



HAL
open science

Optimal Inconsistency Repairing of Pairwise Comparison Matrices Using Integrated Linear Programming and Eigenvector Methods

Haiqing Zhang, Aicha Sekhari, Yacine Ouzrout, Abdelaziz Bouras

► **To cite this version:**

Haiqing Zhang, Aicha Sekhari, Yacine Ouzrout, Abdelaziz Bouras. Optimal Inconsistency Repairing of Pairwise Comparison Matrices Using Integrated Linear Programming and Eigenvector Methods. *Mathematical Problems in Engineering*, 2014, 2014, pp.1-16. 10.1155/2014/989726 . hal-04170150

HAL Id: hal-04170150

<https://hal.science/hal-04170150>

Submitted on 25 Jul 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Research Article

Optimal Inconsistency Repairing of Pairwise Comparison Matrices Using Integrated Linear Programming and Eigenvector Methods

Haiqing Zhang,¹ Aicha Sekhari,¹ Yacine Ouzrout,¹ and Abdelaziz Bouras^{1,2}

¹ DISP Laboratory, Lumière University Lyon 2, 160 boulevard de l'Université 69676, Bron Cedex, France

² Computer Science Department, Faculty of Engineering, Qatar University & ictQATAR, P.O. Box 2731, Doha, Qatar

Correspondence should be addressed to Haiqing Zhang; haiqing.zhang.zhq@gmail.com

Received 1 October 2013; Accepted 18 December 2013; Published 17 March 2014

Academic Editor: Hao-Chun Lu

Copyright © 2014 Haiqing Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Satisfying consistency requirements of pairwise comparison matrix (PCM) is a critical step in decision making methodologies. An algorithm has been proposed to find a new modified consistent PCM in which it can replace the original inconsistent PCM in analytic hierarchy process (AHP) or in fuzzy AHP. This paper defines the modified consistent PCM by the original inconsistent PCM and an adjustable consistent PCM combined. The algorithm adopts a *segment tree* to gradually approach the greatest lower bound of the distance with the original PCM to obtain the middle value of an adjustable PCM. It also proposes a theorem to obtain the lower value and the upper value of an adjustable PCM based on two constraints. The experiments for crisp elements show that the proposed approach can preserve more of the original information than previous works of the same consistent value. The convergence rate of our algorithm is significantly faster than previous works with respect to different parameters. The experiments for fuzzy elements show that our method could obtain suitable modified fuzzy PCMs.

1. Introduction

Analytic Hierarchy Process (AHP) is developed by Saaty [1], which is a multicriterion decision-making methodology widely used in many real problems [2, 3]. The AHP methodology expresses the relative importance of criteria by pairwise comparisons and converts the values of pairwise comparisons to priorities. Fuzzy AHP methodology [4] is an advanced AHP methodology, which is used to tackle the uncertainty and inaccurate problems in multicriteria decision-making process. Fuzzy AHP derives the fuzzy priorities of criteria from pairwise comparisons matrix with triangular (or trapezoidal) fuzzy elements. To make sure the priorities of each criterion are accurate and sensible, consistency of pairwise comparison matrix (PCM) with crisp or fuzzy elements must be achieved.

Several works [5–8] focus on reducing inconsistent PCMs with crisp numbers. Karapetrovic and Rosenbloom [5] revised the single entry of a ratio's value till the consistency of relative matrix was at an acceptable level. Xu and Wei [6]

preserved the initial ratios' value in the pairwise comparison matrix while obtaining satisfactory consistency requirements. Cao et al. [7] developed a heuristic approach, which can preserve more of the original information compared to Xu and Wei [6]. However, for these three works, when the consistency requirement is increased, the computing times will be largely increased and the information on the original matrix cannot be well preserved. Anholcer's [8] work is used to minimize the distance between inconsistent PCMs and their corresponding consistent PCM. The aim of this work is to find out a new modified PCM which is consistent and has the closest resemblance to the optimal one. But the modified new matrix has a long distance with the original matrix according to the analysis of the parameters given in [6, 7].

Some works used to solve the inconsistency of PCMs with fuzzy elements are given in [9–12]. Xu and Wang [9] repaired incomplete and inconsistent fuzzy preference relations by finding out the unusual and false element until the consistency ratio was at a satisfactory level. Leung and Cao [10] proposed a new definition of fuzzy positive reciprocal matrix

by setting deviation tolerances based on an idea of allowing inconsistent information. Morteza and Bafandeh [11] further discussed Leung and Cao's work and proposed a new method of fuzzy consistency tests by direct fuzzification of a QR (quick response) algorithm, which is one of the methods for the eigenvalues calculation of an arbitrary matrix. Wang and Chen [12] applied fuzzy linguistic preference relations to construct consistent PCM's by considering reducing the number of pairwise comparisons. However, these works do not have standard parameters to verify the reliability of their theory so far. Therefore, it is very important to prove the feasibility of a methodology that can reduce the inconsistency of the original matrix and preserve the original matrix's information as much as possible.

In this research work, we propose a modified consistent PCM as a combination of original inconsistent PCM and an adjustable consistent PCM. In order to achieve the modified PCM, this paper is structured in the following way: Section 2 and Section 3 give the basic concepts of the PCM with fuzzy and crisp elements, consistency indices for crisp and fuzzy elements, and parameters to judge the effectiveness of the modified matrix. Section 4 and Section 5 propose an algorithm to obtain the middle value, upper value, and lower value of the adjustable matrix. The main idea of our algorithm is to find the optimum priority vector by solving a linear programming problem and use the eigenvector method to obtain the adjustable matrix based on the optimum priority vector. In Section 6, the algorithm was applied to obtain the adjustable PCM by adopting the same illustrating PCM in [6, 7]; the comparison results show that our algorithm can preserve more original information than Cao et al. [7] and Xu and Wei [6]. In Section 7, the algorithm was used to find the sensible and the closest consistent modified PCM with fuzzy elements. The experiments show that the new modified matrix can satisfy consistent indices' requirements of NI [13] and CCI [14, 15] and also can preserve more original information under parameters of σ_{fuzzy} and δ_{fuzzy} . Section 8 concludes with effective and efficient analysis to show that our algorithm can be performed easily.

2. State of the Art

2.1. Notations and Definitions. The $m \times n$ pairwise comparison matrix \tilde{A} with triangular fuzzy elements can be described in the following:

$$\tilde{A} = \begin{pmatrix} (a_{11}^L, a_{11}^M, a_{11}^U) & \cdots & (a_{1n}^L, a_{1n}^M, a_{1n}^U) \\ \vdots & \ddots & \vdots \\ (a_{m1}^L, a_{m1}^M, a_{m1}^U) & \cdots & (a_{mn}^L, a_{mn}^M, a_{mn}^U) \end{pmatrix}, \quad (1)$$

where for each element $\tilde{a}_{ij} = (a_{ij}^L, a_{ij}^M, a_{ij}^U)$, a_{ij}^L is the lower value, a_{ij}^M is the middle value, and a_{ij}^U is the upper value. In the

particular situation, if $m = n$, when the following condition is satisfied, then \tilde{A} is a reciprocal matrix:

$$\tilde{a}_{ij} = (a_{ij}^L, a_{ij}^M, a_{ij}^U) \quad \text{implies} \quad \tilde{a}_{ji} = \left(\frac{1}{a_{ij}^U}, \frac{1}{a_{ij}^M}, \frac{1}{a_{ij}^L} \right) \quad (2)$$

$$\forall i, j = 1, 2, \dots, n,$$

where $1/\sigma < a_{ij}^L < a_{ij}^M < a_{ij}^U < \sigma$ and $S = [1/\sigma, \sigma]$, $\sigma > 1$. S is an interval of real numbers called fuzzy scale. Crisp numbers (nonfuzzy numbers) are special cases of \tilde{A} when $a_{ij}^L = a_{ij}^M = a_{ij}^U$.

Definition 1. Asit was proposed by Buckley [16], a fuzzy positive reciprocal matrix $\tilde{A} = [\tilde{a}_{ij}]_{n \times n}$ is consistent if and only if $\tilde{a}_{ik} \odot \tilde{a}_{kj} \approx \tilde{a}_{ij}$, $\{\forall i, j, k \mid 1 \leq i, j, k \leq n\}$.

Where the operator \odot is one of the operation rules of triangular fuzzy elements, this operation can be calculated by the following equation:

$$\begin{aligned} \tilde{a}_{ik} \odot \tilde{a}_{kj} &= (a_{ik}^L, a_{ik}^M, a_{ik}^U) \odot (a_{kj}^L, a_{kj}^M, a_{kj}^U) \\ &= (a_{ik}^L \cdot a_{kj}^L, a_{ik}^M \cdot a_{kj}^M, a_{ik}^U \cdot a_{kj}^U) \\ &\approx \tilde{a}_{ij} = (a_{ij}^L, a_{ij}^M, a_{ij}^U). \end{aligned} \quad (3)$$

When positive reciprocal matrix \tilde{A} is crisp numbers A , then the consistent condition is $a_{ik} \cdot a_{kj} = a_{ij}$, $\{\forall i, j, k \mid 1 \leq i, j, k \leq n\}$.

However, this definition is too strict, because it is unrealistic to reach perfect consistency of a PCM (crisp or fuzzy elements). Some works [10, 13, 17–23] have developed consistency indices to accept a certain level of deviations. We will adopt consistency indices in Section 2.2 to determine whether the current PCM matrix is at an acceptable consistency level.

2.2. Consistency Indices. Several consistency indices have been proposed for crisp numbers: for instance, geometric consistency index [17], singular value decomposition method [18], and harmonic consistency index [19]. However, it was proven that all these consistency indices are linear or non-linear transformations of Satty's CR [20]. Therefore, we use Satty's CR in this paper for measuring consistency of crisp numbers.

The consistency index of CR [20] is defined as follows:

$$\text{CR} = \frac{\lambda_{\max} - n}{n - 1} \cdot \frac{1}{\text{RI}}, \quad (4)$$

where λ_{\max} is the principle eigenvalue of A . RI is random index, which can be gotten by searching a defined table. When the value of $0 < \text{CR} < 0.1$, the consistency can be accepted.

Several important works focus on the consistency of pairwise comparison matrix with fuzzy elements. The first one is Leung and Cao [10], who proposed a notion with consideration of a tolerance deviation. However, the notion

is strongly related to Satty's CR and it has shortcomings to calculate consistency of pairwise comparison matrix with fuzzy elements [11]. The second one is Ramík and Korviny's [13] work, which proposes a new consistency index NI to examine fuzzy elements based on the distance of the matrix to a special ratio matrix and compare the properties with CR. This work has been further studied and has been used by several important works [21–23]. We have tested this work's performance. The results indicated that it can satisfy reasonable results with fuzzy elements, although it has some shortcomings [23].

The consistency index of NI [13] is defined as follows:

$$NI_n^\sigma = \gamma_n^\sigma \cdot \max_{i,j} \left\{ \max \left\{ \left| \frac{\omega_i^L}{\omega_j^U} - a_{ij}^L \right|, \left| \frac{\omega_i^M}{\omega_j^M} - a_{ij}^M \right|, \left| \frac{\omega_i^U}{\omega_j^L} - a_{ij}^U \right| \right\} \right\} \quad (1 \leq i, j \leq n), \quad (5)$$

where γ_n^σ is a normalization factor and the values of ω_i^L , ω_i^M , and ω_i^U can be obtained from a_{ij}^L , a_{ij}^M , and a_{ij}^U .

Another successful index that can be extracted from [14, 15] is that they extend GCI (geometric consistency index) [24] to CCI (centric consistency index) to deal with PCM with triangle fuzzy elements. The consistency index CCI is defined as follows:

$$CCI(\tilde{A}) = \frac{2}{(n-1)(n-2)} \times \sum_{i < j} \left(\log \left(\frac{a_{ij}^L + a_{ij}^M + a_{ij}^U}{3} \right) - \log \left(\frac{\omega_i^L + \omega_i^M + \omega_i^U}{\omega_j^L + \omega_j^M + \omega_j^U} \right) \right)^2, \quad (6)$$

where $\tilde{A} = (a_{ij}^L, a_{ij}^M, a_{ij}^U)$ are the elements for fuzzy PCM; $\tilde{\omega} = (\tilde{\omega}_i)$ is a priority vector derived by logarithmic least squares. When $CCI(\tilde{A}) = 0$, \tilde{A} is considered fully consistent. Thresholds remain identical with index GCI as CCI being a fuzzy extension of GCI. The thresholds are provided as follows: CCI = 0.3147 for $n = 3$, CCI = 0.3526 for $n = 4$, and CCI = 0.370 for $n > 4$ based on Aguarón and Moreno-Jiménez [24].

In conclusion, we will adopt CR to examine the consistency of crisp elements and use NI and CCI to examine the consistency of fuzzy elements.

2.3. Parameters to Judge the Effectiveness of Modified Matrix. Next, we are going to describe the necessary parameters which can be used to measure the effectiveness of modified

PCM *B*. Xu and Wei [6] have given two parameters in the following:

$$\delta = \max_{i,j} \{|b_{ij} - a_{ij}|\}, \quad i, j = 1, 2, \dots, n, \quad (7)$$

$$\sigma = \frac{\sqrt{\sum_{i=1}^n \sum_{j=1}^n (b_{ij} - a_{ij})^2}}{n}. \quad (8)$$

σ and δ are used as the parameters of modificatory effectiveness. The authors of [6, 7] argue that a modified matrix that preserves the most information of the original one must satisfy the following condition: $0 < \delta < 2$ and $0 < \sigma < 1$. Extending these two parameters to be suitable to judge modificatory effectiveness of fuzzy elements so the range is identical with σ and δ , σ_{fuzzy} and δ_{fuzzy} are

$$\begin{aligned} \delta_{\text{fuzzy}} &= \max_{i,j} \{|\tilde{b}_{ij} - \tilde{a}_{ij}|\} \\ &= \max_{i,j} \{|b_{ij}^L - a_{ij}^L|, |b_{ij}^M - a_{ij}^M|, |b_{ij}^U - a_{ij}^U|\}, \quad i, j = 1, 2, \dots, n, \\ \sigma_{\text{fuzzy}} &= \left(\max \left\{ \left(\sum_{i=1}^n \sum_{j=1}^n (b_{ij}^L - a_{ij}^L)^2 \right), \left(\sum_{i=1}^n \sum_{j=1}^n (b_{ij}^M - a_{ij}^M)^2 \right), \left(\sum_{i=1}^n \sum_{j=1}^n (b_{ij}^U - a_{ij}^U)^2 \right) \right\} \right)^{1/2} n^{-1}. \end{aligned} \quad (9)$$

Besides σ and δ , two parameters proposed by Xu and Wei [6], we propose a third parameter should be added, which is Condition of Order Preservation (COP) (Table 3) [25]. For example, suppose the original matrix *A* has alternatives (a_1, a_2, a_3 , and a_4); it has that the relationship a_1 dominates a_2 and a_3 dominates a_4 , and the judgments indicate that the extent to which a_1 dominates a_2 is greater than the extent to which a_3 dominates a_4 ; then the priority vector ω should satisfy $\omega(a_1) > \omega(a_2)$ and $\omega(a_3) > \omega(a_4)$ (preservation of order of preference) and $\omega(a_1)/\omega(a_2) > \omega(a_3)/\omega(a_4)$ (preservation of order of intensity of preference).

3. Main Theories to Obtain Modified PCM

3.1. Distance Analysis between Original Matrix and Modified Matrix. To measure the distance from original PCM *A* (or \tilde{A}), several methods can be used [26–30]. The PCM \tilde{A} can be considered as the combination of three matrixes of PCM *A*. If the measuring distance methodologies for crisp *A* have been properly handled, then they can be suitable for fuzzy \tilde{A} as well. We will first discuss the distance methods for crisp numbers. The measurement has logarithmic least square method (LSM) [26, 31], eigenvector method [27, 28], and least

squares method [29, 30]. The optimum eigenvector should be as close as possible to the original eigenvector (derived from the original matrix). The adjustable matrix could strongly resemble the original matrix once the optimum eigenvector has been determined. Therefore, eigenvector can be used to calculate adjustable PCM B' (\tilde{B}').

According to [27, 28, 32], if matrix A is consistent, then we could find positive weights $\omega = \{\omega_1, \omega_2, \dots, \omega_n\}^T$ which can satisfy such condition $a_{ij} = \omega_i/\omega_j, \forall i, j \in (1, 2, \dots, n)$. Therefore, if matrix A is close to consistent, then it must have $(\omega_i/\omega_j) - a_{ij} \approx 0$. On the basis of this idea, we want to obtain the minimum value of the fastest distance between ω_i/ω_j and a_{ij} . The fastest distance between ω_i/ω_j and a_{ij} is defined as follows:

$$f = \max \left(\left| a_{ij} - \frac{\omega_i}{\omega_j} \right| \right). \quad (10)$$

The issue of finding the optimal modified matrix can be portrayed as resolving the minimum value of function f , which can be expressed in the following equation:

$$G(A, \omega) = \min(f) = \min \left(\max \left(\left| a_{ij} - \frac{\omega_i}{\omega_j} \right| \right) \right). \quad (11)$$

This equation can reach the absolute minimum value (lowest point) when even the worst situation of proportion of ω_i/ω_j is closest to a_{ij} . The method to obtain the optimum positive eigenvector ω is to find out all possible constraints and obtain feasible solutions by selecting a suitable linear programming pattern.

3.2. Constraints Analysis for Achieving Modified Matrix. The new modified PCM B (or \tilde{B}) should satisfy three conditions: (1) the consistent value should be at an acceptable level; (2) the farthest distance between new PCM B (or \tilde{B}) and original PCM A (or \tilde{A}) should be as small as possible; (3) the obtained new matrix should have a strong similarity with the original matrix. These three conditions can guarantee a new consistent PCM B (or \tilde{B}) which has the closest and maximal similarity with the original inconsistent PCM A (or \tilde{A}). To suit these conditions, we provide an adjustable PCM which can reduce the inconsistency of original PCM as much as possible. On the basis of the adjustable matrix, we propose the definition of the modified matrix.

Definition 2. The new modified PCM B is defined as a combination of the original PCM A and an adjustable matrix called B' , which is derived from A ; the modified PCM B is defined in

$$B = a_{ij}^\beta (b_{ij}')^{1-\beta}. \quad (12)$$

For fuzzy elements, the new modified PCM \tilde{B} is defined as follows:

$$\tilde{B} = \left\{ \left((a_{ij}^L)^\beta \cdot (b_{ij}'^{L'})^{1-\beta} \right), \left((a_{ij}^M)^\beta \cdot (b_{ij}'^{M'})^{1-\beta} \right), \left((a_{ij}^U)^\beta \cdot (b_{ij}'^{U'})^{1-\beta} \right) \right\}. \quad (13)$$

The properties of the adjustable PCM B (or \tilde{B}) and the way to obtain them will be studied in Section 4.

4. Calculation Processes of Obtaining Adjustable Matrix

4.1. Problem Statement. The aim of this paper is to find out a consistent matrix $\tilde{B} = [\tilde{b}_{ij}]_{n \times n}$ which can preserve the most information of the original matrix $\tilde{A} = [\tilde{a}_{ij}]_{n \times n}$ and be the closest to \tilde{A} . One method to achieve this aim is that setting one element of \tilde{B} is $\tilde{b}_{ij} = (b_{ij}^L, b_{ij}^M, b_{ij}^U)$ and one element of \tilde{A} is $\tilde{a}_{ij} = (a_{ij}^L, a_{ij}^M, a_{ij}^U)$, by making every element of \tilde{B} closely resemble the elements of \tilde{A} , which is $b_{ij}^L \doteq a_{ij}^L, b_{ij}^M \doteq a_{ij}^M$, and $b_{ij}^U \doteq a_{ij}^U$. Therefore, we go to find $b_{ij}^L, b_{ij}^M, b_{ij}^U$ separately. After finding out $b_{ij}^L, b_{ij}^M, b_{ij}^U$, combine them together to get the new matrix \tilde{B} .

On the basis of this idea, we build an adjustable matrix $\tilde{B}' = [\tilde{b}'_{ij}]_{n \times n}$ which can reduce the inconsistency of the original matrix. The new modified matrix is constructed by two parts. (1) One is the original matrix. The function of this matrix is to keep the original information and make sure the two matrixes are in an acceptable distance. (2) One is the adjustable matrix. The function of this matrix is to modify the inconsistency level to make sure the new modified matrix's consistency is based on consistency indexes. The mathematical expression objective is as follows:

$$\tilde{B} = \tilde{A} \odot \tilde{B}'. \quad (14)$$

The mathematical expression objective can be developed as

$$\tilde{B} = \left\{ \left((a_{ij}^L)^\beta \cdot (b_{ij}'^{L'})^{1-\beta} \right), \left((a_{ij}^M)^\beta \cdot (b_{ij}'^{M'})^{1-\beta} \right), \left((a_{ij}^U)^\beta \cdot (b_{ij}'^{U'})^{1-\beta} \right) \right\}. \quad (15)$$

This equation is also suitable when $a_{ij}^L = a_{ij}^M = a_{ij}^U$, which is for crisp elements.

4.2. Stage 1: Specify Formulas to Obtain the Middle Value $b_{ij}^{M'}$ of the Adjustable Matrix \tilde{B}' . First, we go to calculate $b_{ij}^{M'}$ of adjustable matrix \tilde{B}' . $b_{ij}^{M'}$ should have a strong relationship with a_{ij}^M . $\tilde{\omega}^M$ is the eigenvector matrix of a_{ij}^M . $b_{ij}^{M'}$ can be gotten in the following equation:

$$b_{ij}^{M'} \approx \frac{\omega_i^M}{\omega_j^M}. \quad (16)$$

The value can be gotten when the proportion is closest to a_{ij}^M . Equation (17) has feasible solutions which means the following equation reaches the minimum based value on (11):

$$G(a_{ij}^M, \tilde{\omega}^M) = \min \left(\max \left(\left| a_{ij}^M - \frac{\omega_i^M}{\omega_j^M} \right| \right) \right). \quad (17)$$

We can simply write the variable by introducing an additional variable $Z = G(a_{ij}^M, \tilde{\omega}^M)$. Then, the problem is as follows:

$$\begin{aligned} & \min Z \\ & \text{s.t.} \\ & \left| a_{ij}^M - \frac{\omega_i^M}{\omega_j^M} \right| \leq Z, \quad \forall i, j \in (1, 2, \dots, n) \end{aligned} \quad (18)$$

$$\frac{1}{\sigma} \leq \omega_i^M, \quad \omega_j^M \leq \sigma, \quad j = 1, 2, \dots, n. \quad (19)$$

(The calculation steps of (19) are given in the appendix.)

Assume the value of Z is given, and then the constraint (18) can be rewritten as follows:

$$-Z \leq \left(a_{ij}^M - \frac{\omega_i^M}{\omega_j^M} \right) \leq Z, \quad (20)$$

$$(a_{ij}^M - Z) \cdot \omega_j^M - \omega_i^M \leq 0, \quad (21)$$

$$\omega_i^M \leq (a_{ij}^M + Z) \cdot \omega_j^M. \quad (22)$$

Take the reciprocal of constraint (22); then the new constraints (23), (24), and (25) can be gotten:

$$\frac{\omega_j^M}{\omega_i^M} \geq \frac{1}{(a_{ij}^M + Z)}, \quad (23)$$

$$\frac{1}{(a_{ij}^M + Z)} \cdot \omega_i^M - \omega_j^M \leq 0, \quad (24)$$

$$\frac{1}{(a_{ji}^M + Z)} \cdot \omega_j^M - \omega_i^M \leq 0. \quad (25)$$

Combine constraint (21) and constraint (25) and remove one of the inequalities, and then the new inequality can be gotten:

$$-\omega_i^M + \left(\max \left\{ (a_{ij}^M - Z), \frac{1}{(a_{ji}^M + Z)} \right\} \right) \omega_j^M \leq 0. \quad (26)$$

Analogously, a similar inequality can be gotten:

$$-\omega_j^M + \left(\max \left\{ (a_{ji}^M - Z), \frac{1}{(a_{ij}^M + Z)} \right\} \right) \omega_i^M \leq 0. \quad (27)$$

Next, we add slack variable S, T , and objective function $H = S + T$, to change the constraints (26)-(27) to equality constraints

$$\begin{aligned} & -\omega_i^M + \left(\max \left\{ (a_{ij}^M - Z), \frac{1}{(a_{ji}^M + Z)} \right\} \right) \omega_j^M + S = 0, \\ & -\omega_j^M + \left(\max \left\{ (a_{ji}^M - Z), \frac{1}{(a_{ij}^M + Z)} \right\} \right) \omega_i^M + T = 0, \\ & H = S + T, \\ & 0 \leq S, T, H. \end{aligned} \quad (28)$$

Now we specify constraints, propose formulas to calculate ω_i^M, ω_j^M , and add additional stopping parameter X . Then constraints (28) correspond to the following linear programming problem:

$\min X$

s.t.

$$-\omega_i^M + \left(\max \left\{ (a_{ij}^M - Z), \frac{1}{(a_{ji}^M + Z)} \right\} \right) \omega_j^M + S = 0 \quad (29)$$

$$-\omega_j^M + \left(\max \left\{ (a_{ji}^M - Z), \frac{1}{(a_{ij}^M + Z)} \right\} \right) \omega_i^M + T = 0 \quad (30)$$

$$H = S + T \quad (31)$$

$$\omega_1^M + X = 1 \quad (32)$$

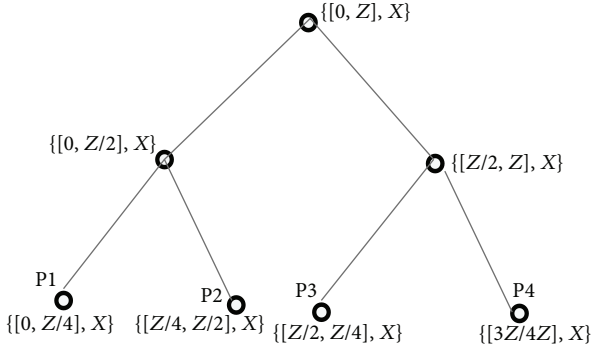
$$\frac{1}{\sigma} \leq \omega_i^M, \quad \omega_j^M \leq \sigma i, \quad j = 1, 2, \dots, n \quad (33)$$

$$0 \leq Z, S, T, H. \quad (34)$$

Note 1. (1) In constraint (29), we define $\omega_1^M = 1$ to normalize vector $\tilde{\omega}^M$. If the stopping parameter $X = 0$, then it means constraint (29) has a solution, and the problem (29)–(34) has a feasible solution; the value of ω_i^M is the optimal solution. If the stopping parameter is $X = 1$ which contradicts with constraint (32), then the problem (29)–(34) is inconsistent; then the value of ω_i^M is not the optimal solution.

(2) The equalities (29)–(34) can be solved by *simplex algorithm* [33]. The main idea of this algorithm is to walk along edges of the polytope to find out extreme points with lower and lower objective values till the minimum value is reached or an unbounded edge is visited. If the extreme point is reached, then the problem (29)–(34) has feasible solutions.

(3) If $Z = Z'$ can make problem (29)–(34) have a feasible solution, then it must have $Z \geq Z'$ that can also make the problem (29)–(34) have a feasible solution. In order to find the *greatest lower bound of Z*, we investigate it in Section 4.3.

FIGURE 1: The structure of the *segment tree*.

(4) If $i = j$, then $a_{ii}^M = 1$; then constraint (29) can be rewritten as follows: $-\omega_i^M + (\max\{a_{ii}^M - Z, 1/(a_{ii}^M + Z)\}) + S = 0$. This equality must always be satisfied. Analogously, constraint (30) is always satisfied. Then the equalities (29)–(34) can be used to find adjustable PCM B' for original PCM A .

(5) In this section, we aim to find out the least absolute worst distance by setting a more precise priority weights range and adding slack variables, and an iterative way to obtain feasible solutions will be proposed in Section 4.3.

4.3. Stage 2: Find Feasible Solutions to Obtain the Middle Value $b_{ij}^{M'}$ of Adjustable Matrix \tilde{B}' . Next, we focus on how to find out the *greatest lower bound* of Z . The problem can be described as storing intervals of $[0, Z_{\max}]$, analyzing the corresponding X value, and finding the *greatest lower bound* of Z that makes $X = 0$. This problem can be solved by *segment tree*.

Substep 1 (sets the initial value). Assume the accuracy level is ξ ; let the initial value of ω_i^M be $(\prod_{j=1}^n a_{ij}^M)^{1/n}$, and $\omega_1^M = 1$; let $Z_{\max} = Z = G(a_{ij}^M, \tilde{\omega}^M)$, and $Z_{\min} = Z = 0$.

Substep 2 (builds a segment tree by using interval $[0, Z]$). For example, in Figure 1, $[0, Z]$ is the interval, and P1, P2, P3, P4 is the list of distinct interval endpoints. We separate intervals into two parts in every division and terminate this process till the value of the interval is less than the accuracy level (ξ). Then obtain the X value in problem (29)–(34) by setting the current Z value. If $X = 0$, then the next Z value is equal to the lower bound of the current node. The calculation steps will end till it reaches endpoints (P1; P2; P3; P4) based on accuracy level ξ .

Substep 3. Select the greatest lower value of Z when $X = 0$; obtain the feasible solution of ω_i^M, ω_j^M to achieve $b_{ij}^{M'}$.

Note 2. The *segment tree* is special for storing intervals. The built time is $O(n \log n)$ for n intervals, and it uses $O(n \log n)$ storage. The reason we adapted to the *segment tree* is because the segments can be stored in any arbitrary manner, it can

easily be adapted to counting queries, and it helps us to query the number of segments that contain a given point.

4.4. Stage 3: Obtain the $b_{ij}^{L'}$ and $b_{ij}^{U'}$ of the Adjustable Matrix \tilde{B}' . Once the optimal solution ($b_{ij}^{M'}$) has been obtained, next, we focus on obtaining the value of $b_{ij}^{L'}$ and $b_{ij}^{U'}$. The modified matrix is a combination of the adjustable matrix and the original matrix based on Definition 2; then the adjustable matrix should have the minimum fuzziness and maximum preservation of the original matrix's pattern. If \tilde{B}' is minimum fuzziness, then fuzziness of \tilde{B} will mostly come from \tilde{A} , and \tilde{B} will be more similar with \tilde{A} . In fact, the minimum fuzziness of \tilde{B}' could reduce uncertainty factors of \tilde{B} . If \tilde{B}' could maximally preserve the pattern of \tilde{A} , then the combination of \tilde{B}' and \tilde{A} could reach the most potential of similarity with \tilde{A} . We propose Theorem 3 to obtain the value of $b_{ij}^{L'}$ and $b_{ij}^{U'}$ based on the above theory.

Theorem 3. *The optimal solution vector is ω_i^M ; then $B_{ij}^{M'} = \omega_i^M / \omega_j^M$, $i = 1, 2, \dots, n$. Set C_L, C_U as arbitrary positive constants. Define the value of ω_i^L, ω_i^U in the following formulas:*

$$\omega_i^L = C_L \cdot \omega_i^M,$$

$$\text{where } C_L = \min \left\{ \frac{\sum_{j=1}^n a_{ij}^M}{\sum_{j=1}^n a_{ij}^L}, \frac{(\prod_{j=1}^n a_{ij}^M)^{1/n}}{(\prod_{j=1}^n a_{ij}^L)^{1/n}} \right\},$$

$$\omega_i^U = C_U \cdot \omega_i^M,$$

$$\text{where } C_U = \max \left\{ \frac{\sum_{j=1}^n a_{ij}^M}{\sum_{j=1}^n a_{ij}^U}, \frac{(\prod_{j=1}^n a_{ij}^M)^{1/n}}{(\prod_{j=1}^n a_{ij}^U)^{1/n}} \right\}$$

Then the value of $b_{ij}^{L'}$ and $b_{ij}^{U'}$ is defined as follows:

$$b_{ij}^{L'} = \frac{\omega_i^L}{\omega_j^L}, \quad b_{ij}^{U'} = \frac{\omega_i^U}{\omega_j^U}. \quad (36)$$

Proof. The value of C_L and C_U should satisfy two conditions: one is minimum fuzziness of ω_i^M , and the other one is maximally maintaining similarity of original matrix \tilde{A} .

By the first condition, we can get

$$\omega_i^L \leq \omega_i^M \leq \omega_i^U,$$

$$\omega_i^L = C_{\min} \omega_i^M,$$

$$\omega_i^U = C_{\max} \omega_i^M. \quad (37)$$

By the second condition, we need to consider the distance between a_{ij}^L, a_{ij}^M , and a_{ij}^U , maintain the relationship among the original matrix, and make the new matrix closest to the original matrix's pattern. It means to find out the smallest coefficient between a_{ij}^L and a_{ij}^M and the smallest coefficient

TABLE 1: The values of parameters for each iterative time.

Iterative time	Z_{\max}	Z_{\min}	Z	ξ	X	ω
1	10.0725	0	5.0362	10.0725	0	ω_1
2	5.0362	2.5181	2.5181	2.5181	1	—
3	3.7772	2.5181	3.7772	1.2591	0	ω_2
4	3.7772	3.1476	3.1476	0.6296	1	—
5	3.7772	3.4624	3.4624	0.3148	1	—
6	3.7772	3.6198	3.6198	0.1574	1	—
7	3.7772	3.6985	3.6985	0.0787	0	ω_3

between a_{ij}^M and a_{ij}^U . We can get the following formulas based on the second condition:

$$C_{\min} = \min \left\{ \frac{\sum_{j=1}^n a_{ij}^M}{\sum_{j=1}^n a_{ij}^L}, \frac{(\prod_{j=1}^n a_{ij}^M)^{1/n}}{(\prod_{j=1}^n a_{ij}^L)^{1/n}} \right\}, \tag{38}$$

$$C_{\max} = \max \left\{ \frac{\sum_{j=1}^n a_{ij}^M}{\sum_{j=1}^n a_{ij}^U}, \frac{(\prod_{j=1}^n a_{ij}^M)^{1/n}}{(\prod_{j=1}^n a_{ij}^U)^{1/n}} \right\}.$$

$$A = \begin{pmatrix} 1.0000 & 5.0000 & 3.0000 & 7.0000 & 6.0000 & 6.0000 & 0.3333 & 0.2500 \\ 0.2000 & 1.0000 & 0.3333 & 5.0000 & 3.0000 & 3.0000 & 0.2000 & 0.1429 \\ 0.3333 & 3.0000 & 1.0000 & 6.0000 & 3.0000 & 4.0000 & 6.0000 & 0.2000 \\ 0.1429 & 0.2000 & 0.1667 & 1.0000 & 0.3333 & 0.2500 & 0.1429 & 0.1250 \\ 0.1667 & 0.3333 & 0.3333 & 3.0000 & 1.0000 & 0.5000 & 0.2000 & 0.1667 \\ 0.1667 & 0.3333 & 0.2500 & 4.0000 & 2.0000 & 1.0000 & 0.2000 & 0.1667 \\ 3.0000 & 5.0000 & 0.1667 & 7.0000 & 5.0000 & 5.0000 & 1.0000 & 0.5000 \\ 4.0000 & 7.0000 & 5.0000 & 8.0000 & 6.0000 & 6.0000 & 2.0000 & 1.0000 \end{pmatrix}. \tag{39}$$

For matrix A , $\lambda_{\max} = 9.669$, $CR = 0.161 > 0.1$, and the principal eigenvector is $\omega = (0.1730 \ 0.0540 \ 0.1881 \ 0.0175 \ 0.0310 \ 0.0363 \ 0.1668 \ 0.3332)^T$.

The value of CR is more than 0.1 . Therefore, we will adopt Algorithm 1 to obtain the new consistency matrix.

Calculation Step 1. Input of the initial value of \tilde{A} and initial value of ξ .

- (1) The original matrix $\tilde{A} = (a_{ij}^L, a_{ij}^M, a_{ij}^U)$; here, crisp number is a specific case of fuzzy number in our model. Set $a_{ij}^L = a_{ij}^M = a_{ij}^U = A$.
- (2) The acceptable precise degree $\xi = 0.1$. (Here, we adopt 0.1 as an example.)

Calculation Step 2. Calculation process.

- (1) Calculate the initial value of Z by (17).

Therefore, we can prove formula (35) exists and prove that it is correct by (38). To be more precise, for instance, in Figure 2, $(\tilde{b}_{ij})'_2$ is not satisfied with condition 1, and then it is not the minimum fuzziness of associated weights. If $(\tilde{b}_{ij})'_1$ is not satisfied with condition 2, then it does not have the same pattern with matrix \tilde{A} . $(\tilde{b}_{ij})'_3$ is the optimal solution. \square

5. An Algorithm to Obtain Modified PCM \tilde{B}

After analysis in Sections 4.2, 4.3, and 4.4, we propose an algorithm to conclude how to obtain the modified PCM \tilde{B} (see Algorithm 1).

6. Numerical Illustration and Comparison with Crisp Numbers

6.1. Calculation of an Illustration by Using Proposed Algorithm. We run the experiments by software Matlab (R2009a) on a personal computer with Intel Core 2.2 GHZ and 4 G RAM. First we test crisp numbers by using Algorithm 1 and then compare them with [6, 7].

The inconsistent matrix in [6, 7] is the following matrix:

- (2) First time $Z = 10.0725$; substitute Z into Algorithm 1; $Z_{\max} = 10.075$, $Z_{\min} = 0$, and $Z_{\max} - Z_{\min} > \xi$; set new $Z = (Z_{\max} - Z_{\min})/2$.
- (3) The second time, $Z = (10.0725 - 0)/2 = 5.0362$; substitute new Z into Algorithm 1; solve the problem (29)–(34); the result is $X = 0$ and $\omega = (1.0000 \ 0.3469 \ 4.0000 \ 0.5262 \ 0.2902 \ 0.1346 \ 0.9375 \ 1.4637)$. We put the following calculation in Table 1.

Calculation Step 3. Description parameters' meaning. The value of ξ is a stopping sign. If current value of ξ is less than 0.1 , stop the calculation process.

The value of X can determine how to change the current Z value. If $X = 0$, it means there is a solution for problem (29)–(34); then save the value of vector ω and set $Z_{\max} = Z$; else $X = 1$; it means there is not a solution for problem (29)–(34); then set $Z_{\min} = Z$, $Z = Z_{\max}$.

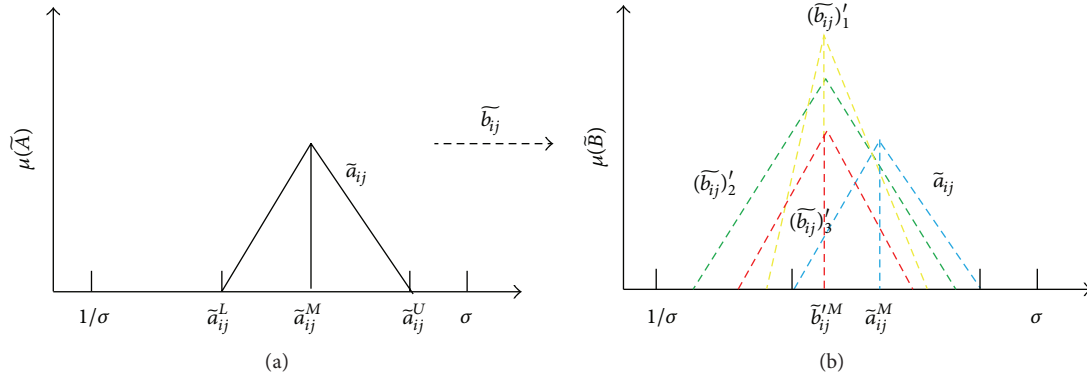


FIGURE 2: Choosing optimal matrix \tilde{B}' which is closest to matrix \tilde{A} .

Algorithm Find Modified Matrix($Z_{\max}, Z_{\min}, \xi, \tilde{A}$)

Input: Z_{\max}, Z_{\min}, ξ , original matrix \tilde{A}

Output: Z_{\max}, Z_{\min} , optimal matrix \tilde{B}

BEGIN

(1) **If** ($Z_{\max} - Z_{\min} < \xi$) & ($X = 0$)

Then

The current ω_i^M is the desired optimal solution of weight vector.

Adopt (16) to get $b_{ij}^{M'}$.

Use Theorem 3 to obtain $b_{ij}^{L'}$ and $b_{ij}^{U'}$.

Adjustable PCM $\tilde{B}' = (b_{ij}^{L'}, b_{ij}^{M'}, b_{ij}^{U'})$.

Obtain Modified PCM \tilde{B} by Definition 2.

Stop.

(2) **Else Search segment tree**

Find the greatest lower value of Z when $X = 0$

Z_{\max} = upper bound of the interval; Z_{\min} = lower bound of the interval

Solve the problem (29)–(34),

Save the new value of ω_i^M

END

ALGORITHM 1: Find modified PCM.

Note 3. Consider

$$\begin{aligned} \omega_1 &= (1.0000 \ 0.3469 \ 4.0000 \ 0.5262 \ 0.2902 \ 0.1346 \ 0.9375 \ 1.4637)^T, \\ \omega_2 &= (1.0000 \ 0.4499 \ 1.2228 \ 0.1259 \ 0.4499 \ 0.4499 \ 0.5501 \ 1.4953)^T, \\ \omega_3 &= (1.0000 \ 0.9097 \ 2.6082 \ 0.2677 \ 0.4345 \ 0.4345 \ 1.1620 \ 3.3244)^T. \end{aligned} \quad (40)$$

Now, we go to discuss the results. When in the 7th iterative time, the gap between Z_{\min} and Z_{\max} is 0.0787, which is less than 0.1; the process stops at this point. In order to better understand, we use the *segment tree* to express the changing process in Figure 3.

In Figure 3, the current Z value is colored with red and bold. The first Z value is 10.0725, and $X = 0$, which means

there is a solution for inequalities (29)–(34); the second Z value is 5.0362, and $X = 0$; by combining the first result, it must have a solution when Z value is between 5.0362 and 10.0725. Therefore, the program goes to search the left subtree of the node; the third Z value is 2.5181, and $X = 1$, which means there is not a solution for inequalities (29)–(34); the program needs to search the right subtree of this

node, because the right child node is bigger than 2.5181. We continue the process till the gap between two child nodes is less than ξ .

The final result of positive vector is $\omega_3 = (1.0000 \ 0.9097 \ 2.6082 \ 0.2677 \ 0.4345 \ 0.4345 \ 1.1620 \ 3.3244)^T$.
On the basis of ω_3 , we construct the new consistency matrix B' as in the following:

$$B' = \begin{pmatrix} 1.0000 & 1.0993 & 0.3834 & 3.7354 & 2.3015 & 2.3015 & 0.8606 & 0.3008 \\ 0.9097 & 1.0000 & 0.3488 & 3.3979 & 2.0936 & 2.0936 & 0.7828 & 0.2736 \\ 2.6082 & 2.8672 & 1.0000 & 9.7424 & 6.0027 & 6.0027 & 2.2445 & 0.7845 \\ 0.2677 & 0.2943 & 0.1026 & 1.0000 & 0.6161 & 0.6161 & 0.2304 & 0.0805 \\ 0.4345 & 0.4777 & 0.1666 & 1.6230 & 1.0000 & 1.0000 & 0.3739 & 0.1307 \\ 0.4345 & 0.4777 & 0.1666 & 1.6230 & 1.0000 & 1.0000 & 0.3739 & 0.1307 \\ 1.1620 & 1.2775 & 0.4455 & 4.3406 & 2.6745 & 2.6745 & 1.0000 & 0.3495 \\ 3.3244 & 3.6546 & 1.2746 & 12.4179 & 7.6512 & 7.6512 & 2.8609 & 1.0000 \end{pmatrix}. \quad (41)$$

Here, we adopt $\beta = 0.8$ as an example in Definition 2. The new matrix B is as follows:

$$B = \begin{pmatrix} 1.0000 & 3.6932 & 1.9881 & 6.1737 & 4.9536 & 4.9536 & 0.4030 & 0.2594 \\ 0.2708 & 1.0000 & 0.3364 & 4.6283 & 2.7917 & 2.7917 & 0.2628 & 0.1627 \\ 0.5030 & 2.9730 & 1.0000 & 6.6108 & 3.4464 & 3.3383 & 4.9288 & 0.2629 \\ 0.1620 & 0.2161 & 0.1513 & 1.0000 & 0.3769 & 0.2994 & 0.1572 & 0.1145 \\ 0.2019 & 0.3582 & 0.2902 & 2.6531 & 1.0000 & 0.5743 & 0.2267 & 0.1588 \\ 0.2019 & 0.3582 & 0.2305 & 3.3397 & 1.7411 & 1.0000 & 0.2267 & 0.1588 \\ 2.4817 & 3.8058 & 0.2029 & 6.3619 & 4.4119 & 4.4119 & 1.0000 & 0.4655 \\ 3.8547 & 6.1468 & 3.8041 & 8.7354 & 6.2989 & 6.2989 & 2.1484 & 1.0000 \end{pmatrix}. \quad (42)$$

6.2. Comparison with References. Xu and Wei [6] defined the original matrix A (the elements are a_{ij}) that can be replaced by the new matrix B (the elements are b_{ij}), which is showed in the following equation:

$$b_{ij} = a_{ij}^\alpha \left(\frac{\omega_i}{\omega_j} \right)^{1-\alpha}, \quad (43)$$

where α is the positive value which is less than, but approaching to, 1.

Cao et al. [7] proposed an equation to obtain the new matrix B , which is showed by the following:

$$b_{ij} = \left(\frac{\omega_i}{\omega_j} \right) \circ D', \quad (44)$$

where \circ is the symbol of Hadamard product. For example, $A = B \circ C$ means $a_{ij} = b_{ij} \times c_{ij}, \forall i, j = 1, \dots, n$, and D' is the modified deviation matrix, which is showed in the following equation:

$$D' = [d'_{ij}] = \gamma D \oplus (1 - \gamma) DI, \quad (45)$$

where D is the deviation matrix and DI is a zero deviation matrix when $[d_{ij}] = 1$. The value of γ is between 0 and 1.

α and γ has different meanings for two papers, but the two parameters should be as close to 1 as possible. In two papers, they mentioned that $\alpha = \gamma = 0.98$ is the most suitable value to get the optimal new matrix. We will discuss β 's meaning and the relationship with α and γ in Section 8. We compare with two references in two situations: one is the required critical ratio (CR) less than 0.1 (Table 2); the other one is the required critical ratio (CR) close to 0 (Table 4).

From Table 2, we reach the outcome that the CR value is lower than [6, 7]; at the same time, the method could achieve lower values of δ and σ in short iterative times. It means that our method can preserve more original information and obtain a more consistent new matrix in short iterative times. When we rank the priority weight ω derived from A , the ranking results are the same except when $\xi = 0.1$ in two similar weights, which means the priority weight ω which is derived from our method is acceptable.

On the basis of Table 2, we go to discover the difference of COP parameter in our method and two references. The preference value between every two alternatives is gotten from priority weight ω of the column "priority weight" in Table 2. For example, the value of $\{a_1, a_2\}$ is obtained in $\omega_{XU} = (1.0000 \ 0.3241 \ 1.0329 \ 0.1034 \ 0.1843 \ 0.2172 \ 0.9804 \ 1.9353)$, which is $1/0.3241 = 3.0855$.

The modified new matrix $B_{XU}, B_{Cao}, B_{Our}(0.05), B_{Our}(0.1)$, and $B_{Our}(0.2)$ is in the following:

$$\begin{aligned}
B_{X_{U}} &= \begin{pmatrix} 1.0000 & 2.9597 & 1.0501 & 9.4467 & 5.2285 & 4.4066 & 1.0207 & 0.5116 \\ 0.3379 & 1.0000 & 0.3548 & 3.1918 & 1.7666 & 1.4889 & 0.3449 & 0.1729 \\ 0.9523 & 2.8185 & 1.0000 & 8.9961 & 4.9791 & 4.1964 & 0.9720 & 0.4872 \\ 0.1059 & 0.3133 & 0.1112 & 1.0000 & 0.5535 & 0.4665 & 0.1080 & 0.0542 \\ 0.1913 & 0.5661 & 0.2008 & 1.8068 & 1.0000 & 0.8428 & 0.1952 & 0.0978 \\ 0.2269 & 0.6717 & 0.2383 & 2.1438 & 1.1865 & 1.0000 & 0.2316 & 0.1161 \\ 0.9797 & 2.8997 & 1.0288 & 9.2553 & 5.1226 & 4.3173 & 1.0000 & 0.5012 \\ 1.9546 & 5.7851 & 2.0525 & 18.4648 & 10.2197 & 8.6133 & 1.9950 & 1.0000 \end{pmatrix}, \\
B_{C_{ao}} &= \begin{pmatrix} 1.0000 & 3.2055 & 0.9198 & 9.8822 & 5.5733 & 4.7634 & 1.0371 & 0.5193 \\ 0.3120 & 1.0000 & 0.2869 & 3.0829 & 1.7387 & 1.4860 & 0.3235 & 0.1620 \\ 1.0872 & 3.4851 & 1.0000 & 10.7443 & 6.0595 & 5.1789 & 1.1275 & 0.5646 \\ 0.1012 & 0.3244 & 0.0931 & 1.0000 & 0.5640 & 0.4820 & 0.1049 & 0.0525 \\ 0.1794 & 0.5751 & 0.1650 & 1.7731 & 1.0000 & 0.8547 & 0.1861 & 0.0932 \\ 0.2099 & 0.6729 & 0.1931 & 2.0746 & 1.1700 & 1.0000 & 0.2177 & 0.1090 \\ 0.9643 & 3.0909 & 0.8869 & 9.5290 & 5.3741 & 4.5932 & 1.0000 & 0.5007 \\ 1.9258 & 6.1729 & 1.7712 & 19.0308 & 10.7329 & 9.1731 & 1.9971 & 1.0000 \end{pmatrix}, \\
B_{O_{ur}}(0.05) &= \begin{pmatrix} 1.0000 & 4.2959 & 2.2710 & 6.7491 & 4.9705 & 4.9705 & 0.4642 & 0.2934 \\ 0.2328 & 1.0000 & 0.3303 & 4.3498 & 2.4082 & 2.4082 & 0.2602 & 0.1582 \\ 0.4403 & 3.0273 & 1.0000 & 6.3266 & 3.0273 & 3.8107 & 4.9705 & 0.2602 \\ 0.1482 & 0.2299 & 0.1581 & 1.0000 & 0.3459 & 0.2748 & 0.1656 & 0.1184 \\ 0.2012 & 0.4152 & 0.3303 & 2.8906 & 1.0000 & 0.5743 & 0.2602 & 0.1789 \\ 0.2012 & 0.4152 & 0.2624 & 3.6387 & 1.7411 & 1.0000 & 0.2602 & 0.1789 \\ 2.1543 & 3.8428 & 0.2012 & 6.0374 & 3.8428 & 3.8428 & 1.0000 & 0.4569 \\ 3.4088 & 6.3228 & 3.8428 & 8.4449 & 5.5892 & 5.5892 & 2.1886 & 1.0000 \end{pmatrix}, \\
B_{O_{ur}}(0.1) &= \begin{pmatrix} 1.0000 & 3.6932 & 1.9881 & 6.1737 & 4.9536 & 4.9536 & 0.4030 & 0.2594 \\ 0.2708 & 1.0000 & 0.3364 & 4.6283 & 2.7917 & 2.7917 & 0.2628 & 0.1627 \\ 0.5030 & 2.9730 & 1.0000 & 6.6108 & 3.4464 & 4.3383 & 4.9288 & 0.2629 \\ 0.1620 & 0.2161 & 0.1513 & 1.0000 & 0.3769 & 0.2994 & 0.1572 & 0.1145 \\ 0.2019 & 0.3582 & 0.2902 & 2.6531 & 1.0000 & 0.5743 & 0.2267 & 0.1588 \\ 0.2019 & 0.3582 & 0.2305 & 3.3397 & 1.7411 & 1.0000 & 0.2267 & 0.1588 \\ 2.4817 & 3.8058 & 0.2029 & 6.3619 & 4.4119 & 4.4119 & 1.0000 & 0.4655 \\ 3.8547 & 6.1468 & 3.8041 & 8.7354 & 6.2989 & 6.2989 & 2.1484 & 1.0000 \end{pmatrix}, \\
B_{O_{ur}}(0.2) &= \begin{pmatrix} 1.0000 & 4.2516 & 2.3133 & 7.1790 & 4.9193 & 4.9193 & 0.4680 & 0.3044 \\ 0.2352 & 1.0000 & 0.3400 & 4.6750 & 2.4082 & 2.4082 & 0.2651 & 0.1658 \\ 0.4323 & 2.9414 & 1.0000 & 6.6066 & 2.9414 & 3.7026 & 4.9193 & 0.2651 \\ 0.1393 & 0.2139 & 0.1514 & 1.0000 & 0.3219 & 0.2557 & 0.1570 & 0.1155 \\ 0.2033 & 0.4152 & 0.3400 & 3.1067 & 1.0000 & 0.5743 & 0.2651 & 0.1876 \\ 0.2033 & 0.4152 & 0.2701 & 3.9107 & 1.7411 & 1.0000 & 0.2651 & 0.1876 \\ 2.1369 & 3.7727 & 0.2033 & 6.3702 & 3.7727 & 3.7727 & 1.0000 & 0.4702 \\ 3.2854 & 6.0312 & 3.7727 & 8.6577 & 5.3315 & 5.3315 & 2.1266 & 1.0000 \end{pmatrix}.
\end{aligned}
\tag{46}$$

The gap of elements in every two new matrixes (B_{XU} , B_{Cao} , $B_{Our}(0.05)$, $B_{Our}(0.1)$ and $B_{Our}(0.2)$) is less than $1.5 < 2$, which proves these five matrixes are considerably close with each other; but, in Table 2, the CR value of our method is less than the other two methods, which means $B_{Our}(0.05)$,

$B_{Our}(0.1)$, and $B_{Our}(0.2)$ are more consistent than B_{XU} and B_{Cao} . Meanwhile, the value of δ and σ for $B_{Our}(0.05)$ is the lowest, which means $B_{Our}(0.05)$ is the best solution for original matrix A . Next, we go to discuss the second situation when CR is close to 0. The analysis results are in Table 4.

The iterative times for [6, 7] 820 and 1619 are quite large, which will cost the running time. The value of δ and σ in these two references is far away from the acceptable value, which means the priority weight of ω_{xu} and ω_{cao} is not acceptable at all; but the iterative times of our method are less than 10 times, the value of δ is less than 4, and the value of σ is less than 2, which means our result is acceptable to some extent.

7. Numerical Illustration and Comparison with Fuzzy Numbers

To judge the effectiveness of Algorithm 1 for fuzzy PCMs, we analyze several parameters including the inconsistency index NI [13], consistency index CCI [14, 15], δ_{fuzzy} , σ_{fuzzy} , COP, iteration times, and running time.

We adopt the same inconsistent matrix with Ramík and Korviny [13], which is

$$\tilde{A} = \begin{pmatrix} (1, 1, 1) & (2, 3, 4) & (4, 5, 6) \\ \left(\frac{1}{4}, \frac{1}{3}, \frac{1}{3}\right) & (1, 1, 1) & (3, 4, 5) \\ \left(\frac{1}{6}, \frac{1}{5}, \frac{1}{4}\right) & \left(\frac{1}{5}, \frac{1}{4}, \frac{1}{3}\right) & (1, 1, 1) \end{pmatrix}. \quad (47)$$

For matrix \tilde{A} , $NI_3^9(\tilde{A}) = 0.219$. It is an inconsistent PCM with fuzzy elements. We adopt Algorithm 1 to obtain the modified new matrix under different parameters. The results are shown in Table 5.

For this matrix, the modified matrices \tilde{B} are quite similar when the accurate degrees are $\xi = 0.1$ and $\xi = 0.01$, and, therefore, we only give the modified matrix \tilde{B} when $\xi = 0.1$ and $\xi = 0.001$ under $\beta = 0.5$ and $\beta = 0.6$:

$$\begin{aligned} \tilde{B}(\xi = 0.1, \beta = 0.5) &= \begin{pmatrix} [1, 1, 1] & [2.021, 2.453, 2.718] & [5.284, 5.480, 5.800] \\ [0.350, 0.407, 0.424] & [1, 1, 1] & [3.202, 3.460, 3.895] \\ [0.154, 0.182, 0.211] & [0.241, 0.288, 0.331] & [1, 1, 1] \end{pmatrix}, \\ \tilde{B}(\xi = 0.001, \beta = 0.5) &= \begin{pmatrix} [1, 1, 1] & [2.017, 2.449, 2.714] & [5.281, 5.477, 5.797] \\ [0.350, 0.408, 0.425] & [1, 1, 1] & [3.205, 3.463, 3.899] \\ [0.154, 0.182, 0.211] & [0.241, 0.288, 0.331] & [1, 1, 1] \end{pmatrix}, \\ \tilde{B}(\xi = 0.1, \beta = 0.6) &= \begin{pmatrix} [1, 1, 1] & [2.016, 2.553, 2.936] & [4.997, 5.380, 5.839] \\ [0.327, 0.391, 0.404] & [1, 1, 1] & [3.1608, 3.562, 4.094] \\ [0.156, 0.185, 0.218] & [0.232, 0.280, 0.331] & [1, 1, 1] \end{pmatrix}, \\ \tilde{B}(\xi = 0.001, \beta = 0.6) &= \begin{pmatrix} [1, 1, 1] & [2.014, 2.550, 2.933] & [4.995, 5.378, 5.837] \\ [0.327, 0.392, 0.405] & [1, 1, 1] & [3.163, 3.565, 4.098] \\ [0.156, 0.185, 0.218] & [0.232, 0.280, 0.331] & [1, 1, 1] \end{pmatrix}. \end{aligned} \quad (48)$$

In Table 5, the inconsistency index NI is less than 0.1 and consistency index CCI is less than 0.3147, which indicates the modified matrix can reach a consistent requirement; δ_{fuzzy} satisfies $0 < \delta_{fuzzy} < 2$ and σ_{fuzzy} satisfies $0 < \sigma_{fuzzy} < 1$, which shows the obtained new matrix is in an acceptable distance with \tilde{A} ; the value of COP, which is gotten from priority vector $\tilde{\omega}$ of \tilde{B} , can preserve order of preference and order of intensity preference, which presents \tilde{B} can maintain the pattern of \tilde{A} (similarity). Meanwhile, Algorithm 1 has high convergence speed based on the less iteration times and running time. The gap among these four matrixes is quite narrow; $\tilde{B}(\xi = 0.1, \beta = 0.5)$ is the best choice when we select the smallest value of the consistency index (NI, CCI), δ_{fuzzy} , and σ_{fuzzy} .

8. Discussion

Tables 2 and 4 conclude the effectiveness of Algorithm 1 by comparing with Cao et al. [7] and Xu and Wei [6] based on PCMs with crisp elements. Algorithm 1 can retain more original information and achieve lower value of δ and σ when both α [6] and γ [7] approach 1. Table 5 summarizes the

effectiveness of Algorithm 1 for PCMs with fuzzy elements. The modified PCMs with fuzzy elements can maintain original information and reach acceptable consistency level as well.

In Algorithm 1, β has different meaning with α and γ . In Cao et al. [7] and Xu and Wei [6], the results gotten by using α and γ equal to 0.98 are significantly better than those by using α and γ less than 0.98. Yet, it is not always the case for β . The true meaning of β is the proportion of the original matrix to adjustable matrix. The approximate perfect range of β is [0.25, 0.8] based on the experiments' results, and there is no clue which value is the best one for β as the inconsistency level of PCMs is different. As a matter of fact, changing β 's value will not add running time, because this operation runs in constant time. As a future research, it would be interesting to figure out the exact perfect range of β . In this section, we focus on efficiency analysis of Algorithm 1 as effective analysis is done in Sections 6 and 7.

8.1. Efficiency Analysis: Algorithm Running Time. To demonstrate the overall efficiency of Algorithm 1, we measure

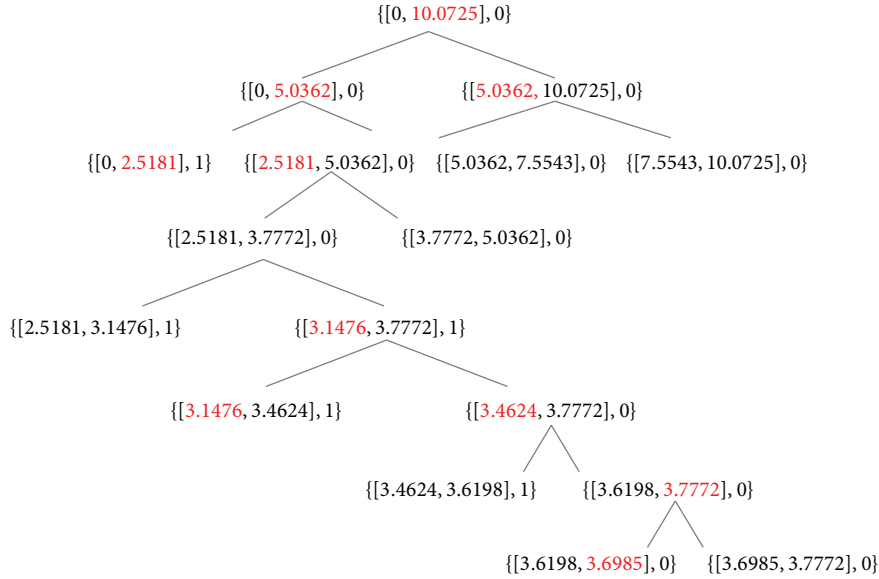


FIGURE 3: An example of iterative times in segment tree structure.

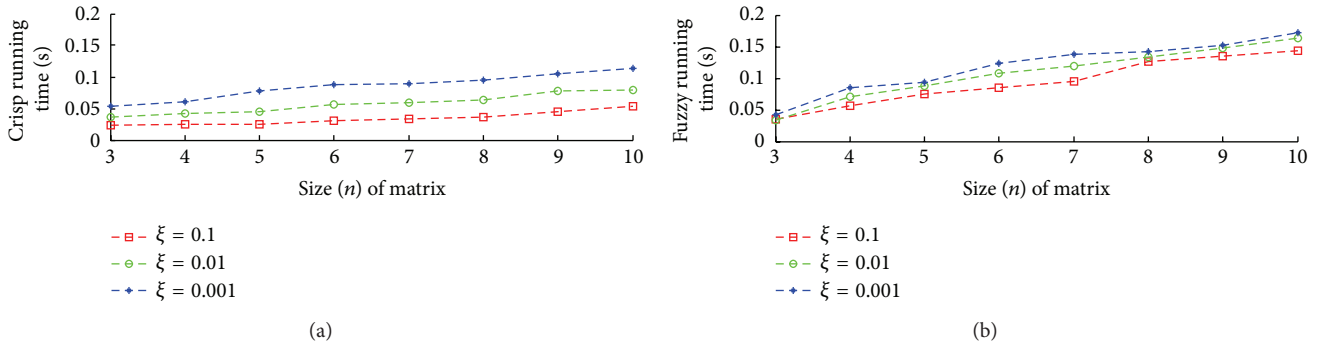


FIGURE 4: Average running time with fuzzy and crisp elements (in seconds).

the average running time and iteration times for crisp and fuzzy elements. The elements of crisp matrix A and fuzzy \tilde{A} are gotten from randomly generated numbers by programming. The size (n) of matrix A and \tilde{A} varies from 3 to 10 (in real life problems, the size is usually no more than 10). We adopt twenty matrixes for each size.

The assumed accuracy level ξ is the following three numbers: 0.1, 0.01, and 0.001. The value of β in the final step ($B = a_{ij}^\beta (b'_{ij})^{1-\beta}$) will not affect the running time. The average running time of crisp elements is shown in Figure 4(a). The average running time of fuzzy elements is shown in Figure 4(b). In all cases, the running times are less than half a second, which is acceptable in real experiments. As expected, the higher the accuracy degree, the longer the running time in the same matrix size. The figure shows the growth rate of fuzzy matrixes' running time is more irregular than crisp matrixes.

8.2. Efficiency Analysis: Convergence Rate. The speed of convergence is one important factor of the efficiency for an iterative method. In Algorithm 1, the only parameter that can influence iteration times is the accuracy degree (ξ). α and γ are the parameters which can influence the number of iterations for [6, 7], respectively. According to the numerical illustration in Section 6, we study the convergence rate with respect to the values of ξ , α , and γ , by using Algorithm 1 and [6, 7], respectively. Meanwhile, we examine the convergence rate for fuzzy elements based on the data in Section 7. Figure 5 shows the results.

The convergence rate of Cao et al. [7] is slower than Xu and Wei [6] when $\gamma = \alpha$. For these two methods, the lower the parameters value, the higher the rate of convergence. Algorithm 1 can approach its limits faster than Cao et al. [7] and Xu and Wei [6]. The maximum iteration time for Algorithm 1 is less than twenty when $\xi = 0.0001$, while the maximum iteration times for Cao et al. [7] and Xu and Wei [6]

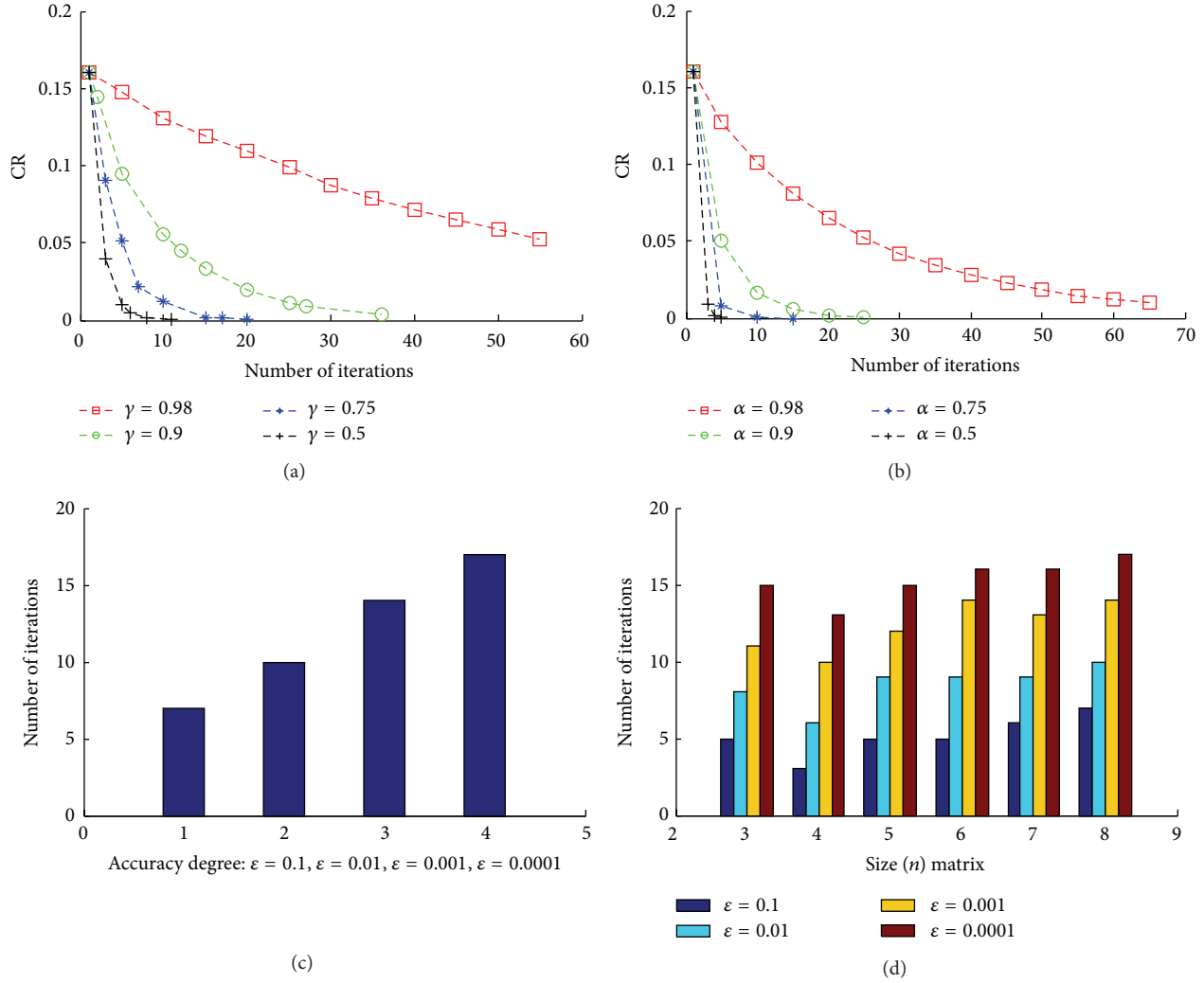


FIGURE 5: Convergence rate of Cao et al. [7], Xu and Wei [6], and Algorithm 1.

are 1619 and 820, respectively. For fuzzy elements, the convergence rate of Algorithm 1 has low growth rate as well.

9. Conclusions

In this work, an algorithm has been proposed to derive a consistent PCM with crisp or fuzzy elements from an inconsistent one. The presented approach used the same numerical example used by [6, 7, 13]. The experiments reveal that the proposed approach could retain more original information and the convergence rate is faster than Cao et al. [7] and Xu and Wei [6] in different values of CR. Two effective criteria, δ and σ , show that the distance between modified PCM and original PCM is acceptable for both crisp and fuzzy elements. A new effectiveness criterion COP reflects that the modified PCM resembles the original PCM (crisp and fuzzy elements). In conclusion, this approach could enhance the quality of vague and inaccuracy data for decision makers and

also could better handle the inconsistency problem of AHP and fuzzy AHP.

In the future work, we will apply this approach to different applications. Meanwhile, efforts should be made to explore the perfect range of β and study the issue of selecting the situation where we can supply a consistency PCM to an inconsistency one based on real-life meaning.

Appendix

The Calculation Steps of (19)

Before we go to find these elements, define the range of every value of $b_{ij}^L, b_{ij}^M, b_{ij}^U$. Given fuzzy reciprocal matrix $\tilde{A} = \{\tilde{a}_{ij}\}_{n \times n} = \{a_{ij}^L, a_{ij}^M, a_{ij}^U\}_{n \times n}$, it has the support $\text{SUPP}(\tilde{a}_{ij}) \subseteq S = [1/\sigma, \sigma]$, where $\sigma > 1, \forall i, j \in \{1, 2, \dots, n\}$. This means $1/\sigma \leq a_{ij}^L \leq a_{ij}^M \leq a_{ij}^U \leq \sigma$.

TABLE 2: $\alpha = \gamma = 0.98$. Iteration times, $CR \leq 0.1$, δ , σ , λ_{\max} , and priority weight results.

Methods	Iteration needed for $CR \leq 0.1$	CR	δ	σ	λ_{\max}	Priority weight
Xu and Wei [6]	12	0.0972	1.845	0.589	8.955	ω_{XU}, R_{XU}
Cao et al. [7]	18	0.0997	1.713	0.448	8.9844	ω_{Cao}, R_{Cao}
Our method ($\xi = 0.05, \beta = 0.8$)	8	0.0964	1.1572	0.4710	8.9017	$\omega_{Our}(0.05)$ $R_{Our}(0.05)$
Our method ($\xi = 0.1, \beta = 0.8$)	7	0.0964	0.7354	0.4689	8.9017	$\omega_{Our}(0.1)$ $R_{Our}(0.1)$
Our method ($\xi = 0.2, \beta = 0.8$)	6	0.0964	1.2273	0.4993	8.9017	$\omega_{Our}(0.2)$ $R_{Our}(0.2)$
ω_{XU}, R_{XU}	$\omega_{XU} = (1.0000 \ 0.3241 \ 1.0329 \ 0.1034 \ 0.1843 \ 0.2172 \ 0.9804 \ 1.9353)$ $R_{XU} = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					
ω_{Cao}, R_{Cao}	$\omega_{Cao} = (1.0000 \ 0.3191 \ 1.0695 \ 0.1019 \ 0.1824 \ 0.2142 \ 0.9699 \ 1.9340)$ $R_{Cao} = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					
$\omega_{Our}(0.05)$ $R_{Our}(0.05)$	$\omega_{Our}(0.05) = (1.0000 \ 0.3430 \ 1.0856 \ 0.1143 \ 0.2182 \ 0.2493 \ 0.8808 \ 1.9064)$ $R_{Our}(0.05) = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					
$\omega_{Our}(0.1)$ $R_{Our}(0.1)$	$\omega_{Our}(0.1) = (1.0000 \ 0.3990 \ 1.2401 \ 0.1249 \ 0.2189 \ 0.2501 \ 1.0147 \ 2.1558)$ $R_{Our}(0.1) = (8 \ 3 \ 7 \ 1 \ 2 \ 6 \ 5 \ 4)$					
$\omega_{Our}(0.2)$ $R_{Our}(0.2)$	$\omega_{Our}(0.2) = (1.0000 \ 0.3466 \ 1.0657 \ 0.1074 \ 0.2205 \ 0.2518 \ 0.8737 \ 1.8375)$ $R_{Our}(0.2) = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					

TABLE 3: COP (condition of order preservation) parameter value.

	Original matrix ω	ω_{XU}	ω_{Cao}	$\omega_{Our}(0.05)$	$\omega_{Our}(0.1)$	$\omega_{Our}(0.2)$
$\{a_1, a_2\}$	3.2055	3.0855	3.2037	2.9152	2.5062	2.8852
$\{a_1, a_3\}$	0.9198	0.9681	0.9197	0.9212	0.8064	0.9383
$\{a_1, a_4\}$	9.8822	9.6712	9.8857	8.7519	8.0060	9.3096
$\{a_1, a_5\}$	5.5733	5.4259	5.5806	4.5830	4.5674	4.5358
$\{a_1, a_6\}$	4.7634	4.6041	4.7658	4.0119	3.9984	3.9706
...

Conclusion Cao et al. [7] are closer to the original matrix ω , but the gap among Cao et al. [7], Xu and Wei [6], and our method is very narrow (less than 1), the three methods almost in the same extends to close to original ω . Thus, our method is acceptable in this parameter.

Given a vector of fuzzy weights $\tilde{\omega} = (\tilde{\omega}_1, \tilde{\omega}_2, \dots, \tilde{\omega}_n)$, which is the corresponding eigenvector matrix $\tilde{\omega}$ closest to original matrix \tilde{A} , the value of eigenvector matrix $\tilde{\omega}$ should satisfy the following formula:

$$\tilde{\omega}_i \approx \left\{ \left(\prod_{j=1}^n a_{ij}^L \right)^{1/n}, \left(\prod_{j=1}^n a_{ij}^M \right)^{1/n}, \left(\prod_{j=1}^n a_{ij}^U \right)^{1/n} \right\} = \left\{ \frac{1}{\sigma}, \frac{1}{\sigma}, \frac{1}{\sigma} \right\} \cdot \left(\frac{1}{\sigma} \cdot \frac{1}{\sigma} \dots \frac{1}{\sigma} \right)^{1/n} \tag{A.1}$$

$$\leq \left\{ \left(\frac{\sigma \cdot \sigma \dots \sigma}{n} \right)^{1/n}, \left(\frac{\sigma \cdot \sigma \dots \sigma}{n} \right)^{1/n}, \left(\frac{\sigma \cdot \sigma \dots \sigma}{n} \right)^{1/n} \right\} = \{\sigma, \sigma, \sigma\},$$

$$\tilde{\omega}_i \approx \left\{ \left(\prod_{j=1}^n a_{ij}^L \right)^{1/n}, \left(\prod_{j=1}^n a_{ij}^M \right)^{1/n}, \left(\prod_{j=1}^n a_{ij}^U \right)^{1/n} \right\}$$

$$\geq \left\{ \left(\frac{1}{\sigma} \cdot \frac{1}{\sigma} \dots \frac{1}{\sigma} \right)^{1/n}, \left(\frac{1}{\sigma} \cdot \frac{1}{\sigma} \dots \frac{1}{\sigma} \right)^{1/n} \right\}$$

The eigenvector of modified matrix \tilde{B} and the adjustable matrix \tilde{B}' should also be in this range; to be clearer, the range of eigenvector is

$$\left\{ \frac{1}{\sigma}, \frac{1}{\sigma}, \frac{1}{\sigma} \right\} \leq \tilde{\omega}_i \leq \{\sigma, \sigma, \sigma\} \quad \{\forall ii \in (1, 2, \dots, n), n \in \mathbb{R}\},$$

$$\left\{ \frac{1}{\sigma}, \frac{1}{\sigma}, \frac{1}{\sigma} \right\} \leq \tilde{\omega}'_i \leq \{\sigma, \sigma, \sigma\} \quad \{\forall ii \in (1, 2, \dots, n), n \in \mathbb{R}\}. \tag{A.2}$$

TABLE 4: $\alpha = \gamma = 0.98$. Iteration times, CR = 0, δ , σ , λ_{\max} , and priority weight results.

Methods	Iteration needed for CR = 0	CR	δ	σ	λ_{\max}	Priority weight
Xu and Wei [6]	820	0	10.4648	1.9069	8	ω_{XU}, R_{XU}
Cao et al. [7]	1619	0	11.0308	2.0714	8	ω_{Cao}, R_{Cao}
Our method ($\xi = 0.05, \beta = 0.2$)	8	0	3.2554	1.4458	8	$\omega_{Our}(0.05), R_{Our}(0.05)$
Our method ($\xi = 0.1, \beta = 0.2$)	7	0	3.5117	1.5088	8	$\omega_{Our}(0.1), R_{Our}(0.1)$
Our method ($\xi = 0.2, \beta = 0.2$)	6	0	3.6985	1.7867	8	$\omega_{Our}(0.2), R_{Our}(0.2)$
ω_{XU}, R_{XU}	$\omega_{XU} = (1.0000 \ 0.3120 \ 1.0873 \ 0.1012 \ 0.1794 \ 0.2100 \ 0.9644 \ 1.9259)$ $R_{XU} = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					
ω_{Cao}, R_{Cao}	$\omega_{Cao} = (1.0000 \ 0.3121 \ 1.0873 \ 0.1012 \ 0.1792 \ 0.2098 \ 0.9642 \ 1.9260)$ $R_{Cao} = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					
$\omega_{Our}(0.05), R_{Our}(0.05)$	$\omega_{Our}(0.05) = (1.0000 \ 0.4109 \ 1.2324 \ 0.1565 \ 0.3665 \ 0.3796 \ 0.6374 \ 1.8322)$ $R_{Our}(0.05) = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					
$\omega_{Our}(0.1), R_{Our}(0.1)$	$\omega_{Our}(0.1) = (1.0000 \ 0.7522 \ 2.0985 \ 0.2235 \ 0.3715 \ 0.3848 \ 1.1224 \ 2.9959)$ $R_{Our}(0.1) = (8 \ 3 \ 7 \ 1 \ 2 \ 6 \ 5 \ 4)$					
$\omega_{Our}(0.2), R_{Our}(0.2)$	$\omega_{Our}(0.2) = (1.0000 \ 0.4499 \ 1.2228 \ 0.1259 \ 0.4499 \ 0.4499 \ 0.5501 \ 1.4953)$ $R_{Our}(0.2) = (8 \ 3 \ 1 \ 7 \ 2 \ 6 \ 5 \ 4)$					

TABLE 5: Effective analysis for Algorithm 1 based on nine parameters.

$\beta = 0.5$	NI [13]	CCI [14, 15]	δ_{fuzzy}	σ_{fuzzy}	COP	Iteration times	Running time (seconds)
$\xi = 0.1$	0.0887	0.1378	1.2841	0.5688	Keep	5	0.033
$\xi = 0.01$	0.0887	0.1378	1.2841	0.5688	Keep	8	0.031
$\xi = 0.001$	0.0886	0.1378	1.2815	0.5691	Keep	11	0.042
$\beta = 0.6$	NI	CCI	δ_{fuzzy}	σ_{fuzzy}	COP	Iteration times	Running time (seconds)
$\xi = 0.1$	0.1007	0.1991	0.9979	0.4692	Keep	5	0.021
$\xi = 0.01$	0.1007	0.1991	0.9979	0.4692	Keep	8	0.034
$\xi = 0.001$	0.1007	0.1991	0.9960	0.4695	Keep	11	0.046

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] T. L. Saaty, *The Analytic Hierarchy Process*, McGraw-Hill, New York, NY, USA, 1980.
- [2] H. Q. Zhang, Y. Ouzrout, A. Bouras, V. Della Selva, and M. Savino, "Selection of optimal product lifecycle management components based on AHP methodologies," in *Proceedings of the International Conference on Advanced Logistics and Transport (ICALT '13)*, pp. 523–528, Sousse, Tunisia, May 2013.
- [3] H. Q. Zhang, Y. Ouzrout, A. Bouras, A. Mazza, and M. Savino, "PLM components selection based on a maturity assessment and AHP methodology," in *Proceedings of the International Conference on Product Lifecycle Management Conference (PLM '13)*, Nantes, France, July 2013.
- [4] F. Torfi, R. Z. Farahani, and S. Rezapour, "Fuzzy AHP to determine the relative weights of evaluation criteria and Fuzzy TOPSIS to rank the alternatives," *Applied Soft Computing Journal*, vol. 10, no. 2, pp. 520–528, 2010.
- [5] S. Karapetrovic and E. S. Rosenbloom, "A quality control approach to consistency paradoxes in AHP," *European Journal of Operational Research*, vol. 119, no. 3, pp. 704–718, 1999.
- [6] Z. S. Xu and C. P. Wei, "A consistency improving method in the analytic hierarchy process," *European Journal of Operational Research*, vol. 116, no. 2, pp. 443–449, 1999.
- [7] D. Cao, L. C. Leung, and J. S. Law, "Modifying inconsistent comparison matrix in analytic hierarchy process: a heuristic approach," *Decision Support Systems*, vol. 44, no. 4, pp. 944–953, 2008.
- [8] M. Anholcer, *Algorithm for Deriving Priorities from Inconsistent Pairwise Comparison Matrices*, 2013.
- [9] Y. J. Xu and H. M. Wang, "Eigenvector method, consistency test and inconsistency repairing for an incomplete fuzzy preference relation," *Applied Mathematical Modelling*, vol. 37, pp. 5171–5183, 2013.
- [10] L. C. Leung and D. Cao, "On consistency and ranking of alternatives in fuzzy AHP," *European Journal of Operational Research*, vol. 124, no. 1, pp. 102–113, 2000.
- [11] M. Morteza and A. R. Bafandeh, "A new method for consistency test in fuzzy AHP," *Journal of Intelligent and Fuzzy Systems*, vol. 25, no. 2, pp. 457–461, 2013.

- [12] T. C. Wang and Y. H. Chen, "Applying fuzzy linguistic preference relations to the improvement of consistency of fuzzy AHP," *Information Sciences*, vol. 178, no. 19, pp. 3755–3765, 2008.
- [13] J. Ramík and P. Korviny, "Inconsistency of pair-wise comparison matrix with fuzzy elements based on geometric mean," *Fuzzy Sets and Systems*, vol. 161, no. 11, pp. 1604–1613, 2010.
- [14] E. Bulut, O. Duru, T. Keçeci, and S. Yoshida, "Use of consistency index, expert prioritization and direct numerical inputs for generic fuzzy-AHP modeling: a process model for shipping asset management," *Expert Systems with Applications*, vol. 39, no. 2, pp. 1911–1923, 2012.
- [15] E. Dopazo, K. Lui, S. Chouinard, and J. Guisse, "A parametric model for determining consensus priority vectors from fuzzy comparison matrices," *Fuzzy Sets and Systems*, 2013.
- [16] J. J. Buckley, "Fuzzy hierarchical analysis," *Fuzzy Sets and Systems*, vol. 17, no. 3, pp. 233–247, 1985.
- [17] J. Aguarón and J. M. Moreno-Jiménez, "The geometric consistency index: approximated thresholds," *European Journal of Operational Research*, vol. 147, no. 1, pp. 137–145, 2003.
- [18] S. I. Gass and T. Rapcsák, "Singular value decomposition in AHP," *European Journal of Operational Research*, vol. 154, no. 3, pp. 573–584, 2004.
- [19] W. E. Stein and P. J. Mizzi, "The harmonic consistency index for the analytic hierarchy process," *European Journal of Operational Research*, vol. 177, no. 1, pp. 488–497, 2007.
- [20] T. L. Saaty, *Multi-Criteria Decision Making-the Analytical Hierarchy Process*, vol. 1, RWS Publications, Pittsburgh, Pa, USA, 1991.
- [21] D. Dubois, "The role of fuzzy sets in decision sciences: old techniques and new directions," *Fuzzy Sets and Systems*, vol. 184, no. 1, pp. 3–28, 2011.
- [22] M. Yang, F. I. Khan, and R. Sadiq, "Prioritization of environmental issues in offshore oil and gas operations: a hybrid approach using fuzzy inference system and fuzzy analytic hierarchy process," *Process Safety and Environmental Protection*, vol. 89, no. 1, pp. 22–34, 2011.
- [23] M. Brunelli, "A note on the article "inconsistency of pair-wise comparison matrix with fuzzy elements based on geometric mean", *Fuzzy Sets and Systems*, vol. 161, 1604–1613, 2010," *Fuzzy Sets and Systems*, vol. 176, no. 1, pp. 76–78, 2011.
- [24] J. Aguarón and J. M. Moreno-Jiménez, "The geometric consistency index: approximated thresholds," *European Journal of Operational Research*, vol. 147, pp. 137–145, 2003.
- [25] C. A. Bana e Costa and J.-C. Vansnick, "A critical analysis of the eigenvalue method used to derive priorities in AHP," *European Journal of Operational Research*, vol. 187, no. 3, pp. 1422–1428, 2008.
- [26] P. de Jong, "A statistical approach to Saaty's scaling method for priorities," *Journal of Mathematical Psychology*, vol. 28, no. 4, pp. 467–478, 1984.
- [27] T. L. Saaty and G. Hu, "Ranking by eigenvector versus other methods in the analytic hierarchy process," *Applied Mathematics Letters*, vol. 11, no. 4, pp. 121–125, 1998.
- [28] T. L. Saaty, "Decision-making with the AHP: why is the principal eigenvector necessary," *European Journal of Operational Research*, vol. 145, no. 1, pp. 85–91, 2003.
- [29] T. S. Almulhim, L. Mikhailov, and D.-L. Xu, "Deriving weights from group fuzzy pairwise comparison judgement matrices," in *Advances in Information Systems and Technologies*, pp. 545–555, Springer, 2013.
- [30] W. Mogi, N. Izumi-cho, and M. Shinohara, "Optimum priority weight estimation method for pairwise comparison matrix," in *Proceedings of the 10th International Symposium on the Analytic Hierarchy/Network Process*, 2009.
- [31] G. Crawford and C. Williams, "A note on the analysis of subjective judgment matrices," *Journal of Mathematical Psychology*, vol. 29, no. 4, pp. 387–405, 1985.
- [32] G. Kou, D. Ergu, Y. Peng, and Y. Shi, "A new consistency test index for the data in the AHP/ANP," in *Data Processing for the AHP/ANP*, pp. 11–27, Springer, 2013.
- [33] W. L. Winston, M. Venkataramanan, and J. B. Goldberg, *Introduction to Mathematical Programming*, vol. 1, Thomson/Brooks/Cole, 2003.

Copyright of Mathematical Problems in Engineering is the property of Hindawi Publishing Corporation and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.