Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER Applied Mathematics and Computation 167 (2005) 252–273

APPLIED MATHEMATICS AND COMPUTATION

www.elsevier.com/locate/amc

Interval weight generation approaches based on consistency test and interval comparison matrices ☆

Ying-Ming Wang [a,b,*], Jian-Bo Yang [a], Dong-Ling Xu [a]

[a] *Manchester School of Management, UMIST, P.O. Box 88, Manchester M60 1QD, UK*
[b] *School of Public Administration, Fuzhou University, Fuzhou, Fujian, 350002, PR China*

Abstract

A simple yet pragmatic method of consistency test is developed to check whether an interval comparison matrix is consistent or not. Linear programming method is used to derive consistent interval weights from consistent interval comparison matrices and to aggregate local interval weights to generate global interval weights. In cases where an interval comparison matrix is inconsistent, an eigenvector method-based nonlinear programming (NLP) approach is developed to generate interval weights that can meet predetermined consistency requirements. A simple and effective preference ranking method is utilized to compare the interval weights of criteria or rank alternatives. Three numerical examples including a hierarchical (AHP) decision problem are provided to illustrate the validity and practicality of the proposed methods.

☆ This research was supported by the National Natural Science Foundation of China (NSFC) under the Grant no.: 70271056 and the Grant no.: 70171035, Fok Ying Tung Education Foundation under the Grant no.: 71080, and also in part by the UK Engineering and Physical Science Research Council under the Grant no.: GR/N65615/01 and the European Commission under the Grant no.: IPS-2000-00030.
* Corresponding author. Address: Manchester School of Management, UMIST P.O. Box 88, Manchester M60 1QD, UK.

E-mail address: msymwang@hotmail.com (Y.-M. Wang).

0096-3003/$ - see front matter
doi:10.1016/j.amc.2004.06.080

Keywords: Analytic hierarchy process; Interval comparison matrix; Consistency test; Weight estimation; Linear and nonlinear programming; Preference ranking

1. Introduction

The estimation of the relative weights of criteria/attributes plays an important role in multiple criteria decision analysis (MCDA). Many methods for generating weights have been proposed. Among other frameworks, pairwise comparison matrices provide a natural framework to elicit preferences from decision makers and have been used in several weight generation methods such as the principal right eigenvector method (EM) [23–26], and the logarithmic least squares method (LLSM) [5,11,27], which is also known as the geometric mean method (GMM) [6,10]. In a conventional pairwise comparison matrix, 1–9 ratio scales are normally used to elicit exact comparisons. However, due to the complexity and uncertainty involved in real world decision problems and the inherent subjective nature of human judgments, it is sometimes unrealistic and infeasible to acquire exact judgments. It is more natural or easier to provide fuzzy or interval judgments for parts or all of the judgments in a pairwise comparison matrix. A number of techniques have been developed to use such a fuzzy or interval comparison matrix to generate weights.

Van Laarhoven and Pedryce [35] considered treating elements in a comparison matrix as fuzzy numbers having triangular membership functions and employed the logarithmic least squares method to generate fuzzy weights. Buckley [8] extended the method to trapezoidal membership functions and hierarchical analysis. Bonder et al. [7] found a fallacy in the normalization procedure of Van Laarhoven and Pedryce's method for generating fuzzy weights and subsequently modified the method. Xu and Zhai [34] discussed the problem of extracting fuzzy weights from a fuzzy judgment matrix also using the logarithmic least squares method based on a distance definition in a fuzzy judgment space. Xu [33] used the same distance definition to develop a fuzzy least-squares priority method. Leung and Cao [18] proposed a fuzzy consistency definition by considering a tolerance deviation and determined fuzzy local and global weights using the extension principle. Buckley et al. [9] directly fuzzified Saaty's original procedure of computing weights in hierarchical analysis to get fuzzy weights in fuzzy hierarchical analysis. Csutora and Buckley [12] presented a Lambda–Max method, which is the direct fuzzification of the $\lambda_{\max}$ method, to find fuzzy weights. However, most of the above fuzzy priority techniques take little account of inconsistent judgments.

Saaty and Vargas [28] proposed interval judgments for the AHP method as a way to model subjective uncertainty and used a Monte Carlo simulation approach to find out weight intervals from interval comparison matrices. They also pointed out difficulties in using this approach. Arbel [1,2] interpreted interval judgments as linear constraints on local priorities and formulated the prioritization process as a linear programming (LP) model. Kress [16] found that Arbel's method is ineffective for inconsistent interval comparison matrices because no feasible region exists in such circumstances. Salo and Hämäläinen [29,31] extended Arbel's approach to hierarchical structures. Their method found the maximum and minimum feasible values for all interval priorities and incorporated the resulting intervals into further synthesis of global interval priorities. Arbel and Vargas [3,4] formulated the hierarchical problem as a nonlinear programming model in which all local priorities in a hierarchy are included as decision variables and also established a connection between Monte Carlo simulation and Arbel's LP approach. Moreno-Jimenez [22] studied the probability distribution of possible rankings of alternatives in a small interval comparison matrix (e.g. $n = 2$ or $n = 3$). Islam et al. [14] used a Lexicographic Goal Programming (LGP) to find out weights from inconsistent pairwise interval comparison matrices and explored its properties and advantages as a weight estimation technique. Haines [13] proposed a statistical approach to extract preferences from interval comparison matrices. Two specific distributions on a feasible region were examined and the mean of the distributions was used as a basis for assessment and ranking. Mikhailov [19–21] developed a fuzzy preference programming (FPP) method to derive crisp priorities from interval or fuzzy comparison matrices and extended the method to the case of group decision making.

As is well known, the test of consistency plays an important role in judgmental modeling. Due to the uncertainty and complexity of real world decision problems and the subjectivity of expert judgments, it is inevitable to generate inconsistent comparisons. High inconsistency is likely to result in unreliable weights and ranking orders for alternatives. Therefore, checking satisfactory consistency is necessary to ensure the rationality of decisions. Only comparison matrices passing the test of satisfactory consistency can be used to generate reliable weights. In other words, only satisfactory weights are reliable and should be used for follow up decision analysis. However, so far little work has been found in literature on how to check whether an interval comparison matrix is consistent or not and how to derive priorities with pre-determined consistency levels from inconsistent interval comparison matrices. This paper is mainly devoted to solving these two key issues and other related problems. A simple yet pragmatic consistency test method is developed to check whether an interval comparison matrix is consistent. If it is consistent, then Arbel's preference programming method is recommended to derive the consistent interval weights due to its simplicity and effectiveness. Otherwise, an eigenvector method (EM)-based nonlinear programming (NLP) approach is developed to generate interval weights with pre-determined consistence requirements being met. In the case of hierarchical analysis, a linear programming approach is used to aggregate local interval weights to generate global ones. A simple and effective preference ranking method is utilized to compare the interval weights of criteria or rank alternatives. Since fuzzy comparison matrices can be easily transformed into interval comparison matrices using α-*level* sets, this paper is focused only on the analysis of interval comparison matrices.

The paper is organized as follows. Section 2 first gives two definitions of consistency for interval comparison matrices and then develops a simple yet pragmatic approach that can be used to check whether an interval comparison matrix is consistent or not without solving any optimization model. In Section 3, Arbel's preference programming approach is recommended to derive consistent interval weights from consistent interval comparison matrices and an EM-based nonlinear programming method is developed to generate interval weights with pre-determined consistency requirements being met from inconsistent comparison matrices. Section 4 discusses the aggregation problem of interval weights. In Section 5, a simple and effective preference ranking method is introduced to compare the interval weights of criteria or rank alternatives. Section 6 provides three numerical examples including a hierarchical (AHP) decision problem to show the validity and practicality of the proposed methods. The paper is concluded in Section 7.

2. An approach of consistency test for interval comparison matrices

Suppose decision maker (DM) provides interval judgments instead of precise judgments for a pairwise comparison. For example, it could be judged that criterion i is between l_{ij} and u_{ij} times as important as criterion j with l_{ij} and u_{ij} being nonnegative real numbers and $l_{ij} \leqslant u_{ij}$. Then, an interval comparison matrix can be represented by

$$A = (a_{ij})_{n\times n} = \begin{bmatrix} 1 & [l_{12}, u_{12}] & \cdots & [l_{1n}, u_{1n}] \\ [l_{21}, u_{21}] & 1 & \cdots & [l_{2n}, u_{2n}] \\ \vdots & \vdots & \vdots & \vdots \\ [l_{n1}, u_{n1}] & [l_{n2}, u_{n2}] & \cdots & 1 \end{bmatrix}, \tag{1}$$

where $l_{ij} = 1/u_{ji}$, $u_{ij} = 1/l_{ji}$ and $l_{ij} \leqslant a_{ij} \leqslant u_{ij}$. About the above interval comparison matrix, we give the following definition and theorem:

Definition 1. Let $A = (a_{ij})_{n\times n}$ is an interval comparison matrix defined by (1) with $l_{ij} \leqslant a_{ij} \leqslant u_{ij}$ and $a_{ii} = l_{ii} = u_{ii} = 1$ for $i, j = 1,\ldots,n$. If the convex feasible region $S_w = \{w = (w_1, \ldots, w_n) | l_{ij} \leqslant w_i/w_j \leqslant u_{ij}, \sum_{i=1}^{n} w_i = 1, w_j > 0, j = 1, \ldots, n\}$ is nonempty, then A is said to be a consistent interval comparison matrix.

Theorem 1. *$A = (a_{ij})_{n\times n}$ is a consistent interval comparison matrix if and only if it satisfies the following inequality constraints*:

$$\max_k(l_{ik}l_{kj}) \leqslant \min_k(u_{ik}u_{kj}), \quad \textit{for all } i,j,k = 1,\ldots,n. \tag{2}$$

Proof. If A is a consistent interval comparison matrix, then the convex feasible region S_w is nonempty, which means that there is no contradiction among the following inequality constraints:

$$l_{ik} \leqslant w_i/w_k \leqslant u_{ik}, \quad i,k = 1,\ldots,n, \tag{3}$$

$$l_{kj} \leqslant w_k/w_j \leqslant u_{kj}, \quad k,j = 1,\ldots,n. \tag{4}$$

Multiplying (3) by (4) leads to the following implied indirect inequalities

$$l_{ik}l_{kj} \leqslant w_i/w_j \leqslant u_{ik}u_{kj}, \quad i,j,k = 1,\ldots,n. \tag{5}$$

Since (5) holds for any $k = 1,\ldots,n$, it follows that $\max_k(l_{ik}l_{kj}) \leqslant \min_k(u_{ik}u_{kj})$ holds for all $i,j,k = 1,\ldots,n$.

Conversely, if (2) holds for $\forall i,j,k$, then $l_{ij} \leqslant w_i/w_j \leqslant u_{ij}$ holds for any $i,j = 1,\ldots,n$. So, S_w cannot be empty. By Definition 1, A is a consistent interval comparison matrix. □

The above Theorem 1 can be used to judge whether or not an interval comparison matrix is consistent without solving any mathematical programming model. It only requires simple algebraic operations. We will show this feature in the examination of Examples 1 and 2 in Section 6. Since an interval comparison matrix is reciprocal in nature, only its upper or lower triangular elements need to be checked.

3. Priority generation methods for interval comparison matrices

3.1. Linear programming method

As mentioned in Section 1, there are several methods that can be used to derive priorities from interval comparison matrices. Arbel's preference programming method is the simplest yet most effective way to derive priorities from consistent interval comparison matrices. The method can generate consistent interval weights that can satisfy all judgments in a consistent interval comparison matrix. So, we recommend using the method if an interval comparison matrix is judged to be consistent using Theorem 1.

The method was originally developed to find the vertices of the convex feasible region, $S_w = \{w = (w_1,\ldots,w_n) | l_{ij} \leqslant w_i/w_j \leqslant u_{ij}, \sum_{i=1}^{n} w_i = 1,\ w_j > 0,\ j = 1,\ldots,n\}$. If all the vertices prefer w_l to w_k ($l,k = 1,\ldots,n$), then any convex linear combination of all the vertices would prefer w_l to w_k. Usually, the priority vector was generated as a convex linear combination of all the vertices. Since the convex combination produces only a point estimation of priorities, we propose to generate interval weights as the final priorities, which can be obtained by solving the following pairs of linear programming (LP) models:

$$\text{Min/Max} \quad w_i \tag{6}$$

$$\text{s.t.} \quad W \in S_W, \tag{7}$$

where $W = (w_1, \ldots, w_n)^{\mathrm{T}}$. The solutions to the above pairs of LP models form the weight intervals denoted by $[w_i^L, w_i^U](i = 1, \ldots, n)$.

3.2. EM-based nonlinear programming method

If an interval comparison matrix is judged to be inconsistent using Theorem 1, it appears particularly important to derive the priorities with pre-determined consistency requirements being met. The literature review shows that this problem has not been addressed. This subsection is devoted to investigating this problem and proposing an Eigenvector Method (EM)-based nonlinear programming method, which can be used to generate satisfactory interval weights from inconsistent interval comparison matrices.

From the principal right eigenvector method, it is known that

$$\widehat{A}W = \lambda_{\max}W, \tag{8}$$

where $\widehat{A}$ is a crisp comparison matrix; $\lambda_{\max}$ is the maximum eigenvalue of the comparison matrix $\widehat{A}$; W is the principal right eigenvector corresponding to $\lambda_{\max}$. The relationship between $\lambda_{\max}$ and the consistency ratio (CR) can be described by

$$\mathrm{CR} = \frac{\mathrm{CI}}{\mathrm{RI}} = \frac{\lambda_{\max} - n}{n-1} \Big/ \mathrm{RI} = \frac{\lambda_{\max} - n}{(n-1)\mathrm{RI}}, \tag{9}$$

where RI is an average random consistency index [25], which depends on the particular AHP scale used. It is suggested that if CR $\leqslant 0.1$ the comparison matrix is believed to have satisfactory consistency and to be acceptable and that if CR > 0.1 it has poor consistency and needs to be revised. Formula (9) may be further written as

$$\lambda_{\max} = n + (n-1)\mathrm{RI} \cdot \mathrm{CR}. \tag{10}$$

Substituting (10) into (8) produces

$$\widehat{A}W = [n + (n-1)\mathrm{RI} \cdot \mathrm{CR}]W. \tag{11}$$

Eq. (11) is derived from crisp comparison matrices and can be extended to interval comparison matrices.

Suppose $\widehat{A} = (\hat{a}_{ij})_{n \times n}$ is a crisp comparison matrix, which is randomly generated from the interval comparison matrix A with $l_{ij} \leqslant \hat{a}_{ij} \leqslant u_{ij}$ and $\hat{a}_{ji} = 1/\hat{a}_{ij}$.

Then Eq. (11) holds for $\widehat{A}$. Thus, the following pairs of nonlinear programming models, which are based on Saaty's principal right eigenvector method, can be developed to generate the weight intervals with satisfactory consistency:

$$\text{Min/Max} \quad w_i \tag{12}$$

$$\text{s.t.} \quad \sum_{j=1}^{i-1} \frac{w_j}{\hat{a}_{ij}} - (n-1)(1 + \mathrm{RI}\cdot\mathrm{CR})w_i + \sum_{j=i+1}^{n} \hat{a}_{ij} w_j = 0, \quad i = 1,\ldots,n, \tag{13}$$

$$\sum_{i=1}^{n} w_i = 1, \tag{14}$$

$$l_{ij} \leqslant \hat{a}_{ij} \leqslant u_{ij}, \quad i = 1,\ldots,n-1; \;\; j = i+1,\ldots,n, \tag{15}$$

$$\mathrm{CR} \leqslant \delta, \tag{16}$$

where (13) is the expansion of (11), (15) is interval constraints, δ is the level of satisfactory consistency (e.g. $\delta \leqslant 0.1$), CR, w_i ($i = 1,\ldots,n$) and $\hat{a}_{ij}(i = 1,\ldots, n-1; j = i+1,\ldots,n)$ are all decision variables. The purpose of imposing constraint condition (16) on the above NLP models is to want to derive the weights satisfying the level of satisfactory consistency. The optimal objective values of the above pair of NLP models consist of the possible interval of w_i, which we denote by $[w_i^L, w_i^U]$. Repeating the above solution process for each weight w_i ($i = 1,\ldots,n$), all the priority intervals that meet the requirement of the satisfactory consistency can be obtained. This will be shown in Section 6 using numerical examples.

The numerical value of δ in (16) may be determined or adjusted according to the actual requirements of decision analysis. Moreover, the following NLP model, which minimizes the inconsistency of an interval comparison matrix, may also be utilized to derive the weights from an inconsistent interval matrix:

$$\text{Min} \quad \mathrm{CR} \tag{17}$$

$$\text{s.t.} \quad \sum_{j=1}^{i-1} \frac{w_j}{\hat{a}_{ij}} - (n-1)(1 + \mathrm{RI}\cdot\mathrm{CR})w_i + \sum_{j=i+1}^{n} \hat{a}_{ij} w_j = 0 \quad i = 1,\ldots,n, \tag{18}$$

$$\sum_{i=1}^{n} w_i = 1, \tag{19}$$

$$l_{ij} \leqslant \hat{a}_{ij} \leqslant u_{ij}, \quad i = 1,\ldots,n-1; \;\; j = i+1,\ldots,n. \tag{20}$$

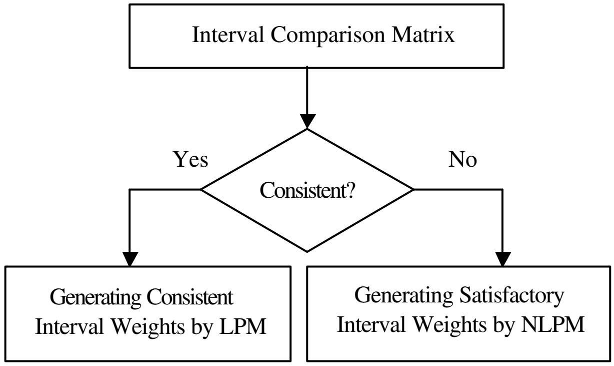

Fig. 1. Process for generating priorities from interval comparison matrices.

Such a NLP model usually leads to one crisp set of weights, which has the minimum inconsistency. The whole process introduced above for generating interval weights from interval comparison matrices is summarized in Fig. 1.

4. Synthesis of interval weights

Suppose the interval weights for upper-level criteria and lower-level alternatives have all been obtained, as shown in Table 1.

Salo and Hämäläinen [31] showed by an example that interval arithmetic is unsuitable for the synthesis of interval weights and has to be rejected. They thus proposed a hierarchical decomposition method that decomposes a hierarchical composition problem into a series of linear programming problems over the feasible regions.

Here, we suggest using the following pair of LP models to capture the lower and upper bounds of a composite weight for each alternative A_i:

$$\text{Min} \quad w_{A_i}^L = \sum_{j=1}^{m} w_{ij}^L w_j \tag{21}$$

$$\text{s.t.} \quad W \in \Omega_W, \tag{22}$$

$$\text{Max} \quad w_{A_i}^U = \sum_{j=1}^{m} w_{ij}^U w_j \tag{23}$$

$$\text{s.t.} \quad W \in \Omega_W, \tag{24}$$

where $W = (w_1, \ldots, w_m)$ and $\Omega_w = \{w = (w_1, \ldots, w_m) \mid w_j^L \leqslant w_j \leqslant w_j^U, \sum_{j=1}^{m} w_j = 1, j = 1, \ldots, n\}$. The above hierarchical composition process results in an interval weight for an alternative A_i, i.e. $[w_{A_i}^L, w_{A_i}^U] (i = 1, \ldots, n)$.

Table 1
Synthesis of interval weights

Alternatives	Criterion 1	Criterion 2	$\cdots$	Criterion M	Composite weights
	$[w_1^L, w_1^U]$	$[w_2^L, w_2^U]$	$\cdots$	$[w_m^L, w_m^U]$	
A_1	$[w_{11}^L, w_{11}^U]$	$[w_{12}^L, w_{12}^U]$	$\cdots$	$[w_{1m}^L, w_{1m}^U]$	$[w_{A_1}^L, w_{A_1}^U]$
A_2	$[w_{21}^L, w_{21}^U]$	$[w_{22}^L, w_{22}^U]$	$\cdots$	$[w_{2m}^L, w_{2m}^U]$	$[w_{A_2}^L, w_{A_2}^U]$
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
A_n	$[w_{n1}^L, w_{n1}^U]$	$[w_{n2}^L, w_{n2}^U]$	$\cdots$	$[w_{nm}^L, w_{nm}^U]$	$[w_{A_n}^L, w_{A_n}^U]$

5. Comparison and ranking of interval weights

In the case of interval comparison matrices, since the judgments are partly or completely imprecise, it is argued that interval weights should be more logical and acceptable to represent such imprecision than an exact priority vector that is only a point estimate. However, interval weights lead to greater complexity and difficulty in comparison and ranking.

In order to compare or rank global interval weights, Salo and Hämäläinen [31] required the decision maker (DM) to provide information on the revision of interval comparison matrices until one interval weight absolutely dominates the others or a pairwise dominance relation is found. However, it is not always feasible to require DM to provide extra information especially when DM fails or is unwilling to do so. Ishibuchi and Tanaka [15] used the comparison rule for interval numbers to define order relations. Their approach is probably the most prominent in the analysis and comparison of interval numbers. But it fails when one interval number is nested in another one. Kunda [17] defined a fuzzy leftness relationship between two interval numbers, which was to some extent able to reflect the degree of one interval number to be superior or inferior to another one. But it required the assumption that all the interval numbers are independent and uniformly distributed. Sengupta and Pal [32] defined an acceptability index, which was also designed to reflect the grade of acceptability of one interval number to be inferior to another one. The index was totally based on the midpoints of interval numbers. The use of midpoints to compare or rank interval numbers, however, was sometimes inconvincible and not easy to be accepted. In view of these reasons, Wang et al. [36] developed a simple yet practical preference ranking method of interval numbers, which makes no assumption and makes no use of the midpoints of interval numbers. The approach is summarized as follows.

Let $a = [a_1, a_2]$ and $b = [b_1, b_2]$ be two interval weights, whose possible relationships are as shown in Fig. 2. We refer to the degree of one interval weight being greater than another one as *the degree of preference*. Accordingly, we have the following definitions and properties.

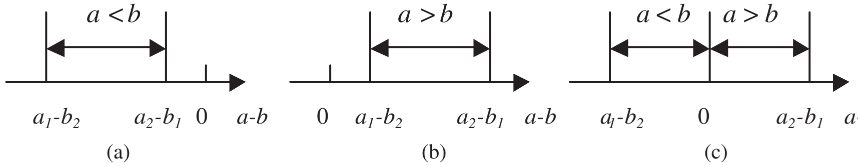

Fig. 2. Relationships between two interval weights a and b.

Definition 2. The degree of preference of a over b (or $a > b$) is defined as

$$P(a > b) = \frac{\max(0, a_2 - b_1) - \max(0, a_1 - b_2)}{(a_2 - a_1) + (b_2 - b_1)}. \tag{25}$$

The degree of preference of b over a (or $b > a$) can be defined in the same way. That is

$$P(b > a) = \frac{\max(0, b_2 - a_1) - \max(0, b_1 - a_2)}{(a_2 - a_1) + (b_2 - b_1)}. \tag{26}$$

It is obvious that $P(a > b) + P(b > a) = 1$ and $P(a > b) = P(b > a) \equiv 0.5$ when $a = b$, i.e. $a_1 = b_1$ and $a_2 = b_2$.

Definition 3. If $P(a > b) > P(b > a)$, then a is said to be superior to b to the degree of $P(a > b)$, denoted by $a \overset{P(a>b)}{\succ} b$; if $P(a > b) = P(b > a) = 0.5$, then a is said to be indifferent to b, denoted by $a \sim b$; If $P(b > a) > P(a > b)$, then a is said to be inferior to b to the degree of $P(b > a)$, denoted by $a \overset{P(b>a)}{\prec} b$.

Property 1. $P(a > b) = 1$ if and only if $a_1 \geqslant b_2$.

Property 2. If $a_1 \geqslant b_1$ and $a_2 \geqslant b_2$, then $P(a > b) \geqslant 0.5$ and $P(b > a) \leqslant 0.5$.

Property 3. If b is nested in a, i.e. $a_1 \leqslant b_1$ and $a_2 \geqslant b_2$, then $P(a > b) \geqslant 0.5$ if and only if $\frac{a_1+a_2}{2} \geqslant \frac{b_1+b_2}{2}$.

Property 4. If $P(a > b) \geqslant 0.5$ and $P(b > c) \geqslant 0.5$, then $P(a > c) \geqslant 0.5$.

The above four properties are very useful in comparing interval weights. Property 1 shows that if two interval weights do not overlap, then the one on the upper end will 100 percent dominate the one on the lower end. Property 2 is consistent with the comparison rule for ranking interval numbers. Property 3 shows how to compare two interval weights when one interval weight is included in the other. Property 4 shows that the preference relations are transitive. With the help of transitivity, a complete ranking order for interval weights could be achieved. The ranking process is outlined below:

Step 1: Calculate the matrix of degree of preference

$$P_D = \begin{array}{c} \\ w_1 \\ w_2 \\ \vdots \\ w_n \end{array} \begin{array}{c} \begin{array}{cccc} w_1 & w_2 & \cdots & w_n \end{array} \\ \begin{bmatrix} - & p_{12} & \cdots & p_{1n} \\ p_{21} & - & \cdots & p_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ p_{n1} & p_{n2} & \cdots & - \end{bmatrix} \end{array}, \tag{27}$$

where

$$p_{ij} = P(w_i > w_j) = \frac{\max(0, w_i^U - w_j^L) - \max(0, w_i^L - w_j^U)}{(w_i^U - w_i^L) + (w_j^U - w_j^L)}, \quad i,j = 1,\ldots n; \; i \neq j. \tag{28}$$

Step 2: Calculate the matrix of preference relation

$$M_{PL} = \begin{array}{c} \\ w_1 \\ w_2 \\ \vdots \\ w_n \end{array} \begin{array}{c} \begin{array}{cccc} w_1 & w_2 & \cdots & w_n \end{array} \\ \begin{bmatrix} - & m_{12} & \cdots & m_{1n} \\ m_{21} & - & \cdots & m_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & \cdots & - \end{bmatrix} \end{array}, \tag{29}$$

where

$$m_{ij} = \begin{cases} 1, & \text{if} \quad p_{ij} > 0.5, \\ 0, & \text{if} \quad p_{ij} \leqslant 0.5, \end{cases} \quad i,j = 1,\ldots n; \; i \neq j. \tag{30}$$

Step 3: Draw a directed diagram

If $m_{ij} = 1$, then draw an arrow from node i to node j. Such an arrow means that interval weight w_i is preferred to interval weight w_j with a degree of preference of p_{ij}, as shown in Fig. 3 for an example.

Step 4: Find a complete preference ranking order for all interval weights from the directed diagram using the property of transitivity. Alternatively, the following simple row-column elimination method can be used to generate a complete preference ranking order. In the matrix of degree of preference given in Eq. (27), first find a row where all elements (except for the diagonal element) are larger than 0.5. If this row corresponds to w_i, then w_i is the most (likely) preferred interval weight. Eliminate the ith row and ith column (thus w_i) in

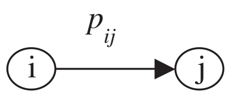

Fig. 3. Preference representation for interval weights w_i and w_j.

the matrix from further consideration. In the reduced matrix, if w_j stands out as the most (likely) preferred interval weight among the remaining intervals, then w_j should be ranked the second or $w_i \succ w_j$ if $p_{ij} > 0.5$, and w_i should be indifferent to w_j or $w_i \sim w_j$ if $p_{ij} = p_{ji} = 0.5$. Eliminate the jth row and jth column and repeat the above process until all intervals are ranked.

We will show the above ranking process in the numerical studies given in the next section.

6. Numerical examples

In this section, three numerical examples are examined to illustrate the proposed methods, their validity and wide potential applications. Comparisons with other existing procedures will also be made.

Example 1. Consider the following interval comparison matrix, which was examined by Arbel and Vargas [3,4] and Haines [13].

$$A = \begin{bmatrix} 1 & [2,5] & [2,4] & [1,3] \\ [\frac{1}{5},\frac{1}{2}] & 1 & [1,3] & [1,2] \\ [\frac{1}{4},\frac{1}{2}] & [\frac{1}{3},1] & 1 & [\frac{1}{2},1] \\ [\frac{1}{3},1] & [\frac{1}{2},1] & [1,2] & 1 \end{bmatrix}.$$

The consistency of A can be tested using Theorem 1. Table 2 shows the results of the consistency tests, from which it is clear that A passes all the consistency tests. So, it is a consistent interval comparison matrix and Arbel's preference programming method can be used to derive the weight intervals. Table 3 records all the vertices of the convex feasible region S_W. It must be pointed out that Arbel and Vargas [4] found only five of six vertices and missed the vertex in the last column of Table 3. The corresponding weight intervals can be generated from Table 3 and are shown in Table 4. They may also be obtained by solving LP models given by (6) and (7). It is clear in Table 4 that criterion 1 is the most important because its minimum weight is greater than the maximum weights of all the other criteria. To give a complete ranking order for the four interval weights, Table 5 records their degrees of preference. The corresponding directed diagram is depicted in Fig. 4, from which it is clear that w_1 is preferred over $w_2 \sim w_4$ to a degree of 100 percent, w_2 over w_3 and w_4 to a degree of 79.47% and 60.83%, respectively, and w_4 over w_3 to a degree of 70.63%. So, the final ranking order should be $w_1 \overset{100\%}{\succ} w_2 \overset{60.83\%}{\succ} w_4 \overset{70.63\%}{\succ} w_3$, which is the most likely ranking order. This ranking order is the same as the ranks given by Arbel and Vargas [3,4] using the average weights of all the vertices

Table 2
Consistency test for Example 1

Judgment element	i	j	k	$l_{ik}l_{kj}$	$u_{ik}u_{kj}$	Consistency test
a_{12}	1	2	1	2	5	$\max(l_{ik}l_{kj}) = 2$
	1	2	3	2/3	4	$\min(u_{ik}u_{kj}) = 3$
	1	2	4	1/2	3	Passed
a_{13}	1	3	1	2	4	$\max(l_{ik}l_{kj}) = 2$
	1	3	2	2	15	$\min(u_{ik}u_{kj}) = 4$
	1	3	4	1	6	Passed
a_{14}	1	4	1	1	3	$\max(l_{ik}l_{kj}) = 2$
	1	4	2	2	10	$\min(u_{ik}u_{kj}) = 3$
	1	4	3	1	4	Passed
a_{23}	2	3	1	2/5	2	$\max(l_{ik}l_{kj}) = 1$
	2	3	2	1	3	$\min(u_{ik}u_{kj}) = 2$
	2	3	4	1	4	Passed
a_{24}	2	4	1	1/5	3/2	$\max(l_{ik}l_{kj}) = 1$
	2	4	2	1	2	$\min(u_{ik}u_{kj}) = 3/2$
	2	4	3	1/2	3	Passed
a_{34}	3	4	1	1/4	3/2	$\max(l_{ik}l_{kj}) = 1/2$
	3	4	2	1/3	2	$\min(u_{ik}u_{kj}) = 1$
	3	4	3	1/2	1	Passed

Note: when $k = j$ and $k = i$, their results are the same. So, only $k = i$ is checked for all examples.

Table 3
The vertices of the feasible region S_w

W	Vertex #1	Vertex #2	Vertex #3	Vertex #4	Vertex #5	Vertex #6
w_1	0.5217	0.5000	0.4000	0.4444	0.4800	0.4615
w_2	0.1739	0.1667	0.2000	0.2222	0.2400	0.2307
w_3	0.1304	0.1667	0.2000	0.1111	0.1200	0.1538
w_4	0.1739	0.1667 (0.174[a])	0.2000	0.2222	0.1600	0.1538

[a] Note: 0.174 is the wrong figure given by Arbel and Vargas [4].

and by Haines [13] using the expected weights, but our ranking order provides more valuable information and is thus more convincing.

Example 2. Consider the following interval comparison matrix, which was investigated by Kress [16] and Islam et al. [14].

$$A = \begin{bmatrix} 1 & [1,2] & [1,2] & [2,3] \\ \left[\frac{1}{2},1\right] & 1 & [3,5] & [4,5] \\ \left[\frac{1}{2},1\right] & \left[\frac{1}{5},\frac{1}{3}\right] & 1 & [6,8] \\ \left[\frac{1}{3},\frac{1}{2}\right] & \left[\frac{1}{5},\frac{1}{4}\right] & \left[\frac{1}{8},\frac{1}{6}\right] & 1 \end{bmatrix}.$$

Table 4
The interval weights generated from the vertices of the feasible region S_w

W	w_1	w_2	w_3	w_4
$w_I^L = \min w_i$	0.4000	0.1667	0.1111	0.1538
$w_i^U = \max w_i$	0.5217	0.2400	0.2000	0.2222

Table 5
The matrix of degrees of preference for interval weights $w_1 \sim w_4$

p_{ij}	w_1	w_2	w_3	w_4
w_1	–	1.0000	1.0000	1.0000
w_2	0	–	0.7947	0.6083
w_3	0	0.2053	–	0.2937
w_4	0	0.3917	0.7063	–

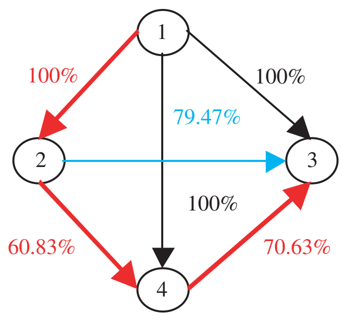

Fig. 4. Preference relations in Example 1.

Kress [16] showed that this interval comparison matrix is inconsistent and hence cannot be solved using Arbel's preference programming method and its variants. This can be confirmed by the consistency test using Theorem 1. Table 6 shows the results of the consistency tests, from which it is clear that A is an inconsistent interval comparison matrix.

Islam et al. [14] used lexicographic goal programming (LGP) to get a point estimate for the priority vector from the upper triangular judgments of A, i.e. $W = (0.3030, 0.4545, 0.1515, 0.0910)^{\mathrm{T}}$, which shows that $w_2 \succ w_1 \succ w_3 \succ w_4$. However, if we use the lower triangular judgments of A, which provide completely the same information as the upper triangular part, then a different point estimate will be obtained. That is $W = (0.3636, 0.3636, 0.1818, 0.0909)^{\mathrm{T}}$, which shows that $w_2 \sim w_1 \succ w_3 \succ w_4$. These two different ranking orders show that the LGP method is in fact defective in theory. In addition, it seems not

Table 6
Consistency test for Example 2

Judgment element	i	j	k	$l_{ik}l_{kj}$	$u_{ik}u_{kj}$	Consistency test
a_{12}	1	2	1	1	2	$\max(l_{ik}l_{kj}) = 1$
	1	2	3	1/5	2/3	$\min(u_{ik}u_{kj}) = 2/3$
	1	2	4	2/5	3/4	Failed
a_{13}	1	3	1	1	2	$\max(l_{ik}l_{kj}) = 3$
	1	3	2	3	10	$\min(u_{ik}u_{kj}) = 1/2$
	1	3	4	1/4	1/2	Failed
a_{14}	1	4	1	2	3	$\max(l_{ik}l_{kj}) = 6$
	1	4	2	4	10	$\min(u_{ik}u_{kj}) = 3$
	1	4	3	6	10	Failed
a_{23}	2	3	1	1/2	2	$\max(l_{ik}l_{kj}) = 3$
	2	3	2	3	5	$\min(u_{ik}u_{kj}) = 5/6$
	2	3	4	1/2	5/6	Failed
a_{24}	2	4	1	1	3	$\max(l_{ik}l_{kj}) = 18$
	2	4	2	4	5	$\min(u_{ik}u_{kj}) = 3$
	2	4	3	18	40	Failed
a_{34}	3	4	1	1	3	$\max(l_{ik}l_{kj}) = 6$
	3	4	2	4/5	5/3	$\min(u_{ik}u_{kj}) = 5/3$
	3	4	3	6	8	Failed

convincing or logical that only a point estimate could result from all interval comparisons. It is argued that the presence of inconsistency in interval comparison matrices should not mean that no acceptable weight vector could be found (Arbel's LP technique) or only an acceptable point priority vector could be generated (Islam et al.'s LGP and Mikhailov's FPP methods). The reason for this is straightforward and rather simple. As is well known in AHP, inconsistency exists widely in comparison matrices. Its existence affects only the degrees of belief rather than the existence of priority vectors. Therefore, it seems more logical and acceptable that interval comparison matrices in general result in satisfactory interval priority vectors.

The proposed EM-based nonlinear programming models (12)–(16) are solved to generate the interval weights under different CR values. The results are presented in Table 7, from which it is clear that when there is no restriction on the inconsistency of the interval comparison matrix the weight intervals are the widest, when the inconsistency is restricted to be less than or equal to 0.1 the weight intervals become narrower, and when the consistency is restricted to minimum the priorities degenerate to be a point estimate. Fig. 5 depicts the corresponding directed diagrams on the preference structures among interval weights under different CR values. It is clear from Fig. 5 that the three cases all lead to the identical ranking: $w_2 \succ w_1 \succ w_3 \succ w_4$, but their degrees of preference are not the same. Although the above ranking order is the same as the

Table 7
The intervals of priorities for Example 2

Priority	min CR = 0.0831			No limitation on CR			With limitation of CR $\leqslant$ 0.1		
	w_i^L	w_i^U	Rank	w_i^L	w_i^U	Rank	w_i^L	w_i^U	Rank
w_1	0.2812	0.2812	2	0.2282	0.3830	2	0.2577	0.3259	2
w_2	0.4132	0.4132	1	0.3264	0.4758	1	0.3878	0.4297	1
w_3	0.2386	0.2386	3	0.1792	0.2819	3	0.2116	0.2579	3
w_4	0.0670	0.0670	4	0.0562	0.0843	4	0.0637	0.0743	4

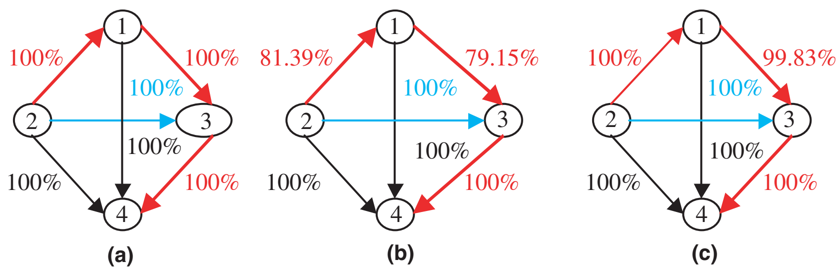

Fig. 5. Preference relations under different CR constraints in Example 2. (a) CR = 0.0831. (b) No limitation on CR. (c) CR $\leqslant$ 0.1.

ranking given by Islam et al. [14], our ranking provides much richer information about the priorities and are thus more convincing.

Example 3. Consider a hierarchy of criteria, which is taken from Islam et al. [14] and shown in Fig. 6. A person is interested in investing his money to any

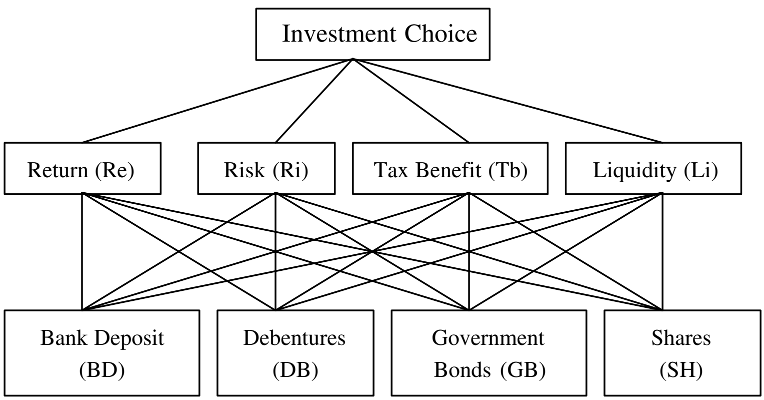

Fig. 6. Hierarchy structure.

one of the four portfolios: bank deposit (BD), debentures (DB), government bonds (GB), and shares (SH). Out of these portfolios he has to choose only one

Table 8
Interval comparison matrix for all the criteria with respect to 'investment choice (Ic)'

Ic	Re	Ri	Tb	Li
Re	1	[3, 4]	[5, 6]	[6, 7]
Ri		1	[4, 5]	[5, 6]
Tb			1	[3, 4]
Li				1

Table 9
Interval comparison matrix for all the alternatives with respect to the criterion 'return (Re)'

Re	BD	DB	GB	SH
BD	1	[1/4, 1/3]	[3, 4]	[1/6, 1/5]
DB		1	[6, 7]	[1/5, 1/4]
GB			1	[1/7, 1/6]
SH				1

Table 10
Interval comparison matrix for all the alternatives with respect to the criterion 'risk (Ri)'

Ri	BD	DB	GB	SH
BD	1	[3, 4]	[4, 5]	[6, 7]
DB		1	[3, 4]	[5, 6]
GB			1	[4, 5]
SH				1

Table 11
Interval comparison matrix for all the alternatives with respect to the criterion 'tax benefits (Tb)'

Tb	BD	DB	GB	SH
BD	1	1	[1/6, 1/5]	[1/4, 1/3]
DB		1	[1/6, 1/5]	[1/4, 1/3]
GB			1	[4, 5]
SH				1

Table 12
Interval comparison matrix for all the alternatives with respect to the criterion 'liquidity (Li)'

Li	BD	DB	GB	SH
BD	1	[3, 4]	6	[6, 7]
DB		1	[3, 4]	[3, 4]
GB			1	[3, 4]
SH				1

Table 13
Local and global priority intervals obtained under the constraint of CR ⩽ 0.1

Portfolio	Re	Ri	Tb	Li	Global priority
	[0.5365, 0.6014]	[0.2481, 0.3083]	[0.0915, 0.1197]	[0.0471, 0.0577]	
BD	[0.0936, 0.1210]	[0.5233, 0.5904]	[0.0736, 0.0922]	[0.5607, 0.6139]	[0.2198, 0.2915]
DB	[0.2209, 0.2691]	[0.2362, 0.2999]	[0.0736, 0.0922]	[0.2119, 0.2678]	[0.2065, 0.2623]
GB	[0.0454, 0.0545]	[0.1135, 0.1486]	[0.5806, 0.6370]	[0.1010, 0.1303]	[0.1146, 0.1557]
SH	[0.5794, 0.6295]	[0.0458, 0.0556]	[0.1992, 0.2541]	[0.0526, 0.0676]	[0.3474, 0.4218]

based upon four criteria: return (Re), risk (Ri), tax benefits (Tb), and liquidity (Li).

The interval comparison matrices for all the criteria as well as for all the alternatives are summarized in Tables 8–12. It can be shown using Theorem 1 that the five interval comparison matrices are all inconsistent. Therefore, the EM-based NLP method is used to generate local interval weights with satisfactory consistency from the five interval comparison matrices. The results are reported in Table 13, where the global interval weights are obtained by solving LP model (21)–(24). The preference relations between the global interval weights are recorded in Fig. 7(a), from which it is clear that $SH \overset{100\%}{\succ} BD \overset{66.67\%}{\succ} DB \overset{100\%}{\succ} GB$. If inconsistency (CR) is restricted to minimum, all the priorities will be degenerated to be point estimates. Table 14 shows the corresponding local and global priorities. The corresponding directed diagram for the global priorities is depicted in Fig. 7(b), from which an absolutely dominant ranking order $SH \overset{100\%}{\succ} BD \overset{100\%}{\succ} DB \overset{100\%}{\succ} GB$ is obtained. So, investing his money to shares (SH) is the best choice for the decision maker (DM).

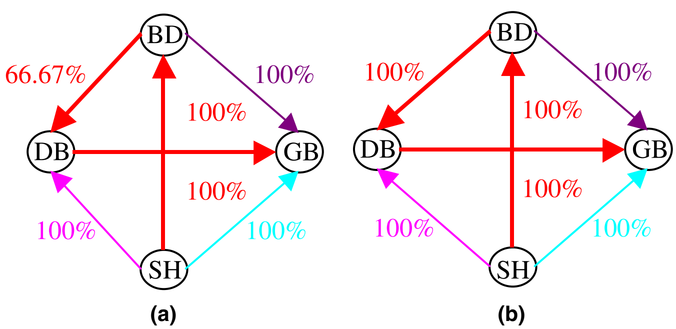

Fig. 7. Preference relations in Example 3. (a) CR ⩽ 0.1. (b) min CR.

Table 14
Local and global priorities obtained through the minimum of CR

Portfolio	Re	Ri	Tb	Li	Global priority	Rank
	0.5600	0.2811	0.1058	0.0531		
BD	0.1085	0.5415	0.0847	0.5831	0.2529	2
DB	0.2440	0.2712	0.0847	0.2454	0.2349	3
GB	0.0508	0.1360	0.6145	0.1132	0.1377	4
SH	0.5967	0.0513	0.2161	0.0583	0.3745	1

7. Concluding remarks

The use of pairwise comparisons to generate relative weights of criteria in multiple criteria decision analysis requires human judgments. Because of the complexity of real world decision problems and the subjective nature of human judgments, interval comparison matrices can provide a more realistic framework to account for such uncertainty than conventional crisp comparison matrices. This is especially the case in a group decision making situation. However, how to derive weights from interval comparison matrices, especially from inconsistency interval comparison matrices, still remains a research topic that needs to be further studied.

In this paper, the definition of consistency for interval comparison matrices was provided and a simple yet pragmatic approach for checking whether or not an interval comparison matrix is consistent was put forward without having to solve any mathematical program. In the case of consistent interval comparison matrices, Arbel's preference programming method was recommended to derive consistent interval weights; otherwise, an EM-based NLP method was developed to generate satisfactory interval weights. The degree of inconsistency or consistency ratio (CR) can be determined or adjusted by the decision maker. This is one of the most favorable features of the proposed method over the other existing weight estimation methods as mentioned in this paper. The synthesis approach of interval weights was also investigated. A simple and effective preference ranking approach was introduced and utilized to compare the interval weights of criteria or rank alternatives. Three numerical examples illustrated the validity and wide applicability of the proposed methods. Since fuzzy comparison matrices may be transformed into interval comparison matrices using α-*cuts*, the proposed methods are also applicable to fuzzy comparison matrices. Therefore, they can be widely used to deal with decision problems that can be modeled using interval, fuzzy and group comparison matrices.

References

[1] A. Arbel, Approximate articulation of preference and priority derivation, European Journal of Operational Research 43 (1989) 317–326.

[2] A. Arbel, A linear programming approach for processing approximate articulation of preference, in: P. Korhonen, A. Lewandowski, J. Wallenius (Eds.), Multiple Criteria Decision Support, Lecture Notes in Economics and Mathematical Systems, vol. 356, Spring-Verlag, Berlin, 1991, pp. 79–86.

[3] A. Arbel, L.G. Vargas, The analytic hierarchy process with interval judgments, in: A. Goicoechea, L. Duckstein, S. Zoints (Eds.), Multiple Criteria Decision Making. Proceedings of the Ninth International Conference held in Fairfax, Virginia, 1990, Springer, New York, 1992, pp. 61–70.

[4] A. Arbel, L.G. Vargas, Preference simulation and preference programming: robustness issues in priority deviation, European Journal of Operational Research 69 (1993) 200–209.
[5] J. Barzilai, W.D. Cook, B. Golany, Consistent weights for judgments matrices of the relative importance of alternatives, Operations Research 6 (3) (1987) 131–134.
[6] J. Barzilai, Deriving weights from pairwise comparison matrices, Journal of Operational Research Society 48 (1997) 1226–1232.
[7] C.G.E. Bonder, J.G. de Graan, F.A. Lootsma, Multicretia decision analysis with fuzzy pairwise comparisons, Fuzzy Sets and Systems 29 (1989) 133–143.
[8] J.J. Buckley, Fuzzy hierarchical analysis, Fuzzy Sets and Systems 17 (1985) 233–247.
[9] J.J. Buckley, T. Feuring, Y. Hayashi, Fuzzy hierarchical analysis revisited, European Journal of Operational Research 129 (2001) 48–64.
[10] G.B. Crawford, The geometric mean procedure for estimating the scale of a judgment matrix, Mathematical Modelling 9 (3–5) (1987) 327–334.
[11] G. Crawford, C.A. Williams, A note on the analysis of subjective judgment matrices, Journal of Mathematical Psychology 29 (1985) 387–405.
[12] R. Csutora, J.J. Buckley, Fuzzy hierarchical analysis: the Lamda–Max method, Fuzzy Sets and Systems 120 (2001) 181–195.
[13] L.M. Haines, A statistical approach to the analytic hierarchy process with interval judgments. (I). Distributions on feasible regions, European Journal of Operational Research 110 (1998) 112–125.
[14] R. Islam, M.P. Biswal, S.S. Alam, Preference programming and inconsistent interval judgments, European Journal of Operational Research 97 (1997) 53–62.
[15] H. Ishibuchi, H. Tanaka, Multiobjective programming in optimization of the interval objective function, European Journal of Operational Research 48 (1990) 219–225.
[16] M. Kress, Approximate articulation of preference and priority derivation—a comment, European Journal of Operational Research 52 (1991) 382–383.
[17] S. Kunda, Min-transitivity of fuzzy leftness relationship and its application to decision making, Fuzzy Sets and Systems 86 (1997) 357–367.
[18] L.C. Leung, D. Cao, On consistency and ranking of alternatives in fuzzy AHP, European Journal of Operational Research 124 (2000) 102–113.
[19] L. Mikhailov, Fuzzy analytical approach to partnership selection in formation of virtual enterprises, Omega 30 (2002) 393–401.
[20] L. Mikhailov, Deriving priorities from fuzzy pairwise comparison judgments, Fuzzy Sets and Systems 134 (2003) 365–385.
[21] L. Mikhailov, Group prioritization in the AHP by fuzzy preference programming method, Computers & Operations Research 31 (2004) 293–301.
[22] J.M. Moreno-Jimenez, A probabilistic study of preference structures in the analytic hierarchy process with interval judgments, Mathematical and Computer Modelling 17 (4/5) (1993) 73–81.
[23] T.L. Saaty, A scaling method for priorities in hierarchical structures, Journal of Mathematical Psychology 15 (1977) 234–281.
[24] T.L. Saaty, The Analytic Hierarchy Process, McGraw-Hill, New York, 1980.
[25] T.L. Saaty, Multicriteria Decision Making: The Analytic Hierarchy Process, RWS Publications, Pittsburgh, PA, 1988.
[26] T.L. Saaty, Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process, RWS Publications, Pittsburgh, PA, 1994.
[27] T.L. Saaty, L.G. Vargas, Comparison of eigenvalue, logarithmic least squares and least squares methods in estimating ratios, Mathematical Modelling 5 (1984) 309–324.
[28] T.L. Saaty, L.G. Vargas, Uncertainty and rank order in the analytic hierarchy process, European Journal of Operational Research 32 (1987) 107–117.

[29] A. Salo, R.P. Hämäläinen, Processing interval judgments in the analytic hierarchy process, in: A. Goicoechea, L. Duckstein, S. Zoints (Eds.), Multiple Criteria Decision Making. Proceedings of the Ninth International Conference held in Fairfax, Virginia, 1990, Springer, New York, 1992, pp. 359–372.
[31] A. Salo, R.P. Hämäläinen, Preference programming through approximate ratio comparisons, European Journal of Operational Research 82 (1995) 458–475.
[32] A. Sengupta, T.K. Pal, On comparing interval numbers, European Journal of Operational Research 127 (2000) 28–43.
[33] R. Xu, Fuzzy least-squares priority method in the analytic hierarchy process, Fuzzy Sets and Systems 112 (2000) 359–404.
[34] R. Xu, X. Zhai, Fuzzy logarithmic least squares ranking method in analytic hierarchy process, Fuzzy Sets and Systems 77 (1996) 175–190.
[35] P.J.M. Van Laarhoven, W. Pedrycz, A fuzzy extension of Saaty's priority theory, Fuzzy Sets and Systems 11 (1983) 229–241.
[36] Y.M. Wang, J.B. Yang, D.L. Xu, A preference aggregation method through the estimation of utility intervals, Computers & Operations Research (in press).