PERGAMON

Mathematical and Computer Modelling 36 (2002) 569–585

MATHEMATICAL AND COMPUTER MODELLING

www.elsevier.com/locate/mcm

An Interactive Approach for Integrated Multilevel Systems in a Fuzzy Environment

HSU-SHIH SHIH
Graduate Institute of Management Science
Tamkang University
Tamsui, Taipei, Taiwan 251, R.O.C.

(Received November 2001; accepted December 2001)

Abstract—This study proposes a total solution of an interactive approach for integrated multilevel systems or multilevel programming problems (MLPPs) in a fuzzy environment. Simulating the actual decision-making process of the hierarchical structure of an organization, MLPP is a practical and useful approach to decentralized planning problems. Because of the complexity of the problems, there are no traditional techniques efficient enough to obtain the numerical solution of a reasonable size problem. Hence, Shih *et al.* [1] propose a fuzzy approach for MLPPs, to simplify the complex structure, which is proven to be feasible and efficient. The imprecise MLPPs will be involved when the coefficients of MLPPs cannot be estimated exactly. Because of such a complicated situation in the real world, we will take advantage of an interactive technique to improve the flexibility and robustness of its decision through progressive articulation of decision information from decision makers (DMs). Roughly speaking, there are two interactive procedures for imprecise MLPPs: inside loop and outside loop. The former is for the preference of the DMs, represented by fuzzy membership functions; the latter for the imprecision of coefficients, described by possibility distributions or cut-off values. Special considerations will be given to the compensatory operator, positive and negative ideal solutions, risk attitude, and ε-constraints. In the final section, linear-programming type and network-flow type of imprecise MLPPs will be solved separately as an integrated multilevel system.

Keywords—Interactive procedure, Multilevel programming problem, Possibilistic measures, Cut-off value, Risk attitude, ε-constraint, Membership function, Compensation.

1. INTRODUCTION

The interactive technique seems to be a step in the right direction to manage complex systems with dynamic consequences, in which decision information can be obtained via its process to relieve decision burden, thus, ensures a rational decision. The technique provides a learning process about the system, whereby the decision makers (DMs) can learn to recognize good solutions, the relative importance of factors, and finally design a high-productivity system [2]. According to Aksoy [3], five major advantages of the interactive technique can be summarized as follows:

The author would like to thank National Science Council, Taiwan, R.O.C., for the financial support under Grant Number: NSC 89-2213-E-214-017. Special thanks are given to Professor E. Stanley Lee at Kansas State University, Kansas, U.S.A. and Professor U. P. Wen at National Tsing Hua University, Hsinchu, Taiwan, for their valuable help.

0895-7177/02/$ - see front matter Typeset by $\mathcal{AMS}$-TEX
PII: S0895-7177(02)00183-8

(a) interactive technique does not require preference information, which is rather difficult for the DMs to provide;
(b) the DMs have greater confidence in the solution obtained;
(c) the algorithm allows an effective division of labor between the DMs and the analysts/machines;
(d) the DMs can clearly learn about their preferences; and
(e) the interactive approach would extenuate problems associated with mismatches between the DMs' perception and the formalization of the problem through a computerized algorithm.

Many techniques have been developed for multiobjective decision making (MODM), fuzzy MODM, and possibilistic MODM to overcome the conflict of multiple noncommensurable objectives [4,5]. Thus, it is natural to consider these techniques for improving decision quality of a more complex multilevel system or multilevel programming problem (MLPP).

Multilevel techniques are developed to solve decentralized planning problems, with multiple DMs in a hierarchical organization, where each unit or department independently seeks its own interests, but is affected by the actions of other units through externalities. The MLPP can be encountered in almost any hierarchical organization such as government agencies, profit or nonprofit organizations, manufacturing plants, and logistic companies. The simplest MLPP formulation, a bilevel case, is illustrated as follows [6]:

$$\max_{\mathbf{x}1} \quad f_1(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{c}_{11}^T \mathbf{x}_1 + \mathbf{c}_{12}^T \mathbf{x}_2 \qquad \text{(upper level)} \tag{1}$$

where $\mathbf{x}_2$ solves,

$$\begin{aligned} \max_{\mathbf{x}2} \quad & f_2(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{c}_{21}^T \mathbf{x}_1 + \mathbf{c}_{22}^T \mathbf{x}_2 \qquad \text{(lower level)} \\ \text{s.t.} \quad & (\mathbf{x}_1, \mathbf{x}_2) \in \mathbf{X} = \{(\mathbf{x}_1, \mathbf{x}_2) \mid \mathbf{A}_1 \mathbf{x}_1 + \mathbf{A}_2 \mathbf{x}_2 \le \mathbf{b} \text{ and } \mathbf{x}_1, \mathbf{x}_2 \ge \mathbf{0}\}, \end{aligned}$$

where $\mathbf{c}_{11}$, $\mathbf{c}_{12}$, $\mathbf{c}_{21}$, $\mathbf{c}_{22}$, and $\mathbf{b}$ are linear vectors, $\mathbf{A}_1$ and $\mathbf{A}_2$ are linear matrices, with $\mathbf{X}$ denotes its constraint set.

Traditional approaches, including vertex enumeration and transformation approaches, cannot provide an efficient algorithm to solving reasonably practical size problems. The simplest bilevel problem with linear form is NP-hard and nonconvex [7]; and more seriously, its solution may not exist under the best of circumstances [8]. Therefore, Shih *et al.* [1] introduce a fuzzy approach for MLPP to simplify the complex nested structure by utilizing the concept of MODM and the degree of satisfaction, in terms of fuzzy membership functions. And these in terms transfer the goals and decisions to a lower level in a top-down fashion for simulating the hierarchical decision making process. The so-called supervised search approach has been proven to be feasible and efficient.

The recommended process generates an auxiliary lower-level problem of expression (1) as follows.

$$\begin{aligned} \text{Max} \quad & \lambda, \\ \text{s.t.} \quad & \mathbf{x} \in \mathbf{X}, \\ & \mu_{f1}(f_1(\mathbf{x})) = \frac{[f_1(\mathbf{x}) - f_1^-]}{[f_1^+ - f_1^-]} \ge \lambda, \\ & \mu_{x1}(\mathbf{x}_1) = \frac{[\mathbf{x}_1 - (\mathbf{x}_1^U - p_1)]}{p_1} \ge \lambda \mathbf{I}, \\ & \mu_{x1}(\mathbf{x}_1) = \frac{[(\mathbf{x}_1^U + p_2) - \mathbf{x}_1]}{p_2} \ge \lambda \mathbf{I}, \\ & \mu_{f1}(f_2(\mathbf{x})) = \frac{[f_2(\mathbf{x}) - f_2^-]}{[f_2^+ - f_2^-]} \ge \lambda, \\ & \lambda \in [0,1] \text{ for degree of satisfaction,} \end{aligned} \tag{2}$$

where p_1 and p_2 are the two-side tolerance for decision vector $\mathbf{x}_1$ on LHS and RHS, respectively. And $\mathbf{x}_1^U$ is the most preferred decision of $\mathbf{x}_1$. f_1^+ and f_1^- represent the positive ideal solution (PIS) and negative ideal solution (NIS), respectively.

Furthermore, through utilizing preference and possibilistic concepts with modification [9], the process has been extended to solve an imprecise minimum-cost flow (MCF) problem, a general form of network flow problems. In addition, the compensatory behavior in the MLPPs is also investigated for managerial decision making [10]. In fact, the new development has been involved in two categories of fuzzy set theory, fuzzy programming and possibilistic programming. And the techniques in both can be beneficial to solving MLPPs. Note that the interactive approach for MLPPs actually involves multiple DMs, other than traditional approaches (e.g., [11,12]), and searches of a compromise in a continuous space.

Although the nested structure of the MLPP, expression (1), has been simplified through fuzzy concept, there are still many realistic decision situations under investigation. Inspired by the achievements of interactive techniques, we will introduce the techniques with dialog to the DMs to get their preference information. Each of the characteristics will be tackled with one concept, respectively, in the fuzzy environment, and the realistic problems can be controlled in a unified process. Furthermore, a series of the dialog between DMs and analysts/machines is continued for accounting for learning, adaptive and dynamic natures of the problems. Accordingly, a preferred or compromised solution can be obtained in the end.

In the following sections, we first consider explicit trade-off information for the interactive procedure. The decision information focusing on preference and vagueness is explored by utilizing the techniques in fuzzy programming and possibilistic programming. Arranging the information in a logical order, an interactive procedure will be proposed. There are mainly two loops for interaction: the inside one for preference and the outside one for vagueness of the problems. Moreover, two types of the imprecise MLPPs are solved separately in illustrated examples. In the final section, we will make some concluding remarks and future directions.

2. DECISION INFORMATION FOR INTERACTION

Interactive procedures have met with a great success over past three decades. Although each interactive technique has specific characteristics in MODM, two main conceptions, a search-oriented conception and a learn-oriented conception, should be distinguished to classify interactive procedures. After investigating ten interactive procedures, Vanderpooten and Vincke [13] infer that the most recent approaches aim at including both aspects. Hwang and Masud [4] point out that some interactive techniques require explicit information regarding the trade-off between the attainment levels of objectives at each stage; others require the implicit trade-off information in the form allowing the DMs to indicate acceptability of the current achievement level. In general, the interactive procedure looks for the trade-off among objectives or searching for a satisfactory nondominated point due to conflicting multiple objectives. Nevertheless, the process will be complicated when fuzzy information is involved (see [5,14]). But the interactive techniques in fuzzy programming and possibilistic programming indeed offer a clue to resolve decentralized planning problems.

Despite many developments of interactive techniques for MODM in the past, only a few devolpments for MLPPs were made until now. Based on the concepts of satisfactoriness of both levels and direction vector for improvement, Shi and Xia [15] suggest an interactive approach for nonlinear bilevel multiobjective decision making problem. But the direction vector is unnecessary for a linear case due to the complexity. At the same time, Sakawa *et al.* [16,17] develop an interactive procedure for MLPPs, but the procedure diverts the difficulties of controlling decision variables and what is more, the procedure is similar to the interactive approach in MODM. Thus, a simple and explicit trade-off within the feasible region would be expected for solving MLPPs in a dynamic environment.

In the following discussions, various aspects of the decision information for interaction include: compromise solutions through PISs/NISs [4], confidence of imprecision by possibilistic theory [18], compensatory operator [19], risk attitude [20], and ε-constraints constraint [21]. The information will be discussed as the necessary elements for interaction.

2.1. Compromise Solution

As multiple conflicting and noncommensurable objectives always exist, the solving procedure is to look at a compromised or satisfactory solution among the nondominated frontier [4]. To reach a compromise, the preference will refer the following:

(a) reference points, i.e., the concept of an ideal or worse system;
(b) distance, i.e., location of alternatives away from reference points; and
(c) normalization, i.e., the process to eliminate nonmeasurability among objectives [22].

These references are common for managing MODM [4].

The above essences are also the core of fuzzy MODM. The reference points PISs/NISs and the normalization play the key roles of establishing fuzzy membership functions in terms of degree of satisfaction in DMs' mind. Then the value of 0 increasing to 1 describes the full dissatisfaction gradually transiting to the full satisfaction, which described by a linear membership function, as in expression (2) [1]. In addition, the reference points have also been the anchors for setting the upper and lower bounds in the interactive steps.

2.2. Confidence of Imprecision

Analogous probability for uncertain information, possibly is one major concept in fuzzy sets, and it originally tries to describe fuzziness or imprecision of natural languages [23]. For comparison of two fuzzy intervals P and Q, Dubois and Prade [18] propose four fundamental indices, $\mathrm{Pos}(\bar{b}_i \geq \underline{R}_i)$, $\mathrm{Pos}(\bar{b}_i > \bar{R}_i)$, $\mathrm{Nec}(\underline{b}_i \geq \underline{R}_i)$, and $\mathrm{Nec}(\underline{b}_i > \bar{R}_i)$, where b_i and R_i are variables whose domains are constrained by μ_P and μ_Q, respectively. The relation among these four: $\mathrm{Pos}(\bar{b}_i \geq \underline{R}_i) \geq \{\max \mathrm{Pos}(\bar{b}_i > \bar{R}_i),\ \mathrm{Nec}(\underline{b}_i \geq \underline{R}_i)\} \geq \{\min\{\mathrm{Pos}(\bar{b}_i > \bar{R}_i),\ \mathrm{Nec}(\underline{b}_i \geq \underline{R}_i)\} \geq \mathrm{Nec}(\underline{b}_i > \bar{R}_i)$, describes the imprecise ranges in terms of confidence of imprecision in DMs' mind. The comparison of fuzzy numbers under these four indices is depicted in Figure 1.

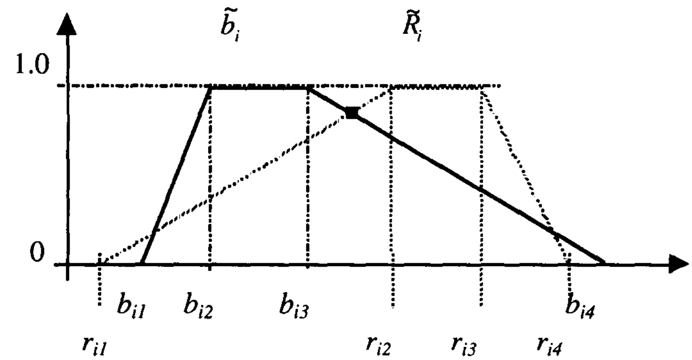
For mathematical programming, Buckley [24] proposes an evaluation, through the operation expressed as triangular fuzzy membership functions, which seek the minimum of all possibilities in its objective and decision spaces. Negi [25] further extends the problem to k-dimension objective space with minimization and maximization based on (strict) exceedance possibility whose distributions are trapezoidal shapes, e.g., $(b_{i1}, b_{i2}, b_{i3}, b_{i4})$ with four numbers. For the i^{th} constraint, with trapezoidal fuzzy numbers (TFNs), $\sum_i \sum_j (a_{ij1}, a_{ij2}, a_{ij3}, a_{ij4}) x_j \leq (b_{i1}, b_{i2}, b_{i3}, b_{i4})$, the measure of exceedance possibility is to look for maximizing δ_i within the range of $b_{i3} \leq \sum_j a_{ij2} x_j$ and $b_{i4} \geq \sum_j a_{ij1} x_j$. Accordingly, for the k^{th} objective with a trapezoidal fuzzy cost, $z_k = \sum_j (c_{kj1}, c_{kj2}, c_{kj3}, c_{kj4}) x_j$, the problem searches for minimizing the LHS of TFNs $\theta_{k1} = (z_k - \sum_j c_{kj1} x_j)/(\sum_j c_{kj2} x_j - \sum_j c_{kj1} x_j)$ or maximizing the RHS of TFNs $\theta_{k2} = (\sum_j c_{kj4} x_j - z_k)/(\sum_j c_{kj4} x_j - \sum_j c_{kj3} x_j)$. The measure of strict exceedance possibility hunts for maximizing δ_i within the range of $b_{i3} \leq \sum_j a_{ij4} x_j$ and $b_{i4} \geq \sum_j a_{ij3} x_j$, and the objective is the same as before. Furthermore, the above four indices can be applied to mathematical programming with TFNs, and the choice will depend on the confidence of imprecision in DMs' minds. Besides, the cut-off values will be another factor affecting the imprecision range. Hence, Negi [25] tries to linearize the process of possibility comparison by a fixed value between 0 and 1, i.e., cut-off value α.

Similarly, the calculation can be expanded for another two possibilistic indices. For the i^{th} constraint with TFNs, $\sum_i \sum_j (a_{ij1}, a_{ij2}, a_{ij3}, a_{ij4}) x_j \leq (b_{i1}, b_{i2}, b_{i3}, b_{i4})$, the measure of exceedance necessity to look for maximizing δ_i within the range of $b_{i1} \leq \sum_j a_{ij2} x_j$ and $b_{i2} \geq \sum_j a_{ij1} x_j$. And the measure of strict exceedance possibility is in search for maximizing δ_i within the range of $b_{i1} \leq \sum_j a_{ij4}x_j$ and $b_{i2} \geq \sum_j a_{ij3}x_j$. The objective is taken from TFNs which are defined the same as before.

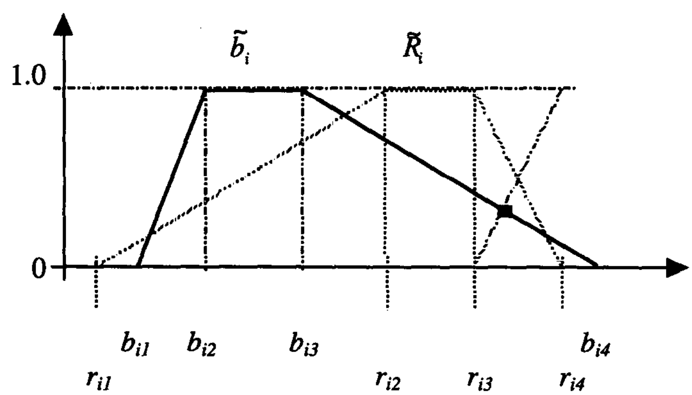
However, there are two problems in applying this formulation for network flow problems. The first one, the number of constraints will be increased a lot, and the second one is that there exists a contradiction in defining the fuzzy capacity constraints. To avoid these difficulties, Shih and Lee [9] adopt the concept of fuzzy range instead of the possibilistic range in formulating the arc capacity, where the value of the fuzzy range is controlled by the same cut-off values. Therefore, two types of problems, linear-programming type and network-flow type, can be considered at the same parameter for manipulating the imprecise of confidence. Then four possibilistic indices and cut-off values will be the information for interaction.

2.3. Compensatory Operator

In fuzzy mathematical programming, Zimmermann [26] has followed the decision as the intersection of goals and constraints, and the best decision will be the union of all decisions, i.e., $\mu_{D^*} = \max\{\min(\mu_G, \mu_C)\}$. This operation is accepted as a general tool for manipulating the linear programming and MODM problems. However, the interpretation of such a decision as an intersection or union will result in no compensation (under-achievement) or full compensation (over-achievement). Managerial decisions always have some kind of compensation between either

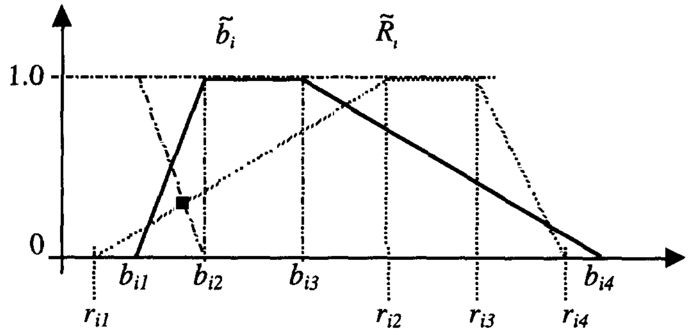


(a) Exceedance possibility, Pos[$\bar{b}_i \geq \underline{R}_i$] > 0.

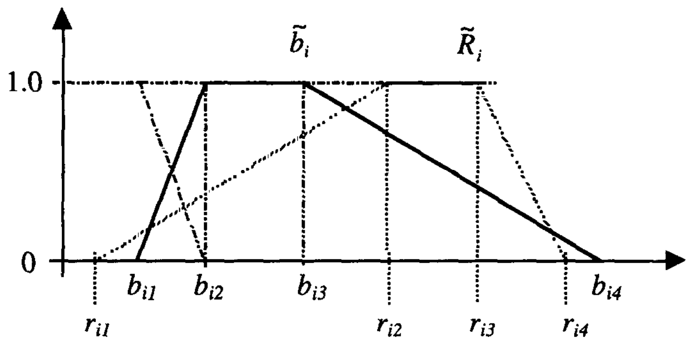


(b) Strict exceedance possibility, Pos[$\bar{b}_i > \bar{R}_i$] > 0.

Figure 1. Fuzzy number comparison through four possibility indices.
(1) Pos[$\bar{b}_i \geq \underline{R}_i$] $= \delta_i = (b_{i4} - r_{i1})/[(b_{i4} - b_{i3}) + (r_{i2} - r_{i1})]$ in (a), where $b_{i3} \leq r_{i2}$ and $r_{i1} \leq b_{i4}$.
(2) Pos[$\bar{b}_i > \bar{R}_i$] $= \delta_i = (b_{i4} - r_{i3})/[(b_{i4} - b_{i3}) + (r_{i4} - r_{i3})]$ in (b), where $b_{i3} \leq r_{i4}$ and $r_{i3} \leq b_{i4}$.
(3) Nec[$\underline{b}_i \geq \underline{R}_i$] $= \delta_i = (b_{i2} - r_{i1})/[(r_{i2} - r_{i1}) + (b_{i2} - b_{i1})]$ in (c), where $b_{i1} \leq r_{i2}$ and $r_{i1} \leq b_{i2}$.
(4) Nec[$\underline{b}_i \geq \bar{R}_i$] $= \delta_i = (b_{i2} - r_{i3})/[(r_{i4} - r_{i3}) + (b_{i2} - b_{i1})]$ in (d), where $b_{i1} \leq r_{i4}$ and $r_{i3} \leq b_{i2}$.



(c) Exceedance necessity, Nec[$\underline{b}_i \geq \underline{R}_i$] > 0.



(d) Strict exceedance necessity, Nec[$\underline{b}_i > \bar{R}_i$] = 0 (no intersection here).

Figure 1. (cont.)

different degrees of goal achievement or decision restrictions [27]. It is calculated through the convex combination of goal achievement or decision restrictions. Afterwards, Werners [19] proposes two operators which both lead to formulations in linear form with respect to the empirical data. However, "fuzzy and" $\mu_{\text{and}}(= \gamma\lambda + (1-\gamma)\sum_i \lambda_i/m,\ i = 1,\dots,m)$ will be easier to handle and is applicable to various types of hierarchical structure of MLPPs [10]. And the grade of compensation is another information for interaction.

2.4. Risk Attitude

Another characteristics are the risk attitude of DMs, which can be divided into three categories: risk-averse, risk-seeking, and risk-neutral [28]. All these are to describe more or less of the uncertainty with respect to the expected value of the payoff. Adopted from utility concept or preference functions, these three classes of behavior can be simply elicited by the fuzzy membership function, four constrains in expression (2); with the power of 2, 1/2, and 1, respectively, [20]. In fact, the risk-neutral behavior is the same formula as in expression (2), which is a linear membership function. And risk-averse characteristic is depicted with a convex shape, and risk-seeking characteristic is depicted with concave shape. The former is much optimistic than the latter in decision making. Then the three characteristics will be integrated into the procedure for choice through different membership functions.

2.5. ε-Constraints

ε-constraint method originally allows the analysts the ability to specify bounds on the k objectives in a sequential manner. Specification by the DMs of desired minimum or maximum levels of the $k-1$ objectives appearing in the constraint set essentially results in a preferred solution [21,29]. Similarly, the concept of ε-constraint can be extended to specify bounds on the decision variables for most desired attainments. Thus, ε-constraint is applicable to MLPPs through interactions with DMs, with the reference values given (e.g., PISs/NISs).

If DMs are dissatisfied with the current solution, the process will add an extra ε-constraint related to objectives or decisions, and solve the new problem. The results from different ε-constraints will be trade-off with each other, and submit to DMs to ensure an acceptable result. Since the objectives and decisions are explicitly considered at the same time, the drawback of previous approaches in not controlling both will be eliminated. Moreover, a promising result under a specific ε-constraint can give us the idea about further improvement. Although there will be many cases directly enumerated and compared in the procedure, it is still better than some sophisticated trade-off techniques on objectives only, and fits into the characteristics of MLPPs.

3. FUZZY APPROACH FOR IMPRECISE MLPPS

After the acquisition of uncertain decision information is made, a new approach will be proposed for the imprecise MLPPs. In fact, the concepts of fuzzy programming and possibilistic programming have been provided to manage different characteristics of multilevel systems. In the following contents, two types of MLPPs are reviewed as an integrated MLPP system.

3.1. MLPPs with Linear-Programming Type

Compared to the crisp case, an imprecise bilevel programming problem with linear-programming type can be illustrated as follows:

$$\max_{\mathbf{x}1} \quad \tilde{f}_1(\mathbf{x}_1,\mathbf{x}_2) = \tilde{c}_{11}^T\mathbf{x}_1 + \tilde{c}_{12}^T\mathbf{x}_2 \qquad \text{(upper level)} \tag{3}$$

where $\mathbf{x}_2$ solves,

$$\begin{aligned} \max_{\mathbf{x}2} \quad & \tilde{f}_2(\mathbf{x}_1,\mathbf{x}_2) = \tilde{c}_{21}^T\mathbf{x}_1 + \tilde{c}_{22}^T\mathbf{x}_2 \qquad \text{(lower level)}, \\ \text{s.t.} \quad & (\mathbf{x}_1,\mathbf{x}_2) \in \mathbf{Y} = \left\{(\mathbf{x}_1,\mathbf{x}_2) \mid \tilde{A}_1\mathbf{x}_1 + \tilde{A}_2\mathbf{x}_2 \le \tilde{b} \text{ and } \mathbf{x}_1,\mathbf{x}_2 \ge \mathbf{0}\right\}, \end{aligned}$$

where $\tilde{c}_{11}, \tilde{c}_{12}, \tilde{c}_{21}, \tilde{c}_{22}$ are the imprecise costs, and $\tilde{b}$ is the imprecise resource. And $\tilde{A}_1$ and $\tilde{A}_2$ are the imprecise technological parameters.

As designed, the upper-level DM defines his/her objective and decisions with possible tolerances and other interactive parameters. This information then restricts the lower-level feasible space, and a new auxiliary problem is listed as follows:

$$\begin{aligned} \text{Max} \quad & \mu_{\text{and}} = \gamma\lambda + \frac{(1-\gamma)(\lambda_1+\lambda_2+\cdots+\lambda_{s+2})}{(s+2)}, \\ \text{s.t.} \quad & \mu_{f1}(f_1(\mathbf{x})) = \left\{\frac{[f_1(\mathbf{x}) - f_1^-]}{[f_1^+ - f_1^-]}\right\}^{ra} \ge (\lambda+\lambda_1), \\ & \mu_{x1}(\mathbf{x}_1) = \left\{\frac{[\mathbf{x}_1 - (\mathbf{x}_1^U - \mathbf{p}_1)]}{\mathbf{p}_1}\right\}^{ra} \ge (\lambda+\lambda_s)\mathbf{I}, \\ & \mu_{x1}(\mathbf{x}_1) = \left\{\frac{[(\mathbf{x}_1^U + \mathbf{p}_2) - \mathbf{x}_1]}{\mathbf{p}_2}\right\}^{ra} \ge (\lambda+\lambda_s)\mathbf{I}, \\ & \mu_{f2}(f_2(\mathbf{x})) = \left\{\frac{[f_2(\mathbf{x}) - f_2^-]}{[f_2^+ - f_2^-]}\right\}^{ra} \ge (\lambda+\lambda_2), \\ & \lambda+\lambda_i \le 1, i = 1,2,\ldots,s+2 \\ & \delta_l \ge \alpha, l = 1,2,\ldots,n \text{ (the number of constraints in } \mathbf{Y}) \\ & \text{with the extra space for } \delta_l, \theta_{k2} \ge \alpha, k = 1 \text{ and } 2 \text{ (the number of objectives)} \\ & \text{with the extra space for } \theta_{k2}, \\ & f_1(\mathbf{x}) \ge \varepsilon_1, \mathbf{x}_1 \ge \varepsilon_s\mathbf{I}, \text{ or } f_2(\mathbf{x}) \ge \varepsilon_2, (\varepsilon\text{-constraints if necessary}) \\ & \alpha,\gamma,\lambda,\lambda_1,\lambda_2,\ldots,\lambda_{s+2},\delta_l,\theta_{k2} \in [0,1], \end{aligned} \tag{4}$$

where $\mathbf{p}_1$ and $\mathbf{p}_2$ are the two-side tolerances for decision vector $\mathbf{x}_1$ (including s elements) on LHS and RHS, respectively. The parameter ra describes the risk attitude of DMs, where $ra \in \{1/2, 1, 2\}$. In addition, ε_1, ε_2, and ε_s will be the extra attainments of f_1, f_2, and $\mathbf{x}_1$, respectively, for parametric variation. Furthermore, the imprecise range will be generated through possibilistic indices and represented by δ_l, $l = 1, 2, \dots, n$. And the maximization problem takes θ_{k2}, $k = 1$ and 2, where $\theta_{12} = [f_1 - \sum_j c_{j3}x_j]/[\sum_j c_{j4}x_j - \sum_j c_{j3}x_j]$ and $\theta_{22} = [f_2 - \sum_j c_{j3}x_j]/[\sum_j c_{j4}x_j - \sum_j c_{j3}x_j]$.

A compromise solution is reached when the upper and lower level DMs are satisfied with the above solution. Otherwise, the process needs to elicit new membership functions for the auxiliary level problem until a compromise to be understood. Combined with a set of control decisions and goals with tolerances, a new formulation is established. This expression can be easily solved by any mathematical programming code, e.g., [30].

3.2. MLPPs with Network-Flow Type

Network flow is a special class of linear programming with integer flows. A general form of the network flow problem is minimum-cost flow (MCF) problem, which deals with broader types of problems, such as transportation, maximum flow, assignment, shortest path, and transshipment problems [31]. It is used to determine a least shipment cost of a commodity through a capacitated network in order to satisfy demands at certain nodes from available supplies at other nodes. Since cost and capacity parameters at each arc cannot be exactly estimated, an imprecise multilevel MCF problem, in comparison with a traditional MCF problem [32], can be formulated as

$$\min_{\mathbf{x}_{ij1}} \quad \tilde{f}^1\left(\mathbf{x}_{ij}^1, \mathbf{x}_{ij}^2\right) = \sum_{(i,j)\in\mathbf{A}} \tilde{c}_{ij}^{11}\mathbf{x}_{ij}^1 + \tilde{c}_{ij}^{12}\mathbf{x}_{ij}^2 \qquad \text{(upper level)} \tag{5}$$

where $\mathbf{x}_2$ solves,

$$\begin{aligned}
\min_{\mathbf{x}_{ij2}} \quad & \tilde{f}^2\left(\mathbf{x}_{ij}^1, \mathbf{x}_{ij}^2\right) = \sum_{(i,j)\in\mathbf{A}} \tilde{c}_{ij}^{21}\mathbf{x}_{ij}^1 + \tilde{c}_{ij}^{22}\mathbf{x}_{ij}^2 \qquad \text{(lower level)} \\
\text{s.t.} \quad & \sum_{\{j:(i,j)\in\mathbf{A}\}} x_{ij} - \sum_{\{j:(j,i)\in\mathbf{A}\}} x_{ji} = b(i), \ \forall\, i \in \mathbf{N}, \\
& \tilde{l}_{ij} \le x_{ij} \le \tilde{u}_{ij}, \ \forall\,(i,j) \in \mathbf{A}, x_{ij} \ge 0, \ \text{and integer}, \forall\,(i,j) \in \mathbf{A},
\end{aligned}$$

where $\tilde{f}^1$ and $\tilde{f}^2$ are two imprecise objective functions, $\tilde{c}_{ij}^{11}$, $\tilde{c}_{ij}^{12}$, $\tilde{c}_{ij}^{21}$, and $\tilde{c}_{ij}^{22}$ are imprecise costs, and $\tilde{l}_{ij}$ and $\tilde{u}_{ij}$ symbolize the imprecise capacity with lower bound and upper bound of each arc. In addition, $\mathbf{x}_{ij} = (\mathbf{x}_{ij}^1, \mathbf{x}_{ij}^2) = \sum\sum x_{ij}, \{i,j\} \in \mathbf{A}$. Four numbers that represent the four corners of the trapezoidal shape describe these fuzzy parameters.

Possibilistic linear programming with TFNs seems well fitted to any fuzzy programming problem. However, it will have two shortcomings in dealing with network flows. Thus, we redefine the imprecise arc capacity as: $\alpha(l_{ij2} - l_{ij1}) + l_{ij1} \le x_{ij} \le u_{ij4} - \alpha(u_{ij2} - u_{ij1})$ $\forall\, i$ and j with cut-off value α, where $(l_{ij1}, l_{ij2}, u_{ij1}, u_{ij2})$ are the four corners of capacity restriction [9]. The compensatory auxiliary problem of expression (5) is described as the following form.

$$\text{Max} \quad \mu_{\text{and}} = \gamma\lambda + \frac{(1-\gamma)(\lambda_1 + \lambda_2 + \cdots + \lambda_{s+2})}{(s+2)}, \tag{6}$$

s.t.

$$\mu_{f1}\left(f^1(\mathbf{x})\right) = \left\{\frac{\left[f^{1+} - f^1(x_{ij})\right]}{\left[f^{1+} - f^{1-}\right]}\right\}^{ra} \ge (\lambda + \lambda_1),$$

$$\mu_{x1}(\mathbf{x}_1) = \left\{\frac{\left[\mathbf{x}_{ij}^1 - \left(\mathbf{x}_{ij}^{1U} - \mathbf{p}_{ij}^1\right)\right]}{\mathbf{p}_{ij}^1}\right\}^{ra} \ge (\lambda + \lambda_s)\mathbf{I}, \qquad \forall\, i \text{ and } j$$

$$\mu_{x1}(\mathbf{x}_1) = \left\{ \frac{\left[(\mathbf{x}_{ij}^{1U} + \mathbf{p}_{ij}^2) - \mathbf{x}_{ij}^1\right]}{\mathbf{p}_{ij}^2} \right\}^{ra} \geq (\lambda + \lambda_s)\mathbf{I}, \qquad \forall\, i \text{ and } j$$

$$\mu_{f2}\left(f^2(\mathbf{x})\right) = \left\{ \frac{\left[f^{2+} - f^2(x_{ij})\right]}{\left[f^{2+} - f^{2-}\right]} \right\}^{ra} \geq (\lambda + \lambda_2),$$

$\lambda + \lambda_i \leq 1,\ i = 1, 2, \dots, s+2$

$\theta_{k1} \geq \alpha,\ k = 1$ and 2 (the number of objectives)

with the extra space for $\theta_{k2}, f^1(\mathbf{x}) \geq \varepsilon_1, \mathbf{x}_1 \geq \varepsilon_s \mathbf{I}$, or $f^2(\mathbf{x}) \geq \varepsilon_2$,

(ε-constraints, if necessary) (6)(cont.)

$\gamma, \lambda, \lambda_1, \lambda_2, \dots, \lambda_{s+2} \in [0, 1]$,

$\alpha(l_{ij2} - l_{ij1}) + l_{ij1} \leq x_{ij} \leq u_{ij4} - \alpha(u_{ij2} - u_{ij1}), \forall\, i$ and j, (constraint space)

$\sum_i \sum_j c_{ij1} x_{ij} \leq f^1 \leq \sum_i \sum_j c_{ij1} x_{ij}$, and $\theta_{11} \geq \alpha$, (objective space)

$\sum_i \sum_j c_{ij1} x_{ij} \leq f^2 \leq \sum_i \sum_j c_{ij1} x_{ij}$, and $\theta_{21} \geq \alpha$,

$\alpha, \gamma, \lambda, \lambda_1, \lambda_2, \dots, \lambda_{s+2}, \theta_{12}, \theta_{22} \in [0, 1]$,

$x_{ij} \geq 0$ and integer,

where $\mathbf{p}^1$ and $\mathbf{p}^2$ are the two-side tolerances for decision vector $\mathbf{x}_1$ (including s elements) on LHS and RHS, respectively. The parameter ra depicts the risk attitude of DMs, where $ra \in \{1/2, 1, 2\}$. In addition, ε_1, ε_2, and ε_s will be the extra attainments of f^1, f^2, and $\mathbf{x}_1$, respectively, for parametric variation. And $x_{ij} = (x_{ij}^1, x_{ij}^2)$, and $i = 1, \dots, m$ and $j = 1, \dots, n$. Accordingly, the minimization problem takes θ_{k1}, $k = 1$ and 2, where $\theta_{11} = [f^1 - \sum_j c_{j1} x_j]/[\sum_j c_{j2} x_j - \sum_j c_{j1} x_j]$ and $\theta_{21} = [f^2 - \sum_j c_{j1} x_j]/[\sum_j c_{j2} x_j - \sum_j c_{j1} x_j]$.

This formulation can be easily solved through any mathematical programming code, e.g., [30].

4. AN INTERACTIVE PROCESS FOR IMPRECISE MLPPS

All decision information is organized in logical order as the input for interaction. On the contrary, the output will be the temporary solution of a specified problem. The interactive process produces a series of dialog between DMs and analysts/machines to adjust the parameters of input and to generate the temporary results. Then the results are presented to DMs, and ask for trade-off among them. The process will be stopped until a compromise or satisfactory solution is acquired. Figure 2 illustrates the flowchart of the proposed approach. In detail, there are two interactive procedures for the imprecise MLPP: inside loop and outside loop. The former is major for the preference of the DM, realized by fuzzy membership functions reflecting goals and decisions attainments, risk attitude and compensation; the latter is for the imprecision of coefficients, described by four possibility indices or cut-off values. In the mean time, ε-constraints are involved for interaction if DMs like to make a further improvement. The procedures of the approach are condensed as the following five steps.

STEP 1. Set the initial decision information. Given the necessary decision information for starting the process, the information includes: reference values of PINs/NISs related to goals and decisions, the grade of compensation γ, the parameter of risk-attitude ra, and a possibilistic index and/or a cut-off value α.

STEP 2. Solve the initial bilevel problems individually. The upper-level and lower-level DMs solve their problems independently. If DMs are satisfied with the solutions, a compromise solution of the system is obtained. Go to stop; otherwise, go to Step 3.

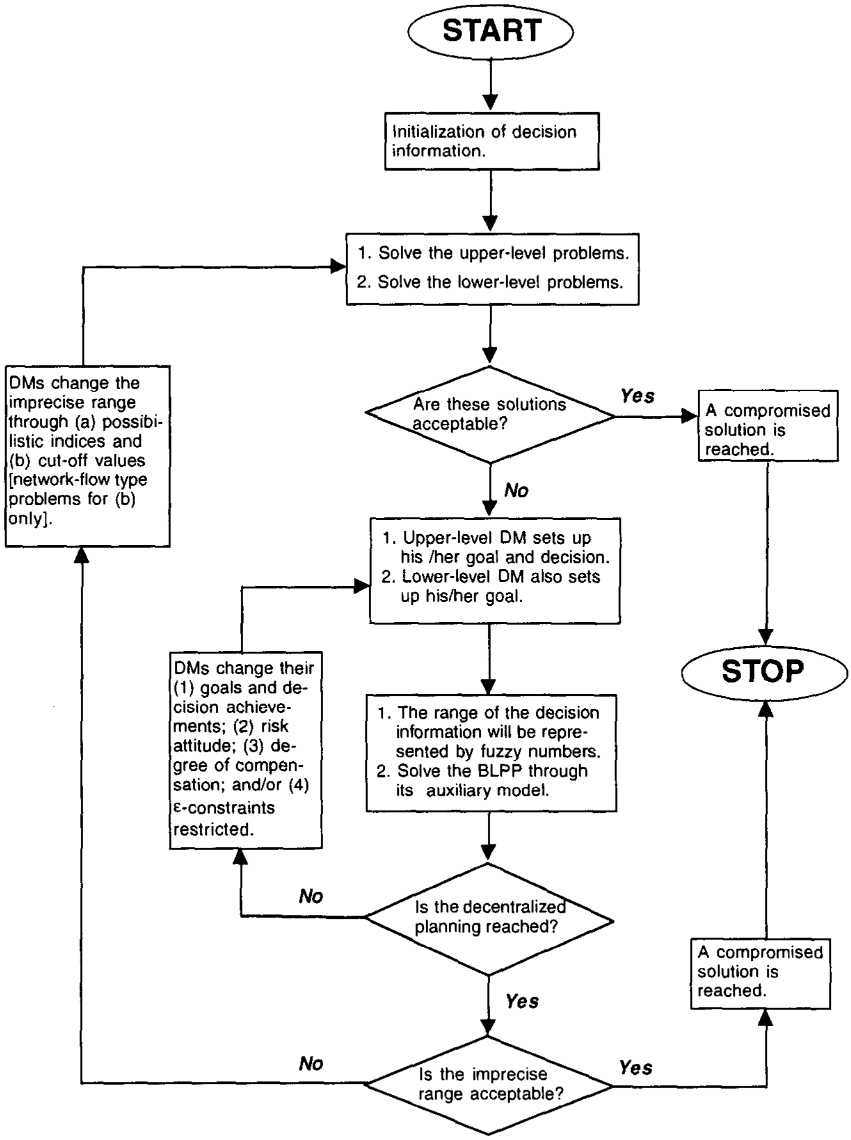


Figure 2. A flow chart for the proposed interactive approach for integrated bilevel systems.

STEP 3. Establish an auxiliary problem of lower levels. With the initial decision information and the solutions of two levels, the tolerance and other requirements of the upper level will transfer to the lower level, thus, establishing an auxiliary problem. After that, go to Step 4.

STEP 4. Check the decentralized planning. Ask the DMs of both levels if they are satisfied with the solution of the previous problem. If yes, a compromise solution of the system is reached, and goes to the next step for adjusting imprecise range. Otherwise, change the membership functions related to goals and decisions achievements, risk attitude, and degree of compensation. In addition, the ε-constraints related to the objectives of both levels and decisions of the upper level will be combined for trade-off. Then go to Step 3.

STEP 5. Modify the imprecise range. If the DMs are not satisfied with the imprecise range, choices of a possibilistic index and a cut-off value will be made to generate a new imprecise bilevel problem. Go to Step 2 and solve the new problem. Otherwise, a compromise solution is obtained, and then goes to stop. In addition, network-type problems have only one parameter, cut-off value, to be adjusted.

To illustrate this approach, let us solve the following two examples.

Table 1. The compared results from different possibilistic indices.

	Exceed. Poss.	Strict Exceed. Poss.	Exceed. Necess.#	Strict Exceed Necess
α	.5000000	.5000000	.4000000	.5000000
μ_{and}	.4753614	.4907555	4989722	.4999185
γ	.5000000	5000000	.5000000	5000000
λ	8523280	.0000000	.9942292	.9992455
λ_1	1475123	9735465	.0053751	.0000000
λ_2	.1476720	1 000000	0057708	.0007544
λ_3	.0000000	9709862	0000000	.0010201
f_1	23.46625	19.10098	14.99406	11.07164
f_2	18 00117	18.37106	14.61517	13 29353
x_1	11.06000	9 430000	7.210000	5.770000
x_2	1.891667	2.822024	2.269907	2 547812

Note.

1 These results are without considering risk attitudes and ε-constraints.

2. # There is no feasible solution at $\alpha \geq 0.5$.

Table 2. The compared results from different ε-constraints (Example 1 with exceedance possibility).

	Exceed. Poss. (Original Problem)	f_1 [-5%]	f_2 [-5%]	x_1 [-5%]
α	.5000000	5000000	.4000000	.5000000
μ_{and}	.4753614	4612924	.4753614	.4168945
γ	.5000000	5000000	5000000	5000000
λ	.8523280	.0000000	.0000000	.0000000
λ_1	.1475123	.9154264	.9998402	.8883504
λ_2	.1476720	1.000000	1.000000	.6471370
λ_3	.0000000	.8523280	.8523280	.9658795
f_1	23.46625	21.48506	23.46625	20.84958
f_2	18.00117	18.00117	18.00117	20.39938
x_1	11.06000	11 06000	11 06000	10.33310
x_2	1.891667	1 891667	1.891667	3.199861

Note

1. [−5%] indicates a ε-constraint being added with −5% of its original achievement

Table 3. The compared results from different ε-constraints (Example 1 with exceedance necessity).

	Exceed. Necess. (Original Problem)	f_1 [-5%]	f_2 [-5%]	x_1 [-5%]
α	.4000000	.4000000	.4000000	.4000000
μ_{and}	.4989722	.4904301	4910178	.4873892
γ	.5000000	.5000000	5000000	.5000000
λ	.9942292	0000000	.0000000	.0000000
λ_1	.0053751	9483514	.9999111	.9885714
λ_2	.0057708	1.000000	1 000000	9445100
λ_3	.0000000	.9942292	9461955	.9912536
f_1	14.99406	14.22527	14.99867	14.82857
f_2	14.61517	14.61517	13.90907	14.57143
x_1	7.210000	7.210000	7.21000	7.142857
x_2	2.269907	2.269907	2.26333	2.285714

Note.

1 [−5%] indicates a ε-constraint being added with −5% of its original achievement.

Table 4. The compared results from different risk attitudes (Example 1 with exceedance possibility).

	Fun(·) (Exceed. Poss.)	Fun(·)$^{0.5}$	Fun(·)^2
α	.5000000	.5000000	.5000000
μ_{and}	.4753614	.4871894	.4543572
γ	.5000000	.5000000	.5000000
λ	.8523280	.0041143	.0000000
λ_1	.1475123	.9958057	.9996805
λ_2	.1476720	.9958856	1.000000
λ_3	.0000000	.9191017	7264630
f_1	23.46625	23.46625	23.46625
f_2	18.00117	18.00117	18.00117
x_1	11.06000	11.06000	11.06000
x_2	1.891667	1.891667	1.891667

Note.

1. Fun(·) represents the original membership function, a linear case.
2. Fun(·)$^{0.5}$ indicates the original membership function with the power of 0.5, so as the function with the power of 2.

Table 5. The compared results from different ε-constraints (Example 1 with exceedance necessity).

	Fun(·) (Exceed. Necess.)	Fun(·)$^{0 5}$	Fun(·)^2
α	.4000000	.4000000	.4000000
μ_{and}	.4989722	.4994854	.4979500
γ	.5000000	.5000000	.5000000
λ	.9942292	.9971104	.0000000
λ_1	.0053751	.0026917	.9992088
λ_2	.0057708	.0028895	.9999998
λ_3	.0000000	.0000000	.9884915
f_1	14.99406	14.99406	14.99407
f_2	14.61517	14.61517	14.61517
x_1	7.210000	7.210000	7.210000
x_2	2.269907	2.269907	2 269907

Table 6. The parameters and structure of an imprecise bilevel MCF problem (Example 2 with 8 nodes and 11 arcs).

(a) Arc information.

Arc No.	1st Objective Imprecise Cost	2nd Objective Imprecise Time	Imprecise Capacity	Note
1	(0.5,1 ,1.5,2)	(2 ,2 ,2.5,3)	(0 ,0 ,9 ,11)	x_{21}
2	(0 ,0 ,0.5,1)	(1 ,2 ,2.5,3)	(0 ,1 ,9 ,13)	x_{23}
3	(5 ,6 ,7 ,8)	(5 ,6 ,7 ,8)	(0 ,0 ,9 ,11)	x_{26}
4	(1.5,2 ,2.5,3)	(2 ,2 ,3 ,3)	(0 ,2 ,15 ,16)	x_{14}
5	(0.5,1 ,1.5,2)	(1.2,2 ,2 ,2.5)	(0 ,0 ,5 ,9.5)	x_{34}
6	(3 ,4 ,5 ,6)	(1.5,2 ,2 ,2.5)	(0 ,0 ,10 ,12)	x_{35}
7	(4 ,5 ,6 ,7)	(6 ,7 ,7.5,8)	(0 ,2 ,10,14.5)	x_{47}
8	(1.5,2,2.5,3.5)	(1 ,2 ,2.5,3)	(0 ,0 ,20 ,22)	x_{56}
9	(6 ,7,8 ,9)	(1 ,2 ,2.5,3)	(0 ,0 ,15 ,17)	x_{57}
10	(7 ,8 ,9 ,10)	(2 ,2.5,3 ,3.5)	(0 ,1 ,10 ,12)	x_{68}
11	(8 ,9 ,10 ,11.5)	(2 ,2.2,3 ,3.5)	(0 ,0 ,15 ,16.5)	x_{78}

(b) Node information.

Node No.	❶	❷	❸	❹	❺	❻	❼	❽
Supply/Demand(−)	10	20	0	−5	0	0	−15	−10

(c) The structure of the MCF problem.

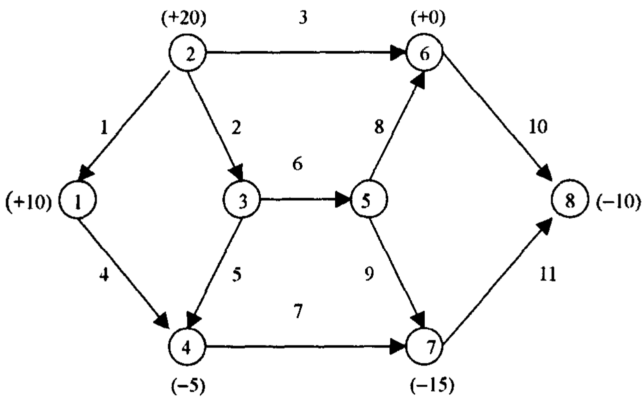


Table 7. The compared results from different cut-off values (Example 2).

Cut-Off Value α	0.5	0.75	1.0
γ	.5000000	.5000000	.5000000
μ_{and}	.4063796	.3411644	.2774995
λ	.0000000	.0000000	.0000000
λ_{14}	1.000000	.8000000	.6000000
λ_{35}	1.000000	.8571429	.7142857
λ_2	.9711400	.8932179	.7835498
λ_3	.2798971	.1789541	.1221603
f_1	241.5000	255.0000	274.0000
f_2	206.6000	218.3750	225.0000
x_{14}	10.00000	11.00000	12.00000
x_{21}	.0000000	1.000000	2.000000
x_{23}	10.00000	10.00000	9.000000
x_{26}	10.00000	9.000000	9.000000
x_{34}	6.000000	5.000000	3.000000
x_{35}	4.000000	5.000000	6.000000
x_{47}	11.00000	11.00000	10.000000
x_{56}	.0000000	1.000000	1.000000
x_{57}	4.000000	4.000000	5.000000
x_{68}	10.00000	10.00000	10.00000
x_{78}	.0000000	.0000000	.0000000
α	.5000000	.7500000	1.000000

5. ILLUSTRATED EXAMPLES

Two types of the problems are illustrated here, i.e., a linear-programming type and a network-flow type of problems, as an integrated imprecise multilevel programming system.

EXAMPLE 1. A trade-off MLPP between exports and imports with TFNs. Four numbers in the parentheses of the following problem figure out the imprecise parameters:

$$\max_{\mathbf{x1}} f_1 = (1.5, 2, 2, 2.5)x_1 + (-1.5, -1, -1, -0.5)x_2 \qquad \text{(upper level)}$$

where x_2 solves,

Table 8. The compared results from different ε-constraints (Example 2).

	Original Problem	$f_1[+5\%]$	$f_2[-5\%]$	$x_{14}[+5\%]$	$x_{35}[-5\%]$
α	.50000	.50000	.50000	.50000	.50000
γ	.50000	.50000	.50000	.50000	.50000
μ_{and}	.40637	.35812	.35812	.37915	.37336
λ	.00000	.00000	.00000	.00000	.00000
λ_{14}	1.00000	1.00000	1.00000	.80000	1.00000
λ_{35}	1.00000	.57142	.57142	1.00000	.75000
λ_2	.97114	.88889	.88889	.96104	.99856
λ_3	.27989	.40462	.40462	.27218	.23823
f_1	241.5000	255.7500	255.7500	243.2500	236.7500
f_2	206.6000	192.0500	192.0500	207.5000	211.4500
x_{14}	10.0000	10.0000	10.0000	11.0000	10.0000
x_{21}	.00000	.00000	.00000	1.00000	.00000
x_{23}	10.0000	10.0000	10.0000	9.00000	10.0000
x_{26}	10.0000	10.0000	10.0000	10.0000	10.0000
x_{34}	6.00000	3.00000	3.00000	5.00000	7.00000
x_{35}	4.00000	7.00000	7.00000	4.00000	3.00000
x_{47}	11.0000	8.00000	8.0000	11.0000	12.0000
x_{56}	.00000	.00000	.00000	.00000	.00000
x_{57}	4.00000	7.0000	7.00000	4.00000	3.00000
x_{68}	10.0000	10.0000	10.0000	10.0000	10.0000
x_{78}	.00000	.00000	.00000	.00000	.00000

Note.
1. [−5%] indicates a ε-constraint being added with −5% of its original achievement.

Table 9. The compared results from different risk attitudes (Example 2).

	Fun(·) (Original Problem)	$\text{Fun}(\cdot)^{0.5}$	$\text{Fun}(\cdot)^2$
α	.5000000	.5000000	.5000000
γ	.5000000	.5000000	.5000000
μ_{and}	.4063796	.4393147	.3824878
λ	.0000000	.0058383	.0598331
λ_{14}	1.000000	.9941616	.9401668
λ_{35}	1.000000	.9941616	.9401668
λ_2	.9711400	.9796260	.9401668
λ_3	.2798971	.5232147	.0000689
f_1	241.5000	241.5000	236.5000
f_2	206.6000	206.6000	210.7000n
x_{14}	10.00000	10.00000	10.00000
x_{21}	.0000000	.0000000	.0000000
x_{23}	10.00000	10.00000	11.00000
x_{26}	10.00000	10.00000	9.000000
x_{34}	6.000000	6.000000	7.000000
x_{35}	4.000000	4.000000	4.000000
x_{47}	11.00000	11.00000	12.000000
x_{56}	.0000000	.0000000	1.000000
x_{57}	4.000000	4.000000	3.000000
x_{68}	10.00000	10.00000	10.00000
x_{78}	.0000000	.0000000	.0000000

Note.
1. Fun(·) represents the original membership function, a linear case.
2. $\text{Fun}(\cdot)^{0.5}$ indicates the original membership function with the power of 0.5, so as the function with the power of 2.

$$\begin{array}{lll} \underset{x2}{\text{Max}} & f_2 = (0.8, 1, 1, 1.4)x_1 + (1.5, 1.5, 2, 2.5)x_2 & \text{(lower level)} \\ \text{s.t.} & & \\ & (2.5, 3, 3, 3.5)x_1 + (-5.5, -5, -5, -4.5)x_2 \leq (10, 25, 25, 26), & \\ & (2.8, 3, 3.2, 3.5)x_1 + (-1.5, -1, -1, -0.5)x_2 \leq (15, 20, 25, 35), & \\ & (2.75, 3, 3.2, 3.2)x_1 + (0.8, 1, 1, 1.3)x_2 \leq (20, 25, 25, 42), & \\ & (2.5, 3, 3, 3.5)x_1 + (3.6, 4, 4, 4.4)x_2 \leq (30, 32, 35, 50), & \\ & (0.8, 1, 1, 1.2)x_1 + (2.6, 3, 3, 3.4)x_2 \leq (14, 15, 16, 25), & \\ & x_1, x_2 \geq 0. & \end{array}$$

The possible range for each objective can be established as: $f_1 \in [0, 23.47]$ and $f_2 \in [0, 21.12]$. The first level control decision x_1 is around 11.6 with negative-sided and positive-sided tolerances 2.06 and 1.94, respectively. With many decision situations, we choose some of them listed in Tables 1–5 for comparison purposes. Table 1 shows the results from four possibilistic indices. Tables 2 and 3 exhibit the different results of ε-constraints of objectives and decision under exceedance possibility and exceedance necessity, respectively. Tables 4 and 5 illustrate the different results related to risk attitude under exceedance possibility and exceedance necessity, respectively. All above results are with the degree of compensation $\gamma = 0.5$ and cut-off value $\alpha = 0.5$, except the third index, with $\alpha = 0.4$. The DMs can choose one compromise solution among them, or make a further search for other desired solutions.

EXAMPLE 2. A bilevel imprecise MCF problem with eight nodes and 11 arcs with TFNs (as shown in Figure 6 with data and structure).

The bilevel network-flow problem is requested to minimize the total cost f^1 for the upper-level DM and to minimize the passing time f^2 for the lower-level DM. Based on the information of PISs and NISs, the possible range for each objective can be established as $f^1 \in [236.5, 409.75]$ and $f^2 \in [176.6, 293.25]$. Assume that the upper-level DM has two control variables, the first control decision x_{14} is around 10 within the interval [4,15]. The second control decision x_{35} is around four within the interval [0, 11]. With many decision situations, we choose some of them listed in Tables 7–9 for comparison. Table 7 shows the results under three cut-off values, 0.5, 0.75, and 1.0, respectively. Table 8 delineates the different results of ε-constraints of objectives and decisions. Table 9 displays the different results related to risk attitude. The degree of compensation $\gamma = 0.5$ is for Tables 7–9, and cut-off value $\alpha = 0.5$ is for Tables 7 and 8. From these listed results, the DMs can choose one compromise solution, or make a further enumeration for examining a compromise solution.

6. CONCLUDING REMARKS

The proposed interactive approach gets ride of the complexity and indeed makes trade-off among objectives and decisions of both levels of MLPPs. It can improve the flexibility and robustness of the original fuzzy approach of MLPPs. And the compromise solution of imprecise MLPP is obtainable with a few iterations under supervised search. Compared to simulation, our proposed approach is rather efficient and would make DMs more confident about the results in a dynamic environment. In addition, the presented algorithm can be directly extended to multilevel cases without increasing their computational complexity. Furthermore, linear-programming type and network-flow type of MLPPs are integrated as a unified approach in the paper.

In the proposed interactive approach, only ε-constraint is explicitly taken as a trade-off of objectives and decisions to fulfill the characteristics of MLPPs. More sophisticated interactive techniques will be expected in the future. Nevertheless, their computational burden will make the problem difficult to handle. Furthermore, under a strict case with more ε-constraints (over-achievement requirement) or less possibilistic measure (strict exceedance necessity), infeasible solutions will be generated. In such a case, DMs must modify his understandings and preference to reach a compromise solution.

The sketch of risk attitude will elicit nonlinear membership functions that depend on the meanings of a particular situation. Thus, many other shapes would be possible for MLPPs after an empirical test. These nonlinear shapes can be transferred to linear shapes through piecewise approximation [33]. Yet, extra work is necessary.

The proposed man-machine interactive procedure can be implemented into decision support systems where all choices and judgments are fulfilled in a computerized procedure (see [14]). It would be more efficient if the procedure can be executed on a computer network with graphic interface [34]. In addition, other nontraditional approaches for MLPPs [35] are also a new direction for future study, with computational efficiency under verification.

REFERENCES

1. H.S. Shih, Y.J. Lai and E.S. Lee, Fuzzy approach for multilevel programming problems, *Computers & Operations Research* **23** (1), 73–91, (1996).
2. Y.J. Lai, IMOST: Interactive multiple objective system technique, *J. of the Operational Research Society* **46**, 958–976, (1995).
3. Y. Aksoy, Interactive multiple objective decision making: A bibliography (1965–1988), *Management Research News* **13**, 1–8, (1990).
4. C.L. Hwang and A.S.M. Masud, *Multiple Objective Decision Making—Methods and Applications*, Springer-Verlag, Berlin, (1979).
5. Y.J. Lai and C.L. Hwang, *Fuzzy Multiple Objective Decision Making*, Springer-Verlag, Berlin, (1994).
6. U.P. Wen and S.T. Hsu, Linear bi-level programming problems—A review, *J. of Operational Research Society* **42**, 125–133, (1991).
7. O. Ben-Ayed and C.E. Blair, Computational difficulties of bilevel linear programming, *Operations Research* **38**, 556–560, (1990).
8. J.F. Bard, Some properties of the bilevel programming problem, *J. of Optimization Theory and Applications* **68** (2), 371–378, (1991).
9. H.S. Shih and E.S. Lee, Fuzzy multi-level minimum cost flow problems, *Fuzzy Sets and Systems* **107** (2), 159–176, (1999).
10. H.S. Shih and E.S. Lee, Compensatory fuzzy multiple level decision making, *Fuzzy Sets and Systems* **114** (1), 71–87, (2000).
11. S. Zionts and J. Wallenius, An interactive programming method for solving the multiple criteria problem, *Management Science* **22**, 652–663, (1976).
12. S. Zionts and J. Wallenius, An interactive multiple objective linear programming method for a class of underlying nonlinear utility functions, *Management Science* **29** (5), 519–529, (1983).
13. D. Vanderpooten and P. Vincke, Description and analysis of some representative interactive multicriteria procedures, *Mathl. Comput. Modelling* **12** (10/11), 1221–1238, (1989).
14. M. Fedrizzi, J. Kacprzyk and M. Roubens, Editors, *Interactive Fuzzy Optimization*, Springer-Verlag, Berlin, (1991).
15. X. Shi and H. Xia, Interactive bilevel multi-objective decision making, *J. of Operational Research Society* **48**, 943–949, (1997).
16. M. Sakawa, I. Nishizaki and Y. Uemura, Interactive fuzzy programming for multilevel linear programming problems, *Computers Math. Applic.* **36** (2), 71–86, (1998).
17. M. Sakawa, I. Nishizaki and M. Hitaka, Interactive fuzzy programming for multi-level 0-1 programming problems through genetic algorithms, *European J. of Operational Research* **114**, 580–588, (1999).
18. D. Dubois and H. Prade, *Possibility Theory*, Plenum Press, New York, (1988).
19. B.M. Werners, Aggregation models in mathematical programming, In *Mathematical Models for Decision Support*, (Edited by G. Mitra), pp. 295–305, Springer-Verlag, Berlin, (1988).
20. H.F. Wang and C.C. Fu, A generalization of fuzzy goal programming with preemptive structure, *Computers and Operations Research* **24** (9), 819–828, (1997).
21. A. Goicoech, D.R. Hansen and L. Duckstein, *Multiobjective Decision Analysis with Engineering and Business Applications*, John Wiley, New York, (1982).
22. Y.J. Lai, T.Y. Liu and C.L. Hwang, TOPSIS for MODM, *European J. of Operational Research* **76**, 486–500, (1994).
23. L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets and Systems* **1**, 3–28, (1978).
24. J.J. Buckley, Possibilistic linear programming with triangular fuzzy numbers, *Fuzzy Sets and Systems* **20**, 135–138, (1988).
25. D.S. Negi, Fuzzy analysis and optimization, Ph.D. Dissertation, Dept. of Industrial Engineering, Kansas State University, Manhattan, KS, (1989).
26. H.J. Zimmermann, Fuzzy programming and linear programming with several objective functions, *Fuzzy Sets and Systems* **1**, 45–55, (1978).

27. H.J. Zimmermann and P. Zysno, Latent connectives in human decision making, *Fuzzy Sets and Systems* **4**, 37–51, (1980).
28. C.A. Holloway, *Decision Making Under Uncertainty: Models and Choices*, Prentice-Hall, Englewood Cliffs, NJ, (1979).
29. M.A. Venkataramanan, J. Mote and D.L. Olson, An interactive MOLP procedure using epsilon-constraints, *Computers Math. Applic.* **17** (12), 1535–1543, (1989).
30. Lingo, The modeling language and optimizer (Extended 6.0), Lindo Systems Inc., Chicago, IL, 1999.
31. K.G. Murty, *Network Programming*, Prentice-Hall, Upper Saddle River, NJ, (1998).
32. R.K. Ahuja, T.L. Magnanti and J.B. Orlin, *Network Flows*, Prentice-Hall, Englewood Cliffs, NJ, (1993).
33. Y.J. Lai and C.L. Hwang, *Fuzzy Mathematical Programming*, Springer-Verlag, Berlin, (1992).
34. W.Y. Ng, An interactive descriptive graphical approach to data analysis for trade-off decisions in multi-objective programming, *Information and Decision Technologies* **17**, 133–149, (1991).
35. B. Liu, Stackelberg-Nash equilibrium for multilevel programming with multiple followers using genetic algorithms, *Computers Math. Applic.* **36** (7), 79–89, (1998).
36. O. Ben-Ayed, Bilevel linear programming, *Computers and Operations Research* **20**, 485–501, (1993).