



An analysis on the implementation of soft computing techniques to solve multi-level mathematical programming problems

Savita Mishra

Department of Mathematics,
The Graduate School College for Women's, Sakchi,
Jamshedpur, Jharkhand, India-831001

Abstract: Multi-level programming techniques are developed to solve decentralized planning problems with multiple decision makers in a hierarchical organization. These become more important for contemporary decentralized organizations where each unit or department seeks its own interests. The multi-level programming problem is strongly NP-hard and non-convex, which implies that the problem is very challenging for most canonical optimization approaches using single-point search techniques to find global optima. In this paper we analyse efficiency of some soft computing approach for solving unmanageable complicated system. The presented approaches achieve efficient and feasible solution in an appropriate time which has been evaluated by comparing to references and test problems.

Keywords: Multi level programming problem, hierarchical structures, scalar optimization problem, analytic hierarchy process, optimal solution, satisfactory solution.

I. Introduction

There are many planning and/or decision making situations that can be properly represented by a multi-level programming model. In the real world, we often encounter decision making situations involving multiple decision makers (DMs). Especially in industrial or governmental decision making situations, those DMs have different interest and decision priority. A multi-level programming problem (MLPP) is one of mathematical optimization models for them. Most of the developments in MLP problems focus on bi-level programming problems (BLPPs)[5,6,7,8] as a class of MLPP. Three-level programming (TLP) is another class of MLP problems in which there are three independent decision-makers (DMs). Each DM attempts to optimize his objective function and is affected by the actions of the other DMs.

Consider a programming problem in which the government is at first level. During the planning period, the government proposes certain goals. In order to optimize the achievement of such goals, it formulates certain policy measures such as taxes and subsidies. The industries at the second level design their course of action keeping such policy measures in mind so that their objectives are fulfilled. The industries supply their products to the consumers in a certain area. The customers at the third level are at liberty to make their purchases from any industries. In doing so, the customers will consider economic criteria such as cost optimization. This is a three level programming problem [9] in which the government's objectives are at least in partial conflict with the two sectors industry and consumers, the policy makers face an optimization problem subject to the optimization problems for industries as well as for the consumers. Various approaches for MLPPs could exist according to situations which the DMs are placed in. Under the assumption that the DMs do not have motivation to cooperate mutually, a Stackelberg solutions is adopted as a reasonable solution for the situation. It is assumed that the DM at the upper level (leader) and the DM at the lower level (follower) completely know their objective functions and the constraints of the problem and they do not have any motivation to cooperate with each other, and the leader first makes a decision and then the follower specifies a decision so as to optimize the objective function of itself with full knowledge of the decision of the leader. Under this assumption, the leader also makes a decision such that the objective function of the leader is optimized. Then, a solution defined as the above mentioned procedure is called a Stackelberg (equilibrium) solutions, which has been employed as a solution concept for two-level mathematical programming problems.

During the past few decades, many methodological developments have been reported for MLPP. However, these methods are proven to be computationally ineffective and can handle only simple MLLPs. Most of these methods are based on concepts of vertex enumeration and transformation approaches. The former is to seek a compromise vertex by simplex algorithm based on adjusting higher level control variables. It is rather inefficient, especially for large size problems. The later involves transforming the lower level

programming problem to be the constraints of the higher-level by its Kuhn-Thucker (K-T) conditions or penalty functions, because non-linear terms appear in constraints, the auxiliary problems become complex and sometimes unmanageable.

To overcome the shortcomings of the traditional methods, the concept of soft computing and artificial intelligence technique like: Genetic algorithm, fuzzy set theory, goal programming etc. was incorporated for large and complex hierarchical optimization problems. In a hierarchical decision making context, it has been realized that each DM should have a motivation to cooperate with the other, and a minimum level of satisfaction of the DM at a lower-level must be considered for the overall benefit of the organization. The use of the concept of membership function of fuzzy set theory to MLPPs for satisfactory decisions was first introduced by Lai [4]. Thereafter, Lai's satisfactory solution concept was extended to solve many such problems. Abo-Sinha [1] extended the fuzzy approach for multi-level programming problems of Shih et al. [10] for solving bi-level and three-level non-linear multi-objective programming problems. The basic concept of these fuzzy programming (FP) approaches is the same as implies that each lower level decision maker optimizes his/her objective function, taking a goal or preference of the first level decision makers into consideration. In the decision process, the membership functions of the fuzzy goals for the decision variables of all the decision makers are taken into consideration and an FP problem is solved with a constraint on an overall satisfactory degree of any upper levels. If the proposed solution is not satisfactory to any upper levels, the solution search is continued by redefining the elicited membership functions until a satisfactory solution is reached.

The main difficulty arises with the fuzzy programming approaches developed so far is that there is a possibility of rejecting the solution again and again by the leader and re-evaluation of the problem is repeatedly needed to reach the satisfactory decision, when the objectives of DMs are over conflicting. Unfortunately, there is a possibility that their method leads to an undesirable final solution because of inconsistency between the fuzzy goals of the objective function and the decision variables. A bibliography of the related references on bi-level and multi-level programming in both linear and non-linear cases, which is updated biannually, can be found in [11]. However, compared to linear MLPPs, only few methodological developments have been reported for non-linear or fractional MLPPs.

The MLP problems that meet either the solution space construction or the model development can be classified into three categories as follows: 1) ill-structured, 2) semi-well-structured or, 3) well structured. Each category has been characterized by specific criteria to indicate its class. Some of these indicator criteria of ill-structured problems are as follows:

1. There is no available solution technique to solve the model.
2. There is no standard mathematical model to represent the problem.
3. There is no ability to involve the qualitative factors in the model.
4. There is no available solution space to pick up the optimal solution.
5. There is a difficulty in measuring the quality of the result solution(s).
6. There is kind of vagueness of the available information that leads to complexity in considering it into the model account.

If some of these criteria exist, then the problem will belong to the second category, which is called semi-ill-structured problems. But, if all of these criteria and others do not exist, then the problem will belong to the third category, which is called well-structured problems. It is clear that there is no problem regarding the third category. Fortunately, the first and second categories represent a rich area for investigation, especially in the era of information technology where all the sciences are interchanged in a complex manner to a degree that one can find difficulty in separating between sciences. In other words, biological sciences, sociology, insects' science, and so on attracted researchers to simulate them by using computer technology.

II. Multi-level Programming problems (MLPPs)

MLPP can be defined as a p-person, nonzero sum game with perfect information in which each player moves sequentially from top to bottom and having linear or non-linear objective functions. Consider a p-level programming problem of maximization-type objective function at each level.

Mathematically, MLPP can be formulated as follows:

$$\begin{aligned}
 \max_{x_1} z_1(x) &= \max_{x_1} \text{imize} && z_1(x_1, x_2, \dots, x_p) \\
 \max_{x_2} z_2(x) &= \max_{x_2} \text{imize} && z_2(x_1, x_2, \dots, x_p) \\
 &\vdots && \\
 &\vdots && \\
 &\vdots && \\
 \max_{x_p} z_p(x) &= \max_{x_p} \text{imize} && z_p(x_1, x_2, \dots, x_p)
 \end{aligned}$$

Subject to

$$x \in S = \{A_1x_1 + A_2x_2 + \dots + A_px_p \ (\geq, =, \leq) B, x \geq 0\}.$$

Here, S is the non empty convex constraint set; M is the total number of constraints.

A_i is the $M \times N_i$ matrix ($i = 1, 2, \dots, p$);

B is M component column vector;

We take, $x = x_1 \cup x_2 \cup \dots \cup x_p$; $N_1 + N_2 + \dots + N_p = N$.

We also assume that $b_{i1}x_1 + b_{i2}x_2 + \dots + b_{ip}x_p + s_i > 0$, ($i = 1, 2, \dots, p$) for all $x \in S$.

This system has interacting decision making units within a hierarchical structure where each level performs its policies after knowing completely the decisions of superior levels.

A multi-level organization has following common features:

**Interactive decision making units within a predominantly hierarchical structure.*

**Execution of decision is sequential, from upper level to lower level.*

**Each unit independently maximizes or minimizes its own benefits, but is affected by the action of other units through externalities.*

**External effect on decision maker's problems can be reflected in both the objective function and the set of feasible decision space.*

III. Some soft computing technique for linear or non-linear MLPPs

There are many search and optimization algorithms in soft-computing and artificial intelligence, the popular ones being uninformed search, heuristic search and evolutionary algorithms etc. Although a lot of research work is done on individual algorithm but not enough research is done on the comparison of these algorithms under different problems.

The implementation of soft computing techniques to handle different problems in MLPP depends on the conditions: the nature of the problem that MLPP suffers from; the availability of the solution techniques and its performance; the environmental factors that affect the problem under study. Here in this section we discuss the flexibility of Weighting Method, Genetic Algorithm, Fuzzy Programming to handle all types of MLPPs.

A. Weighting method for MLPPs

In the weighting problem $\bar{p}(w)$ in the absence explicit preference structure, the strategy is to generate all or representative subsets of non-dominated solutions from which a DM can select the suitable solution.

Let a linear fractional MLPP be represented as:

$$\max_{x_1} z_1(\bar{x}) \quad (1)$$

$$\max_{x_2} z_2(\bar{x}) \quad (2)$$

.

.

$$\max_{x_p} z_p(\bar{x})$$

subject to

$$x \in S = \{A_1x_1 + A_2x_2 + \dots + A_px_p \ (\geq, =, \leq) B, x \geq 0\}$$

$$\bar{x}_1 \geq 0, \bar{x}_2 \geq 0, \dots, x_p(\bar{x}). \quad (3)$$

Where, $z_1(\bar{x}), z_2(\bar{x}), \dots, z_p(\bar{x})$, and $A_ix_i \ (\geq, =, \leq) B$ are linear or non-linear objective functions and linear or non linear constraints respectively. $\bar{x}_1 \geq 0, \bar{x}_2 \geq 0, \dots, x_p(\bar{x})$ are decision vectors under the control of the first level, second and so on p-th level DM.

Let $X =$ set of feasible solutions $= \{ \bar{x} : \bar{x} \in R^n, A_ix_i \ (\geq, =, \leq) B \}$,

$\bar{x}$ = decision vector in n-dimensional Euclidean space $= x = x_1 \cup x_2 \cup \dots \cup x_p$.

Solving the SOP involves finding $\bar{x}^* \in X$ such that $z(\bar{x}^*) \geq z(\bar{x}) \quad \forall \quad \bar{x} \in X$. The point $\bar{x}^*$ is said to be global optimum. If strict inequality holds for the objective functions, then $\bar{x}^*$ is the unique

global optimum. If the inequality holds for some neighborhood of $\bar{x}^*$, then $\bar{x}^*$ is a local or relative optimum while it is strict local optimum if strict inequality holds in a neighborhood of $\bar{x}^*$.

By using the AHP pairwise comparison process, weights or priorities are derived from a set of judgments. While it is difficult to justify weights that are arbitrarily assigned, it is relatively easy to justify judgments and the basis (hard data, knowledge, experience) for the judgments. Suppose already the relative weights of p -objective functions are known, for p -level hierarchical objective functions a complete pairwise comparison matrix A can be expressed as; $A = [a_{ij}]$, $i, j = 1, 2, \dots, p$ is a matrix of size $p \times p$ with the following properties; $a_{ij} > 0$, $a_{ii} = 1$, and $a_{ij} = 1/a_{ji}$, $i, j = 1, 2, \dots, p$ where a_{ij} is the numerical answer given by the each DM for the question "How many times objective i is more important than objective j ?"

$$A = \begin{matrix} & \begin{matrix} z_1 & z_2 & \dots & \dots & z_p \end{matrix} \\ \begin{matrix} z_1 \\ z_2 \\ \vdots \\ \vdots \\ z_p \end{matrix} & \begin{bmatrix} 1 & a_{12} & a_{13} & \dots & a_{1p} \\ a_{21} & 1 & a_{23} & \dots & a_{2p} \\ a_{31} & a_{32} & 1 & \dots & a_{3p} \\ \dots & \dots & \dots & \dots & \dots \\ a_{p1} & a_{p2} & a_{p3} & \dots & 1 \end{bmatrix} \end{matrix}$$

After the normalized matrix, N of pairwise comparison matrix A for a hierarchical p -level structure is designed, the normalized principal eigen vector (priority vector) can be obtained by some ways such as averaging across the rows where, it shows the relative weights for objectives. The weighting problem is to find the p -dimensional weight vector $W = (w_1, w_2, \dots, w_p)$ such that the appropriate ratios of the components of W reflect or, at least, approximate all the a_{ij} values ($i, j = 1, 2, \dots, p$) given by DMs.

Then, the weighting problem for multi-level (p -level) linear or non-linear programming problem is formulated as follows:

$$P(\bar{w}) = \max_{\bar{w} \in W} \sum_{n=1}^p \bar{w}_n z_n(\bar{x}) \quad (4)$$

subject to

$$g_i(x) \leq b_i, \quad i = 1, 2, \dots, m \quad (5)$$

$$\bar{L}_1 \leq \bar{x}_1 \leq \bar{U}_1 \quad (6)$$

$$\bar{L}_2 \leq \bar{x}_2 \leq \bar{U}_2 \quad (7)$$

$$\bar{L}_p \leq \bar{x}_p \leq \bar{U}_p$$

Where, $\bar{w} \in W = \{ \bar{w} : \bar{w} \in R^p, w_n \geq 0, n = 1, 2, \dots, p. \text{ and } \sum_{n=1}^p \bar{w}_n = 1 \}$.

$\bar{U}_p$ and $\bar{L}_p$ are the upper and lower bounds of decision vector provided by the respective DM. Finally the linear or non-linear programming problem (4) - (7), with a single objective function is solved. Here the weighting coefficients convey the importance attached to an objective function. Suppose that the relative importance of the both objective functions is known and is constant. Then the preferred solution is obtained by solving $p(\bar{w})$

where $\bar{w}_p$'s ≥ 0 are the weighting coefficients. The $\bar{w}_p$'s are normalized since, $\sum_{n=1}^p \bar{w}_n = 1$.

This method can be used to generate non-dominated solutions by utilizing various values of $\bar{w}$. In such a case the weighting coefficients $\bar{w}$ do not reflect the relative importance of the objective functions in the proportional sense, but are only parameters varied to locate the non-dominated points.

B. Design of the GA for MLPP

Genetic algorithm (GA) is search algorithms based on the mechanism of natural selection and natural genetics. GA is a stochastic heuristic optimization search technique designed following the natural selection process in biological evolution to arrive at optimal or near optimal solutions to complex decision problems. The primary concept behind the use of GAs is the representation of solutions to a problem in an encoded format. These encoded parameters (alleles) are referred to as genes and these are joined to build strings, which represent a potential solution to the problem. These strings of variables are called the chromosomes. Each gene can be represented by a binary string or a real value. The fitness of a chromosome as a candidate solution to a problem is an expression of the objective function represented by it. The basic idea solving MLPP by GA is: firstly, choose the initial population satisfying the constraints, then the lower-level decision makers make the corresponding optimal reaction and evaluate the individuals according to the fitness function constructed by the feasible degree, until the optimal solution is searched by the genetic operation over and over. It is not easy to know the upper-level objective function of MLPP has no explicit formulation, since it is compounded by the lower-levels solutions functions which has no explicit formulation. Thus, it is hard to express the definition of the derivation of the function in common sense. And it is difficult to discuss the conditions and the algorithms of the optimal solution with the definition. We concerned the GA [12,13] is a numerical algorithm compatible for the optimization problem since it has no special requirements for the differentiability of the function.

The algorithm process using the GA is as follows:

Step 1: Initialization: Set the parameters the population size M , crossover probability P_c , mutation probability P_m , the maximal generation of termination the algorithm T (maximal iteration generation $MAXGEN$), and then set the counter of generation $t=0$;

Step 2: Generating the initial population $P(0)$: The initial population $P(0)$ consists of a set of feasible chromosomes. Initialization of the initial population, M individuals are randomly generated in S , making up of the initial population. After generating sufficient chromosomes, goto next step;

Step 3: Computation of the fitness function.

Step 4: Generate the next generation by genetic operators. Select the individual by roulette wheel selection, crossover and mutate according to generate the next generation.

Step 5: Judge the condition of the termination. When t is larger than the maximal iteration number, stop the GA and output the optimal solution. Otherwise, let $t = t + 1$, turn to Step 3.

C. The Fuzzy Programming Approach (FPA) to MLPPs

The fact that Fuzzy logic touched the vagueness phenomenon led to a large number of its successful applications. One of the main arguments in favour of it consists in its ability to provide a working model of some manifestations of the vagueness phenomenon. Quite often, they rely on a simple model of the meaning of some words of natural languages. From linguist point of view, however, these applications are more or less naive.

Fuzzy set theory was first introduced in 1960s by Zadeh as a way to capture uncertainty and vagueness in complex systems. It was pointed out that Fuzzy set theory is a generalization of the classical set theory. Zimmerman [14] first applied fuzzy programming method to multi-objective linear programming by using the concept given by Bellman and Zadeh [2]. In the last two and half decades several fuzzy programming technique have been developed by various researchers. Lai and Hwang [4], have given a detailed survey on Fuzzy Mathematical Programming and Fuzzy Multiple Objective Decision Making. This technique has been widely applied to many disciplines such as: engineering design, maintenance, managerial decision making, production planning, quality control, reliability, transportation and water resources management. A fuzzy set is characterized by its membership function, which map each element of it to the interval $[0,1]$. This function indicates the degree of belonging for each element of the fuzzy set. Membership functions of fuzzy sets play a crucial role in solving some multi-criteria decision making problems. If it is to be applied to a real life problem, then it has to be selected carefully. The concept of fuzzy membership function is not unique. There are several type of fuzzy membership functions such as linear, triangular, trapezoidal, exponential, hyperbolic and quadratic membership function. Some of the membership functions are linear and some are non-linear. Depending on the nature of the decision making problem a membership function is to be selected. To exploit the benefit provided

by fuzzy programming, we need efficient membership functions which are accurate, flexible, computationally efficient and easy to use.

In the presence of real life data the selected membership function should reflect the knowledge contained in the data in the accuracy sense. Data in the form of membership function value is usually obtained from the decision maker's goals. The membership function should be computationally efficient for the practical purpose. The method may be used in which computer graphics can facilitate the process of constructing the membership function. Once the membership function has been generated, it should be easy to find the membership value.

Large-scale optimization problems can be described by static or dynamic systems. This system can be represented by either linear/non linear algebraic equations/ differential equations.

In fuzzy programming approach to MLLPs, unlike vertex enumeration, we do not pre-assume that the optimal solution exists at the corner points. Rather we find a satisfactory solution, which is more acceptable than an optimal solution. It is difficult to define solid optimality in a multi-person, decision-making process. Potential satisfactory solutions are those, which exist in the non-dominated region. The basic idea is that an upper-level DM sets his goal in terms of membership function and then asks each lower-level DM optima which are calculated in isolation; the lower-level DM's decisions are then submitted and modified by the upper level DM with consideration of the overall benefit for the organization; and the process is continued until a satisfactory solution is reached. Fuzzy programming approach to this problem is a supervised search procedure (supervised by the upper-level DM), which finds a satisfactory solution for a MLPPs. Fuzzy programming approach uses linear or non-linear membership function to find the optimal compromise solution. This process is extremely practical to decentralized system like agriculture, government policy, economic system, finance, welfare, transportation, network designs and is suitable for conflict resolution.

For instance, by adopting a criterion with respect to finance or corporate planning as an objective function at the upper level and employing a criterion regarding production planning as an objective function at the lower level, a bi-level linear fractional programming problem can be formulated and solved for hierarchical decision problems in firms.

In actual decision making situations, we most often make a decision on the basis of vague information or uncertain data. For such decision making problems involving uncertainty, there exist two typical approaches: probability-theoretic approach and fuzzy theoretic one. Assuming that the decision maker has a fuzzy goal for each of the objective functions, having determined the fuzzy goals of the decision maker, we present an interactive fuzzy satisficing method to derive a satisficing solution for the decision maker by updating the reference membership levels.

Fuzzy programming approach is a supervised search procedure under the control of the upper level DM. This is in accordance with a hierarchical system where the upper level DM is the main decision maker. This approach gives a satisfactory solution of the system, which is obtained by considering all the points enclosed, by the solution space. This satisfactory solution is more reasonable as it is obtained by considering the decision variables and the objective functions. Also this approach is interactive since the tolerance limits can only be decided by the DMs. A membership function should be chosen to best represent the objective function. The membership function and tolerance limit are always decided by the experience of the DMs. We also observe from the numerical example that the numerical value of realized degree of satisfaction will always lie within the prescribed range $[0,1]$.

IV. Discussion

Compared with the traditional methods, the method has the following features:

- (1) This class of approach is purely non-differential approach as it does not require any assumption or information regarding the DMs utility function.
- (2) The method has no special requirement for the characters of the function and overcome the difficulty discussing the conditions and the algorithms of the optimal solution with the definition of the differentiability of the function.
- (3) The main advantage of the proposed approach is that the possibility of rejecting the solution again and again and re-evaluation of the problem repeatedly, by redefining the elicited functions, needed to reach to the satisfactory decision does not arise.
- (4) The problem can never be infeasible and unbounded. These method facilitates computation to reduce the complexity in solving problem and is much more effective.
- (5) This is in accordance with a hierarchical system where the upper level DM is the main decision maker.
- (6) GA avoids the use of penalty function to deal with the constraints, by changing the randomly generated initial population into an initial population satisfying the constraints in order to improve the ability of the GA to deal with the constraints. In addition, in order to evaluate each individual, a fitness function was presented, the fitness evaluation, as a sub-procedure of optimization, can partly improve the leader's objective. Since the fittest members of the population are likely to become parents than others, a genetic algorithm tends to generate

improving populations of trial solutions as it proceeds. Mutations occasionally occur so that certain children also can acquire features (sometimes desirable features) that are not possessed by either parent. This helps a genetic algorithm to explore a new, perhaps better part of the feasible region than previously considered.

V. Conclusion

The solution methodologies adapted for multi-level decision problems are generally based on aggregation of decisions obtained for individual decision makers. This approach seems somewhat inadequate when the number of stakeholders is very large. The present study has been performed to have an overview of existing solution methodologies for multi-level decision making approaches. Decision making by single and multiple stakeholders has been considered under both deterministic and uncertain conditions. It has been found that the use of fuzzy set theory to represent various uncertainties associated with decision making situations under multi-objective multiple-participant environment is very promising. Coupled with multi-level methods (e. g. compromise programming and goal programming), fuzzy approach has also the ability to support group decisions, to reflect collective opinions and conflicting objectives.

GA work with an entire population of trial solution and the population is used to create linking paths between its members and to re-launch the search along these paths.

References

- [1] Abo-Sinna M. A. and Baky I. A., Fuzzy goal programming procedure to bilevel multiobjective linear fraction programming problems, *International Journal of Mathematics and Mathematical Sciences*, (2010), 01-15, ID 148975 (2010) 01-15 doi:10.1155/2010/48975.
- [2] Bellman R.E., Zadeh L.A. "Decision Making in Fuzzy Environment", *Management Sciences*, Vol. No.17, 1970, 141-164.
- [3] Holland, J.H. (1975). *Adaptation in natural and artificial systems*, University of Michigan Press, MI.
- [4] Lai and Hwang C.L., *Fuzzy Mathematical Programming, Methods and Applications*, Lecture Notes in Economics and Mathematical Systems, Springer Verlag, Heidelberg, Vol.394, 1992.
- [5] Mishra Savita, Verma Arun Bihari, A Computational Method using Analytic Hierarchy Process for Solutions to bi-level quadratic fractional programming problems, *International Journal of Innovative Research in Technology & Science (IJIRTS)*, 3(4) (2015): pp.38-44.
- [6] Mishra Savita, Ghosh Ajit, "Interactive Fuzzy Programming Approach to Bi-level Quadratic Fractional Programming Problems", 'Annals of Operational Research'; Springer, USA Volume - 143, pp-249-261, April-2006.
- [7] Mishra Savita "Weighting Method for Bi-level Linear Fractional Programming Problems", 'European Journal of Operational Research' Elsevier, Volume-183, pp-296-302, December-2007.
- [8] Mishra Savita, Ghosh Ajit, "Fuzzy Programming Approach to Bi-level Linear Fractional Programming Problems" 'International Journal of Applied Math. Analysis and Applications', Vol. 1 no: 2 (June 2007), Serial Publication.
- [9] Narang Ritu, Arora S.R., "An enumerative algorithm for non-linear multi-level integer programming problem", *Yugoslav Journal of Operations Research*, Volume 19, Number 2, (2009), 263-279.
- [10] H.S. Shih, Y.J. Lai, E.S. Lee, Fuzzy approach for multi-level programming problems, *Comput. Operat. Res.* 23 (1) (1996) 73-91.
- [11] Vicente L. N. and Calamai P. H., Bilevel and multilevel programming: a bibliography review, *Journal of Global Optimization*, 5 (3) (2004), 291-306.
- [12] Wang G., Wan Z. and Wang X, Solving Method for a Class of Bi-level Linear Programming based on Genetic Algorithms, optimization-online.org/DB_FILE/2003/03/617.
- [13] Wanga Guangmin, Wanb Zhongping et al, Genetic algorithm based on simplex method for solving linear-quadratic bilevel programming problem, *Computers and Mathematics with Applications* 56, (2008) 2550-2555.
- [14] Zimmermann R.J., *Fuzzy Programming and Linear Programming with Several Objectives*, Fuzzy Sets and Systems, Vol.1, 1978, 45-55.