c21, c22, c23, c24],
      [c31, c32, c33, c34],
      [c41, c42, c43, c44]]
f1 = lambda fn: (fn.a+fn.b+fn.c+fn.d)/4
f2 = lambda fn: (fn.b+fn.c)/2
f3 = lambda fn: fn.d-fn.a
f4 = lambda fn: fn.b-fn.a
lexC = [[Lex(f1(C[i][j]), f2(C[i][j]), f3(C[i][j]), f4(C[i][j]))
           for j in range(4)] for i in range(4)]
m = Munkres()
indices = m.compute(lexC)
assignment = [0]*4
cost = TrFN()
for row, column in indices:
    assignment[row] = column + 1
    cost += C[row][column]
print('assignment=(%s,%s,%s,%s)'%tuple(assignment))

```

```
print('fuzzy cost=%s'%cost)
```

Listing A3: Python code A3—Example from Table 2.

## References

- Bellman, R.E.; Zadeh, L.A. Decision-making in fuzzy environment. *Manag. Sci.* **1970**, *17*, 141–164. [[CrossRef](#)]
- Lamata, M.; Pelta, D.; Verdegay, J. Optimisation problems as decision problems: The case of fuzzy optimisation problems. *Inf. Sci.* **2017**, *460–461*, 377–388. [[CrossRef](#)]
- Zadeh, L. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [[CrossRef](#)]
- Lodwick, W.A.; Thipwiwatpotjana, P. *Flexible and Generalized Uncertainty Optimization—Theory and Methods*; Studies in Computational Intelligence; Springer International Publishing AG: Cham, Switzerland, 2017; Volume 696. [[CrossRef](#)]
- Zimmermann, H.J. Description and optimization of fuzzy systems. *Int. J. Gen. Syst.* **1976**, *2*, 209–215. [[CrossRef](#)]
- Tanaka, H.; Okuda, T.; Asai, K. On Fuzzy-Mathematical Programming. *J. Cybern.* **1974**, *3*, 37–46. [[CrossRef](#)]
- Verdegay, J.L. Fuzzy mathematical programming. In *Fuzzy Information and Decision Processes*; Gupta, M.M., Sanchez, E., Eds.; North-Holland Publishing Company: Amsterdam, The Netherlands, 1982; pp. 231–237.
- Verdegay, J.L. Progress on Fuzzy Mathematical Programming: A personal perspective. *Fuzzy Sets Syst.* **2015**, *281*, 219–226. [[CrossRef](#)]
- Tanaka, H.; Asai, K. Fuzzy linear programming problems with fuzzy numbers. *Fuzzy Sets Syst.* **1984**, *13*, 1–10. [[CrossRef](#)]
- Campos, L.; Verdegay, J.L. Linear programming problems and ranking of fuzzy numbers. *Fuzzy Sets Syst.* **1989**, *32*, 1–11. [[CrossRef](#)]
- Cadenas, J.M.; Verdegay, J.L. Using ranking functions in multiobjective fuzzy linear programming. *Fuzzy Sets Syst.* **2000**, *111*, 47–53. [[CrossRef](#)]
- Kaur, J.; Kumar, A. Mehar's method for solving fully fuzzy linear programming problems with L-R fuzzy parameters. *Appl. Math. Model.* **2013**, *37*, 7142–7153. [[CrossRef](#)]
- Yang, X.P.; Cao, B.Y.; Zhou, X.G. Solving fully fuzzy linear programming problems with flexible constraints based on a new order relation. *J. Intell. Fuzzy Syst.* **2015**, *29*, 1539–1550. [[CrossRef](#)]
- Pérez-Cañedo, B.; Concepción-Morales, E.R. A method to find the unique optimal fuzzy value of fully fuzzy linear programming problems with inequality constraints having unrestricted L-R fuzzy parameters and decision variables. *Expert Syst. Appl.* **2019**, *123*, 256–269. [[CrossRef](#)]
- Pérez-Cañedo, B.; Verdegay, J.L.; Miranda Pérez, R. An epsilon-constraint method for fully fuzzy multiobjective linear programming. *Int. J. Intell. Syst.* **2020**, *35*, 600–624. [[CrossRef](#)]
- Dubois, D.; Prade, H. Operations on fuzzy numbers. *Int. J. Syst. Sci.* **1978**, *9*, 613–626. [[CrossRef](#)]
- Bojadziev, G.; Bojadziev, M. *Fuzzy Sets, Fuzzy Logic, Applications; Advances in Fuzzy Systems—Applications and Theory*; World Scientific Publishing: Singapore, 1995; Volume 5, p. 283.
- Hanss, M. *Applied Fuzzy Arithmetic*; Springer: Berlin/Heidelberg, Germany, 2005. [[CrossRef](#)]
- Wang, X.; Kerre, E.E. Reasonable properties for the ordering of fuzzy quantities (I). *Fuzzy Sets Syst.* **2001**, *118*, 375–385. [[CrossRef](#)]
- Wang, M.L.; Wang, H.F.; Chih-Lung, L. Ranking fuzzy number based on lexicographic screening procedure. *Int. J. Inf. Technol. Decis. Mak.* **2005**, *4*, 663–678. [[CrossRef](#)]
- Farhadinia, B. Ranking fuzzy numbers based on lexicographical ordering. *Int. J. Appl. Math. Comput. Sci.* **2009**, *5*, 248–251.
- Greco, S.; Ehrgott, M.; Figueira, J.R. Multiple Criteria Decision Analysis. In *International Series in Operations Research & Management Science*; Springer: New York, NY, USA, 2016; Volume 233, p. 1355. [[CrossRef](#)]
- Buckley, J.J.; Feuring, T. Evolutionary algorithm solution to fuzzy problems: Fuzzy linear programming. *Fuzzy Sets Syst.* **2000**, *109*, 35–53. [[CrossRef](#)]
- Ramík, J.; Řimánek, J. Inequality relation between fuzzy numbers and its use in fuzzy optimization. *Fuzzy Sets Syst.* **1985**, *16*, 123–138. [[CrossRef](#)]
- Ebrahimnejad, A.; Verdegay, J.L. Fuzzy Sets-Based Methods and Techniques for Modern Analytics. In *Studies in Fuzziness and Soft Computing*; Springer International Publishing: Cham, Switzerland, 2018; Volume 364, p. 361. [[CrossRef](#)]

26. Arana-Jiménez, M. Nondominated solutions in a fully fuzzy linear programming problem. *Math. Methods Appl. Sci.* **2018**, *41*, 7421–7430. [[CrossRef](#)]
27. Stanojević, B.; Dzitac, S.; Dzitac, I. Solution approach to a special class of full fuzzy linear programming problems. *Procedia Comput. Sci.* **2019**, *162*, 260–266. [[CrossRef](#)]
28. Ehrgott, M. *Multicriteria Optimization*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2005; p. 323. [[CrossRef](#)]
29. Hashemi, S.M.; Modarres, M.; Nasrabadi, E.; Nasrabadi, M.M. Fully fuzzified linear programming, solution and duality. *J. Intell. Fuzzy Syst.* **2006**, *17*, 253–261.
30. Hosseinzadeh Lotfi, F.; Allahviranloo, T.; Alimardani Jondabeh, M.; Alizadeh, L. Solving a full fuzzy linear programming using lexicography method and fuzzy approximate solution. *Appl. Math. Model.* **2009**, *33*, 3151–3156. [[CrossRef](#)]
31. Kaur, J.; Kumar, A. A New Method to Find the Unique Fuzzy Optimal Value of Fuzzy Linear Programming Problems. *J. Optim. Theory Appl.* **2013**, *156*, 529–534. [[CrossRef](#)]
32. Kaur, J.; Kumar, A. Unique fuzzy optimal value of fully fuzzy linear programming problems. *Control Cybern.* **2012**, *41*, 497–508.
33. Kaur, J.; Kumar, A. An Introduction to Fuzzy Linear Programming Problems: Theory, Methods and Applications. In *Studies in Fuzziness and Soft Computing*; Springer International Publishing: Cham, Switzerland, 2016; Volume 340, p. 132. [[CrossRef](#)]
34. Ezzati, R.; Khorram, E.; Enayati, R. A new algorithm to solve fully fuzzy linear programming problems using the MOLP problem. *Appl. Math. Model.* **2015**, *39*, 3183–3193. [[CrossRef](#)]
35. Hosseinzadeh, A.; Edalatpanah, S.A. A New Approach for Solving Fully Fuzzy Linear Programming by Using the Lexicography Method. *Adv. Fuzzy Syst.* **2016**, *2016*, 1–6. [[CrossRef](#)]
36. Ebrahimnejad, A. An effective computational attempt for solving fully fuzzy linear programming using MOLP problem. *J. Ind. Prod. Eng.* **2019**, *36*, 59–69. [[CrossRef](#)]
37. Bhardwaj, B.; Kumar, A. A note on “A new algorithm to solve fully fuzzy linear programming problems using the MOLP problem”. *Appl. Math. Model.* **2015**, *39*, 5982–5985. [[CrossRef](#)]
38. Pérez-Cañedo, B.; Concepción-Morales, E.R.; Edalatpanah, S.A. A Revised Version of a Lexicographical-based Method for Solving Fully Fuzzy Linear Programming Problems with Inequality Constraints. *Fuzzy Inf. Eng.* **2020**, in press. [[CrossRef](#)]
39. Tosarkani, B.M.; Amin, S.H. A possibilistic solution to configure a battery closed-loop supply chain: Multi-objective approach. *Expert Syst. Appl.* **2018**, *92*, 12–26. [[CrossRef](#)]
40. Das, S.K.; Mandal, T.; Edalatpanah, S.A. A mathematical model for solving fully fuzzy linear programming problem with trapezoidal fuzzy numbers. *Appl. Intell.* **2017**, *46*, 509–519. [[CrossRef](#)]
41. Pérez-Cañedo, B.; Rosete, A.; Verdegay, J.L.; Concepción-Morales, E.R. A Fuzzy Goal Programming Approach to Fully Fuzzy Linear Regression. Processing and Management of Uncertainty in Knowledge-Based Systems. In *Communications in Computer and Information Science*; IPMU 2020; Lesot, M.J., Vieira, S., Reformat, M.Z., Carvalho, J.P., Wilbik, A., Bouchon-Meunier, B., Yager, R.R., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 1238, pp. 677–688. [[CrossRef](#)]
42. Pérez-Cañedo, B.; Concepción-Morales, E.R. On LR-type fully intuitionistic fuzzy linear programming with inequality constraints: Solutions with unique optimal values. *Expert Syst. Appl.* **2019**, *128*, 246–255. [[CrossRef](#)]
43. Kuhn, H.H. The hungarian method for the assignment problem. *Nav. Res. Logist. Q.* **1955**, *2*, 83–97. [[CrossRef](#)]
44. Pérez-Cañedo, B.; Concepción-Morales, E.R. A Lexicographic Approach to Fuzzy Linear Assignment Problems with Different Types of Fuzzy Numbers. *Int. J. Uncertain. Fuzziness Knowl. Based Syst.* **2020**, *28*, 421–441. [[CrossRef](#)]
45. Kumar, A.; Gupta, A. Methods for Solving Fuzzy Assignment Problems and Fuzzy Travelling Salesman Problems with Different Membership Functions. *Fuzzy Inf. Eng.* **2011**, *3*, 3–21. [[CrossRef](#)]
46. Baykasoğlu, A.; Subulan, K.; Karaslan, F.S. A new fuzzy linear assignment method for multi-attribute decision making with an application to spare parts inventory classification. *Appl. Soft Comput.* **2016**, *42*, 1–17. [[CrossRef](#)]
47. Yu, P.L. A class of solutions for group decision problems. *Manag. Sci.* **1973**, *19*, 936–946. [[CrossRef](#)]

48. Burkard, R.E.; Hahn, W.; Zimmermann, U. An algebraic approach to assignment problems. *Math. Program.* **1977**, *12*, 318–327. [[CrossRef](#)]
49. Clapper, B. Munkres Implementation for Python. Available online: <http://software.clapper.org/munkres/> (accessed on 2 August 2020).
50. Chen, S.H. Ranking fuzzy numbers with maximizing set and minimizing set. *Fuzzy Sets Syst.* **1985**, *17*, 113–129. [[CrossRef](#)]
51. Abbasbandy, S.; Hajjari, T. A new approach for ranking of trapezoidal fuzzy numbers. *Comput. Math. Appl.* **2009**, *57*, 413–419. [[CrossRef](#)]
52. Rahmani, A.; Hosseinzadeh Lotfi, F.; Rostamy-Malkhalifeh, M.; Allahviranloo, T. A New Method for Defuzzification and Ranking of Fuzzy Numbers Based on the Statistical Beta Distribution. *Adv. Fuzzy Syst.* **2016**, *2016*, 1–8. [[CrossRef](#)]
53. Buckley, J.J.; Feuring, T.; Hayashi, Y. Multi-objective fully fuzzified linear programming. *Int. J. Uncertain. Fuzziness Knowl. Based Syst.* **2001**, *9*, 605–621. [[CrossRef](#)]
54. Jalil, S.A.; Sadia, S.; Javaid, S.; Ali, Q.M. A Solution Approach for Solving Fully Fuzzy Multiobjective Solid Transportation Problem. *Int. J. Agric. Stat. Sci.* **2017**, *13*, 75–84.
55. Nasser, H.; Morteza, M.; Mirmohseni, M. A New Method for Solving Fully Fuzzy Multi Objective Supplier Selection Problem. *Int. J. Res. Ind. Eng.* **2017**, *6*, 214–227. [[CrossRef](#)]
56. Yang, X.P.; Cao, B.Y.; Lin, H.T. Multi-objective fully fuzzy linear programming problems with triangular fuzzy numbers. In Proceedings of the 2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Xiamen, China, 19–21 August 2014; pp. 171–177. [[CrossRef](#)]
57. Pérez-Cañedo, B.; Verdegay, J.L.; Rosete, A.; Concepción-Morales, E.R. Fully Fuzzy Multi-Objective Berth Allocation Problem. In *Hybrid Artificial Intelligent Systems*; Forthcoming; Lecture Notes In Computer Science; Springer: Cham, Switzerland, 2020.
58. Bello, R.; Verdegay, J.L. Rough sets in the Soft Computing environment. *Inf. Sci.* **2012**, *212*, 1–14. [[CrossRef](#)]
59. Villacorta, P.J.; Rabelo, C.A.; Pelta, D.A.; Verdegay, J.L. FuzzyLP: An R Package for Solving Fuzzy Linear Programming Problems. In *Granular, Soft and Fuzzy Approaches for Intelligent Systems*; Studies in Fuzziness and Soft Computing; Kacprzyk, J., Filev, D., Beliakov, G., Eds.; Springer International Publishing: Cham, Switzerland, 2017; Volume 344. [[CrossRef](#)]
60. Biswas, P.; Pal, B.B. A fuzzy goal programming method to solve congestion management problem using genetic algorithm. *Decis. Making: Appl. Manag. Eng.* **2019**, *2*, 36–53. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).