INTERACTIVE FUZZY PROGRAMMING FOR MULTIOBJECTIVE TWO-LEVEL NONLINEAR INTEGER PROGRAMMING PROBLEMS THROUGH GENETIC ALGORITHMS

M.A.K. Azad, K. Kato, H. Katagiri, T. Uno and M. Sakawa

Received February 5, 2007

Abstract. In the present paper, we consider multiobjective two-level nonlinear integer programming problems (MOTLNLIPPs) in which the decision maker (DM) at each level controls his own integer decision variables to optimize multiple objective functions. Various approaches for multi-level programming problems could exist according to situations which the DMs are placed in. In this paper, it is assumed that the DMs have motivation to cooperate with each other and they have own fuzzy goals with respect to their multiple objective functions and partial information on their preferences among them. Under the situation, we propose an interactive fuzzy programming technique through genetic algorithms for MOTLNLIPPs to obtain a satisfactory solution for the DMs. Furthermore, the feasibility of the proposed method is shown by applying it to an illustrative numerical example.

1 Introduction In the real world, we often encounter decision making situations involving multiple decision makers. Especially in industrial or governmental decision making situations, those decision makers have different interest and decision priority. Thus, diversity of evaluation have been a matter of great importance to us and therefore decision makers desire to attain several goals simultaneously. A multiobjective multi-level programming problem is one of mathematical optimization models for them. In this paper, we focus on multiobjective two-level nonlinear integer programming problems (MOTLNLIPPs), in which there exist a decision maker (DM) with integer decision variables at the upper level and another decision maker with integer decision variables at the lower level.

Various approaches for multi-level programming problems could exist according to situations which the DMs are placed in [17]. Under the assumption that the DMs do not have motivation to cooperate mutually, a Stackelberg solution is adopted as a reasonable solution for the situation. It is assumed that the decision maker at the upper level (leader) and the decision maker 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 his own objective function is optimized. Then, a solution defined as mentioned above is called a Stackelberg (equilibrium) solution, which has been employed as a solution concept for two-level mathematical programming problems [3, 7, 11, 12, 25].

On the other hand, under the assumption that the DMs have motivation to cooperate mutually, a satisfactory solution for the DMs is adopted as a reasonable solution for the cooperative situation. For obtaining a satisfactory solution, Lai [8] and Shih, Lai and Lee

2003 *Mathematics Subject Classification.* Primary 90C15, 90C59; Secondary 90C10, 90C29.
Key words and phrases. two-level nonlinear integer programming problem, fuzzy programming, multi-objective programming, genetic algorithm.

[24] proposed solution methods based on fuzzy concepts for multi-level linear programming problems such that decisions of DMs in all levels are sequential and all of the DMs essentially cooperate with each other. In their method, the DMs identify membership functions of fuzzy goals for their objective functions, and especially, the DMs at the upper levels also specifies those of fuzzy goals for decision variables. The DM at the lowest level solves a fuzzy programming problem with constraints on fuzzy goals of the DMs at upper levels. Unfortunately, however, there is a possibility that their method leads a final solution to an undesirable one because of inconsistency between fuzzy goals of the objective function and those of the decision variables. To overcome the problem in the method of Shih et al., eliminating the fuzzy goals for the decision variables, Sakawa et al. have proposed interactive fuzzy programming for multi-level linear programming problems [18, 19, 20]. But they considered only linear case. For obtaining a satisfactory solution to two-level nonlinear integer programming problems the authors [1] proposed an interactive fuzzy programming technique through genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) [15, 16].

In order to take the diversification of social requirements or various multiple objectives of the DM into account, theoretical, methodological or applied researches have been done for multiobjective programming problems involving multiple objective functions conflicting with each other in ordinary single level programming problems [4, 9, 10]. Based on these researches with multiple decision makers, Sakawa et al. [21, 22, 23] have proposed interactive fuzzy programming for multiobjective two-level linear programming problems for obtaining a satisfactory solution for the DMs having multiple objective functions.

Decision making situations in the real world are often formulated as large-scale multiobjective two-level nonlinear integer programming problems (MOTLNLIPPs) involving integer decision variables, nonlinear objective functions and nonlinear constraint functions. Since a general solution method does not exist for nonlinear integer programming problems like the branch and bound method for linear ones, a solution method peculiar to each problem has been proposed. As a general-purpose solution method for nonlinear integer programming problems, we propose the usage of genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) [14, 15, 16].

Under these circumstances, in this paper, for obtaining a satisfactory solution in cooperative relationship between the DMs, interactive fuzzy programming through proposed GADSCRRSU is presented for MOTLNLIPPs, assuming that the DMs have fuzzy goals with respect to their multiple objective functions and also have partial information on their preference [4, 9, 10]. In our proposed method, after identifying membership functions of the fuzzy goals of the two DMs, the DM at the upper level subjectively specifies minimal satisfactory levels for all the fuzzy goals and the DM at the lower level also specifies aspiration levels for all the fuzzy goals. During an interactive process in the proposed method, tentative solutions are obtained and evaluated by using the partial information on preferences of the DMs. Taking into account the overall satisfactory balance between the two levels, the two DMs update some of the minimal satisfactory levels and the aspiration levels, if necessary, in order to derive a satisfactory solution. Furthermore, the feasibility of the proposed method is shown through application of it to illustrative numerical example with different numbers of variables.

2 Problem Formulation

Multiobjective two-level nonlinear integer programming problems in which the decision maker at the upper level (DM1) has t_1 objective functions and the decision maker at the lower level (DM2) has t_2 objective functions are generally formulated as follows:

$$
\begin{array}{lll}
 & \underset{\text{DM1}}{\text{minimize}} & f_1^1(\boldsymbol{x}_1,\ \boldsymbol{x}_2) \\
 & \cdots\ \cdots & \cdots\ \cdots \\
 & \underset{\text{DM1}}{\text{minimize}} & f_1^{t_1}(\boldsymbol{x}_1,\ \boldsymbol{x}_2) \\
(1) & \underset{\text{DM2}}{\text{minimize}} & f_2^1(\boldsymbol{x}_1,\ \boldsymbol{x}_2) \\
 & \cdots\ \cdots & \cdots\ \cdots \\
 & \underset{\text{DM2}}{\text{minimize}} & f_2^{t_2}(\boldsymbol{x}_1,\ \boldsymbol{x}_2) \\
 & \text{subject to} & g_i(\boldsymbol{x}_1,\ \boldsymbol{x}_2) \leq 0,\ i = 1, \ldots, m \\
 & & x_{lj} \in \{0, 1, \ldots, \nu_{lj}\},\ l = 1, 2,\ j = 1, \ldots, n_l,
\end{array}
$$

where $\boldsymbol{x}_l$ is an n_l dimensional integer decision variable column vector for the decision maker at each level, $f_l^{k_l}(\boldsymbol{x}_1,\ \boldsymbol{x}_2)$, $k_l = 1, \ldots, t_l$ and $g_i(\boldsymbol{x}_1,\ \boldsymbol{x}_2)$, $i = 1, \ldots, m$ may be linear or nonlinear. For notational convenience, we use $\boldsymbol{x} = (\boldsymbol{x}_1^T,\ \boldsymbol{x}_2^T)^T$ and denote the feasible region of problem (1) by X. Since the DMs have motivation to cooperate with each other, we denote the solution vector as $\boldsymbol{x}$ without partition.

For example, consider project selection problems in an administrative office at the upper level and several autonomous divisions of a company. In this case, the situation that all the DMs can cooperate with each other seems natural rather than one that all the DMs do not have motivation to cooperate mutually.

Under the hypothesis of cooperation between the DMs, Sakawa et al. [21, 22, 23] proposed interactive fuzzy programming for multiobjective two-level linear programming problems in order to derive satisfactory solutions for the DMs through interactions with the DM at the upper level by introducing fuzzy goals to consider the imprecise nature of DMs' judgments for objective functions.

In this paper, for multiobjective two-level nonlinear integer programming problems, focusing on the case of cooperative relation between the DMs, we present a new interactive fuzzy programming method through genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) in order to derive a satisfactory solution for the DMs.

3 Interactive Fuzzy Programming

In this section, we describe an interactive fuzzy programming method through genetic algorithms based on literatures by Sakawa et al. [21, 22, 23] is summarized as follows.

3.1 Interactive Fuzzy Programming

Considering the ambiguity or fuzziness of the decision makers' judgments on each of the objective functions $f_l^{k_l}(\boldsymbol{x})$ in (1), it seems natural to introduce such fuzzy goals for objective functions as "$f_l^{k_l}(\boldsymbol{x})$ should be subjectively less than or equal to a certain value". In order to identifying such fuzzy goals, first we solve problems to obtain the individual minimum

$$(2) \qquad f_l^{k_l,\min} = \min_{\boldsymbol{x} \in X}\ f_l^{k_l}(\boldsymbol{x}),\ l = 1,\ 2,\ k_l = 1, \ldots, t_l$$

and the individual maximum

$$(3) \qquad f_l^{k_l,\max} = \max_{\boldsymbol{x} \in X}\ f_l^{k_l}(\boldsymbol{x}),\ l = 1,\ 2,\ k_l = 1, \ldots, t_l$$

of each of the objective functions which are referred to when the DMs elicit membership functions prescribing the fuzzy goals for the objective functions $f_l^{k_l}(\boldsymbol{x})$, $l = 1, 2$. Since these problems are single-objective nonlinear integer programming problems and it is difficult to obtain optimal solutions to them, we use genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) which is an extension of genetic algorithms with double strings based on reference solution updating (GADSRSU) for linear 0-1 programming problems [16].

The DMs determine the membership functions $\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}))$, $l = 1, 2$ which are strictly monotone decreasing for $f_l^{k_l}(\boldsymbol{x})$, consulting the variation ratio of degree of satisfaction in the interval between the individual minimum of problem (2) and the individual maximum of problem (3). The domain of the membership functions is in the interval $[f_l^{k_l,\min}, f_l^{k_l,\max}]$, $l = 1, 2, k_l = 1, \ldots, t_l$ and the DM specifies subjectively the value $f_l^{k_l,0}$ of the objective function for which the degree of satisfaction is 0 and the value $f_l^{k_l,1}$ of the objective function for which the degree of satisfaction is 1. For the value undesired (larger) than $f_l^{k_l,0}$, it is defined that $\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x})) = 0$, and for the value desired (smaller) than $f_l^{k_l,1}$, it is defined that $\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x})) = 1$. Here a linear membership function in Figure 1 is considered, which characterizes the fuzzy goal of the DM at each level. The corresponding linear membership function $\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}))$ is defined as:

$$\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x})) = \begin{cases} 1 & f_l^{k_l}(\boldsymbol{x}) < f_l^{k_l,1} \\ \dfrac{f_l^{k_l}(\boldsymbol{x}) - f_l^{k_l,0}}{f_l^{k_l,1} - f_l^{k_l,0}} & f_l^{k_l,1} \le f_l^{k_l}(\boldsymbol{x}) < f_l^{k_l,0} \\ 0 & f_l^{k_l}(\boldsymbol{x}) \ge f_l^{k_l,0} \end{cases} \tag{4}$$

It is assumed that the DMs subjectively specify $f_l^{k_l,0}$ and $f_l^{k_l,1}$.

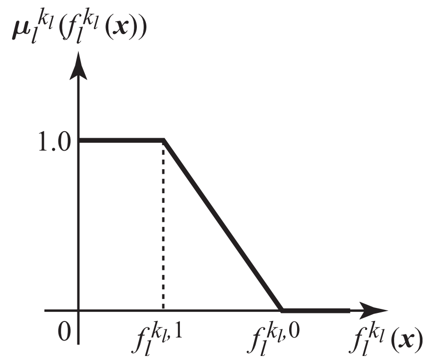

Figure 1: Linear Membership Function

Zimmermann [26] proposed a method for determining the parameters $f_l^{k_l,0}$ and $f_l^{k_l,1}$ of the linear membership function in the following way. That is, using the individual minimum, they are defined as

$$f_l^{k_l,1} = f_l^{k_l,\min} = f_l^{k_l}(\boldsymbol{x}^{lk_l0}) = \min_{\boldsymbol{x}\in X} \; f_l^{k_l}(\boldsymbol{x}),\ l = 1,\ 2,\ k_l = 1, \ldots, t_l \tag{5}$$

together with

$$f_l^{k_l,0} = \max_{l=1,\,2,\,k_l=1,\ldots,t_l} \left\{ f_l^{k_l}(\boldsymbol{x}^{lk_l0}) \right\}. \tag{6}$$

We assume that each DM evaluates a solution $\boldsymbol{x}$ by taking the weighted sum of all of the membership functions, and the aggregated membership function of DMl is represented as

$$\sum_{k_l=1}^{t_l} \omega_l^{k_l} \mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x})), \tag{7}$$

where $\boldsymbol{\omega}_l = (\omega_l^1, \dots, \omega_l^{t_l})$ denotes a weighting coefficient vector satisfying

$$\boldsymbol{\omega}_l \in \left\{ \boldsymbol{\omega}_l \in R^{t_l} \;\middle|\; \sum_{k_l=1}^{t_l} \omega_l^{k_l} = 1,\ \omega_l^{k_l} \geq 0,\ k_l = 1, \dots, t_l \right\}. \tag{8}$$

Moreover, we assume that each DM cannot identify the weighting coefficients precisely and has some partial information on his preference [4, 9, 10]. Suppose that such partial information can be represented by the following two inequalities with respect to DMl:

$$LB_l^{k_l} \leq \omega_l^{k_l} \leq UB_l^{k_l}, \tag{9}$$

$$\omega_l^p \geq \omega_l^q + \epsilon,\ p \neq q, \tag{10}$$

where ϵ is a small nonnegative constant. The upper bound $UB_l^{k_l}$ and the lower bound $LB_l^{k_l}$ are specified for the weight $\omega_l^{k_l}$ to the membership function $\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}))$ of the fuzzy goal for the k_lth objective function like the condition (9). The condition (10) represents an order relation between the pth fuzzy goal and the qth one. Let Ω_l denote a set of weighting coefficient vectors $\boldsymbol{\omega}_l = (\omega_l^1, \dots, \omega_l^{t_l})$ of DMl satisfying the conditions (9) and (10) as well as the condition (8). For example, suppose that DMl has two objectives and thinks μ_l^1 is more important than μ_l^2 but there does not exist a great difference between them. Then DMl could specify the partial information of preference like $\omega_l^2 \geq 0.4$ and $\omega_l^1 \geq \omega_l^2$. As a result, ω_l^1 and ω_l^2 are restricted as $0.5 \leq \omega_l^1 \leq 0.6$ and $0.4 \leq \omega_l^2 \leq 0.5$, respectively.

Having elicited membership functions $\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}))$ for $f_l^{k_l}(\boldsymbol{x})$, $l = 1, 2$ and the partial information on preference about each of the objective functions by the DM at each level, then the original multiobjective two-level nonlinear integer programming problems (1) can be interpreted as a multiobjective two-level membership maximization problem defined by:

$$\begin{array}{rl}
\underset{\text{DM1}}{\text{maximize}} & \mu_1^1(f_1^1(\boldsymbol{x})) \\
\cdots\ \cdots & \cdots\ \cdots \\
\underset{\text{DM1}}{\text{maximize}} & \mu_1^{t_1}(f_1^{t_1}(\boldsymbol{x})) \\
\underset{\text{DM2}}{\text{maximize}} & \mu_2^1(f_2^1(\boldsymbol{x})) \\
\cdots\ \cdots & \cdots\ \cdots \\
\underset{\text{DM2}}{\text{maximize}} & \mu_2^{t_2}(f_2^{t_2}(\boldsymbol{x})) \\
\text{subject to} & g_i(\boldsymbol{x}) \leq 0,\ i = 1, \dots, m \\
& x_j \in \{0, 1, \dots, \nu_j\},\ j = 1, \dots, n
\end{array} \tag{11}$$

Since (11) is a multiobjective two-level membership maximization problem, in general, a complete optimal solution that simultaneously maximizes all the DMs' degree of satisfaction of their objective functions does not always exist when the objective functions conflict with each other. Thus, a satisfactory solution is expected to be obtained from among M-Pareto optimal solution set which is defined for multiobjective programming problems [13, 15, 16].

For deriving an overall satisfactory solution to the formulated problem (11), first the maximizing decision of the fuzzy decision proposed by Bellman and Zadeh [2] is found.

Namely, the following problem is solved for obtaining a solution which maximizes the smallest degree of satisfaction among satisfactory degrees for all of the fuzzy goals of the two DMs:

$$\begin{array}{lll} & \text{maximize} & \min_{l=1,\,2,\,k_l=1,\ldots,t_l} \left\{ \mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x})) \right\} \\ (12) & \text{subject to} & g_i(\boldsymbol{x}) \leq 0,\ i=1,\ldots,m \\ & & x_j \in \{0,1,\ldots,\nu_j\},\ j=1,\ldots,n \end{array}$$

This problem can also be solved by GADSCRRSU.

Let us denote an optimal solution to the problem (12) by $\boldsymbol{x}^*$ and if DM1 is satisfied with this solution, it follows that the optimal solution $\boldsymbol{x}^*$ becomes a satisfactory solution. However, DM1 is not always satisfied with the optimal solution $\boldsymbol{x}^*$. It is quite natural to assume that DM1 expects satisfactory degrees for the membership functions $\mu_1^{k_1}$, $k_1 = 1,\ldots,t_1$ larger than certain minimal satisfactory levels $\hat{\delta}_1^{k_1} \in [0,1]$, $k_1 = 1,\ldots,t_1$, and DM2 also holds certain aspiration levels $\bar{\mu}_2^{k_2}$, $k_2 = 1,\ldots,t_2$ to values of the membership functions $\mu_2^{k_2}$, $k_2 = 1,\ldots,t_2$. To specify the minimal satisfactory levels $\hat{\delta}_1^{k_1}$ and the aspiration levels $\bar{\mu}_2^{k_2}$, it seems reasonable for DM1 and DM2 to consult the optimal solution to the maxmin problem (12) and the related information.

Consequently, if DM1 is not satisfied with the solution $\boldsymbol{x}^*$ to problem (12), then DM1 specifies minimal satisfactory levels $\hat{\delta}_1^{k_1}$ for his membership of all objective functions and also DM2 specifies his aspiration levels $\bar{\mu}_2^{k_2}$ for his membership of all objective functions, then the following minmax problem is formulated:

$$\begin{array}{lll} & \text{minimize} & \max_{k_2=1,\ldots,t_2} \left\{ \bar{\mu}_2^{k_2} - \mu_2^{k_2}(f_2^{k_2}(\boldsymbol{x})) \right\} \\ (13) & \text{subject to} & \mu_1^{k_1}(f_1^{k_1}(\boldsymbol{x})) \geq \hat{\delta}_1^{k_1},\ k_1=1,\ldots,t_1 \\ & & g_i(\boldsymbol{x}) \leq 0,\ i=1,\ldots,m \\ & & x_j \in \{0,1,\ldots,\nu_j\},\ j=1,\ldots,n \end{array}$$

In problem (13), the distance between a vector of the membership values of DM2 and that of the aspiration levels is minimized under the conditions that the membership values of DM1 are larger than or equal to the minimal satisfactory levels specified by DM1. After obtaining a solution to problem (13), on the preference of self, DM2 updates the membership values $\bar{\mu}_2^{k_2}$ representing the aspiration levels and searches a satisfactory solution of self, if necessary.

If an optimal solution to problem (13) exists, it follows that DM1 obtains a satisfactory solution having satisfactory degrees larger that or equal to the minimal satisfactory levels specified by DM1. However, the larger the minimal satisfactory levels $\hat{\delta}_1^{k_1}$ are assessed, the smaller the DM2's satisfactory degrees become. Consequently, a relative difference between the aggregated satisfactory degrees of DM1 and DM2 becomes larger and it cannot be anticipated that the obtained solution becomes a satisfactory solution balancing the aggregated satisfactory degree of DM1 and that of DM2.

To obtain a satisfactory solution acceptable for both DMs, we must evaluate a candidate for the satisfactory solution. By using the aggregated membership functions with weighting coefficients, an optimal solution $\boldsymbol{x}^*$ to problem (13) is evaluated. Because possible weighting coefficients vectors $\boldsymbol{\omega}_l$ belong to Ω_l, the minimum and the maximum of the aggregated membership functions with weighting coefficients of DMl with respect to $\boldsymbol{x}^*$ can be represented by:

$$S_l^{\min} = \min_{\boldsymbol{\omega}_l \in \Omega_l} \sum_{k_l=1}^{t_l} \omega_l^{k_l} \mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}^*)),\ l = 1,\ 2, \tag{14}$$

$$S_l^{\max} = \max_{\boldsymbol{\omega}_l \in \Omega_l} \sum_{k_l=1}^{t_l} \omega_l^{k_l} \mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}^*)),\ l = 1,\ 2. \tag{15}$$

Using the two values $S_l^{\min}$ and $S_l^{\max}$, we define an aggregated satisfactory degree of DMl with respect to $\boldsymbol{x}^*$ as an L-L fuzzy number [5] defined as

$$\tilde{S}_l = (u_l,\ v_l)_{LL}. \tag{16}$$

where

$$u_l = \frac{S_l^{\max} + S_l^{\min}}{2},$$
$$v_l = \frac{S_l^{\max} - S_l^{\min}}{2}.$$

The L-L fuzzy number $\tilde{S}_l$ is represented by the following membership function:

$$\mu_{\tilde{S}_l}(p) = \begin{cases} L\left(\dfrac{S_l^{\max} + S_l^{\min} - 2p}{S_l^{\max} - S_l^{\min}}\right) & \text{if } p \leq (S_l^{\max} + S_l^{\min})/2, \\ L\left(\dfrac{2p - S_l^{\max} - S_l^{\min}}{S_l^{\max} - S_l^{\min}}\right) & \text{if } p > (S_l^{\max} + S_l^{\min})/2, \end{cases} \tag{17}$$

where $L(p) = \max(0,\ 1 - |p|)$. The fuzzy number representing the satisfactory degree of DMl is shown in Figure 2. $S_l^{\min}$ and $S_l^{\max}$ are values for which the satisfaction degrees are 0 and $(S_l^{\min} + S_l^{\max})/2$ is a value for which the satisfaction is 1.

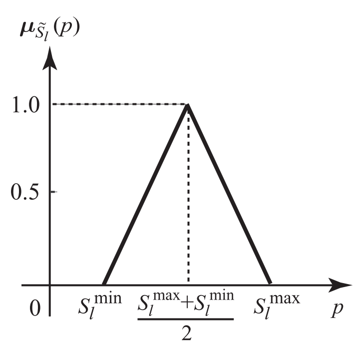

Figure 2: Satisfactory degree $\tilde{S}_l$ of DMl

In order to take account of the overall satisfactory balance between both levels, we define a ratio of satisfactory degrees between both DMs as a quotient of the two L-L fuzzy numbers [5].

$$\tilde{S}_2 \oslash \tilde{S}_1 \cong \tilde{\Delta} = \left(\frac{u_2}{u_1},\ \frac{v_2 u_1 + v_1 u_2}{(u_1)^2}\right)_{LL}. \tag{18}$$

Let $\mu_{\tilde{\Delta}}(p)$ denote a membership function of the ratio $\tilde{\Delta}$ of satisfactory degrees between both DMs.

DM1 is guaranteed to have satisfactory degrees larger than or equal to the minimal satisfactory levels for all of the fuzzy goals because the corresponding constraints are involved in problem (13). To take into account the overall satisfactory balance between both levels as well as the satisfactory degrees of self, it is assumed that DM1 has a fuzzy goal $\tilde{R}$ for the ratio $\tilde{\Delta}$ of satisfactory degrees. The fuzzy goal $\tilde{R}$ is expressed in words such as "the ratio $\tilde{\Delta}$ should be in the vicinity of a certain value q".

We introduce a termination condition that the maxmin value of the ratio $\tilde{\Delta}$ of satisfactory degrees and its fuzzy goal $\tilde{R}$ is larger than or equal to the permissible level $\hat{\delta}_{\tilde{\Delta}}$, i.e.,

$$\alpha \triangleq \max_{p} \min\{\mu_{\tilde{\Delta}}(p),\ \mu_{\tilde{R}}(p)\} \geq \hat{\delta}_{\tilde{\Delta}}, \tag{19}$$

where $\mu_{\tilde{R}}(p)$ denotes a membership function of the fuzzy goal $\tilde{R}$. When the termination condition is not satisfied, or DM1 judges that it is desirable for self to increase his satisfactory degree at the sacrifice of that of DM2 or the reverse is true, DM1 must update some or all of the minimal satisfactory levels.

3.2 Algorithm of the Interactive Fuzzy Programming through GADSCRRSU

We are now ready to present an interactive algorithm for deriving an overall satisfactory solution to multiobjective two-level nonlinear integer programming problems (1) through genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU), which is summarized in the following.

Step 1: Ask the two DMs about partial information of their preference of each objective function.

Step 2: Solve (5) through GADSCRRSU for individual minimum and by using (6) calculate $f_l^{k_l,0}$ for each objective function of all the DMs and ask the DMs to identify their membership functions $\mu_l^{k_l}(f_l^{k_l}(\cdot))$, $l = 1,\ 2,\ k_l = 1, \ldots, t_l$ of the fuzzy goals for their own objective functions.

Step 3: Solve (12) through GADSCRSSU. If DM1 is satisfied with an obtained solution, the solution becomes a satisfactory solution. Otherwise, ask DM1 to specify the minimal satisfactory levels $\hat{\delta}_1^{k_1}$, $k_1 = 1, \ldots, t_1$ by considering the current satisfaction degrees, and also ask DM2 to specify the aspiration levels $\bar{\mu}_2^{k_2}$, $k_2 = 1, \ldots, t_2$. Moreover, elicit the membership $\mu_{\tilde{R}}(p)$ of the fuzzy goal for the ratio of satisfactory degrees from DM1 and ask DM1 to specify the permissible level $\hat{\delta}_{\tilde{\Delta}}$.

Step 4: Solve (13) through GADSCRRSU with the minimal satisfactory levels and aspiration levels. If DM2 is satisfied with an obtained solution, go to step 6. Otherwise, go to step 5.

Step 5: Ask DM2 to update his aspiration levels and return to step 4.

Step 6: The satisfactory degrees $\tilde{S}_l$, $l = 1,\ 2$ and the ratio of satisfactory degrees $\tilde{\Delta}$ corresponding an optimal solution to (13) are shown to DM1. If the solution shown to DM1 satisfies the termination condition and DM1 concludes the solution as a satisfactory solution, the algorithms stops. Otherwise, go to step 7.

Step 7: Ask DM1 to update some of the minimal satisfactory levels by consulting the optimal solution to (13) and the related information. Return to step 4.

4 Genetic Algorithms with Double Strings using Continuous Relaxation based on Reference Solution Updating (GADSCRRSU) Genetic algorithms were initiated by Holland, his colleagues, and his students at the University of Michigan in the 1970s as stochastic search procedures based on the mechanism of natural selection and natural genetics. It should be noticed that genetic algorithms have received much attention as a promising approximate computational method for large-scale optimization problems. Generally in genetic algorithms, an individual is usually represented by binary 0-1 strings. For solving constrained mathematical programming problems through genetic algorithms the most straightforward technique is to transform the constrained problem into an unconstrained problem by penalizing infeasible solutions. The fitness function is defined for preventing to generate infeasible solutions by imposing penalties on individuals that violate the constraints. It is generally recognized that the smaller the feasible region, the harder it is for the penalty function methods to generate feasible solutions, as pointed out in the field of nonlinear optimization. Sakawa et al. [15, 16] proposed genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) for multidimensional integer problems.

In this section, we mention GADSCRRSU proposed as a general solution method for nonlinear integer programming problems defined as

$$\begin{array}{rl} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & g_i(\boldsymbol{x}) \leq 0,\ i = 1, \dots, m \\ & x_j \in \{0, 1, \dots, \nu_j\},\ j = 1, \dots, n. \end{array} \tag{20}$$

In the problem (20), $\boldsymbol{x}$ is an n dimensional integer decision variable vector, $f(\boldsymbol{x})$, $g_i(\boldsymbol{x})$, $i = 1, \dots, m$ are nonlinear functions and ν_j, $j = 1, \dots, n$ is the upper bound of each decision variable.

4.1 Individual Representation The individual representation [15, 16] by double strings shown in Figure 3 is adopted in GADSCRRSU. In the figure, each of $s(j)$, $j =$

Indices	$s(1)$	$s(2)$	$\cdots$	$s(j)$	$\cdots$	$s(n)$	$= \boldsymbol{s}$
Values	$y_{s(1)}$	$y_{s(2)}$	$\cdots$	$y_{s(j)}$	$\cdots$	$y_{s(n)}$	

Figure 3: Double Strings Representation

$1, \dots, n$ is the index of an element in a solution vector and each of $y_{s(j)} \in \{0, 1, \dots, \nu_j\}$, $j = 1, \dots, n$ is the value of the element, respectively.

4.2 Decoding Algorithm Let N be the total number of population (*pop_size*). The individuals $\boldsymbol{s}$ with the dimensions n are generated randomly. Unfortunately, however, the direct mapping of the individual can generate infeasible solutions [15, 16]. To eliminate such solutions, a decoding algorithm of double strings for nonlinear integer programming problems (20) using a reference solution $\boldsymbol{x}^0$, which is a feasible solution and used as the origin of decoding, is constructed as follows.

Decoding algorithm using reference solution:

In the algorithm, it is assumed that a feasible solution $\boldsymbol{x}^0$ is obtained in advance. Let n, and N be the number of variables and number of individuals in the population, respectively.

Step 1: If the index of an individual to be decoded is in $\{1, \ldots, \lfloor N/2 \rfloor\}$, go to step 2. Otherwise, go to step 8.

Step 2: Let $j := 1$, $\boldsymbol{x} := \{0, \ldots, 0\}$, $l := 1$.

Step 3: Let $x_{s(j)} := y_{s(j)}$.

Step 4: If $g_i(\boldsymbol{x}) \leq 0$, $i = 1, \ldots, m$, let $l := j$, $j := j + 1$, and go to step 5. Otherwise, let $j := j + 1$, and go to step 5.

Step 5: If $j \leq n$, go to step 3. Otherwise, go to step 6.

Step 6: If $l > 0$, go to step 7. Otherwise, go to step 8.

Step 7: By substituting $x_{s(j)} := y_{s(j)}$, $1 \leq j \leq l$ and $x_{s(j)} := 0$, $l < j \leq n$, we obtain a feasible solution $\boldsymbol{x}$ corresponding to the individual $\boldsymbol{s}$ and stop.

Step 8: Let $j := 1$, $\boldsymbol{x} := \boldsymbol{x}^0$.

Step 9: Let $x_{s(j)} := y_{s(j)}$. If $y_{s(j)} = x^0_{s(j)}$, let $j := j+1$, and go to step 11. If $y_{s(j)} \neq x^0_{s(j)}$, go to step 10.

Step 10: If $g_i(\boldsymbol{x}) \leq 0$, $i = 1, \ldots, m$, let $j := j + 1$, and go to step 11. Otherwise, let $\boldsymbol{x}_{s(j)} := \boldsymbol{x}^0_{s(j)}$, $j := j + 1$, and go to step 11.

Step 11: If $j \leq n$, go to step 9. Otherwise, we obtain a feasible solution $\boldsymbol{x}$ corresponding to the individual $\boldsymbol{s}$ and stop.

This decoding algorithm enables us to decode each of the individuals represented by the double strings to the corresponding feasible solution. However, the diversity of the solution $\boldsymbol{x}$ greatly depends on the reference solution, because solutions obtained by the decoding algorithm using reference solution tend to concentrate around the reference solution. To overcome such situations, the reference solution updating procedure [15, 16] is adopted here.

4.3 Fitness Nature obeys the principle of Darwinian "survival of the fittest", the individuals with high fitness values will, on average, reproduce more often than those low fitness values. For obtaining satisfactory solution for the DMs to multiobjective two-level nonlinear integer programming problems (1) through GADSCRRSU, the objective function value is used as the fitness value f of an individual $\boldsymbol{s}$. When the variance of fitness in a population is small, it is often observed that the ordinary roulette wheel selection does not work well because there is little difference between the probability of a good individual surviving and that of a bad one surviving [15, 16]. In order to overcome this problem, the linear scaling [15, 16] is adopted here. The new fitness $f'_l(\boldsymbol{s})$, $l = 1, 2$ of the DMl is obtained by using the following linear scaling

$$f'_l(\boldsymbol{s}) := a_l f_l(\boldsymbol{s}) + b_l \tag{21}$$

where $f_l(\boldsymbol{s})$, $l = 1, 2$ are the fitness values of the DMs at all levels with respect to each decoded individual $\boldsymbol{s}$.

4.4 Genetic Operators For obtaining satisfactory solution for the DMs to multiobjective two-level nonlinear integer programming problems (1) through GADSCRRSU, four genetic operators such as reproduction, partially matched crossover (PMX), bit reverse mutation and inversion [15, 16] are adopted here.

4.5 Usage of Continuous Relaxation In order to find an approximate optimal solution with high accuracy in reasonable time, we need some schemes such as the restriction of the search space to a promising region, the generation of individuals near the optimal solution and so forth. From the point of view, the information about an optimal solution to the corresponding continuous relaxation problem

$$\begin{array}{rl} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & g_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m \\ & 0 \leq x_j \leq \nu_j,\ j = 1, \ldots, n \end{array} \tag{22}$$

is used in the generation of the initial population and the bit reverse mutation. When this problem is convex, we can obtain a global optimal solution by some convex programming technique, e.g., the sequential quadratic programming. Otherwise, i.e., when it is nonconvex, because it is difficult to find a global optimal solution, we search an approximate optimal solution by some approximate solution method such as genetic algorithms or simulated annealing. Here GENOCOP V [6] is used to find the solution of corresponding continuous relaxation problem (22).

4.6 Computational Procedures of GADSCRRSU Now the genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) for solving nonlinear integer programming problems (20) are summarized in the following.

Step 0: Determine values of the parameters used in GADSCRRSU: the population size N, the minimal search generation $I_{\min}$, the maximal search generation $I_{\max} > I_{\min}$, the convergence criterion ϵ, the degree of use of information about solutions to nonlinear programming relaxation problem R, the parameter for feasible solution θ, the parameter for reference solution updating η, the upper bound of each decision variable ν, the scaling constant c_{mult}, the probability of crossover p_c, the generation gap G, the probability of mutation p_m, the probability of inversion p_i and set the generation counter r at 0.

Step 1: Generate the initial population consisting of N individuals based on the information of a solution to the continuous relaxation problem (22).

Step 2: Decode each individual (genotype) in the current population and calculate its fitness based on the corresponding solution (phenotype).

Step 3: If the termination condition is fulfilled, stop. Otherwise, let $r := r + 1$ and go to step 4.

Step 4: Apply the reproduction operator based on the elitist expected value selection, after carrying out linear scaling.

Step 5: Apply the crossover operator, called PMX (Partially Matched Crossover) for double strings.

Step 6: Apply the mutation operator based on the information of an optimal solution to the continuous relaxation problem (22).

Step 7: Apply the inversion operator. Go to step 2.

5 Numerical Example Here, we consider the following multiobjective two-level nonlinear integer programming problem in order to test the proposed algorithm.

$$
\begin{aligned}
&\underset{\text{DM1}}{\text{maximize}} && f_1^1(\boldsymbol{x}) = \prod_{j=1}^{n} \left[1 - (1 - r_j)^{x_j}\right] \\
&\underset{\text{DM1}}{\text{minimize}} && f_1^2(\boldsymbol{x}) = \sum_{j=1}^{n} c_j x_j \\
&\underset{\text{DM2}}{\text{minimize}} && f_2^1(\boldsymbol{x}) = \sum_{j=1}^{n} d_j \left[x_j + \exp\left(\frac{x_j}{4}\right)\right] \\
&\underset{\text{DM2}}{\text{minimize}} && f_2^2(\boldsymbol{x}) = \sum_{j=1}^{n} q_j x_j \, \exp\left(\frac{x_j}{4}\right) \\
&\text{subject to} && g_1(\boldsymbol{x}) = \sum_{j=1}^{n} p_j x_j^2 - P \leq 0 \\
& && g_2(\boldsymbol{x}) = \sum_{j=1}^{n} w_j x_j \, \exp\left(\frac{x_j}{4}\right) - W \leq 0 \\
& && x_j \in \{1, 2, \ldots, 10\}, \; j = 1, \ldots, n
\end{aligned} \tag{23}
$$

The numerical experiments were performed on a personal computer (processor: Intel 1 GHz, memory: 512 MB, OS: Windows 2000) using Visual C/C++ compiler (version 6.0). The parameter values used in GADSCRRSU for solving (23) were set as follows: $N = 100$, $I_{\min} = 100$, $I_{\max} = 1000$, $\epsilon = 0.005$, $R = 0.9$, $\theta = 5.0$, $\eta = 0.1$, $\sigma = 2.0$, $\tau = 3.0$, $\nu = 10$, $c_{\text{mult}} = 1.8$, $p_c = 0.9$, $G = 0.9$, $p_m = 0.05$, $p_i = 0.03$, and $P = 200$. Several problems with different numbers of variables were considered to test the proposed algorithm for solving (23). The data were generated randomly. In the following section, the result has been discussed briefly when $n = 15$ and the data are shown in Table 1.

Table 1: Data for problem (23) with 15 variables

	Values of elements of coefficient vectors														
$\boldsymbol{r}$	0.659	0.573	0.574	0.603	0.634	0.710	0.543	0.593	0.922	0.656	0.786	0.639	0.621	0.703	0.776
$\boldsymbol{c}$	4.900	17.700	12.500	20.900	17.200	26.600	25.700	26.600	10.000	33.800	16.900	17.600	28.900	12.800	27.100
$\boldsymbol{d}$	8.650	4.980	3.340	2.820	9.480	9.330	3.420	8.480	0.890	5.720	1.110	6.710	8.070	7.370	9.590
$\boldsymbol{q}$	4.540	1.240	4.500	4.290	0.830	2.090	4.450	4.890	1.560	2.380	0.640	3.340	2.270	4.070	1.320
$\boldsymbol{p}$	3.000	1.000	10.000	5.000	5.000	9.000	3.000	8.000	4.000	10.000	5.000	1.000	10.000	3.000	8.000
$\boldsymbol{w}$	0.385	0.672	0.934	0.254	0.892	0.284	0.347	0.578	0.627	0.570	0.505	0.850	0.389	0.912	0.284
P	1781.250														
W	414.871														

5.1 Result and Discussion First, in step 1, suppose that DM1 and DM2 specify the partial information of preference as follows

$$
\begin{aligned}
\Omega_1 &\equiv \{\omega_1 \in R^2 | \omega_1^1 \geq \omega_1^2, \, \omega_1^2 \geq 0.2, \, \omega_1^1 + \omega_1^2 = 1, \, \omega_1^1 \geq 0, \, \omega_1^2 \geq 0\}, \\
\Omega_2 &\equiv \{\omega_2 \in R^2 | \omega_2^2 \geq \omega_2^1, \, \omega_2^1 \geq 0.2, \, \omega_2^1 + \omega_2^2 = 1, \, \omega_2^1 \geq 0, \, \omega_2^2 \geq 0\}.
\end{aligned}
$$

In step 2, the individual minimum and maximum of each objective function of both the DMs are calculated by using Zimmermann method and are shown in Table 2. After calculating individual minimum and maximum of each objective function of both the DMs, the corresponding linear membership functions are specified subjectively by the DMs.

Table 2: Calculated individual minimum and maximum

Decision Maker	Objective	$f_l^{k_l,\,\min}$	$f_l^{k_l,\,\max}$	Time (Sec.)
DM1	f_1^1	0.0019	0.8985	34.68
	f_1^2	299.2000	1456.7000	
DM2	f_2^1	205.4709	770.6370	
	f_2^2	54.4555	835.4906	

Since DM1's first objective function is a maximization type, the linear membership function in equation (24) in Figure 4 is used to specify the fuzzy goal of the DM1's 1st objective function.

$$\mu_1^1(f_1^1(\boldsymbol{x})) = \begin{cases} 0 & f_1^1(\boldsymbol{x}) < f_1^{1,0} \\ \dfrac{f_1^1(\boldsymbol{x}) - f_1^{1,0}}{f_1^{1,1} - f_1^{1,0}} & f_1^{1,0} \leq f_1^1(\boldsymbol{x}) < f_1^{1,1} \\ 1 & f_1^1(\boldsymbol{x}) \geq f_1^{1,1} \end{cases} \tag{24}$$

On the other hand, since DM1's second objective and DM2's objective functions are mini-

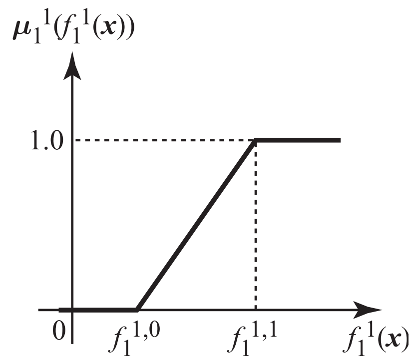

Figure 4: Linear membership function for DM1's 1st objective

mization type, the linear membership function in equation (4) in Figure 1 is used to specify the fuzzy goal of the corresponding objective function. The corresponding linear membership functions of both the DMs for their objective functions are shown in Table 3.

After specifying the linear membership functions, in step 3, in order to obtain a satisfactory solution to the multiobjective two-level nonlinear integer programming problem (23), the maxmin problem (12) is formulated and solved through GADSCRRSU, where the smallest degree of satisfaction among satisfactory degrees for all of the fuzzy goals of both DMs is maximized and the results are shown in Table (4) and the solution is proposed to DM1.

Suppose that DM1 at the upper level is not satisfied with the obtained solution and specifies the minimal satisfactory levels for the membership functions $\mu_1^{k_1}(f_1^{k_1}(\boldsymbol{x}))$, $k_1 = 1, 2$

Table 3: Parameters for linear membership functions

Decision Maker	Objective	$f_l^{k_l,0}$	$f_l^{k_l,1}$
DM1	f_1^1	0.10	0.85
	f_1^2	1400.00	350.00
DM2	f_2^1	675.00	250.00
	f_2^2	550.00	100.00

Table 4: Optimal values to the maxmin problem (Iteration 1)

Decision Maker	Objective	$f_l^{k_l}(\boldsymbol{x}^*)$	$\mu_l^{k_l}(f_l^{k_l}(\boldsymbol{x}^*))$	Time(sec.)
DM1	f_1^1	0.4858	0.5144	
	f_1^2	860.1000	0.5142	6.77
DM2	f_2^1	454.2847	0.5193	
	f_2^2	313.2190	0.5262	

at $\hat{\delta}_1^1 = 0.60$ and $\hat{\delta}_1^2 = 0.40$ by consulting the smallest satisfactory degree of both levels and the partial information Ω_1, and DM2 at the lower level also sets the aspiration levels to the membership functions $\mu_2^{k_2}(f_2^{k_2}(\boldsymbol{x}))$, $k_2 = 1, 2$ at $\bar{\mu}_2^1 = 0.60$ and $\bar{\mu}_2^2 = 0.80$. Moreover, suppose that DM1 thinks the ratio $\tilde{\Delta}$ should be in the vicinity of about 0.9 and identifies the membership function of the fuzzy goal $\tilde{R}$ for the ratio $\tilde{\Delta}$ of satisfactory degrees as

$$\mu_{\tilde{R}}(p) = \begin{cases} \max\{0,\ 10p - 8\}, & p < 0.9 \\ \max\{0,\ -10p + 10\}, & p \geq 0.9. \end{cases} \tag{25}$$

The fuzzy goal $\tilde{R}$ corresponds to the fuzzy number $(0.9,\ 0.1)_{LL}$, $L(p) = \max\{0,\ 1 - |p|\}$. Suppose that DM1 determines the permissible level $\hat{\delta}_{\tilde{\delta}}$ at 0.85, and then the termination condition is represented as $\alpha \triangleq \max_p \min\{\mu_{\tilde{\Delta}}(p),\ \mu_{\tilde{R}}(p)\} \geq \hat{\delta}_{\tilde{\Delta}} = 0.85$.

Then, in step 4, the minmax problem (13) is formulated and solved through GAD-SCRRSU and the results are shown in Table (5). Suppose that the DM2 is not satisfied

Table 5: Iteration 2

Decision Maker	Objective	$f_l^{k_l}(\boldsymbol{x}^*)$	$\mu_l^{k_l}(f(\boldsymbol{x}^*))$	Time(sec.)
DM1	f_1^1	0.5507	0.6009	
	f_1^2	951.4000	0.4272	2.80
DM2	f_2^1	495.0954	0.4233	
	f_2^2	284.2084	0.5906	

with the obtained solution from iteration 2 and he raises his first aspiration level to 0.7 and the the minmax problem (13 is again formulated and solved, and the results with other related information are shown in Table (6).

Table 6: Iteration 3

	DM1	DM2
Individual satisfactory degree	$f_1^1 = 0.5550$ $\mu_1^1 = 0.6067$ $f_1^2 = 948.7000$ $\mu_1^2 = 0.4298$	$f_2^1 = 481.2942$ $\mu_2^1 = 0.4558$ $f_2^2 = 298.4707$ $\mu_2^2 = 0.5590$
$(\boldsymbol{\omega}_1^{\min}, \boldsymbol{\omega}_2^{\min})$	$\omega_1^{1\,\min} = 0.5000$ $\omega_1^{2\,\min} = 0.5000$	$\omega_2^{1\,\min} = 0.5000$ $\omega_2^{2\,\min} = 0.5000$
Minimum of the weighted membership value	$S_1^{\min} = 0.5183$	$S_2^{\min} = 0.5074$
$(\boldsymbol{\omega}_1^{\max}, \boldsymbol{\omega}_2^{\max})$	$\omega_1^{1\,\max} = 0.8000$ $\omega_1^{2\,\max} = 0.2000$	$\omega_2^{1\,\max} = 0.2000$ $\omega_2^{2\,\max} = 0.8000$
Maximum of the weighted membership value	$S_1^{\max} = 0.5713$	$S_2^{\max} = 0.5383$
Aggregated satisfactory degree	$(0.5448,\ 0.0265)_{LL}$	$(0.5228,\ 0.0255)_{LL}$
Ratio of satisfactory degrees maxmin value α	$(0.9597,\ 0.0752)_{LL}$ $\alpha = 0.6592$	
Time (sec.)	5.96	

In step 6, the obtained solution and related information from iteration 3 are proposed to DM1. But the maxmin value α between the ratio of satisfactory degrees and the fuzzy goal for the ratio is 0.6592 and it is smaller than the permissible level $\hat{\delta}_{\tilde{\Delta}} = 0.85$. Therefore, DM1 must update some of the minimal satisfactory levels. Suppose that DM1 raises the second minimal satisfactory level to 0.45 and the revised minmax problem (13) is solved and the results and the related information are shown in Table (7). In this iteration the value of α is 0.8285 and it is still smaller than the permissible level. So the DM1 must update the minimal satisfactory levels. Suppose DM1 raises the first minimal satisfactory level to 0.62 and the revised minmax problem (13) is solved and the results and the related information are shown in Table (8). In this iteration the value of α is 0.8835 and it is greater than the permissible level and also the memberships of DM1 are greater than the minimal satisfactory levels. So the obtained solution becomes satisfactory for both the DMs and the interactive process is terminated.

Through the application of the proposed method to this test problem (23), we could find one of satisfactory solutions. Furthermore, we applied the proposed method to (23) with different numbers of variables. In all applications, we could also find one of satisfactory solutions. Figure 5 shows the relation between the number of variables and the total computational time. From the figure, it seems that the computational time increases polynomially as the number of variables increases. These results indicate the feasibility and efficiency of the proposed interactive fuzzy programming for MOTLNLIPPs.

6 Conclusion In this paper, focusing on multiobjective two-level nonlinear integer programming problems, an interactive fuzzy programming procedure for them through genetic algorithms with double strings using continuous relaxation based on reference solution updating (GADSCRRSU) is presented. In the proposed method, the decision maker at the upper level subjectively specifies the minimal satisfactory levels for his all of the fuzzy goals and the decision maker at the lower level also specifies the aspiration levels for his all of the

Table 7: Iteration 4

	DM1	DM2
Individual satisfactory degree	$f_1^1 = 0.5522$ $\mu_1^1 = 0.6029$ $f_1^2 = 925.5000$ $\mu_1^2 = 0.4519$	$f_2^1 = 484.1852$ $\mu_2^1 = 0.4490$ $f_2^2 = 305.3659$ $\mu_2^2 = 0.5436$
$(\boldsymbol{\omega}_1^{\min}, \boldsymbol{\omega}_2^{\min})$	$\omega_1^{1\min} = 0.5000$ $\omega_1^{2\min} = 0.5000$	$\omega_2^{1\min} = 0.5000$ $\omega_2^{2\min} = 0.5000$
Minimum of the weighted membership value	$S_1^{\min} = 0.5274$	$S_2^{\min} = 0.4963$
$(\boldsymbol{\omega}_1^{\max}, \boldsymbol{\omega}_2^{\max})$	$\omega_1^{1\max} = 0.8000$ $\omega_1^{2\max} = 0.2000$	$\omega_2^{1\max} = 0.2000$ $\omega_2^{2\max} = 0.8000$
Maximum of the weighted membership value	$S_1^{\max} = 0.5727$	$S_2^{\max} = 0.5247$
Aggregated satisfactory degree	$(0.5500,\ 0.0226)_{LL}$	$(0.5105,\ 0.0142)_{LL}$
Ratio of satisfactory degrees maxmin value α	$(0.9281,\ 0.0640)_{LL}$ $\alpha = 0.8285$	
Time (sec.)	3.58	

Table 8: Iteration 5

	DM1	DM2
Individual satisfactory degree	$f_1^1 = 0.5653$ $\mu_1^1 = 0.6204$ $f_1^2 = 926.5000$ $\mu_1^2 = 0.4510$	$f_2^1 = 506.4666$ $\mu_2^1 = 0.4295$ $f_2^2 = 358.8285$ $\mu_2^2 = 0.5288$
$(\boldsymbol{\omega}_1^{\min}, \boldsymbol{\omega}_2^{\min})$	$\omega_1^{1\min} = 0.5000$ $\omega_1^{2\min} = 0.5000$	$\omega_2^{1\min} = 0.5000$ $\omega_2^{2\min} = 0.5000$
Minimum of the weighted membership value	$S_1^{\min} = 0.5357$	$S_2^{\min} = 0.4792$
$(\boldsymbol{\omega}_1^{\max}, \boldsymbol{\omega}_2^{\max})$	$\omega_1^{1\max} = 0.8000$ $\omega_1^{2\max} = 0.2000$	$\omega_2^{1\max} = 0.2000$ $\omega_2^{2\max} = 0.8000$
Maximum of the weighted membership value	$S_1^{\max} = 0.5865$	$S_2^{\max} = 0.5090$
Aggregated satisfactory degree	$(0.5611,\ 0.0254)_{LL}$	$(0.4941,\ 0.0149)_{LL}$
Ratio of satisfactory degrees maxmin value α	$(0.8806,\ 0.0664)_{LL}$ $\alpha = 0.8835$	
Time (sec.)	2.07	

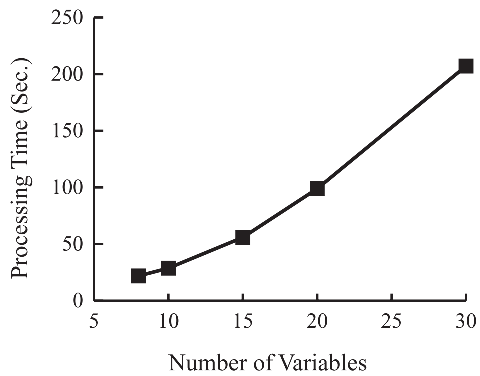

Figure 5: Total Computational Time

fuzzy goals according to the partial information of their preferences. Taking into account the overall satisfactory balance between the two decision makers, we have derived a satisfactory solution by interactive fuzzy programming and obtained the total computational time for various size problems.

References

[1] M.A.K. Azad, M. Sakawa, K. Kato and H. Katagiri, Interactive fuzzy programming for two-level nonlinear integer programming problems through genetic algorithms, Asia Pacific Management Review: An International Journal, vol. 10, no. 1, pp. 70-77, 2005.

[2] R.E. Bellman and L.A. Zadeh, Decision making in a fuzzy environment, Management Science, vol. 17, no. 4, pp. 141-164, 1970.

[3] W.F. Bialas and M.H. Karwan, Two-level linear programming, Management Science, vol. 30, no. 8, pp. 1004-1020, 1984.

[4] E. Carrizosa, E. Conde, F.R. Fernandez and J. Puerto, Multi-criteria analysis with partial information about the weighting coefficients, European Journal of Operational Research, vol. 81, no. 2, pp. 291-301, 1995.

[5] D. Dubois and H. Prade, Fuzzy Sets and Systems: Theory and Applications, Academic Press, New York, USA, 1980.

[6] GENOCOP V, ftp://ftp.uncc.edu/coe/evol/zbyszek/genocop5.0.tar.Z

[7] R. Jan and M. Chern, Nonlinear integer bilevel programming, European Journal of Operational Research, vol. 72, no. 3, pp. 574-587, 1994.

[8] Y.J. Lai, Hierarchical optimization: a satisfactory solution, Fuzzy Sets and Systems, vol. 77, no. 3, pp. 321-335, 1996.

[9] B. Malakooti, Identifying nondominated alternatives with partial information for multiple-objective discrete and linear programming problems, IEEE Transactions on Systems, Man, and Cybernatics, vol. SMC-19, no. 1, pp. 95-107, 1989.

[10] A.M. Marmol, J. Puerto and F.R. Fernandez, The use of partial information on weights in multicriteria decision problems, Journal of Multi-criteria Decision Analysis, vol. 7, no. 6, pp. 322-329, 1998.

[11] I. Nishizaki, M. Sakawa and K. Kato, Computational methods through genetic algorithms for obtaining Stackelberg solutions to two-level zero-one programming problems, Proceedings of 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation, pp. 2750-2755, 2000.

[12] I. Nishizaki and M. Sakawa, Computational methods through genetic algorithms for obtaining Stackelberg solutions to two-level mixed zero-one programming problems, Cybernetics and Systems: An International Journal, vol. 31, no. 2, pp. 203-221, 2000.

[13] M. Sakawa, Fuzzy Sets and Interactive Multiobjective Optimization, Plenum Press, New York, USA, 1993.

[14] M. Sakawa, Large Scale Interactive Fuzzy Multiobjective Programming, Physica-Verlag, Heidelberg, 2000.

[15] M. Sakawa, Genetic Algorithms and Fuzzy Multiobjective Optimization, Kluwer Academic Publishers, Boston, USA, 2002.

[16] M. Sakawa and K. Kato, Genetic algorithms with double strings for 0-1 programming problems, European Journal of Operational Research, vol. 144, no. 3, pp. 581-597, 2003.

[17] M. Sakawa and I. Nishizaki, Cooperative and Noncooperative Multi-Level Programming, Springer, New York, 2009.

[18] M. Sakawa, I. Nishizaki and Y. Uemura, Interactive fuzzy programming for multi-level linear programming problems, Computers and Mathematics with Applications, vol. 36, no. 2, pp. 71-86, 1998.

[19] M. Sakawa, I. Nishizaki and Y. Uemura, Interactive fuzzy programming for multi-level linear programming problems with fuzzy parameters, Fuzzy Sets and Systems, vol. 109, no. 1, pp. 3-19, 2000.

[20] M. Sakawa, I. Nishizaki and M. Hitaka, Interactive fuzzy programming for multi-level 0-1 programming problems through genetic algorithms, European Journal of Operational Research, vol. 114, no. 3, pp. 580-588, 1999.

[21] M. Sakawa, I. Nishizaki and Y. Oka, Interactive fuzzy programming for multiobjective two-level linear programming problems with partial information of preference, International Journal of Fuzzy Systems, vol. 2, no. 2, pp. 79-86, 2000.

[22] M. Sakawa, I. Nishizaki and Y.X. Shen, Interactive fuzzy programming for multiobjective two-level linear fractional programming problems with partial information of preference, International Journal of Fuzzy Systems, vol. 3, no. 3, pp. 452-461, 2001.

[23] Y.X. Shen, M. Sakawa, K. Kato and I. Nishizaki, Interactive fuzzy programming using partial information about preference through genetic algorithms for multiobjective two-level 0-1 programming problems, Proceedings of the 5th Czech-Japan Seminar on Data Analysis and Decision Making under Uncertainity, pp. 59-64, 2002.

[24] H.S. Shih, Y.J. Lai and E.S. Lee, Fuzzy approach for multi-level programming problems, Computers and Operations Research, vol. 23, no. 1, pp. 73-91, 1996.

[25] K. Shimizu, Y. Ishizuka and J.F. Bard, Nondifferentiable and Two-Level Mathematical Programming, Kluwer Academic Publishers, Boston, USA, 1997.

[26] H.-J. Zimmermann, Fuzzy programming and linear programming with several objective functions, Fuzzy Sets and Systems, vol. 1, vol. 1, pp. 45-55, 1978.

Md. Abul Kalam Azad
The R & D Centre ALGORITMI, University of Minho
Campus Azurem, 4800 Guimaraes, PORTUGAL
E-mail: akazad@dps.uminho.pt