

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSI*net*

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Formulation and Modeling Approaches for Piecewise Linear Membership Functions in Fuzzy Nonlinear Programming

Bo Wen

School of Information Science and Technology, Beijing University of Chemical Technology,
No. 15, North Third Road East, Beijing, 100029, China

Abstract: Attractive issues related with membership functions in fuzzy nonlinear programming have been discussed in depth in this study. Initially, an approach to constructing unilateral and bilateral membership functions is introduced to deal with soft constraints. In order to facilitate solution of fuzzy nonlinear programming problems, a procedure towards linearization of membership functions is presented, along with a novel modeling methodology to formulate fuzzy nonlinear programming with piecewise linear membership functions. Therein, taking advantage of compatible configuration of binary variables, relations between the sub-membership-functions are explicitly characterized, which could efficiently help prevent constraint-free problems and achieve satisfied solutions. Finally, a numerical example is employed to demonstrate the benefits of the contribution.

Key words: Fuzzy programming, unilateral and bilateral membership functions, piecewise linear membership functions, binary items

INTRODUCTION

At present, the crisp optimization problem already has mature mathematical programming methodologies but these approaches become powerless when facing variety of sources of uncertainty in industrial applications. Alternatively, since fuzzy decision and fuzzy linear programming theories were founded by Bellman and Zadeh (1970) and Zimmermann (1978), respectively, fuzzy programming has been extensively studied and applied to dealing with optimization problems with uncertainty (Zimmermann, 1987; Roubens and Teghem, 1991; Shi and Yu, 1992; Liu and Shi, 1994).

Membership functions are usually employed to represent Decision-Makers (DMs)' understanding and acceptance of uncertainty as well as subjective preferences when dealing with soft constraints in fuzzy programming problems. Generally, membership functions can be trapezoidal or triangular which are easily configured but rarely well accommodated with human's subjective activities. Instead, Gaussian functions are acknowledged to be excellent alternatives in this issue. However, nonlinear expressions of Gaussian membership functions readily lead to huge difficulties in solving mathematical programming which possibly result in intractable problems such as achievement of local optima.

Piecewise Linear Functions (PLFs) were firstly introduced by Garvin *et al.* (1957), which have been used in the areas of drilling, production, manufacturing, marketing and distribution. Subsequently, Stone (1961) introduced construction methods of PLFs. After that, PLFs have been studied extensively, even being used to solve large-scale engineering problems in recent years (Wen and Ma, 2008; Effati and Abbasiyan, 2010; Toriello and Vielma, 2012; Sun *et al.*, 2012). Since, the pioneer work of Narasimhan (1980) and Hannan (1981), Piecewise Linear Membership Functions (PLMFs) have been introduced into the fuzzy programming and decision-making community. Depending on the number of sub-segments assigned, PLFs could approximate any nonlinear functions in expected fidelity. Additionally, too much attention on local description could be avoided in simulating human's satisfaction, which helps make achievement of solutions easier and far from local convergence of fuzzy programming. Nonetheless, PMLFs have found less popularity in applications. For instance, Chang (2010) presented an approach to build PMLFs through selecting the maximum error, which, however, over-relies on computations instead of DM's preferences.

Due to the special form, the modeling methodologies of fuzzy programming with PMLF are the problem which researchers interesting. There are three existing main methodologies: The first one is the binary variable

representation which proposed by Yang and Ignizio (1991), they use binary variables to distinguish the different sub-segments of PLMFs and lay a basis for this methodology; the second one is proposed by Hu and Fang (1999), they converted the problems to a one constraint programming by employing the concepts of constraint surrogation and maximum entropy; the last one is proposed by Li and Yu (2000), they distinguish different sub-segments by changing the symbol which controlled by the absolute values. In the three methodologies, the second one is too complicated to be used in real life and the third one needs adding extra binary variables which will increasing the computational burden. Due to the drawbacks, only the first one had been studied extensively.

After the proposed of Yang and Ignizio (1991), Li and Yu (1999) showed that the method of Yang and Ignizio (1991) demanded extra binary variables in the face of complex PLMFs, thereby solving this problem by adding extra satisfaction degrees. Lin and Chen (2002) suggested that the relationship between sub-segments could be expressed by intersections and unions, so that extending the method of Yang and Ignizio (1991) to handle fuzzy programming problem with complex PLMFs. They also pointed out three shortcomings of Li and Yu's (1999) method, i.e., no generalized procedure or algorithms available, in need of extra binary variables sometimes and actual aspiration level unachievable directly. Besides, Chang (2007) employed a couple of binary numbers to represent the relationship among binary variables, which became the first generalized algorithm for fuzzy programming with PLMFs. However, in some cases, the combination of binary variables would even lead to constraint-free problems in solving.

The contribution of this study is twofold. First, an approach to manage discrete points of Gaussian membership functions and construct PLMFs according to the reference points selected by DMs is explicitly introduced. Second, a novel method to distribute binary variables in formulating fuzzy programming with PLMFs is suggested, which could not only avoid traditional demerits aforementioned but also make problem solving easier by utilization of few binary items in some situations.

FORMULATION OF NONLINEAR FUZZY MEMBERSHIP FUNCTIONS

Practically, a mathematical programming problem may involve "soft constraints" (Narasimhan, 1980) which refer to the constraints allowing admissible ranges of violations, such as:

$$\begin{aligned} & Z_{\max} \\ \text{s.t. } & g_i(x_i) \leq b_i \\ & h_j(x_j) = b_j \end{aligned} \tag{1}$$

where, membership functions μ_z , μ_g and μ_h associated with the goal and constraints are suggested to measure a DM's satisfaction degree λ , characterized by $\lambda(x_i) = \mu_z(x_i) \cap \mu_g(x_i) \cap \mu_h(x_i)$. Therefore, we have:

$$\begin{aligned} \text{Max}_x \lambda(x_i) &= \text{Max}_x \text{Min} \{ \mu_z(x_i), \mu_g(x_i), \mu_h(x_i) \} \\ &= \text{Min} \{ \mu_z(x_i^*), \mu_g(x_i^*), \mu_h(x_i^*) \} \end{aligned} \tag{2}$$

Usually, two kinds of soft constraints are available for a mathematical programming problem, inequality and equality. For an equality constraint, the membership decreases whatever the left-hand side is higher or lower than the limit, which demands a bilateral membership function; while for an inequality constraint, a unilateral membership function could be employed, because the membership decreases when the left-hand side goes beyond the limits. Considering the inequality constraints $g_i(x_i) \leq b_i$ in problem (1), the satisfaction degree achieve 1 when g_i become less than or equal to b_i , while it will descend when g_i are greater than b_i . As depicted in Fig. 1, where the maximum relaxation of the right-hand side is supposed to be $b_i + \delta_i$, unilateral membership functions are obviously accommodated. As for the equality constraints, $h_j(x_j) = b_j$, the satisfaction becomes 1 only when h_j are equal to b_j ; it descend whether h_j deviate to the left or right. In this sense, as shown in Fig. 2, bilateral membership functions are available where the maximum relaxation of left and right sides are supposed to be $b_j - \delta_j$ and $b_j + \delta_j$, respectively.

Traditionally, unilateral and bilateral membership functions can be expressed by sub-section functions, such as:

$$\mu_{g_i} = \begin{cases} 1 & g_i \leq b_i \\ f(g_i) & b_i \leq g_i \leq b_i + \delta_i \\ 0 & g_i \geq b_i + \delta_i \end{cases} \tag{3}$$

and:

$$\mu_{h_j} = \begin{cases} f_1(h_j) & b_j - \delta_j \leq h_j \leq b_j \\ f_2(h_j) & b_j \leq h_j \leq b_j + \delta_j \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Which are apparently susceptible to expressional and computational difficulties relatively. In response, we introduce an approach to build continuous unilateral and bilateral membership functions in a novel way.

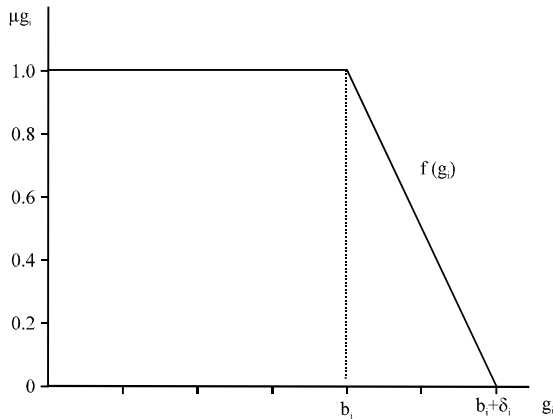


Fig. 1: Bilateral membership function

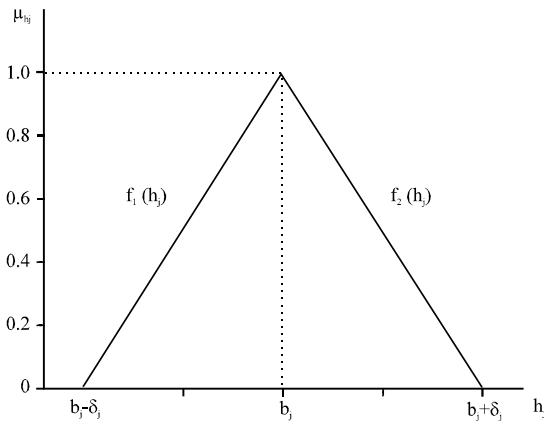


Fig. 2: Unilateral membership function

Suppose that a_i indicates the expectation of k_i . In the presence of $k_i \leq a_i$ there is $\mu_{k_i} = f_1(k_i)$, otherwise $\mu_{k_i} = f_2(k_i)$. A function $y(k_i)$ specified to change with k_i on different sides a_i . Here, we normalize $k_i - a_i$ to the interval $(-1, 1)$ by:

$$\frac{\arctan(k_i - a_i)}{\pi / 2}$$

In so doing, the bilateral membership functions can be formulated as follow:

$$y(k_i) = \left(\left[\frac{\arctan(k_i - a_i)}{\pi / 2} \right] + 1 \right) * f_2(k_i) - \left[\frac{\arctan(k_i - a_i)}{\pi / 2} \right] * f_1(k_i) \quad (5)$$

where, $[\bullet]$ denotes the Gaussian rounding functions, which can be characterized by:

$$\begin{cases} [m] = 0 & 0 \leq m < 1 \\ [m] = -1 & -1 \leq m < 0 \end{cases} \quad (6)$$

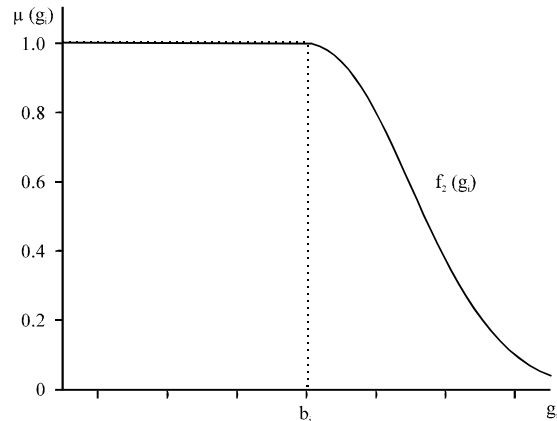


Fig. 3: Unilateral Gaussian membership function

And $f_1(k_i)$ and $f_2(k_i)$ are the functions corresponding to left and right sides of point a_i , respectively. If $k_i \leq a_i$, then there is $\mu_{k_i} = f_1(k_i)$; if $a_i < k_i$, then there is $\mu_{k_i} = f_2(k_i)$. Therefore, the asymmetric unilateral and bilateral membership functions have been established consequently.

It is acknowledged that Gaussian functions serve best imitations to human's subjective preferences as well as take advantage of nonlinearity, smoothness and continuous differentiates, which have been extensively applied in intelligent technologies such as neural networks and fuzzy expert systems (Hannan, 1981; Chang, 2010).

Gaussian membership functions are characterized by:

$$g(x) = e^{-\frac{(x-a)^2}{2p^2}}$$

where, a and p are the center point and width factor, respectively. Therefore, the unilateral and bilateral Gaussian membership functions of the constraints corresponding to problem (1) can be specified as follows:

$$\mu(g_i) = \left(\left[\frac{\arctan(g_i - b_i)}{\pi / 2} \right] + 1 \right) * e^{-\frac{(g_i - b_i)^2}{2p_i^2}} - \left[\frac{\arctan(g_i - b_i)}{\pi / 2} \right] * 1 \quad (7)$$

$$\mu(h_j) = \left(\left[\frac{\arctan(h_j - b_j)}{\pi / 2} \right] + 1 \right) * e^{-\frac{(h_j - b_j)^2}{2p_j^2}} - \left[\frac{\arctan(h_j - b_j)}{\pi / 2} \right] * e^{-\frac{(h_j - b_j)^2}{2p_j^2}} \quad (8)$$

Accordingly, Fig. 3 and 4 show visualization of unilateral and bilateral Gaussian membership functions.

PIECEWISE LINEAR FUZZY MEMBERSHIP FUNCTIONS

Despite that Gaussian fuzzy membership functions could simulate human's preferences in high fidelity, the

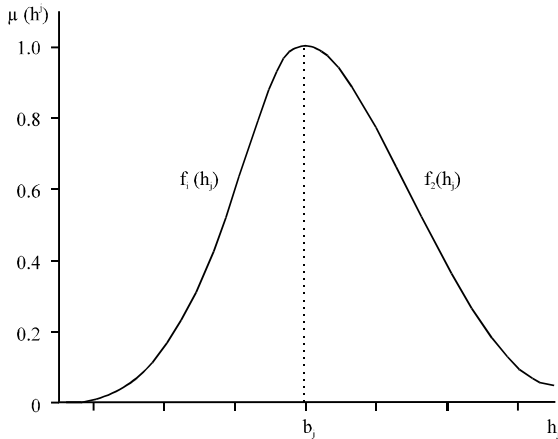


Fig. 4: Bilateral gaussian membership function

inherent characteristics of nonlinearity would lead to computational complexity of the corresponding optimization problems. Motivated by this fact, the nonlinear membership functions are suggested to be segmented into piecewise linear ones. As an example shown in Fig. 5, a basic S-shaped PLMF is characterized by three parameters, μ_{11} , μ_{12} and μ_{13} , whose expressions are presented as follows:

$$\mu_{11}(z_1(x)) = \begin{cases} 1 & \text{if } z_1(x) \geq p_{18} \\ 1 - \frac{p_{18} - z_1(x)}{p_{18} - p_{12}} & \text{if } p_{12} \leq z_1(x) \leq p_{18} \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

$$\mu_{12}(z_1(x)) = \begin{cases} 1 & \text{if } z_1(x) \geq p_{16} \\ 1 - \frac{p_{16} - z_1(x)}{p_{16} - p_{13}} & \text{if } p_{13} \leq z_1(x) \leq p_{16} \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

$$\mu_{13}(z_1(x)) = \begin{cases} 1 & \text{if } z_1(x) \geq p_{17} \\ 1 - \frac{p_{17} - z_1(x)}{p_{17} - p_{11}} & \text{if } p_{11} \leq z_1(x) \leq p_{17} \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

The convex and concave points between sub-segments can be approximated by unions and intersections, respectively, thereby resulting in $\mu_i = \mu_{i1} \cup (\mu_{i2} \cap \mu_{i3})$. It is conceivable that the reference points between sub-segments such as the convex point k_1 and concave point k_2 depicted in Fig. 5 are decisive to the PLMFs. Therefore, the most important step to build PLMFs is to find the appropriate reference points.

We provide a general procedure towards constructing PLMFs as follows. Firstly, calculate the discrete points of integral membership functions using

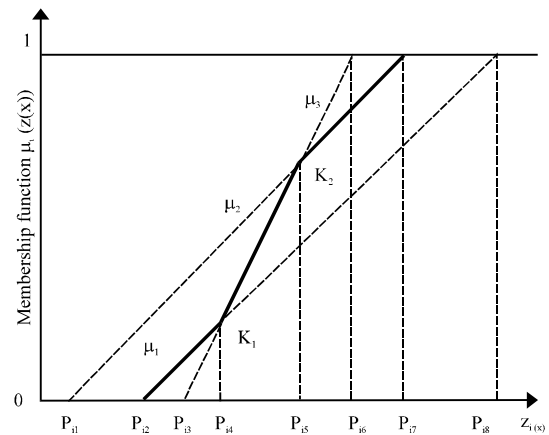


Fig. 5: A S-shaped piecewise linear membership function

Eq. 5. Secondly, choose reference points according to DM's preferences. Finally, build the PLMFs by connecting the key points with straight lines. It is practical to consider that the approximating accuracy of PLFs to the originals relies on the number of sub-segments. Fewer sub-segments may lead to easy calculation but low accuracy, while more sub-segments may result in complicated calculation but high accuracy. To compromise the trade-off relations between accuracy and calculation, DMs are encouraged to consider following suggestions:

- Carefully choose the key points and the relevance, such as boundary points, extreme points and inflection points, etc
- In the presence of linear parts involved in the original membership functions, specify the two endpoints of the lines as key points
- More key points should be selected for the constraints sensitive to the optimum solutions
- Consider fewer key points with small membership values because they are rarely concerned with in practice

FORMULATION OF FUZZY PROGRAMMING

Problem statement: Consider the PLMF described in Fig. 5. As mentioned above, we have $\mu_1 = \mu_{11} \cup (\mu_{12} \cap \mu_{13})$ in which “ \cup ” and “ \cap ” indicate union and intersection operators, respectively. Taking advantage of the binary tree as presented in Fig. 6, intersection and union, respectively, can be identified as a “parallel” relation that does not need to be distinguished and an “or” relation that demands extra binary variables for mathematical representation.

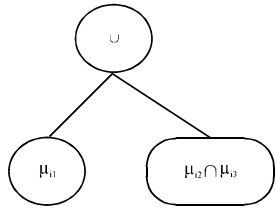


Fig. 6: A binary tree representation

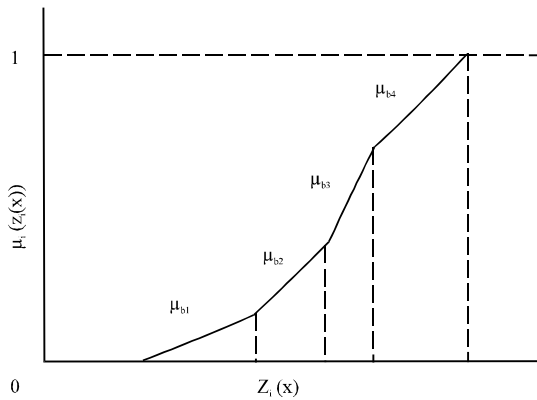


Fig. 7: Piecewise linear membership functions

Accordingly, in an attempt to achieve the greatest of satisfaction, the statement of corresponding fuzzy programming problem could be addressed as follow (Yang and Ignizio, 1991):

$$\begin{aligned} & \lambda_{\max} \\ \text{s.t. } & \lambda \leq \frac{a_{18} - z_1(x)}{a_{18} - a_{12}} + M(1 - \delta_1), \lambda \leq \frac{a_{16} - z_1(x)}{a_{16} - a_{13}} + M\delta_1, \\ & \lambda \leq \frac{a_{17} - z_1(x)}{a_{17} - a_{11}} + M\delta_1, \delta_1 = 0, 1, i = 1, 2, \dots, i \end{aligned} \quad (12)$$

where, λ indicates the satisfaction degree, M is a big positive number and δ_i is an additional binary variable.

We should mention that it is less sufficient to use only one binary variable for modeling in the presence of more than one convex points available for a PLMF. Taking the membership function shown in Fig. 7 as an example, two convex points and one concave point available implies two unions and one intersection are involved, demanding two more binary variables for modeling. Accordingly, the fuzzy optimization problem can be formulated as follows (Lin and Chen, 2002):

$$\begin{aligned} & \lambda_{\max} \\ \text{s.t. } & \lambda \leq \mu_{b1} + \delta_1 M + \delta_2 M, \lambda \leq \mu_{b2} + (1 - \delta_1) M \\ & \lambda \leq \mu_{b3} + \delta_1 M, \lambda \leq \mu_{b4} + \delta_1 M, \\ & \delta_1, \delta_2 = 0, 1, X \in F \end{aligned} \quad (13)$$

where, δ_1 and δ_2 are additional binary variables. It is clear that more binary variables are demanded in the case of more complex PLMFs.

Approaches: We would like to introduce several related definitions before getting to the approaches:

- **Definition 1:** δ_i or $(1 - \delta_i)$ is called a binary item, $\delta_i \in \{0, 1\}$
- **Definition 2:** A binary item group contains all binary-items used to describe a piecewise linear constraint
- **Definition 3:** Binary item groups expressed in a form of matrix refers to a binary item matrix, in which, a row corresponds to a binary item group, implying that the numbers of matrix rows and columns equal to those of the sub-segments and binary items, respectively

A crucial step of formulating fuzzy programming with PLMFs is to associate each sub-segment with a constraint, highlighting the assignment of the binary item groups required. In what follows, two approaches are introduced to facilitate the treatment of PLMFs. The first suggests a simple procedure to accommodate the binary item matrix, while the second presents a general formula to deal with the problems.

Considering that two sub-segments connected by a concave point can be described by one binary item and have same upper bound, only PLMFs with convex points are concerned with so as to facilitate the discussion.

Approach 1: The main idea behind this approach is, initially, to establish a binary item matrix based on the numbers of both sub-segments and binary items required, followed by deleting some useless items then, eventually resulting in the binary item groups responsible for describing each sub-segments of the PLMFs considered. Finally, we could formulate the soft constraints through the binary item matrix. In general, the detailed procedures can be addressed as follows:

- Step 1:** Compute θ , the number of binary items required, by $\theta = \lceil \ln n / \ln 2 \rceil$, where n indicates the number of concave points associated with a PLMF
- Step 2:** Establish the binary item matrix according to θ and the number of sub-segments q ($q = n + 1$) by applying the following rules: In the 1st column δ_1 and $(1 - \delta_1)$ are written alternately every 1 row; in the 2nd column, δ_2 and $(1 - \delta_2)$ are

Table 1: Binary item matrix

Variables	1	2	3	4	...	θ
1	δ_1	δ_2	δ_3	δ_4	...	δ_θ
2	$1-\delta_1$	δ_2	δ_3	δ_4	...	δ_θ
3	δ_1	$1-\delta_2$	δ_3	δ_4	...	δ_θ
4	$1-\delta_1$	$1-\delta_2$	δ_3	δ_4	...	δ_θ
5	δ_1	δ_2	$1-\delta_3$	δ_4	...	δ_θ
6	$1-\delta_1$	δ_2	$1-\delta_3$	δ_4	...	δ_θ
7	δ_1	$1-\delta_2$	$1-\delta_3$	δ_4	...	δ_θ
8	$1-\delta_1$	$1-\delta_2$	$1-\delta_3$	δ_4	...	δ_θ
9	δ_1	δ_2	δ_3	$1-\delta_4$...	δ_θ
10	$1-\delta_1$	δ_2	δ_3	$1-\delta_4$...	δ_θ
11	δ_1	$1-\delta_2$	δ_3	$1-\delta_4$...	δ_θ
...
$2^{\theta-1}$	δ_θ
$2^{\theta-1}+1$	$1-\delta_\theta$
...
Q	$1-\delta_\theta$

Table 2: An intermediate binary item matrix

Variables	1	2	3
1	δ_1	δ_2	δ_3
2	$1-\delta_1$	δ_2	δ_3
3	δ_1	$1-\delta_2$	δ_3
4	$1-\delta_1$	$1-\delta_2$	δ_3
5	δ_1	δ_2	$1-\delta_3$

Table 3: Final binary item matrix

Variables	1	2	3
1	δ_1	δ_2	δ_3
2	$1-\delta_1$	δ_2	δ_3
3	δ_1	$1-\delta_2$	
4	$1-\delta_1$	$1-\delta_2$	
5	δ_1	δ_2	$1-\delta_3$

written alternately every 2 rows; in the 3rd column, δ_3 and $(1-\delta_3)$ are written alternately every 4 rows; ..., in the θ^{th} column, δ_θ and $(1-\delta_\theta)$ are written alternately every $2^{\theta-1}$ rows, ultimately resulting in a binary item matrix with q rows and θ columns, as shown in Table 1

Step 3: Delete the binary items located at row $q-2^{\theta-1}+1$ to $2^{\theta-1}$ in the last column

Step 4: Formulate each sub-segment as a constraints associated with the satisfaction degree λ , in which, each binary item in a row of the matrix is multiplied with a big positive number in turn and then added on the corresponding membership function

For instance, a procedure towards dealing with a PLMF with 4 concave points is given as follows:

Step 1: The number of binary items required is measured by $\theta = \lceil \ln 4 / \ln 2 \rceil = 3$

Step 2: Taking account of 5 sub-segments involved, we build a binary item matrix with 5 rows and 3 columns, as shown in Table 2

Step 3: Rows 2 to 4 of the last column were deleted, eventually attaining the matrix as shown in Table 3

Step 4: The statement of fuzzy programming is then addressed as follows:

$$\begin{aligned}
 &\lambda_{\max} \\
 \text{s.t. } &\lambda \leq \mu_1 + M\delta_1 + M\delta_2 + M\delta_3 \\
 &\lambda \leq U_1 + M\delta_1 + M\delta_2 + M\delta_3 \\
 &\lambda \leq \mu_2 + M(1-\delta_1) + M\delta_2 \\
 &\lambda \leq U_2 + M(1-\delta_1) + M\delta_2 \\
 &\lambda \leq \mu_3 + M\delta_1 + M(1-\delta_2) \\
 &\lambda \leq U_3 + M\delta_1 + M(1-\delta_2) \\
 &\lambda \leq \mu_4 + M(1-\delta_1) + M(1-\delta_2) \\
 &\lambda \leq U_4 + M(1-\delta_1) + M(1-\delta_2) \\
 &\lambda \leq \mu_5 + M\delta_1 + M\delta_2 + M(1-\delta_3) \\
 &\lambda \leq U_5 + M\delta_1 + M\delta_2 + M(1-\delta_3)
 \end{aligned} \tag{14}$$

where, U_i is the upper bound of μ_i .

In contrast, we also deal with this problem using Ching-Ter's methodology, whose corresponding models are characterized by:

$$\begin{aligned}
 &\lambda_{\max} \\
 \text{s.t. } &\lambda \leq \mu_1 b_1 + M\delta_1 + M\delta_2 + M\delta_3 \\
 &\lambda \leq \mu_2 b_2 + U_1(1-b_2) + M(1-\delta_1) + M\delta_2 + M\delta_3 \\
 &\lambda \leq \mu_3 b_3 + U_2(1-b_3) + M\delta_1 + M(1-\delta_2) + M\delta_3 \\
 &\lambda \leq \mu_4 b_4 + U_3(1-b_4) + M(1-\delta_1) + M(1-\delta_2) + M\delta_3 \\
 &\lambda \leq \mu_5 b_5 + U_4(1-b_5) + M\delta_1 + M\delta_2 + M(1-\delta_3) \\
 &\lambda \leq U_1 + b_2 M, \lambda \leq U_2 + b_3 M, \lambda \leq U_3 + b_4 M \\
 &\lambda \leq U_4 + b_5 M
 \end{aligned} \tag{15}$$

where, b_i is also binary variables. Because δ_i and b_i are binary variables which can take values of 0 and 1 arbitrarily, when $b_2 = b_3 = b_4 = b_5 = 1$, in three particular situations, $\delta_1 = \delta_2 = 1, \delta_3 = 1, \delta_1 = 0, \delta_2 = 0, \delta_3 = 1$ and $\delta_1 = 1, \delta_2 = 1, \delta_3 = 1$, (18) can change to be of following type:

$$\begin{aligned}
 &\lambda_{\max} \\
 \text{s.t. } &\lambda \leq \mu_1 b_1 + M, \lambda \leq \mu_2 + M, \lambda \leq \mu_3 + M \\
 &\lambda \leq \mu_4 + M, \lambda \leq \mu_5 + M, \lambda \leq U_1 + M \\
 &\lambda \leq U_2 + M, \lambda \leq U_3 + M, \lambda \leq U_4 + M
 \end{aligned} \tag{16}$$

Considering that M is a big positive number, satisfaction degree λ becomes unconstrained. While this constraint-free problem would never happen by our methodologies regard to any change of binary variables.

Approach 2: Alternatively, fuzzy programming with PLMFs can be formulated directly by means of the following orthodox equations:

$$\begin{aligned}
 &\lambda_{\max} \\
 \text{s.t. } &\lambda \leq \mu_q(z_i(x)) + \sum_{i=1}^{q-1} \frac{1-(z_i(x))^{2^i}}{2^i} (1-\delta_i) \frac{1-(z_i(x))^{2^i}}{2^i} M + [(n+q)2^{n+q} + \lceil 2^{n+q} q \rceil] \delta_i \frac{1-(z_i(x))^{2^i}}{2^i} (1-\delta_i) \frac{1-(z_i(x))^{2^i}}{2^i} M \\
 &\lambda \leq U_q + \sum_{i=1}^{q-1} \frac{1-(z_i(x))^{2^i}}{2^i} (1-\delta_i) \frac{1-(z_i(x))^{2^i}}{2^i} M + [(n+q)2^{n+q} + \lceil 2^{n+q} q \rceil] \delta_i \frac{1-(z_i(x))^{2^i}}{2^i} (1-\delta_i) \frac{1-(z_i(x))^{2^i}}{2^i} M
 \end{aligned} \tag{17}$$

where, U_{ij} is the upper bound of μ_{ij} , n is the number of convex points, $\theta = \lceil \ln n / \ln 2 \rceil$ is the number of binary variables demanded, q is the number of sub-sections of the PLMF.

For example, Eq. 17 accounts for a PLMFs with $\theta = 4$ and $n = 8$, where, μ_1 to μ_9 are membership functions of each sub-segment respectively. In this case, unconstrained λ associated with the corresponding fuzzy programming problem could be avoided. Besides, it is observed that $2^{\theta-q}$ constraints are free of the fourth binary items, which implies that $\theta-1$ binary items are demanded only. Undoubtedly, reduction of the number of binary items would help make solving the optimization problem easier:

$$\begin{aligned} &\lambda_{\max} \\ \text{s.t. } &\lambda \leq \mu_1 + M\delta_1 + M\delta_2 + M\delta_3 + M\delta_4, \lambda \leq U_1 + M\delta_1 + M\delta_2 + M\delta_3 + M\delta_4 \\ &\lambda \leq \mu_2 + M(1-\delta_1) + M\delta_2 + M\delta_3, \lambda \leq U_2 + M(1-\delta_1) + M\delta_2 + M\delta_3 \\ &\lambda \leq \mu_3 + M\delta_1 + M(1-\delta_2) + M\delta_3, \lambda \leq U_3 + M\delta_1 + M(1-\delta_2) + M\delta_3 \\ &\lambda \leq \mu_4 + M(1-\delta_1) + M(1-\delta_2) + M\delta_3, \lambda \leq U_4 + M(1-\delta_1) + M(1-\delta_2) + M\delta_3 \\ &\lambda \leq \mu_5 + M\delta_1 + M\delta_2 + M(1-\delta_3), \lambda \leq U_5 + M\delta_1 + M\delta_2 + M(1-\delta_3) \\ &\lambda \leq \mu_6 + M(1-\delta_1) + M\delta_2 + M(1-\delta_3), \lambda \leq U_6 + M(1-\delta_1) + M\delta_2 + M(1-\delta_3) \\ &\lambda \leq \mu_7 + M\delta_1 + M(1-\delta_2) + M(1-\delta_3), \lambda \leq U_7 + M\delta_1 + M(1-\delta_2) + M(1-\delta_3) \\ &\lambda \leq \mu_8 + M(1-\delta_1) + M(1-\delta_2) + M(1-\delta_3), \lambda \leq U_8 + M(1-\delta_1) + M(1-\delta_2) + M(1-\delta_3) \\ &\lambda \leq \mu_9 + M\delta_1 + M\delta_2 + M\delta_3 + M(1-\delta_4), \lambda \leq U_9 + M\delta_1 + M\delta_2 + M\delta_3 + M(1-\delta_4) \end{aligned} \quad (18)$$

AN ILLUSTRATIVE EXAMPLE

Consider the following numerical example:

$$\begin{aligned} \text{Max } & f(x) = -10x_1^2 - 10x_2^2 - 20x_3^2 + 5x_4^2 + 50x_1 + 50x_2 + 210x_3 + 10x_4 \\ \text{s.t. } & g_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 \leq 700 \\ & g_2(x) = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 \leq 1044 \\ & h(x) = 2x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 = 502, x_i \geq 0 \quad i=1,2,3,4 \end{aligned} \quad (19)$$

where, all constraints are of soft type. For later comparison, we initially obtained the optimum solutions of the corresponding crisp optimization problem as: $x^*_1 = 15.04, x^*_2 = 2.25, x^*_3 = 6.05, x^*_4 = 20.08, z^* = 1305.6$. In regard to the soft constraints, we construct unilateral and bilateral Gaussian membership functions, shown as follows:

$$\mu_{g_1}(x_1) = \left(\left[\frac{\arctan(g_1 - 700)}{\pi / 2} \right] + 1 \right) * e^{-\frac{(g_1 - 700)^2}{2p_1^2}} - \left[\frac{\arctan(g_1 - 700)}{\pi / 2} \right] * 1 \quad (20)$$

$$\mu_{g_2}(x_1) = \left(\left[\frac{\arctan(g_2 - 1044)}{\pi / 2} \right] + 1 \right) * e^{-\frac{(g_2 - 1044)^2}{2p_2^2}} - \left[\frac{\arctan(g_2 - 1044)}{\pi / 2} \right] * 1 \quad (21)$$

$$\mu_h(x_1) = \left(\left[\frac{\arctan(h - 502)}{\pi / 2} \right] + 1 \right) * e^{-\frac{(h - 502)^2}{2p_{3L}^2}} - \left[\frac{\arctan(h - 502)}{\pi / 2} \right] * e^{-\frac{(h - 502)^2}{2p_{3R}^2}} \quad (22)$$

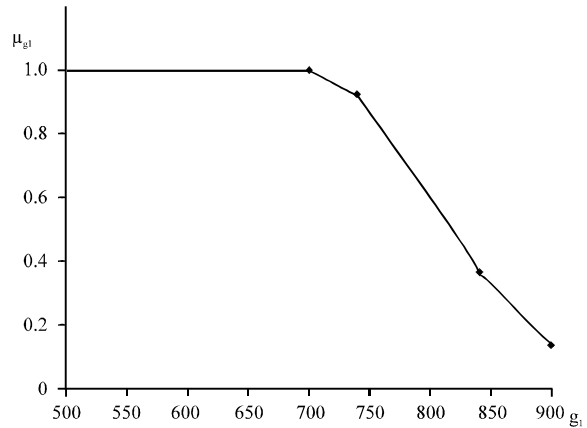


Fig. 8: Piecewise linear membership functions of μ_{g_1}

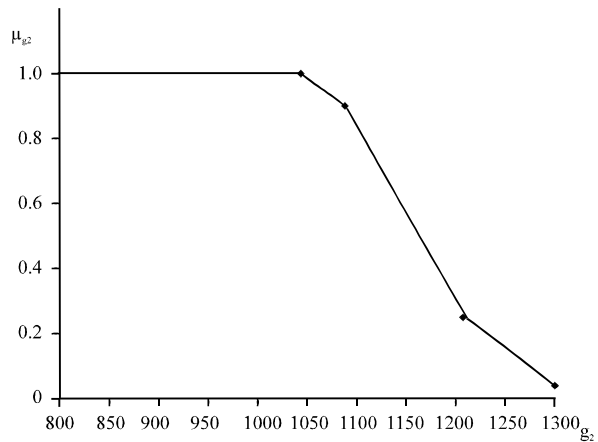


Fig. 9: Piecewise linear membership functions of μ_{g_2}

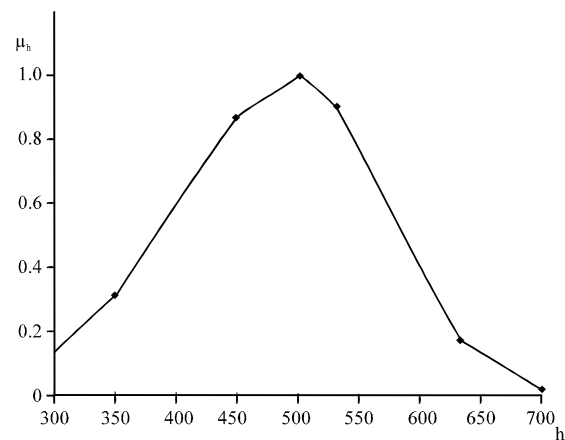


Fig. 10: Piecewise linear membership functions of μ_h

where, $p_1 = p_2 = p_{3L} = 100$ and $p_{3R} = 70$. The reference points for the PLMFs were specified as shown in Table 4. Then we got the PLMFs presented as Eq. 23, 24 and 25, whose profile curves are shown in Fig. 8-10 and the upper

Table 4: Reference points of the piecewise linear membership functions

Membership functions	Reference points
μ_{g_1}	(500,1) (700,1) (740,0.9231) (842,0.3649) (900,0.1353)
μ_{g_2}	(800,1) (1044,1) (1090,0.8996) (1210,0.2521) (1300,0.03775)
μ_h	(300,0.13) (350,0.315) (450,0.8735) (502,1) (534,0.9) (633,0.1736) (700,0.01831)

Table 5: Binary variables

Constraints	1	2	3
Binary variables	δ_1 (1- δ_1)	δ_2 (1- δ_2)	δ_3 δ_4 (1- δ_3) δ_4 δ_3 (1- δ_4)

Table 6: Solutions of the fuzzy programming problems

Types of membership functions	Optimum results
Piecewise linear membership functions	$x_1^* = 14.78$ $x_2^* = 2.249$ $x_3^* = 6.048$ $x_4^* = 20.402$ $\lambda^* = 0.96$ $z^* = 1440.296$
Regular Gaussian membership functions	$x_1^* = 14.58$ $x_2^* = 0.1$ $x_3^* = 0.1$ $x_4^* = 21.6$ $\lambda^* = 0.786$ $z^* = 1079$

bounds of each PLMFs are: $U_{11} = 1$, $U_{12} = 0.3649$, $U_{21} = 1$, $U_{22} = 0.2521$, $U_{31} = 0.315$, $U_{32} = 1$, $U_3 = 0.1736$:

$$\mu_{g_1} \begin{cases} 1 & g_1 \leq 700 \\ -0.0019g_1 + 2.3291 & 700 \leq g_1 \leq 740 \\ -0.0055g_1 + 4.9959 & 740 \leq g_1 \leq 842 \\ -0.004g_1 + 3.7353 & 842 \leq g_1 \leq 900 \end{cases} \quad (23)$$

$$\mu_{g_2} \begin{cases} 1 & g_2 \leq 1044 \\ -0.0022g_2 + 3.2976 & 1044 \leq g_2 \leq 1090 \\ -0.0054g_2 + 6.7861 & 1090 \leq g_2 \leq 1210 \\ -0.0024g_2 + 3.1577 & 1210 \leq g_2 \leq 1300 \end{cases} \quad (24)$$

$$\mu_h \begin{cases} 0.0037h - 0.98 & 300 \leq h \leq 350 \\ 0.0056h - 1.6465 & 350 \leq h \leq 450 \\ 0.0024h - 0.2048 & 450 \leq h \leq 502 \\ -0.0031h + 2.5554 & 502 \leq h \leq 534 \\ -0.0072h + 4.7312 & 534 \leq h \leq 633 \\ -0.0023h + 1.6283 & 633 \leq h \leq 700 \end{cases} \quad (25)$$

It is clearly observed that 1, 1 and 2 convex points are involved in the three PLMFs, respectively. We calculate the number of binary variables required as $\theta_1 = [\ln 1/\ln 2] = 1$, $\theta_2 = [\ln 1/\ln 2] = 1$, $\theta_3 = [\ln 1/\ln 2] = 2$, which means 4 binary variables are needed to describe the relationship between the sub-segments. The binary variables are listed in Table 5.

Thus, the fuzzy programming formulations are created as follows:

$$\begin{aligned} &\lambda_{\max} \\ \text{s.t. } &\lambda \leq 1 + M\delta_1, \lambda \leq -0.0019g_1 + 2.3291 + M\delta_1, \lambda \leq -0.0055g_1 + 4.9959 + M\delta_1 \\ &\lambda \leq -0.004g_1 + 3.7353 + M(1 - \delta_1), \lambda \leq 0.3649 + M(1 - \delta_1), \lambda \leq 1 + M\delta_2 \\ &\lambda \leq -0.0022g_2 + 3.2976 + M\delta_2, \lambda \leq -0.0054g_2 + 6.7861 + M\delta_2 \end{aligned}$$

$$\begin{aligned} &\lambda \leq -0.0024g_3 + 3.1577 + M(1 - \delta_3), \lambda \leq 0.2521 + M(1 - \delta_3) \\ &\lambda \leq 0.0037h - 0.98 + M\delta_3 + M\delta_4, \lambda \leq 0.315 + M\delta_3 + M\delta_4, \\ &\lambda \leq 0.0056h - 1.6465 + M(1 - \delta_3) + M\delta_4, \lambda \leq 0.0024h - 0.2048 + M(1 - \delta_3) + M\delta_4 \\ &\lambda \leq -0.0031h + 2.5554 + M(1 - \delta_3) + M\delta_4, \lambda \leq -0.0072h + 4.7312 + M(1 - \delta_3) + M\delta_4 \\ &\lambda \leq 1 + M(1 - \delta_3) + M\delta_4, \lambda \leq -0.0023h + 1.6283 + M\delta_3 + M(1 - \delta_4), \\ &\lambda \leq 0.1736 + M\delta_3 + M(1 - \delta_4), \lambda \leq z/1500, \\ &z = -10x_1^2 - 10x_2^2 - 20x_3^2 + 5x_4^2 + 50x_1 + 50x_2 + 210x_3 + 10x_4, \\ &g_1 = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4, g_2 = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 \\ &h = 2x_1^2 + x_2^2 + x_3^2 + 2x_4^2 - x_2 - x_4, x_i \geq 0 \quad i=1,2,3,4 \end{aligned} \quad (26)$$

The above nonlinear programming problem was solved using the ALPHA ECP solver of GAMS which is always employed to deal with mixed-integer nonlinear programming problems. After 340 iterations, the optimum solution was achieved at $\delta_1 = 0$, $\delta_2 = 0$, $\delta_3 = 1$ and $\delta_4 = 0$, as show in the first line of Table 6. In this situation, Eq. 24 can be simplified and re-arranged as follows:

$$\begin{aligned} \text{Max } &f = \lambda \\ \text{s.t. } &\omega_1 = \lambda \leq 1, \omega_2 = \lambda + 0.0022(x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4) \leq 3.2976 \\ &\omega_3 = \lambda - 0.0024(2x_1^2 + x_2^2 + x_3^2 + 2x_4^2 - x_2 - x_4) \leq -0.2048 \\ &\omega_4 = \lambda - (-10x_1^2 - 10x_2^2 - 20x_3^2 + 5x_4^2 + 50x_1 + 50x_2 + 210x_3 + 10x_4)/1500 \leq 0 \\ &x_i \geq 0 \quad i=1,2,3,4 \end{aligned} \quad (27)$$

At the optimal point x_i^* and λ^* , the active constraints must satisfy KKT conditions which are also known as the first order necessary conditions of optimality. Regarding Eq. 25, the KKT conditions are expressed by:

$$\begin{aligned} \frac{\partial f}{\partial \lambda} - \sum_{i=1}^4 \eta_i \frac{\partial \omega_i}{\partial \lambda} &= 0, \frac{\partial f}{\partial x_j} - \sum_{i=1}^4 \eta_i \frac{\partial \omega_i}{\partial x_j} = 0 \\ \eta_1 (\omega_1 - 1) &= 0, \eta_2 (\omega_2 - 3.2976) = 0 \\ \eta_3 (\omega_3 + 0.2048) &= 0, \eta_4 (\omega_4 - 0) = 0 \end{aligned} \quad (28)$$

where, $\eta_1, \eta_2, \eta_3, \eta_4$ are the vectors of KKT multipliers corresponding to each constraint. To perform algebraic calculations, the solutions of Eq. 26 could be obtained as $\eta_1 = 0$, $\eta_2 = 0.243$, $\eta_3 = 0.447$ and $\eta_4 = 0.31$, which implies the optimality of the solutions.

For comparison, we also formulated this problem using regular Gaussian membership functions, the solutions are presented in Table 6. Due to the characteristics of PLMFs, the nonlinearity of the model didn't be deepened which obtained better solutions.

CONCLUSION

A method of formulating unilateral and bilateral membership functions was presented. Taking advantage of options of reference points in Gaussian membership functions, an approach able to convert unilateral and bilateral membership functions into piecewise linear ones

is subsequently introduced. By means of compatible configuration of binary variables, the PLMFs can be represented more efficiently, which helps the corresponding fuzzy optimization formulations prevent constraint-free problems. Compared with conventional modeling methodologies, this contribution demands fewer binary items in some scenarios, possibly reducing the complexity of optimization calculations.

REFERENCES

- Bellman, R.E. and L.A. Zadeh, 1970. Decision-making in a fuzzy environment. *Manage. Sci.*, 17: B141-B164.
- Chang, C.T., 2007. Binary behavior of fuzzy programming with piecewise linear membership functions. *IEEE Trans. Fuzzy Syst.*, 15: 342-349.
- Chang, C.T., 2010. An approximation approach for representing S-Shaped membership functions. *IEEE Trans. Fuzzy Syst.*, 18: 412-424.
- Effati, S. and H. Abbasiyan, 2010. Solving fuzzy linear programming problems with piecewise linear membership function. *Appl. Applied Math.*, 5: 504-533.
- Garvin, W.W., H.W. Crandall, J.B. John and R.A. Spellman, 1957. Applications of linear programming in the oil industry. *Manage. Sci.*, 3: 407-427.
- Hannan, E.L., 1981. Linear programming with multiple fuzzy goals. *Fuzzy Sets Syst.*, 6: 235-248.
- Hu, C.F. and S.C. Fang, 1999. Solving fuzzy inequalities with piecewise linear membership functions. *IEEE Trans. Fuzzy Syst.*, 7: 230-235.
- Li, H.L. and C.S. Yu, 1999. Comments on fuzzy programming with nonlinear membership functions. *Fuzzy Sets Syst.*, 101: 109-113.
- Li, H.L. and C.S. Yu, 2000. A fuzzy multiobjective program with quasiconcave membership functions and fuzzy coefficients. *Fuzzy Sets Syst.*, 109: 59-81.
- Lin, C.C. and A.P. Chen, 2002. Generalization of Yang *et al.*'s method for fuzzy programming with piecewise linear membership functions. *Fuzzy Sets Syst.*, 132: 347-352.
- Liu, Y.H. and Y. Shi, 1994. A fuzzy programming approach for solving a multiple criteria and multiple constraint level linear programming problem. *Fuzzy Sets Syst.*, 65: 117-124.
- Narasimhan, R., 1980. Goal programming in a fuzzy environment. *Decision Sci.*, 11: 325-336.
- Roubens, M. and J. Teghem Jr., 1991. Comparison of methodologies for fuzzy and stochastic multi-objective programming. *Fuzzy Sets Syst.*, 42: 119-132.
- Shi, Y. and P.L. Yu, 1992. Selecting optimal linear production systems in multiple criteria environments. *Comput. Oper. Res.*, 19: 585-608.
- Stone, H., 1961. Approximation of curves by line segments. *Math. Comput.*, 15: 40-47.
- Sun, W., G.H. Huang, Y. Lv and G. Li, 2012. Waste management under multiple complexities: Inexact piecewise-linearization-based fuzzy flexible programming. *Waste Manage.*, 32: 1244-1257.
- Toriello, A. and J.P. Vielma, 2012. Fitting piecewise linear continuous functions. *Eur. J. Oper. Res.*, 219: 86-95.
- Wen, C. and X. Ma, 2008. A max-piecewise-linear neural network for function approximation. *Neurocomputing*, 71: 843-852.
- Yang, T. and J.P. Ignizio, 1991. Fuzzy programming with nonlinear membership functions: Piecewise linear approximation. *Fuzzy Sets Syst.*, 41: 39-53.
- Zimmermann, H.J., 1978. Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets Syst.*, 1: 45-55.
- Zimmermann, H.J., 1987. *Fuzzy Sets, Decision Making and Expert Systems*. 2nd Edn., Springer, New York, USA., ISBN-13: 978-0898381498, Pages: 335.