An Interactive Satisficing Method Based on Possibilistic and Stochastic Programming Models for Fuzzy Random Multiojbective Integer Programming Problems

Cahit Perkgoz[1,2], Hideki Katagiri[1], Masatoshi Sakawa[1], and Kosuke Kato[1]

Graduate School of Engineering, Hiroshima University[1]
Kagamiyama 1-4-1, Higashi-Hiroshima, Hiroshima, 739-8527 Japan[1]
email: cperkgoz@msl.sys.hiroshima-u.ac.jp[2]

Abstract- In this paper, we consider a multiobjective integer programming problem involving fuzzy random variable coefficients. Introducing a fuzzy goal for each objective function, we focus on a degree of possibility that each objective function satisfies the corresponding fuzzy goal. Since the degree of possibility varies randomly, we formulate the multiobjective integer programming problem to minimize the variances of degrees of possibility with constraints with respect to the expectation of the objective function values. In order to find a satisficing solution for a decision maker, we propose an interactive satisficing method based on the reference point method and solve the formulated problem using the branch-and-bound method or a tabu search method.

1. Introduction

In classical mathematical programming, the coefficients of objectives or constraints in problems are assumed to be completely known. However, in real systems, they are rather uncertain than constant. In order to deal with such uncertainty, stochastic programming [1] and fuzzy programming [2,3] were considered. They are useful tools for the decision making under a stochastic environment or a fuzzy environment, respectively.

Most researches in respect to mathematical programming take account of either fuzziness or randomness. However, in practice, decision makers face with the situations where both fuzziness and randomness exist. For instance, in the case where some expert estimates coefficients of objective functions or constraints with uncertainty, they are not always given as random variables or fuzzy sets but as the values including both fuzziness and randomness. Fuzzy random variables [4,5] are one of the mathematical concepts dealing with fuzziness and randomness simultaneously. Recently, several authors considered linear programming problems involving fuzzy random variables [6–9]. In this research, we consider multiobjective integer programming problem using the concept of possibility measures [11] and the V-model in stochastic programming [12]. In Section 2, we consider a multiobjective integer programming problem with fuzzy random variable coefficients and formulate it as a multiobjective integer programming problem where each objective function is the variance of a degree of possibility with respect to a fuzzy goal. Section 3 proposes an interactive satisficing method for the problem to obtain the satisficing solution for a decision maker. Next, to solve the integer programming problems, tabu search method [13,14] is summarized in Section 4. Finally, in Section 5, we conclude this paper and discuss further research.

2. Formulation

In this paper, we consider the following multiobjective integer programming problem:

$$\left.\begin{array}{ll} \min \tilde{\bar{\boldsymbol{C}}}_i \boldsymbol{x}, & i=1,\ldots,k \\ \text{s. t. } A\boldsymbol{x} \le \boldsymbol{b} & \\ \quad \boldsymbol{x}_j \in \{0,1,\ldots,v_j\}, & j=1,\ldots,n \end{array}\right\} \quad (1)$$

where $\boldsymbol{x} = (x_1,\ldots,x_n)^t$ is a decision vector and $\tilde{\bar{\boldsymbol{C}}}_i = (\tilde{\bar{C}}_{i1},\ldots,\tilde{\bar{C}}_{in})$ is a coefficient vector. Let A be an $m \times n$ matrix and $\boldsymbol{b}$ an $m \times 1$ vector. Each $\tilde{\bar{C}}_{ij}$ is a fuzzy random variable with the following membership function:

$$\mu_{\tilde{\bar{C}}_{ij}}(t) = \max\left\{0, 1 - \frac{|t-\bar{c}_{ij}|}{\alpha_{ij}}\right\},\ i=1,\ldots,k,\quad j=1,\ldots,n \quad (2)$$

where $\bar{c}_{ij}$ denotes a random variable (or a scenario variable) whose realization under the scenario s_i is c_{ijs_i}, and the number of scenarios s_i corresponding to the ith objective function is S_i. Let p_{is_i} be the probability that each scenario s_i occurs. We assume that $\sum_{s_i=1}^{S_i} p_{is_i} = 1$ holds. Each α_{ij} denotes the spread of a fuzzy number. This type of fuzzy random variable is equivalent to a *hybrid number*, which was introduced by Kaufman and Gupta [15].

Since the coefficients of objective functions are the symmetric triangular fuzzy random variables, each objective function also becomes the same type of fuzzy random variable $\bar{\tilde{Y}}_i$ with the following membership function:

$$\mu_{\tilde{\bar{Y}}_i}(y) = \max \left\{ 0 \ , \ 1 - \frac{\left| y - \sum_{j=1}^{n} \bar{c}_{ij} x_j \right|}{\sum_{j=1}^{n} \alpha_{ij} x_j} \right\}, \ i = 1, \ldots, k. \tag{3}$$

Considering the imprecision or fuzziness of the decision maker's judgment, for each objective function of problem (1), we introduce the fuzzy goal $\tilde{G}_i$ with the membership function expressed as

$$\mu_{\tilde{G}_i}(y) = \begin{cases} 0, & y > g_i^0 \\ \dfrac{y - g_i^0}{g_i^1 - g_i^0}, & g_i^1 \leq y \leq g_i^0 \\ 1, & y < g_i^1, \qquad i = 1, \ldots, k. \end{cases} \tag{4}$$

Since the membership function $\mu_{\tilde{\bar{Y}}_i}$ is regarded as a possibility distribution, the degree of possibility $\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)$ that the objective function value satisfies the fuzzy goal $\tilde{G}_i$ is

$$\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i) = \sup_y \min \left\{ \mu_{\tilde{\bar{Y}}_i}(y), \ \mu_{\tilde{G}_i}(y) \right\}, \ i = 1, \ldots, k. \tag{5}$$

Accordingly, we consider the following multiobjective problem:

$$\left.\begin{array}{ll} \max \Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i), & i = 1, \ldots, k \\ \text{s. t. } A\boldsymbol{x} \leq \boldsymbol{b} & \\ \quad \boldsymbol{x}_j \in \{0, 1, \ldots, v_j\}, & j = 1, \ldots, n \end{array}\right\} \tag{6}$$

In this research, we calculate $g_i^{\max}$ and $g_i^{\min}$ defined by

$$g_i^{\max} = \max_{s_i} \max_{\boldsymbol{x} \in X} \sum_{j=1}^{n} c_{ijs_i} x_j, \quad i = 1, \ldots, k,$$

$$g_i^{\min} = \min_{s_i} \min_{\boldsymbol{x} \in X} \sum_{j=1}^{n} c_{ijs_i} x_j, \quad i = 1, \ldots, k,$$

where $X \triangleq \{\boldsymbol{x} | A\boldsymbol{x} \leq \boldsymbol{b}, \ 0 \leq \boldsymbol{x}_j \leq v_j \ j = 1, \ldots, n\}$. Assume that g_i^1 and g_i^0 are determined by a decision maker so as to satisfy the condition that $g_i^{\min} \geq g_i^1$ and $g_i^{\max} \leq g_i^0$. Then, by using (3) and (4), the degree of possibility is represented as follows:

$$\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i) = \frac{\sum_{j=1}^{n} \{\alpha_{ij} - \bar{c}_{ij}\} x_j + g_i^0}{\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0}, \ i = 1, \ldots, k.$$

Since the degree of possibility in problem (6) varies randomly, the problem is regarded as a stochastic programming problem. Katagiri et al.[10] proposed a fuzzy random multiobjective linear programming model, which is to maximize the expected degree of possibility that objective function values satisfy the respective fuzzy goals. This model is useful for decision making under fuzzy stochastic environments; however, in the obtained solution based on this model, there is a case where the degree of possibility corresponding to a certain scenario is fairly small because the variance of the degree of possibility is unconsidered. Therefore, in this research, we propose the model to minimize the variances of degrees of possibility subject to satisfying constraints with respect to the expectations. Then the problem to be considered is formulated as follows:

$$\left.\begin{array}{l} \min Var[\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)], \ i = 1, \ldots, k \\ \text{s. t. } A\boldsymbol{x} \leq \boldsymbol{b}, \ \boldsymbol{x}_j \in \{0, 1, \ldots, v_j\}, \ j = 1, \ldots, n \\ \quad E\left[\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)\right] \geq \delta_i, \ i = 1, \ldots, k \end{array}\right\} \tag{7}$$

where $E[\cdot]$ and $Var[\cdot]$ denote expectation and variance, respectively. The expectations and the variances of degrees of possibility are calculated as follows:

$$E[\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)] = \frac{\sum_{s_i=1}^{S_i} p_{s_i} \left[\sum_{j=1}^{n} \{\alpha_{ij} - c_{ijs_i}\} x_j + g_i^0 \right]}{\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0},$$

$$V[\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)] = \frac{1}{\left(\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0 \right)^2} V\left[\sum_{j=1}^{n} \bar{c}_{ij} x_j \right].$$

Let V_i denote the variance-covariance matrix of $\bar{\boldsymbol{c}}_i$. Then the problem to minimize the variances of degrees of possibility is formulated as

$$\left.\begin{array}{l} \min \dfrac{1}{\left(\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0 \right)^2} \boldsymbol{x}^T V_i \boldsymbol{x}, \ i = 1, \ldots, k \\ \text{s. t. } A\boldsymbol{x} \leq \boldsymbol{b}, \ x_j \in \{0, 1, \ldots, v_j\}, \ j = 1, \ldots, n \\ \quad \sum_{j=1}^{n} \left\{ \sum_{s_i=1}^{S_i} p_{is_i} c_{ijs_i} + (\delta_i - 1)\alpha_{ij} \right\} x_j \\ \qquad \leq (1 - \delta_i) g_i^0 + \delta_i g_i^1, \ i = 1, \ldots, k. \end{array}\right\} \tag{8}$$

The variance-covariance matrix is expressed by

$$V_i = \begin{bmatrix} v_{11}^i & v_{12}^i & \cdots & v_{1n}^i \\ v_{21}^i & v_{22}^i & \cdots & v_{2n}^i \\ \vdots & \vdots & \ddots & \vdots \\ v_{n1}^i & v_{n2}^i & \cdots & v_{nn}^i \end{bmatrix}, \quad i = 1, \ldots, k,$$

where

$$v_{jj}^i = V[\bar{c}_{ij}] = \sum_{s_i=1}^{S_i} p_{is_i} \{c_{ijs_i}\}^2 - \left\{ \sum_{s_i=1}^{S_i} p_{is_i} c_{ijs_i} \right\}^2, \quad j = 1, \ldots, n,$$

$$v_{jl}^i = Cov[\bar{c}_{ij}, \bar{c}_{il}] = E[\bar{c}_{ij}, \bar{c}_{il}] - E[\bar{c}_{ij}] E[\bar{c}_{il}], \quad j \neq l, \ l = 1, \ldots, n$$

and

$$E[\bar{c}_{ij}, \bar{c}_{il}] = \sum_{s_i=1}^{S_i} p_{is_i} c_{ijs_i} c_{ils_i}$$

In (8), since

$$\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0 > 0$$

and $\boldsymbol{x}^T V_i \boldsymbol{x} \geq 0$, an Pareto optimal solution set of the following problem is equivalent to that of (7).

$$\left.\begin{array}{l} \min\ z_i(\boldsymbol{x}) \triangleq \sqrt{V[\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)]},\ i = 1, \ldots, k \\ \text{s. t. } A\boldsymbol{x} \leq \boldsymbol{b},\ x_j \in \{0, 1, \ldots, v_j\},\ j = 1, \ldots, n \\ \qquad E\left[\Pi_{\tilde{\bar{Y}}_i}(\tilde{G}_i)\right] \geq \delta_i,\ i = 1, \ldots, k \end{array}\right\} \quad (9)$$

In the next section, we consider a method for solving problem (9).

3. Interactive Satisficing Method

3.1 Interactive algorithm based on the reference point method

Since problem (7) has several objective functions, there does not generally exist the solution optimizing all functions. Therefore, in this section, we discuss the interactive decision making based on the reference point method [16] to obtain a Pareto optimal solution.

For each of the multiple conflicting objective functions, assume that the decision maker can specify the so-called reference point $\bar{\boldsymbol{z}} = (\bar{z}_1, \ldots, \bar{z}_k)$ which reflects in some sense the desired values of the objective functions of the decision maker. Also assume that the decision maker can change the reference point interactively due to learning or improved understanding during the solution process. When the decision maker specifies the reference point $\bar{\boldsymbol{z}} = (\bar{z}_1, \ldots, \bar{z}_k)$, the corresponding Pareto optimal solution, which is, in the minimax sense, nearest to the reference point or better than that if the reference point is attainable, is obtained by solving the following minimax problem:

$$\left.\begin{array}{l} \min\ \max_{1 \leq i \leq k} \{z_i(\boldsymbol{x}) - \bar{z}_i\} \\ \text{s. t. } A\boldsymbol{x} \leq \boldsymbol{b},\ x_j \in \{0, 1, \ldots, v_j\},\ j = 1, \ldots, n \\ \qquad \sum_{j=1}^{n} \left\{ \sum_{s_i=1}^{S_i} p_{is_i} c_{ijs_i} + (\delta_i - 1)\alpha_{ij} \right\} x_j \\ \qquad\qquad \leq (1 - \delta_i) g_i^0 + \delta_i g_i^1,\ i = 1, \ldots, k. \end{array}\right\} \quad (10)$$

Consequently, we construct an algorithm for obtaining a satisficing solution of a decision maker through interaction is described as follows:

[An interactive satisficing method for fuzzy random multiobjective integer programming problems]

Step 1: Calculate $g_i^{\min}$ and $g_i^{\max}$, $i = 1, \ldots, k$.

Step 2: Ask a decision maker to set g_i^0 and g_i^1 based on the values calculated in step 1,

Step 3: Set the initial reference probability point as $\bar{\boldsymbol{z}} = \boldsymbol{0}$.

Step 4: For the given reference probability levels, solve the minimax problem (10).

Step 5: If the decision maker is satisfied with the current solution $\boldsymbol{x}^c$, then stop. Otherwise, update $\bar{\boldsymbol{z}}$ and return to Step 4.

3.2 Exact solution method for the minimax problem

This section shows an exact solution method for the minimax problem (10). For simplicity, we define N_i and Q_i as

$$z_i(\boldsymbol{x}) - \bar{z}_i = \frac{\sqrt{\boldsymbol{x}^t V_i \boldsymbol{x}} - \bar{z}_i \left(\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0 \right)}{\sum_{j=1}^{n} \alpha_{ij} x_j - g_i^1 + g_i^0} \triangleq \frac{N_i(\boldsymbol{x})}{Q_i(\boldsymbol{x})}.$$

Then, in minimax problem (10), the numerators of objective functions are all convex functions and the denominators are all affine functions. Hence, it follows that all objective functions in (7) are quasi-convex functions. Accordingly, an strict optimal solution of the following continuous relaxation problem of (10) is obtained by using Borde's method [17]:

$$\left.\begin{array}{l} \min\ \max_{1 \leq i \leq k} \{z_i(\boldsymbol{x}) - \bar{z}_i\} \\ \text{s. t. } A\boldsymbol{x} \leq \boldsymbol{b},\ 0 \leq x_j \leq v_j,\ j = 1, \ldots, n \\ \qquad \sum_{j=1}^{n} \left\{ \sum_{s_i=1}^{S_i} p_{is_i} c_{ijs_i} + (\delta_i - 1)\alpha_{ij} \right\} x_j \\ \qquad\qquad \leq (1 - \delta_i) g_i^0 + \delta_i g_i^1,\ i = 1, \ldots, k. \end{array}\right\} \quad (11)$$

The algorithm for solving problem (11) is as follows:

[An algorithm for solving the continuous relaxation problem of the minimax problem]

Step 1: Set $\lambda \leftarrow 0$ and find a feasible solution. Let the solution be $\boldsymbol{x}^\lambda$.

Step 2: Calculate q^λ defined by

$$q^\lambda = \max_{1 \leq i \leq k} \left\{ \frac{N_i(\boldsymbol{x}^\lambda)}{Q_i(\boldsymbol{x}^\lambda)} \right\}$$

and solve the following problem by using Borde's method [17]:

$$\left.\begin{array}{l} \min\ Z \\ \text{s.t. } \frac{1}{Q_i(\boldsymbol{x}^\lambda)} \left\{ N_i(\boldsymbol{x}) - q^\lambda Q_i(\boldsymbol{x}) \right\} \leq Z,\ i = 1, \ldots, k, \\ \qquad \boldsymbol{x} \in \Omega \end{array}\right\} \quad (12)$$

where Ω is a set of feasible solutions in problem (8). Let an optimal solution of (12) be $\boldsymbol{x}^c$. Go to Step 3.

Step 3: If $Z = 0$, then stop. Otherwise, set $\boldsymbol{x}^\lambda \leftarrow \boldsymbol{x}^c$, $\lambda \leftarrow \lambda + 1$ and return to Step 2.

The fact that the continuous relaxation problem of (10) is solved means that problem (10) is solved by the branch-and-bound method. It should be noted that an optimal solution of (10) is at least a weak Pareto optimal solution of (7).

3.3 Tabu Search Method

In the previous section, we have shown an exact solution method for the minimax problem, which is useful for the case of small number of decision variables. However, for a large-scale problem, the proposed method is not practical because the computational time exponentially increases dependent on the number of decision variables. For a large-scale problem, we shall propose a solution algorithm based on a tabu search (TS). A tabu search was first introduced by Glover [13,14] and has been developed as a general solution method for integer (or discrete) programming problems. In order to outline some basic concepts necessary for constructing solution method based on a tabu search, we consider the following problem:

$$\left.\begin{array}{l} \min\ f(x) \\ \text{s. t. } g_w(\boldsymbol{x}) \leq \boldsymbol{b},\ w = 1, \ldots, m \\ \qquad \boldsymbol{x} \in\ X. \end{array}\right\} \tag{13}$$

In (13), the objective function $f(\boldsymbol{x})$ may be linear or nonlinear and $\boldsymbol{x}$ is an n dimensional integer decision variable vector. $g_w(\boldsymbol{x}) \leq \boldsymbol{b}$ are the m constraints which may include linear or nonlinear inequalities.

Basic operation principle of tabu search method depends on neighborhood moves, that proceed from one solution to another at each iteration. During the solution process, some moves are forbidden which are called as *tabu*. Let $\boldsymbol{x}^{now}$ be the current solution at each iteration, and $\boldsymbol{x}^{best}$ the best solution found so far, *iter* the current iteration number, and *tabu*(*iter*) the set of tabu moves at the current iteration. We denote the feasible region of (13) by $X^{'}$. A tabu search method for solving integer programming problems may be expressed as follows.

Step 1: Initialize *iter*=0 and $tabu(iter) = \emptyset$; Select a starting solution $\boldsymbol{x}^{now} \in X^{'}$.

Step 2: Record the current best known solution by setting $\boldsymbol{x}^{best} = \boldsymbol{x}^{now}$ and define $MinCost = f(\boldsymbol{x}^{best})$.

Step 3: Select a list of moves from the neighborhood of $\boldsymbol{x}^{now}$ $(N(\boldsymbol{x}^{now}))$ randomly where $N(\boldsymbol{x}^{now}) \subset X^{'}$ and evaluate each of them.

Step 4: From the list selected in step 3, choose an appropriate move which has the best evaluation and does not belong to *tabu*(*iter*) or which qualifies to be selected as a result of being admissible by aspiration. If the choice criteria employed cannot be satisfied by any member of $N(\boldsymbol{x}^{now})$, or any other termination condition is fulfilled , stop.

Step 5: Increase *iter* by 1 and update *tabu*(*iter*). Reset $\boldsymbol{x}^{now} = \boldsymbol{x}^{next}$, and if $f(\boldsymbol{x}^{now}) < MinCost$, then go to Step 2, otherwise go to step 3.

Neighborhood of $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\}$ consists of changing the current value of one of x_j, $j \in \{1, 2, \ldots, n\}$. Given a solution $\boldsymbol{x}$, the neighborhood $N(\boldsymbol{x})$ can be defined as

$$N(\boldsymbol{x}) = \{change(\boldsymbol{x}, j) \mid j \in \{1, 2, \ldots, n\}\}$$

where

$$change((x_1, \ldots, x_n), j) = (x_1, \ldots, x_{j-1}, \bar{x}_j, x_{j+1}, \ldots, x_n),$$

$x_j \neq \bar{x}_j \in \{1, 2, \ldots, v_j\}$ and $N(\boldsymbol{x}) \subset X^{'}$

The new solutions are selected from the neighborhood of the current solution, if they are not forbidden. To avoid local minimum, a tabu list (*tabu*(*iter*)) is constructed. Initially, tabu list is empty and constructed in consecutive iterations of search. It is updated during the search process by adding the last move to k forbidden moves at each iteration, while the oldest move is removed from the list. Forbidden moves are determined depending on the short and long term memory of the search process.

The short term memory is a recency based memory structure, which determines some restrictions during the generation of the next solution. Such a restrictive mechanism prevents the search process from revisiting a local minimum in short term and decrease the chance of cycling in the long term. The number of iterations of the restrictions being active depends on the parameter called *tabu tenure*, which is the number of iterations a tabu restriction remains in force. The value of *tabu tenure* can be decided according to the restriction being strong or weak. A strong restriction will have a shorter *tabu tenure* than a weak one.

The long term memory is a frequency based memory structure to attain a diversification effect on the search process. During the process, some regions could be visited less than the others. The solutions have been found are recorded in this memory from where the most often used solutions can be known. By using long term memory and having some related restrictions, the frequently visited regions could also be explored.

The next move during the process is the best move from the neighborhood of the current solution. Although the next solution is better than the current one, it could not be determined from the best move, if it is in the tabu list (*tabu*(*iter*)) at the current iteration. Tabu restrictions can be violated under certain circumstances. For example, when a tabu move would result in a solution better than any visited so far, its tabu classification may be overridden. Such conditions are called aspiration criteria. A tabu move is taken if it satisfies the aspiration criteria.

4. Conclusion

In this paper, we have proposed the model to minimize the variances of degrees of possibility for a multiobjective integer programming problem including fuzzy random variable coefficients. After transforming the formulated problem into the deterministic equivalent multiobjective quasi-convex programming problem, we have constructed an interactive satisficing method for fuzzy random multiobjective integer programming problem. Furthermore, we have shown an exact solution algorithm and a tabu search algorithm for solving the minimax problem.

Although we dealt with only a degree of possibility in this paper, we can also consider the model to minimize the variances of degrees of necessity in a similar manner. In future, we will try to consider the models based on other stochastic programming models such as the probability maximization model and the fractile criterion optimization model.

References

1. S.Vajda, *Probabilistic Programming*, Academic Press, 1972.
2. M.Sakawa, *Fuzzy Sets and Interactive Multiobjective Optimization*, Plenum, New York, 1993.
3. M.Inuiguchi, J.Ramik, Possibilistic Linear Programming: A Brief Review of Fuzzy Mathematical Programming and a Comparison with Stochastic Programming in Portfolio Selection Problem, *Fuzzy Sets and Systems*, vol.111, pp. 3–28, 2000.
4. H.Kwakernaak, Fuzzy Random Variable-1: Definitions and Theorems, *Information Sciences*, vol.15, pp. 1–29, 1978.
5. M.L.Puri, D.A.Ralescu, Fuzzy Random Variables, *Journal of Mathematical Analysis and Applications*, vol.114, pp. 409–422, 1986.
6. G.Y.Wang, Q.Zhong, Linear Programming with Fuzzy Random Variable Coefficients, *Fuzzy Sets and Systems*, vol.57, pp. 295–311, 1993 .
7. M.K.Luhandjula, M.M.Gupta, On Fuzzy Stochastic Optimization, *Fuzzy Sets and Systems*, vol.81, pp. 47–55, 1996.
8. H.Katagiri, H.Ishii, Chance Constrained Bottleneck Spanning Tree Problem with Fuzzy Random Edge Costs, *Journal of the Operations Research Society of Japan*, vol.43, pp. 128–137, 2000.
9. H.Katagiri, H.Ishii, Linear Programming Problem with Fuzzy Random Constraint, *Mathematica Japonica*, vol.52, pp. 123–129, 2000.
10. H.Katagiri, M.Sakawa, H.Ishii, Multiobjective Fuzzy Random Linear Programming Using E-model and Possibility Measure, *Proceedings of Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Vancouver, Canada, July 25-28*, pp. 2295–2300, 2001.
11. L.A.Zadeh, Probability Measure of Fuzzy Events, *Journal of Mathematical Analysis and Applications*, vol.23, pp. 421–427.
12. A.Charnes, W.W. Cooper, Deterministic Equivalents for Optimizing and Satisficing under Chance Constraints, *Operations Research*, vol.11, pp. 18-39, 1955.
13. F.Glover, Tabu Search-Part I., *ORSA Journal on Computing*, vol.1, No.3, pp. 190–206, 1989.
14. F.Glover, Tabu Search-Part II., *ORSA Journal on Computing*, vol.2, pp. 4–32, 1990.
15. A.Kaufman, M.M.Gupta, *Introduction to Fuzzy Arithmetic: Theory and Applications*, Van Nostrand Reinhold Company, 1985.
16. A.P.Wierzbicki, *The Use of Reference Objectives in Multiobjective Optimization, in: G. Frande and T. Gal (eds.), Multiple Criteria Decision Making: Theory and Application*, Springer-Verlag, 1980.
17. J.Borde, J.P.Crouzeix, Convergence of a Dinkelbach-type Algorithm in Generalized Fractional Programming, *Zeitschrift fur Operations Research*, vol.31, pp. 31–54, 1987.