E-ISSN: 2717-3453 | P-ISSN: 2783-1442

Journal of Fuzzy Extension and Applications

www.journal-fea.com

J. Fuzzy. Ext. Appl. Vol. 1, No. 4 (2020) 252–271.

Paper Type: Research Paper

Fuzzy Programming Approach to Bi-level Linear Programming Problems

Eshetu Dadi Gurmu [1,*], Tagay Takele Fikadu[1]

[1] Department of Mathematics, Wollega University, Nekemte, Ethiopia; eshetudadi1@gmail.com; tagay4@gmail.com.

Citation:

Gurmu, E. D., & Fikadu, T. T. (2020). Fuzzy programming approach to Bi-level linear programming problems. *Journal of fuzzy extension and application, 1* (4), 252-271.

Received: 11/07/2020 Reviewed: 09/08/2020 Revised: 17/10/2020 Accept: 25/11/2020

Abstract

In this study, we discussed a fuzzy programming approach to bi-level linear programming problems and their application. Bi-level linear programming is characterized as mathematical programming to solve decentralized problems with two decision-makers in the hierarchal organization. They become more important for the contemporary decentralized organization where each unit seeks to optimize its own objective. In addition to this, we have considered Bi-Level Linear Programming (BLPP) and applied the Fuzzy Mathematical Programming (FMP) approach to get the solution of the system. We have suggested the FMP method for the minimization of the objectives in terms of the linear membership functions. FMP is a supervised search procedure (supervised by the upper Decision Maker (DM)). The upper-level decision-maker provides the preferred values of decision variables under his control (to enable the lower level DM to search for his optimum in a wider feasible space) and the bounds of his objective function (to direct the lower level DM to search for his solutions in the right direction).

Keywords: Fuzzy set, Fuzzy function, Fuzzy linear programming, Bi Level programming.

1 | Introduction

Decision making problems in decentralized organizations are often modeled as stackelberg games, and they are formulated as bi-level mathematical programming problems. A bi-level problem with a single decision maker at the upper level and two or more decision makers at the lower level is referred to as a decentralized bi-level programming problem. Real-world applications under non cooperative situations are formulated by bi-level mathematical programming problems and their effectiveness is demonstrated.

The use of fuzzy set theory for decision problems with several conflicting objectives was first introduced by Zimmermann. Thereafter, various versions of Fuzzy Programming (FP) have been investigated and widely circulated in literature. The use of the concept of tolerance membership function of fuzzy set theory to Bi-Linear Programming Problems (BLPPs) for satisfactory decisions

Corresponding Author: eshetudadi1@gmail.com

10.22105/JFEA.2021.281500.1061

was first introduced by Lai in 1996 [1]. Shih and Lee further extended Lai's concept by introducing the compensatory fuzzy operator for solving BLPPs [2]. Sinha studied alternative BLP techniques based on Fuzzy Mathematical Programming (FMP).

The basic concept of these FMP approaches is the same as Fuzzy Goal Programming (FGP) approach which implies that the lower level DMs optimizes, his/her objective function, taking a goal or preference of the higher level DMs in to consideration. In the decision process, considering the membership functions of the fuzzy goals for the decision variables of the higher level DM, the lower level DM solves a FMP problem with a constraint on an overall satisfactory degree of the higher level DMs. If the proposed solution is not satisfactory, to the higher level DMs, the solution search is continued by redefining the elicited membership functions until a satisfactory solution is reached [2]. The main difficulty that arises with the FMP approach of Sinha is that there is possibility of rejecting the solution again and again by the higher level DMs and re-evaluation of the problem is repeatedly needed to reach the satisfactory decision, where the objectives of the DMs are over conflicting [2].

Taking in to account vagueness of judgments of the decision makers, we will present interactive fuzzy programming for bi-level linear programming problems. In the interactive method, after determining the fuzzy goals of the decision makers at both levels, a satisfactory solution is derived by updating some reference points with respect to the satisfactory level. In the real world, we often encounter situations where there are two or more decision makers in an organization with a hierarchical structure, and they make decisions in turn or at the same time so as to optimize their objective functions. In particular, consider a case where there are two decision makers; one of the decision makers first makes a decision. Such a situation is formulated as a bi-level programming problem. Although a large number of algorithms for obtaining stackelberg solutions have been developed, it is also known that solving the mathematical programming problems for obtaining stackelberg solution is NP-hard [3]. From such difficulties, a new solution concept which is easy to compute and reflects structure of bi-level programming problems had been expected [4] proposed a solution method, which is different from the concept of stackelberg solutions, for bi-level linear programming problems with cooperative relationship between decision makers. Sakawa and Nishizaki [5] present interactive fuzzy programming for bi-level linear programming problems. In order to overcome the problem in the methods of [4], after eliminating the fuzzy goals for decision variables, they formulate the bi-level linear programming problem.

In their interactive method, after determining the fuzzy goals of the decision makers at all the levels, a satisfactory solution is derived efficiently by updating the satisfactory degree of the decision maker at the upper level with considerations of overall satisfactory balance among all the levels. By eliminating the fuzzy goals for the decision variables to avoid such problems in the method of [4]-[6] develop interactive fuzzy programming for bi-level linear programming problems. Moreover, from the viewpoint of experts' imprecise or fuzzy understanding of the nature of parameters in a problem-formulation process, they extend it to interactive fuzzy programming for bi-level linear programming problems with fuzzy parameters [5]. Interactive fuzzy programming can also be extended so as to manage decentralized bi-level linear programming problems by taking in to consideration individual satisfactory balance between the upper level DM and each of the lower level DMs as well as overall satisfactory balance between the two levels [7]. Moreover, by using some decomposition methods which take advantage of the structural features of the decentralized bi-level problems, efficient methods for computing satisfactory solutions are also developed [7] and [8].

Recently, [9]-[11] considered the L-R fuzzy numbers and the lexicography method in conjunction with crisp linear programming and designed a new model for solving FFLP. The proposed scheme presented promising results from the aspects of performance and computing efficiency. Moreover, comparison between the new model and two mentioned methods for the studied problem shows a remarkable agreement and reveals that the new model is more reliable in the point of view of optimality. Also, an author in [12]-[15] has been proposed a new efficient method for FFLP, in order to obtain the fuzzy optimal solution with unrestricted variables and parameters. This proposed method is based on crisp nonlinear programming and has a simple structure.

Furthermore, several authors deal with the modeling and optimization of a bi-level multi-objective production planning problem, where some of the coefficients of objective functions and parameters of constraints are multi-choice. They has been used a general transformation technique based on a binary variable to transform the multi-choices parameters of the problem into their equivalent deterministic form [16]-[21].

In this study, we discuss a procedure for solving bi-level linear programming problems through linear FMP approach. In order to reach the optimal solution of bi-level linear programming problems, using fuzzy programming approach, the report contains section three chapters. In section two we describe the basic concept of fuzzy set, and linear programming using fuzzy approach. In section three the basic concept of bi level linear programming characteristics and general model of mathematical formulation of bi -level linear programming problems are presented. In section, four the procedure for solving bi-level linear programming problems and FMP solution approach are discussed.

2| Preliminary

2.1| Fuzzy Set Theory

Fuzzy set theory has been developed to solve problems where the descriptions of activities and observations are imprecise, vague, or uncertain. The term "fuzzy" refers to a situation where there are no well-defined boundaries of the set of activities or observations to which the descriptions apply. For example, one can easily assign a person 180 cm tall to the class of tall men". But it would be difficult to justify the inclusion or exclusion of a 173 cm tall person to that class, because the term "tall" does not constitute a well- defined boundary. This notion of fuzziness exists almost everywhere in our daily life, such as a "class of red flowers," a "class of good shooters," a "class of comfortable speeds for travelling," a "number close to 10,"etc.These classes of objects cannot be well represented by classical set theory. In classical set theory, an object is either in a set or not in a set. An object cannot partially belong to a set .In fuzzy set theory, we extend the image set of the characteristic function from the binary set $B = \{0,1\}$ which contains only two alternatives, to the unit interval $U = [0,1]$ which has an infinite number of alternatives. We even give the characteristic function a new name, the membership function, and a new symbol μ, instead of χ. The vagueness of language, and its mathematical representation and processing, is one of the major areas of study in fuzzy set theory.

2.2| Definition of Fuzzy and Crisp Sets

Definition 1. Let X be a space of points (objects) called universal or referential set .An ordinary (a crisp) subset A in X is characterized by its characteristic function X_A as mapping from the elements of X to the elements of the set $\{0,1\}$ defined by;

$$X_A(x) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A \end{cases}.$$

Where $\{0, 1\}$ is called a valuation set. However, in the fuzzy set t, the membership function will have not only 0 and 1 but also any number in between. This implies that if the valuation set is allowed to be the real interval $[0,1]$, A is called a fuzzy set.

Definition 2. If X is a collection of objects denoted by x, then a fuzzy set A is a set of ordered pairs denoted by $A = \{(x, \mu_A(x)) \mid x \in X\}$. Where $\mu_A(x): X \to [0,1]$ is called membership function or degree of membership (degree of compatibility or degree of truth).

Gurmu and Fikadu|J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

Definition 3. A fuzzy set A in a non empty set X is categorized by its membership function $\mu_A(x): X \to [0,1]$ and $\mu_A(x)$ is called the degree of membership of element x in fuzzy set A for each x is an element of X that makes values in the interval $[0,1]$.

Definition 4. Let X be a universal set and A is a subset of X. A fuzzy set of A in x is a set of ordered pairs $A = \{(x, \mu_A(x)) \mid x \in X\}$ where, $\mu_A(x) \to [0,1]$ is called the membership function at x in membership, the value one is used to represent complete membership and value zero is used to represent intermediate degree of membership.

Example 1. let $X = \{a, b, c\}$ and define the fuzzy set A as follows:

$$\mu_A(a) = 1.0, \qquad \mu_A(b) = 0.7, \qquad \mu_A(c) = 0.4,$$

$$A = \{(a, 1.0), (b, 0.7), (c, 0.4)\}.$$

Note. The statement, $\mu_A(b) = 0.7$ is interpreted as saying that the membership grade of 'b' in the fuzzy set A is seven-tenths. i.e. the degree or grade to which b belongs to A is 0.7.

Definition 5. A fuzzy set $A = \varnothing$ if and only if it is identically zero on X.

Definition 6. If two fuzzy sets A andfuzzy set B are equal then $A = B$, if and only if $A(x) = B(x), \ \forall x \in X$.

2.3| Fuzzy Linear Programming

Crisp linear programming is one of the most important operational research techniques. It is a problem of maximizing or minimizing a crisp objective function subject to crisp constraints (crisp linear-inequalities and/or equations). It has been applied to solve many real world problems but it fails to deal with imprecise data, that is, in many practical situations it may not be possible for the decision maker to specify the objective and/or the constraint in crisp manner rather he/she may have put them in "fuzzy sense". So many researchers succeeded in capturing such vague and imprecise information by fuzzy programming problem. In this case, the type of the problem he/she put in the fuzziness should be specified, that means, there is no general or unique definition of fuzzy linear problems. The fuzziness may appear in a linear programming problem in several ways such as the inequality may be fuzzy (p1–FLP), the objective function may be fuzzy (P2-FLP) or the parameters c, A, b may be fuzzy (P3-FLP) and so on.

Definition 7. If an imprecise aspiration level is assigned to the objective function, then this fuzzy objective is termed as fuzzy goal. It is characterized by its associated membership function by defining the tolerance limits for achievement of its aspired level.

We consider the general model of a linear programming

$$\begin{aligned} &\max \ C^T x, \\ &\text{s.t.} \\ &A_i x \le b_i \quad (i = 1,2,3, \dots m), \\ &x \ge 0, \end{aligned} \tag{1}$$

Where A_i is an n-vector C is an n-column vector and $x \in \mathbb{R}^n$.

To a standard linear programming *Problem (1)* above, taking in to account the imprecision or fuzziness of a decision maker's judgment, Zimmermann considers the following linear programming problem with a fuzzy goal (objective function) and fuzzy constraints.

$$C^T x \lesssim Z_0, \quad (1a)$$

$$A_i x \lesssim b_i \quad (i = 1,2,3, \dots m), \quad (1b) \qquad (2)$$

$$x \geq 0.$$

Where the symbol $\lesssim$ denotes a relaxed or fuzzy version of the ordinary inequality $<$. From the decision maker's preference, the fuzzy goal (1a) and the fuzzy constraints (1b) mean that the objective *function* $C^T x$ should be "essentially smaller than or equal to" a certain level Z_0, and that the values of the constraints AX should be "essentially smaller than or equal to" b, respectively. Assuming that the fuzzy goal and the fuzzy constraints are equally important, he employed the following unified formulation.

$$Bx \lesssim b',$$

$$x \geq 0.$$

Where $B = \begin{bmatrix} C \\ A_i \end{bmatrix}$ *and* $b' = \begin{bmatrix} Z_0 \\ b_i \end{bmatrix}$.

Definition 8. Fuzzy decision is the fuzzy set of alternatives resulting from the intersection of the fuzzy constraints and fuzzy objective functions. Fuzzy objective functions and fuzzy constraints are characterized by their membership functions.

2.4 | Solution Techniques of Solving Some Fuzzy Linear Programming Problems

The solution techniques for fuzzy linear programming problems follow the following procedure. We consider the following linear programming problem with fuzzy goal and fuzzy constraints (the coefficients of the constraints are fuzzy numbers).

Where $\widetilde{a_{ij}}$ and $\widetilde{b_i}$ are fuzzy numbers with the following linear membership functions:

$$\mu_{ij}(x) = \begin{cases} 1, & \text{if } x \leq a_{ij}, \\ \dfrac{a_{ij} + d_{ij} - x}{d_{ij}}, & \text{if } a_{ij} < x < a_{ij} + d_{ij}, \\ 0, & \text{if } x \geq a_{ij} + d_{ij}. \end{cases}$$

$$\mu_{\widetilde{b_i}}(x) = \begin{cases} 1, & \text{if } x \leq b_i, \\ \dfrac{b_i + p_i - x}{p}, & \text{if } b_i < x < b_i + p_i, \\ 0, & \text{if } x \geq b_i + p_i. \end{cases}$$

and $x \in R, dij > 0$ is the maximum tolerance for the corresponding constraint coefficients and p_i is the maximum tolerance for the i^{th} constraint. For defuzzification of the problem, we first fuzzify the objective function. This is done by calculating the lower and upper bounds of the optimal values. These optimal values z_l and z_u can be defined by solving the following standard linear programming problems, for which we assume that both of them have finite optimal values.

Gurmu and Fikadu | J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

$$\begin{aligned} & z_1 = \max \sum_{j=1}^{n} c_j x_j, \\ & \text{s. t.} \\ & \qquad \sum_{j=1}^{n} (a_{ij} + d_{ij}) x_j \le b_i, \ \ 1 \le i \le m)\ x_j \ge 0, \\ & \qquad \text{and} \\ & \qquad z_2 = \max \sum_{j=1}^{n} c_j x_j\,, \\ & \qquad \text{s. t.} \\ & \qquad \sum_{j=1}^{n} a_{ij} x_j \le b_i + p_i, \ \ 1 \le i \le m) x_j \ge 0. \end{aligned} \tag{3}$$

Let $z_l = min(z_1, z_2)$ and $z_u = max(z_1, z_2)$. The objective function takes values between z_l and z_u while the constraint coefficients take values between a_{ij} and $a_{ij} + d_{ij}$and the right-hand side numbers take values between b_i and $b_i + p_i$ .Then, the fuzzy set optimal values, G, which is a subset of R^n is defined by:

$$\mu_G(x) = \begin{cases} 0, & \text{if } \sum_{j=1}^{n} c_j x_j \le z_l, \\ \dfrac{\sum_{j=1}^{n} c_j x_j - z_l}{z_u - z_l}, & \text{if } z_l < \sum_{j=1}^{n} c_j x_j \le z_u. \\ 1, & \text{if } \sum_{j=1}^{n} c_j x_j \ge z_u. \end{cases}$$

The fuzzy set of the i^{th} constraint, C_i , which is a subset of R^n is defined by:

$$\mu_{ci}(x) = \begin{cases} 0, & \text{if } b_i \le \sum_{j=1}^{n} a_{ij} x_j\,, \\ \dfrac{b_i - \sum_{j=1}^{n} a_{ij} x_j}{\sum_{j=1}^{n} d_{ij} x_j + p_i}, & \text{if } \sum_{j=1}^{n} a_{ij} x_j < b_i < \sum_{j=1}^{n} \left(a_{ij} x_j + d_{ij}\right) x_j + p_i. \\ 1, & \text{if } b \ge \sum_{j=1}^{n} \left(a_{ij} x_j + d_{ij}\right) x_j + p_i. \end{cases}$$

Using the above membership functions $\mu_{ci}(x)$ and $\mu_G(x)$and following Bellmann and Zadeh approach, we construct the membership function $\mu_D(x)$as follows: $\mu_D(x) = \text{mini}(\mu_G(x), \mu_{ci}(x))$.

Where $\mu_D(x)$ is the membership function of the fuzzy decision set. The min. section is selected as the aggregation operator. Then the optimal decision x^* is the solution of $x^* = \arg(\max \text{mini}\{\mu_G(x), \mu_{ci}(x)\}$.

Then, *Problem (1)* is reduced to the following crisp problem by introducing the auxiliary variable λ which indicates the common degree of satisfaction of both the fuzzy constraints and objective function.

$$\max \lambda,$$
$$\text{s.t.}$$
$$\mu_G(x) \geq \lambda,$$
$$\mu_{ci}(x) \geq \lambda,$$
$$x \geq 0, 0 \leq \lambda \leq 1,\ 1 \leq i \leq m.$$

This problem is equivalent to the following non-convex optimization problem

$$\max \lambda,$$
$$\lambda(z_1 - z_2) - \sum_{j=1}^{n} c_j x_j - z_1 \leq 0,$$
$$\sum_{j=1}^{n} \left(a_{ij} + \lambda d_{ij}\right) x_j + \lambda p_i - b_i \leq 0\,,$$
$$x \geq 0, 0 \leq \lambda \leq 1, \qquad 1 \leq i \leq m\,.$$

Which contains the cross product terms λx_j that makes non- convex. Therefore, the solution of this problem requires the special approach such as fuzzy decisive method adopted for solving general non-convex optimization problems. Here solving the above linear programming problem gives us an optimum $\lambda^* \in [0,1]$. Then the solution of the problem is any $x \geq 0$ satisfying the problem constraint with $\lambda = \lambda^*$.

3| Bi-Level Programming

3.1| Basic Definitions

3.1.1| Decision making

Decision making is a process of choosing an action (solution) from a set of possible actions to optimize a given objective.

3.1.2| Decision making under multi objectives

In most real situation a decision maker needs to choose an action to optimize more than one objective simultaneously. Most of these objectives are usually conflicting. For example, a manufacturer wants to increase his profit and at the same time want to produce a product with better quality. Mathematically a multi objective optimization with k objectives, for a natural number $K > 1$, can be given as:

$$\max F(x) = (f_1(x), \qquad f_2(x), \dots, f_k(x))\,,$$
$$\text{s.t.}$$
$$x \in S \subseteq \mathbb{R}^n.$$

3.1.3| Hierarchical decision making

An optimization problem which has other optimization problems in the constraint set and has a decision maker for each objective function controlling part of the variables is called multi-level optimization problem. If there are only two nested objective functions then it is called a bi-level optimization problem. The decision maker at the first level, with objective function f_1, is called the leader and the other decision makers are called the followers. A solution is supposed to fulfill all the feasibility conditions and optimize each objectives it is uncommon to find a solution which makes all the decision makers happy. Hence to choose an action the preference of the decision makers for all the levels or objectives play a big role.

3.1.4| Bi-level programming (BLP)

is a mathematical programming problem that solves decentralized planning problems with two DMs in a two level or hierarchical organization. It has been studied extensively since the 1980s. It often represents an adequate tool for modeling non-cooperative hierarchical decision process, where one player optimizes over a subset of decision variables, while taking in to account the independent reaction of the other player to his or course of action. In the real world, we often encounter situations where there are two or more decision makers in an organization with a hierarchical structure, and they make decisions in turn or at the same time so as to optimize their objective functions. In particular, consider a case where there are two decision makers; one of the decision makers first makes a decision, and then the other who knows the decision of the opponent makes a decision. Such a situation is formulated as a bi-level programming problem. We call the decision maker who first makes a decision the leader, and the other decision maker the follower. For bi-level programming problems, the leader first specifies (decides) a decision and then the follower determines a decision so as to optimize the objective function of the follower with full knowledge of the decision of the leader. According to this rule, the leader also makes a decision so as to optimize the objective function of self. This decision making process is extremely practical to such decentralized systems as agriculture, government policy, economic systems, finance, warfare, transportation, network designs, and is especially for conflict resolution.

Bi-level programming is particularly appropriate for problems with the following characteristics:

- ***Interaction:*** *Interactive decision-making units within a predominantly hierarchical structure.*
- ***Hierarchy:*** *Execution of decision is sequential, from upper to lower level.*
- ***Full information:*** *Each DM is fully informed about all prior choices when it is his turn to move.*
- ***Nonzero sum:*** *The loss for the cost of one level is unequal to the gain for the cost of the other level. External effect on a DM's problem can be reflected in both the objective function and the set of feasible decision space.*
- *Each DM controls only a subset of the decision variables in an organization.*

3.2| Mathematical Formulation of a Bi-Level Linear Programming Problem (BLPP)

For the bi-level programming problems, the leader first specifies a decision and then the follower determines a decision so as to optimize the objective function of self with full knowledge of the decision of the leader. According to this rule, the leader also makes a decision so as to optimize the objective function of self. The solution defined as the above mentioned procedure is a stackelberg solution.

A bi-level LPP for obtaining the stackelberg solution is formulated as:

$$\begin{aligned} &\max_{x_1.} z_1(x_1, x_2) = c_1x_1 + d_1x_2, \\ &\text{Where } x_2 \text{ solves} \\ &\max_{x_2,} z_2(x_1, x_2) = c_2x_1 + d_2x_2, \\ &\text{s.t.} \\ &Ax_1 + Bx_2 \leqq b. \end{aligned} \tag{4}$$

Where $c_i, i = 1,2$ are n_1-dimensional row coefficient vector$d_i, i = 1,2$, are n_2- dimensional row coefficient vector, A is an mxn1 coefficient matrix, B is a mxn_2 coefficient matrix, b-is an m-dimensional column constant vector. In the bi-level linear programming problem above$z_1(x_1, x_2)$, and $z_2(x_1, x_2)$represent the objective functions of the leader and the follower, respectively, and x_1 and x_2 represent the decision variables of the leader and the follower respectively. Each decision maker knows the objective function of self and the constraints. The leader first makes a decision, and then the follower makes a decision so as to maximize the objective function with full knowledge of the decision of the leader. Namely, after the leader chooses x_1, he solves the following linear programming problem:

$$\max_{x_2} z_2(x_1, x_2) = c_2x_1 + d_2x_2, \tag{5}$$

$$\text{s.t.}$$
$$Bx_2 \leq b - Ax_1,$$

$$x_2 \geqq 0.$$

And chooses an optimal solution $x_2(x_1)$ to the problem above as a rational response. Assuming that the follower chooses the rational response, the leader also makes a decision such that the objective function $z_1(x_1, x_2(x_1))$ is maximized.

3.3 | BLP Problem Description

The linear bi-level programming problem is similar to standard linear programming, except that the constraint region is modified to include a linear objective function constrained to be optimal with respect to one set of variables. The linear BLPP characterized by two planners at different hierarchical levels each independently controlling only a set of decision variables, and with different conflicting objectives. The lower- level executes its policies after and in view of, the decision of the higher level , and the higher level optimizes its objective independently which is usually affected by the reactions of the lower level. Let the control over all real-valued decision variables in the vector $x = (x_1^1, x_1^2, \dots, x_1^{N(1)}, x_2^1, x_2^2, \dots, x_2^{N(2)})$ be partitioned between two planners ,hereafter known as level-one(the superior or top planner) and level-two(the inferior or bottom planner).Assume that the level-one has control over the vector $x = (x_1^1, x_1^2, \dots, x_1^{N(1)})$, the first $N(1)$ components of the vector x, and that the level-two has control over the vector $x = (x_2^1, x_2^2, \dots, x_2^{N(2)})$ the remaining $N(2)$ components .Further, assume that $f_1, f_2: R^{N(1)} x\, R^{N(2)} \to R^1$ linear. Then, the linear BLPP can be formulated as:

$$\max_{x_1.} f_1(x_1, x_2) = c_1x_1 + d_1x_2, \tag{6}$$
$$\text{Where } x_2 \text{ solves}$$

$$\max_{x_2,} f_2(x_1, x_2) = c_2x_1 + d_2x_2, \tag{7}$$
$$\text{s.t.} (x_1, x_2) \in S.$$

Where $S \subseteq R^{N(1)+N(2)}$ is the feasible choices of (x_1, x_2), and is closed and bounded. For any fixed choice of x_1, level-two will choose a value of x_2 to maximize the objective function $f_1(x_1, x_2)$. Hence, for each feasible value of x_1, level-two will react with a corresponding value of x_2. This induces a functional reaction ship between the decisions of level-one and the reactions of level-two. Say, $x_2 = W(x_1)$ .We will assume that the reaction function, $W(.)$, is completely known by level one.

Gurmu and Fikadu|J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

Definition 9. The set $Wf_2(S)$ given by $Wf_2(S) = \{(x_1^*, x_2^*) \in S: f_2(x_1^*, x_2^*) = max\, f_2(x_1^*, x_2^*)$ is the set of rational reactions of $f_2$over S. Hence level-one is really restricted to choosing a point in the set of rational reactions of f_2 over S. So, if level-one wishes to maximize its objective function, $f_1(x_1, x_2)$,by controlling only the vector x_1, it must solve the following mathematical programming problem:

$$\begin{aligned} &\max f_1(x_1, x_2), \\ &\text{s.t.} \\ &(x_1, x_2) \in Wf_2(S). \end{aligned} \tag{8}$$

For convenience of notation and terminology, we will refer to $S^1 = Wf_2(S)$ as the level-one feasible region or in general, the feasible region, and $S^1 = S$ as the level two feasible regions.

The following are the basic concepts of the bi-level linear programming problem of *Eq. 3*:

The feasible region of the bi-level linear programming problem: $S = \{(x_1, x_2): Ax_1 + Bx_2 \leqq b\}$.

The decision space (feasible set) of the follower after x_1 is specified by the leader: $S(x_1) = \{x_2 \geqq 0: Bx_2 < b - Ax_1, x_1 \geqq 0\}$.

The decision space of the leader: $S_x = \{x_1 \geqq 0$ there is an x_2 such that $Ax_1 + Bx_2 \leqq b, x_2 \geqq 0\}$.

The set of rational responses of the follower for x_1 specified by the leader

$$R(x_1) = \begin{cases} x_2 \geqq 0: x_2 \in \arg\max z_1(x_1, x_2) \\ \qquad\qquad x_2 \in S(x_1) \end{cases}.$$

Inducible region: $IR = \{(x_1, x_2): (x_1, x_2) \in S, x_2 \in R(x_1)\}$.

Stackleberg solution: $\left\{(x_1, x_2): (x_1, x_2) \in \arg\max z_1(x_1, x_2), (x_1, x_2) \in R(x_1)\right\}$.

Computational methods for obtaining stackelberg solution to bi-level linear programming problems are classified roughly in to three categories. These are

The vertex enumeration approach [2]. This takes advantage of the property that there exists a stackelberg solution in a set of extreme points of the feasible region. The solution search procedure of the method starts from the first best point namely an optimal solution to the upper level problem which is the first best solution, is computed, and then it is verified whether the first best solution is also an optimal solution to the lower level problem. If the first best point is not the stackelberg solution, the procedure continues to examine the second best solution to the problem of the upper level, and so forth.

The Kuhn-Tucker approach. In this approach, the leader's problem with constraints involving the optimality conditions of the follower's problem is solved.

The penalty function approach. In this approach, a penalty term is appended to the objective function of the leader so as to satisfy the optimality of the follower's problem.

Fuzzy approach:-that will be discussed in detail under the next chapter.

4| Fuzzy Approach to Bi-Level Linear Programming Problems

4.1| Fuzzy Bi-Level Linear Programming

As discussed under chapter two, a bi-level linear programming problem is formulated as:

$$\max \mathrm{f}_1(\mathrm{x}_1, \mathrm{x}_2) = \mathrm{c}_{11}\mathrm{x}_1 + \mathrm{c}_{12}\mathrm{x}_2,$$

$$\mathrm{x}_1. \tag{9}$$

Where x_2 solves

$$\max \mathrm{f}_2(\mathrm{x}_1, \mathrm{x}_2) = \mathrm{c}_{21}\mathrm{x}_1 + \mathrm{c}_{22}\mathrm{x}_2,$$

$$\text{s.t.}$$

$$\mathrm{A}_1\mathrm{x}_1 + \mathrm{A}_2\mathrm{x}_2 \leq \mathrm{b},$$

$$(\mathrm{x}_1, \mathrm{x}_2) \geq 0.$$

Where $x_i, i = 1{,}2$ is an n_i-dimensional decision variable column vector ;

$C_{i1}, i = 1{,}2$ is an n_1-dimensional constant column vector;

$C_{i2}, i = 1{,}2$ is an n_2-dimensional constant column vector;

b-is an m-dimensional constant column vector, and

$A_i, i = 1{,}2$ is an mxni coefficient matrix.

For the sake of simplicity, we use the following notations:

$X = (x_1, x_2) \in R^{n_1+n_2}, C_i = (C_{i1}, C_{i2}), i = 1{,}2$ and $A = [A_1, A_2]$ and Let DM_1 denotes the decision maker at the upper level and DM_2 denotes the decision maker at the lower level. In the bi-level linear programming problem (7) above, $f_1(x_1, x_2)$ and $f_2(x_1, x_2)$represent the objective functions of DM_1 and DM_2 respectively; and x_1 and x_2 represent the decision variables of DM_1 and DM_2 respectively.

Instead of searching through vertices as the k^{th} best algorithm, or the transformation approach based on Kuhn-Tucker conditions, we here introduce a supervised search procedure (supervised by DM_1) which will generate (non dominated) satisfactory solution for a bi-level programming problem. In this solution search, DM_1 specifies(decides) a fuzzy goal and a minimal satisfactory level for his objective function and decision vector and evaluates a solution proposed by DM_2, and DM_2 solves an optimization problem, referring to the fuzzy goal and the minimal satisfactory level of DM_1. The DM_2 then presents his/her solution to the DM_1. If the DM_1 agrees to the proposed solution, a solution is reached and it is called a satisfactory solution here. If he/she rejects this proposal, then DM_1 will need to re-evaluate and change former goals and decisions as well as their corresponding leeway or tolerances until a satisfactory solution is reached. It is natural that decision makers have fuzzy goals for their objective functions and their decision variables when they take fuzziness of human judgments in to consideration .For each of the objective functions $f_i(x)$, $i = 1{,}2$, assume that the decision makers have fuzzy goals such as "the objective function $f_i(x)$ should be substantially less than or equal to some value q_i " and the range of the decision on x_i , $i = 1{,}2$,should be " around $x_i^*$with its negative and positive – side tolerances $p_i^-$and p_i^+,respectively.

Gurmu and Fikadu|J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

We obtain optimal solution of each DM$_1$ and DM$_2$ calculated in isolation. If the individual optimal solution x_i^0 , $i =$ 1.2; are the same then a satisfactory solution of the system has been attained. But this rarely happens due to conflicting objective functions of the two DMs. The decision-making process then begins at the first level. Thus, the first-level DM provides his preferred ranges for $f_1$and decision vector x_1 to the second level DM. This information can be modeled by fuzzy set theory using membership functions.

4.2| Fuzzy Programming Formulation of BLPPs

To formulate the fuzzy programming model of a BLPP, the objective functions f_i, (i = 1,2) and the decision vectors x_i, (i = 1,2) would be transformed in to fuzz goals by means of assigning an aspiration level (the optimal solutions of both of the DMs calculated in isolation can be taken as the aspiration levels of their associated fuzzy goals) to each of them. Then, they are to be characterized by the associated membership functions by defining tolerance limits for achievement of the aspired levels of the corresponding fuzzy goals.

4.3| Fuzzy Programming Approach for Bi-Level LPPs

In the decision making context, each DM is interested in maximizing his or her own objective function, the optimal solution of each DM when calculated in isolation would be considered as the best solution and the associated objective value can be considered as the aspiration level of the corresponding fuzzy goal because both the DMs are interested of maximizing their own objective functions over the same feasible region defined by the system of constraints. Let x_i^B be the best (optimal) solution of the i^{th} level DM. It is quite natural that objective values which are equal to or larger than $f_i^B = f_i(x_i^B) = max\ f_i(x)$, i = 1,2. , $x \in S$ should be absolutely satisfactory to the i^{th} level DM. If the individual best (optimal) solution $x_i^B, i = 1,2$ are the same, then a satisfactory optimal solution of the system is reached. However, this rarely happens due to the conflicting nature of the objectives. To obtain a satisfactory solution, higher level DM should give some tolerance (relaxation) and the relaxation of decision of the higher level DM depends on the needs, desires and practical situations in the decision making situation .Then the fuzzy goals take the form $f_i(x) \lesssim f_i(x_i^B),\ i = 1,2, x_i \cong x_i^B$.

To build membership functions, goals and tolerance should be determined first. However, they could hardly be determined without meaningful supporting data. Using the individual best solutions, we find the values of all the objective functions at each best solution and construct a payoff matrix

$$\begin{bmatrix} & f_1(x) & f_2(x) \\ x_1^0 & f_1(x_1^0) & f_2(x_1^0) \\ x_2^0 & f_1(x_2^0) & f_2(x_2^0) \end{bmatrix}.$$

The maximum value of each column $\left(f_i(x_i^0)\right)$ gives upper tolerance limit or aspired level of achievement for the ith objective function where $f_i^u = f_i(x_i^0) = \max f_i(x_i^0)$, i = 1,2.

The minimum value of each column gives lower tolerance limit or lowest acceptable level of achievement for the ith objective function where $f_i^L = min\ f_i(x_i^0), i = 1.2$. For the maximization type objective function, the upper tolerance limit $f_t^u, t = 1,2$, are kept constant at their respective optimal values calculated in isolation but the lower tolerance limit f_i^L are changed. The idea being that $f_i(x) \to f_t^u$, then the fuzzy objective goals take the form $f_i(x) \lesssim f_i(x_i^u), i =$ 1,2. And the fuzzy goal for the control vector x_i is obtained a $x_i \cong x_i^u$. Now, in the decision situation, it is assumed that all DMs that are up to i^{th} motivation to cooperate each other to make a balance of decision powers, and they agree to give a possible relaxation of their individual optimal decision. The i^{th} level DM must adjust his/her goal by assuming the lowest acceptable level of achievement f_i^L based on indefiniteness of the decentralized organization. Thus, all values of $f_i(x) \geq f_t^u$ are absolutely acceptable (desired) to objective function $f_i(x)$ satisfactory to the ith level DM. All values of $f_i(x)$f with $f_i(x) \leq f_t^L$ are absolutely unacceptable (undesired) to the objective function $f_i(x)$ for $i = 1,2$. Based on this interval of tolerance, we can establish the following linear membership functions for the defined fuzzy goals as *Fig.1* below.

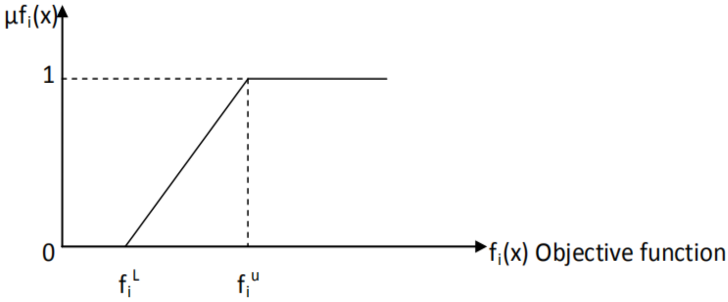

Fig. 1. Membership function of maximization-type objective function.

$$\mu_i(f_i(x)) = \begin{cases} 1, & \text{if } f_i(x) \geq f_i^u, \\ \dfrac{f_i(x) - f_i^L}{f_i^u \geq f_i^L}, & \text{if } f_i^L \leq f_i(x) \leq f_i^u, i = 1,2 \\ 0, & \text{if } f_i(x) \leq f_i^L. \end{cases} \quad (10)$$

By identifying the membership functions $\mu_1(f_1(x))$and $\mu_2(f_2(x))$for the objective functions $f_1(x)$ and $f_2(x)$, and following the principle of the fuzzy decision by Bellman and Zadeh, the original bi-level linear programming *Problem (9)* can be interpreted as the membership function maxmin problem defined by:

$$\begin{aligned} & \max\ \min\{\mu_i(f_i(x)),\ \ i = 1,2\}, \\ & \text{s.t.} \\ & A_1x_1 + A_2x_2 \leq b,\ \ x_1, x_2 \geq 0. \end{aligned} \quad (11)$$

Then the linear membership functions for decision vector x_1 can be formulated as:

$$\mu_{x1}(f_1(x)) = \begin{cases} \dfrac{x_1 - (x_1^0 - e_1^-)}{e_1^-}, & \text{if } x_1^0 - e_1^- \leq x_1 \leq x_1^0 \\ \dfrac{(x_1^0 + e_1^+) - x_1}{e_1^+}, & \text{if } x_1^0 \leq x_1 \leq (x_1^0 + e_1^+) \\ 0, & \text{if otherwise.} \end{cases} \quad (12)$$

Where x_1^0 is the optimal solution of first level DM;

e_1^- the negative tolerance value on x_1;

e_1^+ the positive tolerance value on x_1.

To derive an overall satisfactory solution to the membership function maximization *Problem (11)*, we first find the maximizing decision of the fuzzy decision proposed by Bellman and Zadeh [22]. Namely, the following problem is solved for obtaining a solution which maximizes the smaller degree of satisfaction between those of the two decision makers:

Gurmu and Fikadu|J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

$$\max\ \min\{\mu_1(f_1(x)),\ \mu_2(f_2(x)),\ \mu_{x1}(x_1)\},$$
$$\text{s.t.} \tag{13}$$
$$A_1x_1 + A_2x_2 \le b,\ \ x_1, x_2 \ge 0.$$

By introducing an auxiliary variable λ, this problem can be transformed into the following equivalent problem:

$$\max\ \lambda,$$
$$\text{s.t. } \mu_1(f_1(x)) \ge \lambda,$$
$$\mu_2(f_2(x)) \ge \lambda, \tag{14}$$
$$\mu_{x1}(x_1) \ge \lambda,$$
$$A_1x_1 + A_2x_2 \le b,\ \ x_1, x_2 \ge 0.$$

Solving *Problem (14)*, we can obtain a solution which maximizes the smaller satisfactory degree between those of both decision makers. It should be noted that if the membership function$\mu_i(f_i(x)), i = 1.2$ are linear membership functions such as *Eq. (10)*, *Problem (14)* becomes a linear programming problem. Let x^*denotes an optimal solution to *Problem (14)*. Then we define the satisfactory degree of both decision makers under the constraints as

$$\lambda^* = \min\{\mu_1(f_1(x^*)),\ \mu_2(f_2(x^*))\}. \tag{15}$$

If DM1 is satisfied with the optimal solution x^*, it follows that the optimal solution x^*becomes a satisfactory solution; however DM1 is not always satisfied with the solution x^*. It is quite natural to assume that DM1 specifies (decides) the minimal satisfactory level $\delta \in [0,1]$ for his membership function subjectively. Consequently, DM2 optimizes his objective under the new constraints as the following problem:

$$\max\ \mu_2(f_2(x)),$$
$$\text{s.t.} \tag{16}$$
$$\mu_1(f_1(x)) \le \delta$$
$$A_1x_1 + A_2x_2 \le b,\ \ x_1, x_2 \ge 0.$$

If an optimal solution to *Problem (16)* exists, it follows that DM1 obtains a satisfactory solution having a satisfactory degree larger than or equal to the minimal satisfactory level specified (decided) by DM1's own self. However, the larger the minimal satisfactory level is assessed, the smaller DM2's satisfactory degree becomes. Consequently, a relative difference between the satisfactory degrees of DM1 and DM2 becomes larger than it is feared that overall satisfactory balance between both levels cannot be maintained. To take account of overall satisfactory balance between both levels, DM1 needs to compromise (agree) with DM2 on DM1' s own minimal satisfactory level. To do so, the following ratio of the satisfactory degree of DM2 to that of DM1 is defined as:

$$\Delta = \frac{\mu_2\left(f_2(x^*)\right)}{\mu_1\left(f_1(x^*)\right)}. \tag{17}$$

This is originally introduced by Lai [6].

Let $\Delta > \Delta^L$ denote the lower bound and the upper bound of Δ specified by DM1. If $\Delta > \Delta^U, i.e\ \mu_2\left(f_2(x^*)\right) > \Delta^U \mu_1\left(f_1(x^*)\right)$, then DM1 updates (improves) the minimal satisfactory level δ by increasing δ . Then DM1 obtains a larger satisfactory degree and DM2 accepts a smaller satisfactory degree. Conversely, if $\Delta > \Delta^L, i.e\ \mu_2\left(f_2(x^*)\right) < \Delta^i \mu_1\left(f_1(x^*)\right)$, then DM1 updates the minimal satisfactory level δ by decreasing δ, and DM1 accepts a smaller satisfactory degree and DM2 obtains a larger satisfactory degree.

At an iteration l , let $\mu_1\left(f_1(x^l)\right)$, $\mu_2\left(f_2(x^l)\right), \lambda^l$ and $\Delta^l = \frac{\mu_2\left(f_2(x_1^l)\right)}{\mu_1\left(f_1(x^l)\right)}$ denote DM1's and DM2's satisfactory degrees, a satisfactory degree of both levels and the ratio of satisfactory degrees between both DMs, respectively, and let a corresponding solution be l^x at the iteration. The iterated interactive process terminates if the following two conditions are satisfied and DM1 concludes the solution as a satisfactory solution.

4.3.1| Termination conditions of the interactive processes for bi-level linear programming problems

DM1's satisfactory degree is larger than or equal to the minimal satisfactory level δ specified by DM1, i.e. $\mu_1\left(f_1(x^l)\right) \geq \delta$.

The ratio $\Delta^l$of satisfactory degrees lies in the closed interval between the lower and upper bounds specified by DM1, i.e. $\Delta^l \in [\Delta\ min, \Delta max]$.

Condition (i) is DM1's required condition for solutions, and Condition (ii) is provided in order to keep overall satisfactory balance between both levels. Unless the conditions are satisfied simultaneously, DM1 needs to update the minimal satisfactory level δ.

Procedure for updating the minimal satisfactory level δ.

If Condition (i) is not satisfied, then DM1 decreases the minimal satisfactory level by δ.

If the ratio Δ^l exceeds its upper bound, then DM1 increases the minimal satisfactory level δ. Conversely, if the ratio $\Delta^l$is below its lower bound, then DM1 decreases the minimal satisfactory level δ.

4.4| Algorithm of Interactive Fuzzy Programming for BLPPs

Step 1. Find the solution of the first level and second level independently with the same feasible set given.

Step 2. Do these solutions coincide?

- *If yes, an optimal solution is reached.*
- *If No, go to Step 3.*

Gurmu and Fikadu|J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

Step 3. Define a fuzzy goal, construct a payoff matrix, and then find upper tolerance limit f_t^u and lower tolerance limit f_t^L.

Step 4. Build member ship functions for maximization objective functions $\mu f_i(f_i(x))$ and decision vector x_1 using *Eqs. (8)* and *(10)*, respectively.

Step 5. set $\ell = 1$ and solve the auxiliary *Problems (14)*. If DM1 is satisfied with the optimal solution, the solution becomes a satisfactory solution x^*. Otherwise, ask DM1 to specify (decide) the minimal satisfactory level δ together with the lower and the upper bounds $[\Delta\ min, \Delta\ max]$ of the ratio of satisfactory degrees Δ^l with the satisfactory degree $\lambda^*$of both decision makers and the related information about the solution in mind.

Step 6. Solve *Problem (16)*, in which the satisfactory degree of DM1 is maximized under the condition that the satisfactory degree of DM1 is larger than or equal to the minimal satisfactory level δ, and then an optimal solution x^l to *Problem (16)* is proposed to DM1 together with λ^l, $\mu_1(f_1(x^l))$, $\mu_2(f_2(x^l))$ and Δ^l.

Step 7. If the solution x^l satisfies the termination conditions and DM1 accepts it, then the procedure stops, and the solution x^l is determined to be a satisfactory solution.

Step 8. Ask DM1 to revise the minimal satisfactory level δ in accordance with the procedure for updating minimal satisfactory level. Return to Step 7.

Example 2. Solve (Linear BLPP)

$$
\begin{aligned}
&\max f_1(x) = 5x_1 + 6x_2 + 4x_3 + 2x_4,\\
&x_1, x_2.\\
&\text{Where } x_3, x_4 \text{ solves}\\
&\max f_2(x) = 8x_1 + 9x_2 + 2x_3 + 4x_4,\\
&x_3, x_4,\\
&\text{s.t.}\\
&3x_1 + 2x_2 + x_3 + 3x_4 \le 40,\\
&x_1 + 2x_2 + x_3 + 2x_4 \le 30,\\
&2x_1 + 4x_2 + x_3 + 2x_4 \le 35,\\
&x_1, x_2, x_3, x_4 \ge 0.
\end{aligned}
\tag{18}
$$

Solution.

Step 1. Find the solution of the top-level and lower-level independently with the same feasible set. i.e.

$$\max f_1(x) = 5x_1 + 6x_2 + 4x_3 + 2x_4,$$

$$\begin{aligned}
&\text{s.t.}\\
&3x_1 + 2x_2 + x_3 + 3x_4 \le 40,\\
&x_1 + 2x_2 + x_3 + 2x_4 \le 30,\\
&2x_1 + 4x_2 + x_3 + 2x_4 \le 35,\\
&x_1, x_2, x_3, x_4 \ge 0.
\end{aligned} \tag{19}$$

Then we find the optimal solution

$f_1 = 125\ at\, x_1^0 = (5, 0, 25, 0);$

$f_2 = 118.125\ at\, x_2^0 = (11.25, 3.125, 0, 0);$

But this is not a satisfactory solution (since $x_1^0 \neq x_2^0$).

Step 2. Define fuzzy goals, construct the payoff matrix and we need to find the upper and lower tolerance limit.

Objective function as: $f_1 \lesssim 125, f_2 \lesssim 118.125.$

Decision variables as: $x_1 \cong 5, x_2 \cong 0;$

$$\text{Payoff matrix}=\begin{bmatrix} & f_1(x_1^0) & f_2(x_2^0) \\ x_1^0 & 125 & 90 \\ x_2^0 & 75 & 118.125 \end{bmatrix}.$$

Upper tolerance limits are $f_1^u = 125, f_2^u \lesssim 118.125.$

Lower tolerance limits are $f_1^L = 75, f_2^L \lesssim 90.$

Step 3. Build membership functions for:

Objective functions as

$$\mu f_1\big(f_1(x)\big) = \begin{cases} 1, & \text{if } f_1(x) \ge 125 \\ \dfrac{f_1(x) - 75}{125 - 75}, & \text{if } 75 \le f_1(x) \le 125\,. \\ 0, & \text{if } f_1(x) \le 75 \end{cases}$$

Decision variable function as

$$\mu f_2\big(f_2(x)\big) = \begin{cases} 1, & \text{if } f_2(x) \ge 118.125 \\ \dfrac{f_2(x) - 90}{118.125 - 90}, & \text{if } 90 \le f_2(x) \le 119.125. \\ 0, & \text{if } f_2(x) \le 90 \end{cases}$$

Let the upper level DM specifies (decides) $x_1 = 5$ with 2.5 (negative) and 2.5 (positive) tolerances and $x_2 = 0$ with 0 (negative) and 3 (positive) tolerance values.

Gurmu and Fikadu | J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

$$\mu x_1(x_1)=\begin{cases}\dfrac{x_1-(5-2.5)}{2.5}, & \text{if } 2.5\le x_1\le 5\\ \dfrac{(5+2.5)-x_1}{2.5}, & \text{if } 5\le x_1\le 7.5'\\ 0, & \text{otherwise}\end{cases}$$

$$\mu x_2(x_2)=\begin{cases}x_2, & \text{if } x_2\le 3\\ \dfrac{3-x_2}{3}, & \text{if } 0\le x_2\le 3\\ 0, & \text{otherwise}\end{cases}.$$

Step 4. Solve the auxiliary problem

$$\begin{aligned}&\max \lambda,\\ &\text{s.t.}\\ &\mu f_1\big(f_1(x)\big)\ge\lambda,\\ &\mu f_2\big(f_2(x)\big)\ge\lambda,\\ &\mu x_1(x_1)\ge\lambda,\\ &3x_1+2x_2+x_3+3x_4\le 40,\\ &x_1+2x_2+x_3+2x_4\le 30,\\ &2x_1+4x_2+x_3+2x_4\le 35,\\ &x_1,x_2,x_3,x_4\ge 0.\end{aligned} \tag{20}$$

The result of the first iteration including an optimal solution to the problem is

$$x_1^1=6.41, x_2^1=1.95, x_3^1=10.52, x_4^1=1.42, \text{ and } \lambda^1=0.316, f_1^1(x)=88.67, f_2^1(x)=95.55, \mu_1\big(f_1(x)\big)=0.2734.$$

Suppose that DM1 is not satisfied with the solution obtained in iteration 1, and then let him specify (decide) the minimal satisfactory level at $\delta=0.3$ and the bounds of the ratio at the interval $[\Delta min, \Delta\, max]=[0.3, 0.4]$, taking account of the result of the first iteration. Then, the problem with the minimal satisfactory level is written as:

$$\begin{aligned}&\max \mu f_2(f_2(x)),\\ &\text{s.t.}\\ &\mu f_1(f_1(x))\ge 0.3,\\ &x\in S.\end{aligned} \tag{21}$$

Applying simplex algorithm, the result of the second iteration including an optimal solution to *Problem (21)* is

$$x_1^2 = 6.71, x_2^2 = 2.05, x_3^2 = 10.52, x_4^2 = 1.42,$$

and

$$\lambda^2 = 0.316,$$

$$f_1^2(x) = 90.77, f_2^2(x) = 98.85, \mu_{f1}\big(f_1(x)\big) = 0.3154, \tag{22}$$

and

$$\Delta^2 = 0.3165.$$

Therefore, this solution satisfies the termination conditions.

5 | Conclusion

The fuzzy mathematically programming approach is simple to implement, interactive and applicable to BLPP. The satisfactory solution obtained is realistic. We can take any membership function other than linear. The results will hold good, however, the problem will become a non linear programming problem. We observe that even though the decision making process is from higher to lower level, the lower level becomes the most important. This is because the decision vector under the control of the lower level DM is not given any tolerance limits. Hence this decision vector either remains unchanged or closest to its valued obtained in isolation. But at higher level, the decision vectors are given some tolerance and hence they are free to move within the tolerance limits. The tolerance levels can also be considered as variables and if the DMs cooperate then the entire system as a whole can be optimized. We can easily apply the same approach to non linear BLPPs.

References

[1] Shih, H. S., Lai, Y. J., & Lee, E. S. (1996). Fuzzy approach for multi-level programming problems. *Computers & operations research*, *23*(1), 73-91.

[2] Bialas, W. F., & Karwan, M. H. (1984). Two-level linear programming. *Management science*, *30*(8), 1004-1020.

[3] Wen, U. P., & Hsu, S. T. (1991). Linear bi-level programming problems—a review. *Journal of the operational research society*, *42*(2), 125-133.

[4] Shih, H. S., Lai, Y. J., & Lee, E. S. (1996). Fuzzy approach for multi-level programming problems. *Computers and operations research*, *23*(1), 73-91.

[5] Sakawa, M., & Nishizaki, I. (2009). *Cooperative and noncooperative multi-level programming* (Vol. 48). Springer Science & Business Media.

[6] Lai, Y. J. (1996). Hierarchical optimization: a satisfactory solution. *Fuzzy sets and systems*, *77*(3), 321-335.

[7] Sinha, S. (2003). Fuzzy programming approach to multi-level programming problems. *Fuzzy sets and systems*, *136*(2), 189-202.

[8] Zimmermann, H. J. (1978). Fuzzy programming and linear programming with several objective functions. *Fuzzy sets and systems*, *1*(1), 45-55.

[9] Shamooshaki, M. M., Hosseinzadeh, A., & Edalatpanah, S. A. (2015). A new method for solving fully fuzzy linear programming problems by using the lexicography method. *Applied and computational mathematics*, *1*, 53-55.

[10] Pérez-Cañedo, B., Concepción-Morales, E. R., & Edalatpanah, S. A. (2020). A revised version of a lexicographical-based method for solving fully fuzzy linear programming problems with inequality constraints. *Fuzzy information and engineering*, 1-20.

[11] Hosseinzadeh, A., & Edalatpanah, S. A. (2016). A new approach for solving fully fuzzy linear programming by using the lexicography method. *Advances in fuzzy systems*. https://doi.org/10.1155/2016/1538496

Gurmu and Fikadu|J. Fuzzy. Ext. Appl. 1(4) (2020) 252-271

[12] Das, S. K., Edalatpanah, S. A., & Mandal, T. (2018). A proposed model for solving fuzzy linear fractional programming problem: numerical point of view. *Journal of computational science*, *25*, 367-375.

[13] Elsisy, M. A., & El Sayed, M. A. (2019). Fuzzy rough bi-level multi-objective nonlinear programming problems. *Alexandria engineering journal*, *58*(4), 1471-1482.

[14] Edalatpanah, S. A., & Shahabi, S. (2012). A new two-phase method for the fuzzy primal simplex algorithm. *International review of pure and applied mathematics*, *8*(2), 157-164.

[15] Najafi, H. S., Edalatpanah, S. A., & Dutta, H. (2016). A nonlinear model for fully fuzzy linear programming with fully unrestricted variables and parameters. *Alexandria engineering journal*, *55*(3), 2589-2595.

[16] Kamal, M., Gupta, S., Chatterjee, P., Pamucar, D., & Stevic, Z. (2019). Bi-Level multi-objective production planning problem with multi-choice parameters: a fuzzy goal programming algorithm. *Algorithms*, *12*(7), 143.

[17] Maiti, S. K., & Roy, S. K. (2020). Analysing interval and multi-choice bi-level programming for Stackelberg game using intuitionistic fuzzy programming. *International journal of mathematics in operational research*, *16*(3), 354-375.

[18] Yue, Q., Wang, Y., Liu, L., Niu, J., Guo, P., & Li, P. (2020). Type-2 fuzzy mixed-integer bi-level programming approach for multi-source multi-user water allocation under future climate change. *Journal of hydrology*, *591*, 125332. https://doi.org/10.1016/j.jhydrol.2020.125332

[19] Liu, J., Xue, W., Pang, G., & Guo, Y. (2019, December). Application of gray discrete bi-level linear programming model in the double emergency management network. *2019 12th international symposium on computational intelligence and design (ISCID)* (Vol. 1, pp. 232-235). *IEEE.*

[20] Rizk-Allah, R. M., & Abo-Sinna, M. A. (2020). A comparative study of two optimization approaches for solving bi-level multi-objective linear fractional programming problem. *OPSEARCH*, 1-29.

[21] Han, J., Liu, Y., Luo, L., & Mao, M. (2020). Integrated production planning and scheduling under uncertainty: A fuzzy bi-level decision-making approach. *Knowledge-based systems*, 106056. https://doi.org/10.1016/j.knosys.2020.106056

[22] Bellman, R. E., & Zadeh, L. A. (1970). Decision-making in a fuzzy environment. *Management science*, *17*(4), B-141.