Granular Computing (2024)9:42
https://doi.org/10.1007/s41066-024-00459-5

ORIGINAL PAPER

Closed-form solutions of consistency ratio in best worst method minmax optimization model: max of edge error matrix and minmax edge error determinant methods

Kevin Kam Fung Yuen[1]

Received: 20 December 2023 / Accepted: 17 January 2024

Abstract

The Best Worst Method (BWM), a reduced version of the AHP, is a recent multi-criteria decision-making tool based on pairwise comparisons with reference to the best and worst criteria. Consistency Ratio (CR) measurement for the rating quality and prioritizations is still a controversial topic. Firstly, the computation for the current CR of BWM must rely on a software optimization solver to find the optimal values, and the solver may not always guarantee the exact optimal solutions, especially if the computational cost settings are not large enough for higher number of criteria. Secondly, much effort to evaluate optimization algorithms is needed to find the best solutions with the least computational resources due to diverse solvers possibly leading to different results with different performances. Thirdly, optimization programming code is not trivial to be implemented for general BWM users. To address these issues, this paper presents the closed-form solutions, Max of Edge Error Matrix (MEEM) (Eq. (44) of Theorem 4) and Minmax Edge Error Determinant (MEED) (Algorithm 1), to replace the BWM optimization models to directly calculate the CR values. Two simulations have been performed with a basic laptop using a single process. One simulation of twenty thousand random pairs of vectors took 26.34 h to perform to verify that the approximate results are higher than or very close to the exact closed-form values of both methods when high computational cost is allocated for the solver to increase the precision. Another simulation of one million random pairs of vectors only took 1.27 h to perform to verify that the MEED and MEEM methods always produce the same results for the number of criteria up to nine. The computational time for the exact results is dramatically reduced when the solver is not needed. The advantages of the proposed solutions include the following: the software to solve the optimization model to obtain CR is unnecessary, and the proposed calculation is extremely efficient to obtain the exact accuracy. The two-step optimization model can preserve the fixed Minmax Edge Error to find the weights which add up to one, which is the condition to determine if the model reaches exact optimal solutions. As the CR optimization model produces multiple versions of weights, which are recommended not to be used, the new method does not need to compute the unnecessary weight values to get the Minmax Edge Error. With the provision of equations leading to closed forms, users can understand the properties of CR in much clearer perspectives. Due to the computational efficiency and explainability, the proposed closed forms can replace the CR optimization model to compute CR efficiently and accurately for all diverse applications using BWM.

Keywords Optimization · Minmax problem · Best worst method · Pairwise comparisons · Decision sciences

1 Introduction

The first recognizable description of comparative judgements in the literature was credited to Ramon Llull, the 13th-century mystic and philosopher (Faliszewski et al. 2010; Koczkodaj et al. 2016). Thurstone (1927) underlined significant principles of comparative judgement. Saaty (1977, 1980, 1990) developed the Analytic Hierarchical Process (AHP) leading to a considerable impact on the

✉ Kevin Kam Fung Yuen
kevinkf.yuen@gmail.com

[1] Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China

Published online: 01 April 2024

pairwise comparison research. Whilst the AHP based on the paired ratio scale potentially leads to misapplications, Yuen (2009, 2012, 2014) proposed the Cognitive Network Process (CNP) based on the paired differential scale for pairwise comparisons, and some recent applications were presented in (Guan et al. 2019; Yuen 2022). Rezaei (2015) introduced a merely reduced form of AHP, Best Worst Method (BWM), which is gaining growing attention from researchers. The consistency is considered improved over the AHP simply because the reduced version requires fewer inputs from users, reducing the chances of making mistakes.

Mi et al. (2019) published a survey paper for BWM based on 124 publications. To list some recent articles related to BWM after 2019, Hafezalkotob et al. (2020) integrated interval values, MULTIMOORA, Borda rule and BWM for hybrid vehicle engine selection. Chen and Ming (2020) integrated rough–fuzzy approach, BWM and data envelopment analysis for smart product service module selection. Faizi et al. (2021) combined Hamacher aggregation operations for intuitionistic 2-tuple linguistic sets to BWM. Ali and Rashid (2021) applied BWM to robot selection. Oztas and Erdem (2021) applied conjoint analysis and BWM to framework selection for developing optimization algorithms. Yucesan and Gul (2021) proposed neutrosophic BWM for failure prioritization and control. Dong et al. (2021) integrated the triangular fuzzy numbers and Wan et al. (2021) integrated interval-valued trapezoidal fuzzy numbers to BWM. Jafarzadeh Ghoushchi et al. (2021) incorporated the BWM into the importance-necessity concept. Kheybari and Ishizaka (2022) proposed behavioural BWM. Ma et al. (2023) combined a granular computing-based method in the BWM framework. Qin et al. (2023) proposed a consensus model for the group BWM. Malakoutikhah et al. (2022) combined the fuzzy BWM and fuzzy cognitive map to model the factors affecting unsafe behaviours. Wan and Dong (2022), Chen et al. (2023) and Dong and Wan (2024) integrated intuitionistic fuzzy sets into BWM.

Regarding the studies of consistency measurement, the first BWM study (Rezaei 2015) collected 322 pairs of vectors for BWM and 322 matrices for AHP from 46 valid respondents (university students) for selecting the mobile phone from 4 alternatives based on 6 criteria. However, the weights produced by Rezaei, (2015) are problematic as Example 5 of this study shows that the same objective value (Model (3) or (4)) can lead to multiple versions of non-reproducible weights, and therefore, the conclusion of the hypothesis testing was unreliable. Rezaei (2016) present the linear model (Model (4) or (5)) that can produce a unique version of weights for the optimal objective value with two numerical examples to roughly show the calculation concepts. Although MS Excel Solver (https://bestworstmethod.com/software/) refers to Rezaei (2015, 2016), the software and tools were not mentioned in Rezaei (2015, 2016), whilst different numerical solvers with different algorithms of different seeds in different software packages/applications usually performed differently, no clear relationship between Model (3) or (4) and Model (5) or (4) are shown.

Regarding the recent simulation studies, Wu et al. (2022) randomly generated a total of 80,000 pairs for the criteria size from 3 to 10. The simulation was implemented in Matlab including the *fmincon* function. When the order size increases, the percentage of consistency ratio more than 0.1 decreases. (Liang et al. 2022) generated 10,000 pairs of ordinal-inconsistent vectors and 10,000 pairs of ordinal-consistent vectors (total size is not directly mentioned in the article). Liang et al., (2020) studied the consistency issues for the thresholds by generating a set of only 20,000 random pairs of vectors (A_{BO} and A_{OW}) for each case from three to nine criteria based on one- to nine-point scales. Mazurek et al. (2021) conducted a Monte Carlo Simulation of no more than 6000 instances to compare three prioritization methods, the Geometric Mean and Eigenvalue Methods implemented in C# and the BWM implemented in MS Excel Solver. For the simulations above, the results may not be reliable as the sample size is very small, which may be limited by the significant computational time using the solver. On the other hand, two proposed closed forms take only 1.27 h to complete one million instance computation with a basic laptop in a single processing. The detail of the simulation is presented in Sect. 9.

According to the literature survey to date on Aug 2023 (the release date of this first preprint in http://dx.doi.org/10.2139/ssrn.4551188), no study has provided closed-form solutions for the CR of BWM. To address these problems, the following contributions of this study have been made.

- Section 2 presents a comprehensive review to discuss the limitations and drawbacks of BWM and the motivations of this study. To understand the properties of the optimization models which are not mentioned in the previous studies, equivalent forms of two types of Minmax models without absolute functions are proposed in Sect. 3.
- The concepts and principles of individual edge errors and co-edge errors are presented in Sects. 4 and 5, respectively.
- For the closed forms of the consistency ratio, the max of edge error matrix method is presented in Sect. 6, whilst the Minmax edge error determinant method is presented in Sect. 7.
- Several two-step optimization models with several forms are presented in Sect. 8. The models provide more alternative forms for the different solvers of

different algorithms to implement to guarantee the precision that the largest Minmax edge error is minimized by observing the expected unique optimal objective value.

- In Sect 9, six numerical examples are demonstrated for the usability of the proposed solutions. Two stochastic simulations are conducted to verify the reliability of the proposed solutions. Three supplementary files for the examples and simulations are attached and can be found in Yuen (2023).
- Section 10 discusses the results, concludes remarks and provides future research recommendations. The summary of notations is presented in Appendix section.

2 Review and motivations

2.1 Backgrounds of BWM

The BWM was proposed in Rezaei (2015, 2016) and the calculation steps are illustrated as follows:

1. Define a set of decision criteria.

2. Define the best and the worst criteria. If the best or the worst criterion is more than one, any one among them can arbitrarily be chosen.

3. Evaluate a Best-to-Others vector with numbers between 1 and 9. Reciprocals are not used.

$$A_B = \left(a_{B1}, \ldots, a_{Bj}, \ldots, a_{Bn}\right). \tag{1}$$

4. Evaluate an others-to-worst vector with numbers between 1 and 9.

$$A_W = \left(a_{1W}, \ldots, a_{jW}, \ldots, a_{nW}\right)^T. \tag{2}$$

5: Solve the optimization model below to obtain the weights $w = \{w_j\}$.

$$\begin{aligned} &\min \max_j \left\{ \left| \frac{w_B}{w_j} - a_{Bj} \right|, \left| \frac{w_j}{w_W} - a_{jW} \right| \right\} \\ &\text{S.T.} \quad \sum_j w_j = 1, \\ &\qquad w_j \geq 0, \forall j. \end{aligned} \tag{3}$$

The solution of the model above is transferred to the form below.

$$\begin{aligned} &\min \xi \\ &\text{S. T.} \left| \frac{w_B}{w_j} - a_{Bj} \right| \leq \xi, \forall j \\ &\qquad \left| \frac{w_j}{w_W} - a_{jW} \right| \leq \xi, \forall j \\ &\qquad \sum_j w_j = 1, w_j \geq 0, \forall j. \end{aligned} \tag{4}$$

Solving Problem (4) to obtain ξ^*. Alternatively,

$$\begin{aligned} &\min \max_j \left\{ \left| w_B - a_{Bj} w_j \right|, \left| w_j - a_{jW} w_W \right| \right\} \\ &\text{S. T.} \sum_j w_j = 1, w_j \geq 0, \forall j. \end{aligned} \tag{5}$$

The model above is transferred to the form below.

$$\begin{aligned} &\min \xi \\ &\text{S. T.} \left| w_B - a_{Bj} w_j \right| \leq \xi, \forall j \\ &\qquad \left| w_j - a_{jW} w_W \right| \leq \xi, \forall j \\ &\qquad \sum_j w_j = 1, w_j \geq 0, \forall j. \end{aligned} \tag{6}$$

The consistency ratio (CR) is the ratio of ξ^* to the consistent index (CI).

$$CR = \frac{\xi^*}{CI}. \tag{7}$$

Rezaei (2015, 2016) did not discuss the threshold values for consistency of the paired vectors until his team (Liang et al. 2020) recently discussed this topic. The discussion of thresholds for CR is beyond the scope of this research, although there are a lot of problems, which will discuss in the future study.

2.2 Problems and motivations

Several problems are identified as follows: Firstly, Models (3)–(6) have ill definitions for $w_j \geq 0$. Considering $\frac{w_B}{w_j}$, w_j cannot be 0, i.e. $w_j \neq 0$, due to the divided-by-zero error. Since $\sum_j w_j = 1$ and $w_j \neq 0$, the improper constraint $w_j \geq 0$ should be changed to the constraint as below.

$$0 < w_j < 1, \forall j, \text{i.e.} w_j \in (0, 1), \forall j. \tag{8}$$

$(0, 1)$ means from 0 to 1 exclusive. Equation (8) can avoid divided-by-zero error for the optimizer based on numerical solution.

Secondly, ξ should be bounded, i.e. $CI \geq \xi \geq 0$ for Model (4) and $\xi \geq 0$ for Model (5). Whilst the upper bound of Model (6) is beyond discussion in this study, the upper bound of Model (4) can refer to the simple algebraic form in Eq. (42) to guide the solver not to try the negative values for ξ, but the values between the lower bound of zero and the upper bound less than or equal to the Consistency Index value with respect to the best-over-worst score a_{BW}, which is the worst case for ξ^* when a_{BW} is used for a_{Bj} and a_{jW}.

Thirdly, the absolute function cannot be used for some optimization algorithms. To remove the absolute function, Model (4) is equivalent to Model (17) shown in Sect. 3. In addition, unlike the AHP's reciprocal matrix, none of the matrix operations are used in BWM; transposition in Eq. (2) is completely useless or confused for Models (3) to

(7), especially when the models are implemented into programming codes, and therefore, A_W is defined without transposition in this study. Regarding the use of subscripts, the notations *i,j,k* and others are chosen in order.

Fourthly, whilst Model (3) is not equivalent to Model (5), Model (4) is not equivalent to Model (6), as they produce different objective values for the same notationξ. Only the ξ produced by solving Model (4) isξ^*. In this paper, Model (3) or (4) is called *CR* optimization model. The simulation results in this study show that Models (3) and (4) produce different versions of weights for the same objective value ξ^*, whilst Models (5) and (6) should produce unique results. Two model categories are completely not equivalent. As the examples are shown in Sect. 9, it is recommended that the weights are produced by Model (5) or (6), whilst Model (3) or (4) produces ξ^* for consistency only, although Models (5) and (6) are very different from Models (3) and (4) in nature, especially for their weights and objective values produced, although even the recent paper (Liang et al. 2020) with the original BWM author did not point out this.

In this study, the closed-form solutions for CR shown in Eq. (7) are developed, and thus, the optimization solver software applications to solve Model (3) or (4) are not necessary, as exact precision of ξ^* can be obtained by much simpler proposed closed forms. The measurement of prioritization weights is beyond the scope of this study. Normally, most BWM prioritization methods are the reduced forms of the AHP's prioritization methods, where the comparisons of AHP prioritizations can be found in Yuen (2009, 2010).

3 Equivalent forms without absolute functions

To understand more properties of ξ^*, absolute functions for Models (4) and (6) are firstly removed, and the following proposition holds.

Proposition 1 (Equivalent Forms of Absolute Function) *Let x be any real number. If* $|x| \leq \xi$ *and* $\xi \geq 0$*, the following two inequalities hold.*

$$x \leq \xi, \tag{9}$$

$$-x \leq \xi. \tag{10}$$

As the absolute value of any real number is positive, $\xi \geq 0$*. With respect to* $|x| \leq \xi$*, if either* $x \geq 0$ *or* $x \leq 0$*, Eqs.* (9) and (10) *hold. For example, let* $x = 4, \xi = 5$*. According to Eqs.* (9) and (10)*, either* $4 \leq 5$ *or* $-4 \leq 5$ *is true. Similarly, if* $x = -4$*, either* $-4 \leq 5$ *or* $-(-4) \leq 5$ *is true. In short, If* $|\pm 4| \leq 5$*, then* $\pm 4 \leq 5$*.*

Analogue to the example above, let $x = \frac{w_B}{w_i} - a_{Bi}$. The inequality with the absolute function

$$\left| \frac{w_B}{w_i} - a_{Bi} \right| \leq \xi \tag{11}$$

is equivalent to two forms without the absolute function below:

$$\frac{w_B}{w_i} - a_{Bi} \leq \xi, \tag{12}$$

$$-\frac{w_B}{w_i} + a_{Bi} \leq \xi. \tag{13}$$

Similarly, the inequality with the absolute function

$$\left| \frac{w_i}{w_W} - a_{iW} \right| \leq \xi \tag{14}$$

is equivalent to two forms without the absolute function below:

$$\frac{w_i}{w_W} - a_{iW} \leq \xi, \tag{15}$$

$$-\frac{w_i}{w_W} + a_{iW} \leq \xi. \tag{16}$$

Finally, Model (4) is equivalent to

$$\begin{aligned} &\min \xi \\ &\text{S.T.} \frac{w_B}{w_i} - a_{Bi} \leq \xi, \forall i \\ &\quad -\frac{w_B}{w_i} + a_{Bi} \leq \xi, \forall i \\ &\quad \frac{w_i}{w_W} - a_{iW} \leq \xi, \forall i \\ &\quad -\frac{w_i}{w_W} + a_{iW} \leq \xi, \forall i \\ &\quad \sum_i w_i = 1, 0 \leq \xi \leq CI, 0 < w_i < 1, \forall i. \end{aligned} \tag{17}$$

Similarly, Model (6) is equivalent to

$$\begin{aligned} &\min \xi \\ &\text{S. T.} w_B - w_i a_{Bi} \leq \xi, \forall i \\ &\quad - w_B + w_i a_{Bi} \leq \xi, \forall i \\ &\quad w_i - w_W a_{iW} \leq \xi, \forall i \\ &\quad - w_i + w_W a_{iW} \leq \xi, \forall i \\ &\quad \sum_i w_i = 1, \xi \geq 0, 0 < w_i < 1, \forall i. \end{aligned} \tag{18}$$

For the advantage, Model (17) is easier to be implemented with most software packages than Model (4) as treating absolute function is not needed anymore for some optimization algorithms. Rezaei (2015, 2016) did not define a meaningful name for the notation ξ or ξ^* in Model (4). On the basis of the mathematical formation in the Model (4), ξ^* is called the Minmax Edge Error (MEE) in this paper, which is the minimum of maximum absolute distance between $\frac{w_B}{w_j}$ and a_{Bj}, as well as $\frac{w_j}{w_W}$ and a_{jW}, for all

j. More properties are explored for the edge errors in the following sections.

4 Individual edge errors

If a pair of best and worst vectors, A_B and A_W, are perfectly consistent, the following equality holds, and vice versa.

$$a_{Bi}a_{iW} = a_{BW}, \forall i \in \{1, \ldots, n\}. \tag{19}$$

The inconsistency is induced if one of two conditions below is satisfied.

$$a_{Bi}a_{iW} > a_{BW}, \exists i; \tag{20}$$

$$a_{Bi}a_{iW} < a_{BW}, \exists i. \tag{21}$$

Let an Individual Edge Error (IEE) for c_i in comparison with c_B and c_W be ξ_i. If $a_{Bi}a_{iW} > a_{BW}$, a_{Bi} and a_{iW} are decreased by ξ_i, respectively, and a_{BW} is increased by ξ_i, to achieve an equality as below:

$$(a_{Bi} - \xi_i)(a_{iW} - \xi_i) = (a_{BW} + \xi_i). \tag{22}$$

Expand the form above to have a quadratic equation below:

$$\xi_i^2 - (1 + a_{Bi} + a_{iW})\xi_i + (a_{Bi}a_{iW} - a_{BW}) = 0. \tag{23}$$

On the other hand, if $a_{Bi}a_{iW} < a_{BW}$, a_{Bi} and a_{iW} are decreased by ξ_i, respectively, and a_{BW} is increased byξ_i, to achieve an equality as below:

$$(a_{Bi} + \xi_i)(a_{iW} + \xi_i) = (a_{BW} - \xi_i). \tag{24}$$

Expand the form above to have a quadratic equation as below:

$$\xi_i^2 + (1 + a_{Bi} + a_{iW})\xi_i + (a_{Bi}a_{iW} - a_{BW}) = 0. \tag{25}$$

To find the ξ_i for the two equality relationships above, Theorem 1 holds.

Theorem 1 (Individual Edge Error). *The Individual Edge Error (IEE) for c_j in comparisons with c_B and c_W is obtained as below:*

$$\xi_i = \frac{1}{2}\left|(1 + a_{Bi} + a_{iW}) - \sqrt{(1 + a_{Bi} + a_{iW})^2 - 4(a_{Bi}a_{iW} - a_{BW})}\right|, \forall i. \tag{26}$$

Precisely, the absolute form above is equivalent to the piecewise equation below with specifying conditions:

$$\xi_i = \begin{cases} \dfrac{-(1 + a_{Bi} + a_{iW}) + \sqrt{(1 + a_{Bi} + a_{iW})^2 - 4(a_{Bi}a_{iW} - a_{BW})}}{2}, a_{Bi}a_{iW} - a_{BW} < 0 \\ \dfrac{(1 + a_{Bi} + a_{iW}) - \sqrt{(1 + a_{Bi} + a_{iW})^2 - 4(a_{Bi}a_{iW} - a_{BW})}}{2}, a_{Bi}a_{iW} - a_{BW} > 0 \\ 0, a_{Bi}a_{iW} = a_{BW} \end{cases}, \forall i. \tag{27}$$

Proof.

$$\text{Let } \gamma_{i1} = 1, \tag{28}$$

$$\gamma\prime_{i2} = \begin{cases} \gamma_{i2} \\ -\gamma_{i2} \end{cases} = \begin{cases} (1 + a_{Bi} + a_{iW}), & \gamma_{i3} \le 0 \\ -(1 + a_{Bi} + a_{iW}), & \gamma_{i3} \ge 0 \end{cases}, \tag{29}$$

$$\gamma_{i3} = a_{Bi}a_{iW} - a_{BW}, \forall i \in [1, \ldots, n]. \tag{30}$$

To substitute the above equations to Eqs. (23) and (25),

$$\gamma_{i1}\xi_i^2 + \gamma\prime_{i2}\xi_i + \gamma_{i3} = 0. \tag{31}$$

To solve the above quadratic equation to have two roots,

$$\xi\prime_i = \frac{-\gamma\prime_{i2} \pm \sqrt{\gamma\prime_{i2}^2 - 4\gamma_{i1}\gamma_{i3}}}{2\gamma_{i1}} = \frac{1}{2}\left(-\gamma\prime_{i2} \pm \sqrt{\gamma\prime_{i2}^2 - 4\gamma_{i1}\gamma_{i3}}\right). \tag{32}$$

As the edge error is positive and minimized,

$$\xi_i = \min(|\xi\prime_i|). \tag{33}$$

Precisely, without absolute function, as

$$\frac{1}{2}\left(-\gamma_{i2} - \sqrt{\gamma_{i2}^2 - 4\gamma_{i1}\gamma_{i3}}\right) \ge \frac{1}{2}\left(-\gamma_{i2} + \sqrt{\gamma_{i2}^2 - 4\gamma_{i1}\gamma_{i3}}\right), \gamma_{i3} \le 0, \tag{34}$$

$$\frac{1}{2}\left(\gamma_{i2} - \sqrt{\gamma_{i2}^2 - 4\gamma_{i1}\gamma_{i3}}\right) \le \frac{1}{2}\left(\gamma_{i2} + \sqrt{\gamma_{i2}^2 - 4\gamma_{i1}\gamma_{i3}}\right), \gamma_{i3} \ge 0, \tag{35}$$

the $\min(\left|\xi'_i\right|)$ is

$$\xi_i = \begin{cases} \dfrac{1}{2}\left(-\gamma_{i2} + \sqrt{{\gamma_{i2}}^2 - 4\gamma_{i1}\gamma_{i3}}\right), \gamma_{i3} \le 0 \\ \dfrac{1}{2}\left(\gamma_{i2} - \sqrt{{\gamma_{i2}}^2 - 4\gamma_{i1}\gamma_{i3}}\right), \gamma_{i3} \ge 0 \end{cases}. \tag{36}$$

Therefore, Eqs. (26) and (27) hold. □

The piecewise form of three cases of Eq. (27) can be reduced to two cases by removing the third case with adding equality to the conditions, e.g. in Eq. (36). The advantage of the piecewise form of three cases of Eq. (27) is that the Case 3 can be determined immediately.

Individual Edge Error (IEE) for c_j compared with c_B and c_W is independent of the rating scores with respect to the other criteria compared. However, if another criterion is involved, an impact to the MEE may be induced, and therefore, the concept of co-edge errors is established in the next section.

5 Co-edge errors

If the rating scores are perfectly consistent, the following equality holds.

$$a_{Bi}a_{iW} = a_{Bj}a_{jW} = a_{BW}. \tag{37}$$

If there is inconsistency, the equality above does not hold. In addition to the IEE introduced in the previous section, the concept of co-edge error ξ_{ij} is introduced. Co-edge error ξ_{ij} is the multiplication discrepancy between $a_{Bi}a_{iW}$ and $a_{Bj}a_{jW}$ satisfying the following two conditions.

If $a_{Bi}a_{iW} \leq a_{Bj}a_{jW}$, both a_{Bi} and a_{iW} are increased by ξ_{ij}, and both a_{Bj} and a_{jW} are decreased by ξ_{ij}, to achieve an equality as below:

$$\left(a_{Bi} + \xi_{ij}\right)\left(a_{iW} + \xi_{ij}\right) = \left(a_{Bj} - \xi_{ij}\right)\left(a_{jW} - \xi_{ij}\right). \tag{38}$$

If $a_{Bi}a_{iW} \geq a_{Bj}a_{jW}$, both a_{Bi} and a_{iW} are decreased by ξ_{ij}, and both a_{Bj} and a_{jW} are decreased by ξ_{ij}, to achieve an equality as below:

$$\left(a_{Bi} - \xi_{ij}\right)\left(a_{iW} - \xi_{ij}\right) = \left(a_{Bj} + \xi_{ij}\right)\left(a_{jW} + \xi_{ij}\right). \tag{39}$$

To find the ξ_{ij} for the equality relationships above, Theorem 2 holds.

Theorem 2 (Co-Edge Error) *A Co-Edge Error (CEE) with respect to c_i and c_j is computed by the form below:*

$$\xi_{ij} = \begin{cases} \dfrac{a_{Bi}a_{iW} - a_{Bj}a_{jW}}{a_{Bi} + a_{iW} + a_{Bj} + a_{jW}}, a_{Bi}a_{iW} \geq a_{Bj}a_{jW} \\ \dfrac{a_{Bj}a_{jW} - a_{Bi}a_{iW}}{a_{Bi} + a_{iW} + a_{Bj} + a_{jW}}, a_{Bi}a_{iW} \leq a_{Bj}a_{jW} \end{cases}. \tag{40}$$

Or the piecewise equation above is equivalent to the below form without specifying conditions:

$$\xi_{ij} = \frac{\left|a_{Bi}a_{iW} - a_{Bj}a_{jW}\right|}{a_{Bi} + a_{iW} + a_{Bj} + a_{jW}}. \tag{41}$$

Proof. If $a_{Bi}a_{iW} \leq a_{Bj}a_{jW}$, rearrange Eq. (38) as below:

$$\xi_{ij}^2 + (a_{Bi} + a_{iW})\xi_{ij} + a_{Bi}a_{iW} = \xi_{ij}^2 - \left(a_{Bj} + a_{jW}\right)\xi_{ij} + a_{Bj}a_{jW}$$

$$\left(a_{Bi} + a_{iW} + a_{Bj} + a_{jW}\right)\xi_{ij} = a_{Bj}a_{jW} - a_{Bi}a_{iW}$$

$$\xi_{ij} = \frac{a_{Bj}a_{jW} - a_{Bi}a_{iW}}{\left(a_{Bi} + a_{iW} + a_{Bj} + a_{jW}\right)}.$$

If $a_{Bi}a_{iW} \geq a_{Bj}a_{jW}$, rearrange Eq. (39) as below:

$$\xi_{ij}^2 - (a_{Bi} + a_{iW})\xi_{ij} + a_{Bi}a_{iW} = \xi_{ij}^2 + \left(a_{Bj} + a_{jW}\right)\xi_{ij} + a_{Bj}a_{jW}$$

$$a_{Bi}a_{iW} - a_{Bj}a_{jW} = \left(a_{Bi} + a_{iW} + a_{Bj} + a_{jW}\right)\xi_{ij}$$

$$\xi_{ij} = \frac{a_{Bi}a_{iW} - a_{Bj}a_{jW}}{a_{Bi} + a_{iW} + a_{Bj} + a_{jW}}.$$

Therefore, Form (40) holds. As $\xi_{ij} \geq 0$, Eq. (41) also holds. □

6 Max of edge error matrix method

The consistency ratio (CR) is the ratio of Minimax Edge Error (MEE or ξ^*) to the consistent index (CI), i.e. $CR = \frac{\xi^*}{CI}$. The closed-form solution of CI is derived by Eq. (42) of Theorem 3, whilst the closed-form solution of Minmax Edge Error ξ^* is derived by Eq. (44) of Theorem 4. The details are presented as below.

6.1 Closed-form solution of consistency index

The BWM applications only refer to a CI table stated in (Rezaei 2015, 2016) for the CI with respect to a_{BW}. In fact, the CI can be derived from the simple form in Eq. (42) of Theorem 3 stated as below.

Theorem 3 (Consistency Index of BWM) *The Consistency Index of a_{BW} is computed by the form below:*

$$CI_{a_{BW}} = CI(a_{BW}) = \frac{1}{2}\left(1 + 2a_{BW} - \sqrt{8a_{BW} + 1}\right). \tag{42}$$

Proof. From Eq. (27), if $a_{Bi}a_{iW} - a_{BW} > 0$, then.

$$\xi_i = \frac{(1 + a_{Bi} + a_{iW}) - \sqrt{(1 + a_{Bi} + a_{iW})^2 - 4(a_{Bi}a_{iW} - a_{BW})}}{2}.$$

Substitute $a_{Bi} = a_{iW} = a_{BW}$, the worst case, to the form above to obtain $CI(a_{BW})$ below:

$$CI(a_{BW}) = \frac{(1 + a_{BW} + a_{BW}) - \sqrt{(1 + a_{BW} + a_{BW})^2 - 4(a_{BW}a_{BW} - a_{BW})}}{2}$$

$$CI(a_{BW}) = \frac{(1 + 2a_{BW}) - \sqrt{(1 + 2a_{BW})^2 - 4a_{BW}^2 + 4a_{BW}}}{2}$$

$$CI(a_{BW}) = \frac{(1 + 2a_{BW}) - \sqrt{4a_{BW}^2 + 4a_{BW} + 1 - 4a_{BW}^2 + 4a_{BW}}}{2}$$

$$CI(a_{BW}) = \frac{1}{2}\left((1 + 2a_{BW}) - \sqrt{8a_{BW} + 1}\right).$$

Thus, Eq. (42) holds. □

To have the CR, CI is obtained as above, and the formulation and calculation of ξ^* are discussed in the following section.

6.2 Closed-form solution of minmax edge error

ξ^* can be obtained by finding the largest value of the elements in the Edge Error Matrix (EEM) denoted by $\ddot{\xi}$ of the form below:

$$\ddot{\xi} = \left\{\ddot{\xi}_{ij} : \ddot{\xi}_{ij} = \begin{cases} \xi_i, i = j \\ \xi_{ij}, i \neq j \end{cases}, \forall i, j \in [1, \ldots, n]\right\}. \quad (43)$$

The size of largest values is at least one. $\ddot{\xi}$ is the $n \times n$ symmetric matrix. The diagonal elements where $i = j$ are filled byξ_i,$\forall i$, which are obtained by Eq. (26) or (27). The non-diagonal elements are filled by ξ_{ij}, $i \neq j$,$\forall i$, $\forall j$, which are obtained by Eq. (40) or (41). Elements in the upper triangular matrix of $\ddot{\xi}$, denoted by$\ddot{\xi}^+$, are the same as their corresponding positions in lower triangular matrix of$\ddot{\xi}$, denoted by $\ddot{\xi}^-$. Thus, $\ddot{\xi}$ is a symmetric matrix, i.e.$\xi_{ij} = \xi_{ji}, \forall i, j$. Theorem 4 is established on top of Theorems 1 and 2 to compute the EEM.

Theorem 4 (Max of Edge Error Matrix Method: $\xi^* = max(\ddot{\xi}^+)$) *The Minmax edge error is the largest value of the set of all individual edge errors and all co-edge errors, which can be formed by the upper triangular edge error matrix.*

$$\xi^* = max\left(\ddot{\xi}^+\right) = max(\{\xi_i\}, \{\xi_{ij}\})$$
$$= max\left\{\ddot{\xi}_{ij}^+ : \ddot{\xi}_{ij}^+ = \begin{cases} \xi_i, i = j \\ \xi_{ij}, i \neq j \end{cases}, \forall j \in [i, \ldots, n], \forall i \in [1, \ldots, n]\right\}$$

$$\xi_i = \frac{1}{2}\left|(1 + a_{Bi} + a_{iW}) - \sqrt{(1 + a_{Bi} + a_{iW})^2 - 4(a_{Bi}a_{iW} - a_{BW})}\right|, \forall i \quad (44)$$

$$\xi_{ij} = \frac{|a_{Bi}a_{iW} - a_{Bj}a_{jW}|}{a_{Bi} + a_{iW} + a_{Bj} + a_{jW}}, \forall j \in [i + 1, \ldots, n], \forall i \in [1, \ldots, n].$$

Proof. ξ_i and ξ_{ij} have been proved in Theorems 1 and 2, respectively, with alternative forms without absolute function. In order to have Theorem 4, it is essential to show how Theorems 1 and 2 are related to Model (4). Let $(i^*, j^*) = argmax\left(\ddot{\xi}^+\right)$, which occurs either only at i^*, where $i^* = j^*$, or between i^* and j^*, where $i^* \neq j^*$. If ξ^* occurs at i^*, there are two scenarios shown in Cases 1 and 2. If ξ^* occurs at between i^* and j^*, there are two scenarios shown in Cases 3 and 4.

Case 1 ($a_{Bi^*}a_{i^*W} < a_{BW}$):

If ξ^* occurs at i^* such that $a_{Bi^*}a_{i^*W} < a_{B(i \neq i^*)}a_{(i \neq i^*)W}, \forall i$, substitute $\xi^* = \xi_{i^*} = \xi_i$ to Eq. (24) to have

$$(a_{Bi^*} + \xi^*)(a_{i^*W} + \xi^*) = (a_{BW} - \xi^*), \quad (45)$$

substituted by

$$\frac{w_B}{w_{i^*}} = a_{Bi^*} + \xi^* \text{ and } \frac{w_{i^*}}{w_W} = a_{i^*W} + \xi^* \quad (46)$$

to have

$$\left(\frac{w_B}{w_{i^*}}\right)\left(\frac{w_{i^*}}{w_W}\right) = \frac{w_B}{w_W} = (a_{BW} - \xi^*). \quad (47)$$

Therefore, if $\xi^* = \xi_{i^*} \geq \xi_i, i \neq i^*, \forall i$, then Eq. (47) equals Eq. (45). We can see the relationship among w_{i^*},w_B, w_W and a_{Bi^*},a_{i^*W}, a_{i^*W} and ξ^*.

Case 2 ($a_{Bi^*}a_{i^*W} > a_{BW}$):

ξ^* occurs at i^* such that $a_{Bi^*}a_{i^*W} > a_{B(i\neq i^*)}a_{(i\neq i^*)W}, \forall i$. From Eq. (22),

$$(a_{Bi^*} - \xi^*)(a_{i^*W} - \xi^*) = (a_{BW} + \xi^*), \tag{48}$$

substituted by

$$\frac{w_B}{w_{i^*}} = a_{Bi^*} - \xi^* \text{ and } \frac{w_{i^*}}{w_W} = a_{i^*W} - \xi^* \tag{49}$$

to have

$$\left(\frac{w_B}{w_{i^*}}\right)\left(\frac{w_{i^*}}{w_W}\right) = \frac{w_B}{w_W} = (a_{BW} + \xi^*). \tag{50}$$

Therefore, the relationship between Eqs. (48) and 50) is shown.

Case 3 ($a_{Bi^*}a_{i^*W} < a_{BW} < a_{Bj^*}a_{j^*W}$):

ξ^* occurs at $i*$ and j^* such that $a_{Bi^*}a_{i^*W} < a_{Bj^*}a_{j^*W}$. From Eq. (38),

$$(a_{Bi^*} + \xi^*)(a_{i^*W} + \xi^*) = \left(a_{Bj^*} - \xi^*\right)\left(a_{j*W} - \xi^*\right), \tag{51}$$

substituted by

$$\frac{w_B}{w_{i^*}} = a_{Bi^*} + \xi^*, \frac{w_i*}{w_W} = a_{i^*W} + \xi*, \frac{w_B}{w_{j*}} = a_{Bj*} - \xi^* \text{ and } \frac{w_j*}{w_W} = a_{j*W} - \xi^* \tag{52}$$

to have

$$\left(\frac{w_B}{w_{i^*}}\right)\left(\frac{w_{i^*}}{w_W}\right) = \left(\frac{w_B}{w_{j^*}}\right)\left(\frac{w_{j^*}}{w_W}\right) = \frac{w_B}{w_W} = \left(a_{Bj^*} - \xi^*\right)\left(a_{j^*W} - \xi^*\right) = (a_{Bi^*} + \xi*)(a_{i^*W} + \xi^*). \tag{53}$$

This means that the above equality holds only if $i = i^*$ and $j = j^*$.

Case 4 ($a_{Bi*}a_{i*W} > a_{BW} > a_{Bj*}a_{j*W}$):

ξ^* occurs at $j*$ and $i*$ such that $a_{Bi^*}a_{i^*W} > a_{Bj*}a_{j*W}$. From Eq. (39),

$$(a_{Bi^*} - \xi^*)(a_{i^*W} - \xi^*) = \left(a_{Bj^*} + \xi^*\right)\left(a_{j^*W} + \xi^*\right), \tag{54}$$

substituted by

$$\frac{w_B}{w_{i^*}} = a_{Bi*} - \xi^*, \frac{w_i*}{w_W} = a_{i^*W} - \xi^*, \frac{w_B}{w_{j*}} = a_{Bj^*} + \xi^* \text{ and } \frac{w_j*}{w_W} = a_{j^*W} + \xi^* \tag{55}$$

to have

$$\left(\frac{w_B}{w_{i^*}}\right)\left(\frac{w_{i^*}}{w_W}\right) = \left(\frac{w_B}{w_{j^*}}\right)\left(\frac{w_{j^*}}{w_W}\right) = \frac{w_B}{w_W} = (a_{Bi^*} - \xi^*)(a_{i^*W} - \xi^*) = \left(a_{Bj^*} + \xi^*\right)\left(a_{j^*W} + \xi^*\right). \tag{56}$$

The above equality holds if $i = i^*$ and $j = j^*$.

In short, Minmax Edge Error occurs if one of the above four conditions holds such that $\xi^* = max\left(\{\xi_i\}, \{\xi_{ij}\}\right)$. □

Theorem 4 shows the exhaustive approach to find the Minmax Edge Error, i.e. $IEE = max(EEM)$. Firstly, Individual Edge Errors for all criteria are calculated. Secondly, co-edge errors for all pairs among the criteria are calculated. Finally, MEE is the maximal value of the individual edge errors and co-edge errors. The computation size of Individual Edge Errors is n, whilst the computation size of co-edge errors is $\frac{1}{2}(n-1)(n-2)$, and $\xi_{ij} = \xi_{ji}, \forall i,j$. The advantage of the exhaustive method is that the Edge Error Matrix can provide an overview of the edge errors. If only Minmax Edge Error is introduced, a smarter method may be needed to avoid unnecessary computation, and the determinant method for MEE is introduced in the next section by investigating more properties of Theorem 4.

7 Minmax edge error determinant method

Theorem 4 is the exhaustive search for the upper triangle EEM that requires $\frac{1}{2}n(n-1)$ calculations. The MEE determinant method is proposed to find the ξ^* by initially determining locations inducing ξ^*. Algorithm 1 is presented to show the determinant method to find MEE. Algorithm 1 only concerns the MEE without providing the information for the other edge errors shown in the edge error matrix. For the definitions in this paper, the best criterion is located at B, the worst criterion is located at W, and the rest should be called "others" if the rest size is plural, or "another" if the size is singular.

For the comparison located at best criterion, i.e. $i = B$,

$$a_{BB}a_{BW} - a_{BW} = 0. \tag{57}$$

For the comparison located at best criterion, i.e. $i = W$,

$$a_{BW}a_{WW} - a_{BW} = 0. \tag{58}$$

By Theorem 1, both IEE results in 0, i.e. $\xi_B = \xi_W = 0$. By Theorem 2, both CEE results in 0, i.e. $\xi_{BW} = \xi_{WB} = 0$. In addition, $\xi_i \geq |\xi_{iW}|$ or $\xi_i \geq |\xi_{Bi}|$. Thus, when finding the locations of criteria for ξ^*, rating scores at B in A_B and W in A_W can be removed. In other words, only the others in A_B and A_W are considered. If $n = 3$, the size of the others is 1,

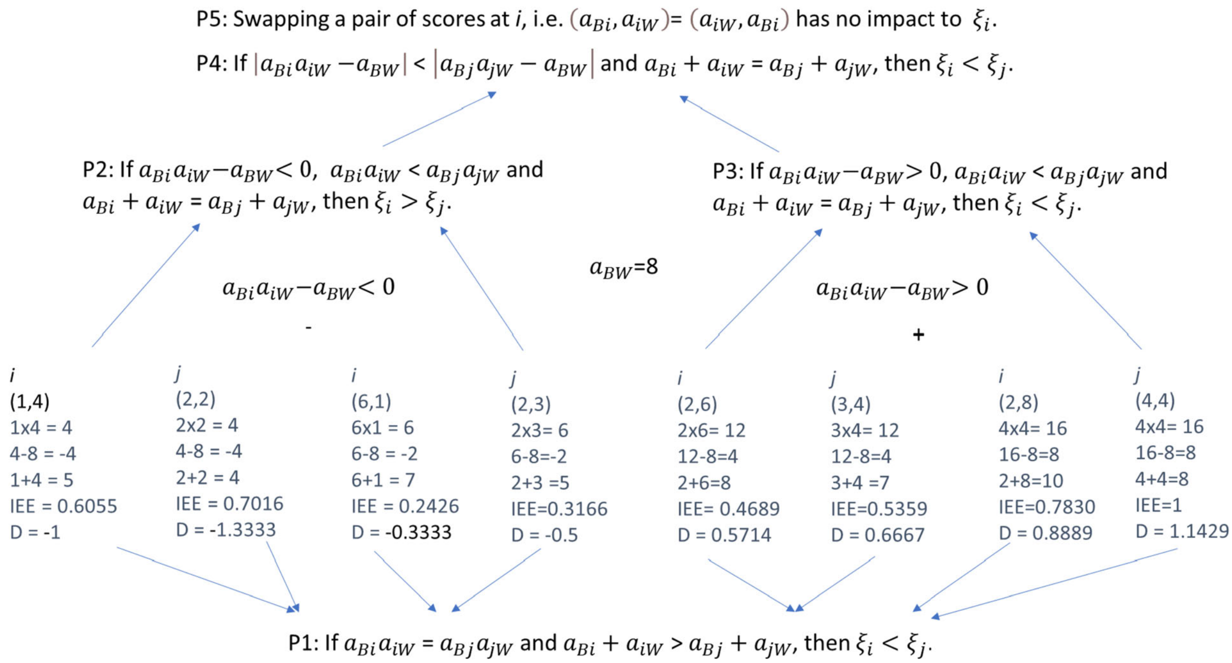

Fig. 1 Examples of the relationships of five properties of IEE

and thus, MEE is the IEE based on paired comparisons related to the other criterion. When $n > 3$, both IEE and CEE are considered.

According to Eq. (27) of Theorem 1, five properties of IEE concerning inequality and equality are further inducted as below. To better understand the properties, examples of pair scores for their relationships are demonstrated in Fig. 1, which is further studied in Example 6 of Sect. 9.

1. If $a_{Bi}a_{iW} = a_{Bj}a_{jW}$ and $a_{Bi} + a_{iW} > a_{Bj} + a_{jW}$, then $\xi_i < \xi_j$.
2. If $a_{Bi}a_{iW} - a_{BW} < 0$, $a_{Bi}a_{iW} < a_{Bj}a_{jW}$ and $a_{Bi} + a_{iW} = a_{Bj} + a_{jW}$, then $\xi_i > \xi_j$.
3. If $a_{Bi}a_{iW} - a_{BW} > 0$, $a_{Bi}a_{iW} < a_{Bj}a_{jW}$ and $a_{Bi} + a_{iW} = a_{Bj} + a_{jW}$, then $\xi_i < \xi_j$.
4. If $|a_{Bi}a_{iW} - a_{BW}| < |a_{Bj}a_{jW} - a_{BW}|$ and $a_{Bi} + a_{iW} = a_{Bj} + a_{jW}$, then $\xi_i < \xi_j$.
5. Swapping a pair of scores at i, i.e. (a_{Bi}, a_{iW})= (a_{iW}, a_{Bi}), has no impact to ξ_i.

From the properties above, $a_{Bk}a_{kW}$ and $a_{Bk} + a_{kW}$ are the core parts for the determination. These properties are essentially used to determine the location of MEE. The highest IEE is more likely to be MEE as the highest CEE is also required to check. In addition, considering Eqs. (40) and (41) from Theorem 2 and the simulation results performed in Sect. 9, the determinant values of others are defined below:

$$D_k = \frac{a_{Bk}a_{kW} - a_{BW}}{a_{Bk} + a_{kW} - 1}, \forall k \in [1, \ldots, p], a_{BW} \leq 9. \tag{59}$$

The value p is the size of the other criteria excluding the best and worst criteria. In short, when the numerator, $a_{Bk} + a_{kW}$, is higher and the pure value of denominator, $|a_{Bk}a_{kW} - a_{BW}|$, is lower, chance of ξ_k to be ξ^* is higher. The -1 is selected based on the simulation results of one million random samples in Sect. 9. $+1$ or 0 does not produce accurate MEE results. Without absolute value, the determinant value has direction, and so the *minimum* function is used to address Property 2 when D_k is negative and *maximum* function is used to address Property 3 when D_k is positive. According to the simulation, the highest CEE is located between a pair with the least negative D_i and the highest positive D_j, when $a_{BW} \geq 9$. When $a_{BW} > 9$, Eq. (59) may not be valid. Finally, Algorithm 1 is formed to determine ξ^* with the least computation.

For Algorithm 1, when $n > 3$, the extra little computation is to find the D to avoid computing all CEEs and IEEs. If $a_{Bj}a_{jW} < a_{Bi}a_{iW}$ and $a_{Bi}a_{iW} - a_{BW} < 0$, then $\xi_j > \xi_i$, and Eq. (60) is used. If $a_{Bj}a_{jW} > a_{Bi}a_{iW}$ and $a_{Bi}a_{iW} - a_{BW} > 0$, then $\xi_j > \xi_i$, and Eq. (61) is used. If $a_{Bi}a_{iW} - a_{BW} < 0$ and $a_{Bj}a_{jW} - a_{BW} > 0$, then ξ^* is determined by Eq. (62). Case 3 includes Cases 1 and 2 including Case 0. Basically, Case 3 can be used for general purpose, but not the fastest one to determine ξ^* when the cases fit for Cases 0 to 2. The demonstrations are shown in Sect. 9.

Algorithm 1 ($detMEE(A_B, A_W)$: Determinant Method for ξ^*).

Input: paired vectors: A_B and A_W such that $a_{BW} \leq 9$

Step 1: exclude B and W in in A_B and A_W to form $A\prime_B$ and $A\prime_W$, where the new size of each vector is$p = n - 2$. The criteria indices of $A\prime_B$ and $A\prime_W$ are updated

Step 2: if $p = 1$, i.e., $n = 3$, compute $\xi^* = \xi_1$, and go to return ξ^*. Otherwise, go to the next step

Step 3: compute $D = \left\{\frac{a_{Bk}a_{kW} - a_{BW}}{a_{Bk} + a_{kW} - 1} : \forall k \in [1, \ldots, p]\right\}$

Step 4: compute ξ^* with respect to different cases

Case 0: if all elements in D are zero, i.e. $min(D) = min(D) = 0$, then $\xi^* = 0$

Case 1: if all elements in D are positive, i.e. $min(D) \geq 0$, find the index set of highest values in D to calculate the MEE

$\xi^* = \max_{i \in S^+}(\xi_i), S^+ = argmax(D)$ (60)

Case 2: if all elements in D are negative, i.e. $max(D) \leq 0$, find the index set of lowest values in D to calculate the MEE

$\xi^* = \max_{i \in S^-}(\xi_i), S^- = argmin(D)$ (61)

Case 3: if elements in D contain both positive and negative numbers, i.e. $min(D) < 0$ and $max(D) > 0$, calculate the MEE

$\xi^* = max\left(\{\xi_i : i \in \{S^+ \cup S^-\}\}, \{\xi_{ij} : i \in S^+, j \in S^-\}\right)$, $S^+ = argmax(D), S^- = argmin(D)$ (62)

Return ξ^*

8 Two-step optimization models

As the closed-form solution of Minmax Edge Error can be obtained by Theorem 4 or Algorithm 1, the Minmax Model (3) and its equivalent Model (4) can be improved and converted to two-step optimization models to find the weights with the exact optimal objective value of Model (3) or (4). The goal of two-step optimization is to find the Minmax Edge Error in the first step and find the weights in the second step. The details of two-step optimization model are shown in Algorithm 2.

Algorithm 2 (Two-step optimization model with known ξ^* for weights).

Input: A_B and A_W

Step 1: get ξ^* by Eq. (44) of Theorem 4, or Algorithm 1

Step 2: compute the parameters by

$$\begin{aligned} & min\left|1 - \textstyle\sum_i w_i\right| \\ & S.T.\ \left|\frac{w_B}{w_i} - a_{Bi}\right| \leq \xi^*, \forall i \\ & \left|\frac{w_i}{w_W} - a_{iW}\right| \leq \xi^*, \forall i \\ & 0 < w_i < 1, \forall i \quad (63) \end{aligned}$$

Return: w

For Algorithm 2, as ξ^* is known after step 1, the constraint $\sum_i w_i = 1$ is changed to the objective function $\min\left|1 - \sum_i w_i\right|$, in which the expected objective value (or difference) must reach to zero. There are several alternative forms for Model (63) in step 2. Alternatively, the absolute error can be changed to the squared error form below:

$$\min\left(1 - \sum\nolimits_j w_i\right)^2. \quad (64)$$

Both absolute and squared forms can produce the weights leading to the same ξ^*, the objective value of Model (63) or (64) is expected to be minimized to zero (or very close to zero due to rounded or truncated errors from floating-point computation). Analogue to Model (17) without absolute function, the Model (63) can be changed to the equivalent model below:

$$\begin{aligned} & \min(\text{or}\max) \sum_j w_i \\ & S.T. w_B/w_i - a_B i \leq \xi^*, \forall i \\ & -\frac{w_B}{w_i} + a_{Bi} \leq \xi^*, \forall i \\ & \frac{w_i}{w_W} - a_{iW} \leq \xi^*, \forall i \\ & -\frac{w_i}{w_W} + a_{iW} \leq \xi^*, \forall i \\ & \sum_i w_i = 1, 0 < w_i < 1, \forall i. \end{aligned} \quad (65)$$

Using either *min* or *max* function, the expected objective value must ideally be one or very close to one due to little precision errors from floating-point operations, as the constraint $\sum_i w_i = 1$ is set. Alternatively, Model (63) is changed to the quadratic form below:

$$\begin{aligned} & \min\left(1 - \sum_j w_i\right)^2 \\ & S.T.\ w_B/w_i - a_B i \leq \xi^*, \forall i \\ & -\frac{w_B}{w_i} + a_{Bi} \leq \xi^*, \forall i \\ & \frac{w_i}{w_W} - a_{iW} \leq \xi^*, \forall i \\ & -\frac{w_i}{w_W} + a_{iW} \leq \xi^*, \forall i \\ & 0 < w_i < 1, \forall i. \end{aligned} \quad (66)$$

Similarly, Model (63) can be changed to the form below:

$$\min\left|1-\sum_j w_i\right|$$
$$S.T.|w_B - a_B i w_i| \le \xi^* w_i, \forall i \quad (67)$$
$$|w_i - a_{iW} w_W| \le \xi^* w_i, \forall i$$
$$0 < w_j < 1, \forall j.$$

The above model is equivalent to a quadratic programming form below:

$$\min\left(1-\sum_j w_i\right)^2$$
$$S.T. w_B - a_{Bi} w_i \le \xi^* w_i, \forall i$$
$$- w_B + a_{Bi} w_i \le \xi^* w_i, \forall i \quad (68)$$
$$w_i - a_{iW} w_W \le \xi^* w_W, \forall i$$
$$- w_i + a_{iW} w_W \le \xi^* w_W, \forall i$$
$$0 < w_i < 1, \forall i.$$

Models (67) and (68) can preserve the same ξ^* from Model (3), which is also verified by the simulation results in Sect. 9. In principle, since the objective value of Model (3) or (4) can be obtained by Theorem 4 or Algorithm 1, by observing the expected result of objective values, Models (61)–(68) could more precisely generate the weights leading to the exact ξ^* than Model (3) or (4). Models (63)–(68) can preserve the exact ξ^* if expected objective value of 0 for Model (63)–(64) and (66)–(68) is reached, and the objective value of 1 for Model (65) is reached. The choice of models may depend on the optimization algorithms. According to the simulation presented in Sect. 9, the Minmax of Model (3) and its equivalent Model (4) do not produce the unique version of weight vector with the same ξ^* if $n > 3$. Similarly, the optimization Models (63)–(68) can also produce multiple versions of weights as they are merely inherited from Models (3) and (4) with the given ξ^*. With the use of optimization Models (3)–(4) and (61)–(68), we can understand more about the properties for ξ^* and w_i, although the weights produced by them are unreliable to be used.

To measure the prioritization performance, Eqs. (69)–(71) are proposed as below. The difference between the weight ratio scores and corresponding rating scores for A_B and A_W can be shown as below:

$$\widehat{\xi} = \begin{bmatrix} \xi_B \\ \xi_W \end{bmatrix} = \begin{bmatrix} \left\{\frac{w_B}{w_i} - a_{Bi}\right\} \\ \left\{\frac{w_i}{w_W} - a_{iW}\right\} \end{bmatrix}. \quad (69)$$

$\widehat{\xi}$ is the Weight-Ratio-To-Edge-Error Matrix (WRTEEM) of size $2 \times n$. The first row is a vector of the best-over-all edge errors denoted by ξ_B and the second row is a vector of the all-over-worst edge errors denoted by ξ_W.

$$\overline{\xi} = \max\left(\left|\widehat{\xi}\right|\right) = \max\left[\left|\begin{matrix} \xi_B \\ \xi_W \end{matrix}\right|\right] = \max\begin{bmatrix} \left|\frac{w_B}{w_i} - a_{Bi}\right| \\ \left|\frac{w_i}{w_W} - a_{iW}\right| \end{bmatrix}. \quad (70)$$

To verify weights produced by the optimization model, the following equality is used to measure whether the weights lead to ξ^*.

$$\xi^* = \overline{\xi} = \max_i\left\{\left|\frac{w_B}{w_i} - a_{Bj}\right|\left|\frac{w_i}{w_W} - a_{jW}\right|\right\}. \quad (71)$$

9 Simulations and discussions

To facilitate the demonstration, analysis and discussion, the following examples are set by arranging or defining $B = 1$ for the best criterion, $W = 2$ for the worst criterion, the indices of other criteria are $3, \ldots, n$. As mentioned earlier, A_W in Eq. (2) is revised and defined without transposition, i.e. T, in this paper, due to redundancy or confusion for calculation. The R programming language is used for the implementation in this study. The NLOPT_GN_ISRES algorithm of the nloptr package (Johnson 2023) is used as the optimization solver after testing a broad range of optimization algorithms behind, but beyond discussion in this paper. The hardware used is the basic model, OMEN by Laptop 15 with Intel i7-7700HQ CPU and 16 GB RAM, and hence, the simulations of this paper are reproducible without high-end hardware requirement. The simulation dataset based on R and a program of MS Excel version for six numerical examples during the current study are available in the supplementary files and more files may be further included and updated in Yuen (2023).

9.1 Numerical examples

Six examples are used to demonstrate the usability of Algorithms 1 and 2 and Theorems 1–4. For the implementation using an optimization solver, Model (3), instead of Model (4), is used due to simplicity and fewer constraints to be implemented. To increase the precision for Model (3), the maximum number of function evaluations is set to 150,000. If the maximum number is not large enough, e.g. just a few thousands, the objective values may not research to the exact or ideal optimal solution. For Examples 2–4 and 6, the numerical approximate solution of Model (3) is almost the same as the closed-form solution of Theorem 4 or Algorithm 1 with the precision of at least 8 decimal places. The Excel file for the calculation of the following examples is available in Supplementary 1.

Table 1 *CI* values for $a_{BW} \in [1, \ldots, 12]$

a_{BW}	2	3	4	5	6	7	8	9	10	11	12
CI	0.4384	1	1.6277	2.2984	3	3.7251	4.4689	5.228	6	6.7830	7.5756

Example 1. To compute the Consistency Index of $a_{BW} = 8$, substitute $a_{BW} = 8$ to Eq. (42) of Theorem 3 to get the *CI* value below.

$$CI_8 = \frac{1}{2}\left((1 + 2 \times 8) - \sqrt{8 \times 8 + 1}\right) = 4.4689.$$

Similarly, *CI* values of four significant decimal digits for $a_{BW} \in [2, \ldots, 12]$ are shown in Table 1. The partial results are the same to the presentations rounded to two significant decimal digits for $a_{BW} \in [2, \ldots, 9]$ shown in Rezaei (2015, 2016).

Example 2. Given $A_B = (1, 8, 3)$ and $A_W = (8, 1, 2)$. By Algorithm 1, $p = 3 - 2 = 1$ and $3 \times 2 - 8 < 0$, the MEE and CR are computed as below:

$$\begin{aligned}\xi^* &= \xi_1 \\ &= \frac{1}{2}\left(-(1 + 3 + 2) + \sqrt{(1 + 3 + 2)^2 - 4(3 \times 2 - 8)}\right) \\ &= 0.3166;\end{aligned}$$

$$CR = \frac{0.3166}{4.4689} = 0.071.$$

For another pair, given $A_B = (1, 8, 5)$ and $A_W = (8, 1, 4)$. As $5 \times 4 - 8 > 0$,

$$\begin{aligned}\xi^* &= \xi_1 \\ &= \frac{1}{2}\left((1 + 5 + 4) - \sqrt{(1 + 5 + 4)^2 - 4(5 \times 4 - 8)}\right) \\ &= 1.3944;\end{aligned}$$

$$CR = \frac{1.3944}{4.4689} = 0.312.$$

If the rating scores at the same location of the other criterion of A_B and A_W are exchanged, i.e. $A_B = (1, 8, 4)$ and $A_W = (8, 1, 5)$, MEE is also the same as above, as the equation explains this scenario.

Example 3. Given $A_B = (1, 8, 2, 2)$ and $A_W = (8, 1, 2, 3)$. By Algorithm 1, $p = 4 - 2 = 2$. By removing elements in criteria B and W, $A\prime_B = (2, 2)$ and $A\prime_W = (2, 3)$. $D = \left(\frac{2\times2-8}{2+2-1}, \frac{2\times3-8}{2+3-1}\right) = \left(-\frac{4}{3}, -\frac{1}{2}\right)$. Since $min(D) \le max(D) \le 0$, Case 2 in Algorithm 1 is applied. As $S^- = \text{argmin}(D) = \text{argmin}\left(-\frac{4}{3}, -\frac{1}{2}\right) = \{1\}$ and$2 \times 2 - 8 < 0$, the MEE is

$$\begin{aligned}\xi^* &= \max_{i\in\{1\}}(\xi_i) \\ &= \max\left(\frac{1}{2}\left(-(1 + 2 + 2) + \sqrt{(1 + 2 + 2)^2 - 4(2 \times 2 - 8)}\right)\right) \\ &= 0.7016.\end{aligned}$$

For another pair, given $A_B = (1, 8, 3, 3, 5)$ and $A_W = (8, 1, 3, 5, 4)$. As $A'_B = (3, 3, 5)$ and $A\prime_W = (3, 5, 4)$, $D = (0.2, 1, 1.5)$. Since $\max(D) \ge \min(D) \ge 0$, Case 1 is applied. As $S^+ = \text{argmax}(D) = \{3\}$ and $5 \times 4 - 8 > 0$, the MEE is

$$\begin{aligned}\xi^* &= \min_{i\in\{3\}}(\xi_i) \\ &= max\left(\frac{1}{2}\left((1 + 5 + 4) - \sqrt{(1 + 5 + 4)^2 - 4(5 \times 4 - 8)}\right)\right) \\ &= 1.3944.\end{aligned}$$

If more other criteria are included and evaluated, e.g. $A_B = (1, 8, 3, 3, 2, 3, 5)$ and $A_W = (8, 1, 3, 5, 4, 3, 4)$ such that $\min(D) \ge 0$ and $\max(D) = 1.5$ at $(5, 4)$, ξ^* is still 1.3944, without any changes. Similarly, if $A_B = (1, 8, 2, 2, 1, 6)$ and $A_W = (8, 1, 2, 3, 5, 1)$ such that $\max(D) \le 0$ and $\min(D) = -\frac{4}{3}$ at $(2, 2)$, ξ^* is still 0.7016 and not changed. It is concluded that the number or size of criteria is not related to MEE for Cases 1 and 2.

Example 4. Given $A_B = (1, 8, 2, 2)$ and $A_W = (8, 1, 2, 7)$. By Algorithm 1, as $A'_B = (2, 2)$ and $A'_W = (2, 7)$, $D = (-1.33, 0.75)$, $\min(D) = -1.33 < 0$ and $\max(D) = 0.75 > 0$, Case 3 is met. As $S^+ = 2$ and $S^- = 1$,

$$\begin{aligned}\xi^* &= \max(\{\xi_1, \xi_2\}, \{\xi_{12}\}) \\ &= \max(\{0.6411, 0.7516\}, \{0.7692\}) = 0.7692.\end{aligned}$$

Now the third elements in A_B and A_W are changed to 1 and 4, but their multiplication is still 4, i.e. $A_B = (1, 8, 1, 2)$ and $A_W = (8, 1, 4, 7)$. However, the new determinant value at (1,4) is -1. The new MEE is

Table 2 The edge error matrix for $A_B = (1, 8, 2, 2)$ and $A_W = (8, 1, 2, 7)$

	c_1	c_2	c_3	c_4
c_1	0	0	0.3077	0.3333
c_2	0	0	0.3077	0.3333
c_3	0.3077	0.3077	0.7016	0.7692
c_4	0.3333	0.3333	0.7692	0.6411

Table 3 The edge error matrix for $A_B = (1, 8, 1, 2)$ and $A_W = (8, 1, 4, 7)$

	c_1	c_2	c_3	c_4
c_1	0	0	0.2857	0.3333
c_2	0	0	0.2857	0.3333
c_3	0.2857	0.2857	0.6056	0.7143
c_4	0.3333	0.3333	0.7143	0.6411

Table 4 The edge error matrix for $A_B = (1, 8, 2, 1, 2, 7)$ and $A_W = (8, 1, 2, 4, 7, 2)$

	c_1	c_2	c_3	c_4	c_5	c_6
c_1	0	0	0.3077	0.2857	0.3333	0.3333
c_2	0	0	0.3077	0.2857	0.3333	0.3333
c_3	0.3077	0.3077	0.7016	0	0.7692	0.7692
c_4	0.2857	0.2857	0	0.6056	0.7143	0.7143
c_5	0.3333	0.3333	0.7692	0.7143	0.6411	0
c_6	0.3333	0.3333	0.7692	0.7143	0	0.6411

Table 5 Multiple weight solutions with respect to the same ξ^* for Example 5

Model	Test case	w_1	w_2	w_3	w_4	ξ^*
(3)	1	0.5234	0.0717	0.1937	0.2112	0.7016
(3)	2	0.5154	0.0706	0.1908	0.2231	0.7016
(3)	3	0.5282	0.0724	0.1955	0.2040	0.7016
(63)	1	0.5293	0.0725	0.1959	0.2023	0.7016
(63)	2	0.5061	0.0693	0.1873	0.2373	0.7016
(63)	3	0.5228	0.0716	0.1935	0.2121	0.7016
(67)	1	0.5289	0.0725	0.1958	0.2029	0.7016
(67)	2	0.5321	0.0729	0.1970	0.1980	0.7016
(67)	3	0.5483	0.0751	0.2030	0.1736	0.7016
(6)	1–3	0.4667	0.0667	0.2	0.2667	1

$$\xi^* = \max(\{\xi_1, \xi_2\}, \{\xi_{12}\}) \\ = \max(\{0.6411, 0.6056\}, \{0.7143\}) = 0.7143.$$

If the pair is mixed with more numbers, e.g. $A_B = (1, 8, 2, 1, 2, 7)$ and $A_W = (8, 1, 2, 4, 7, 2)$. By Algorithm 1, as $A'_B = (2, 1, 2, 7)$ and $A'_W = (2, 4, 7, 2)$, $D = (-1.33, -1, 0.75, 0.75)$, $\min(D) = -1.33 < 0$ and $\max(D) = 0.75 > 0$, Case 3 is satisfied. As $S^+ = \{3, 4\}$ and $S^- = \{1\}$,

$$\xi^* = \max(\{0.6411, .6411, 0.7516\}, \{0.7692, 0.7692\}) \\ = 0.7692.$$

Table 6 Weight-ratio-to-edge-error matrices for the cases in Table 5 for Example 5

Model	Test Case	$\widehat{\xi}$	$i = 1$	$i = 2$	$i = 3$	$i = 4$
(3)	1	ξ_B	0	−0.7016	0.7016	0.4787
(3)	1	ξ_W	−0.7016	0	0.7016	−0.0555
(3)	2	ξ_B	0	−0.7016	0.7016	0.3100
(3)	2	ξ_W	−0.7016	0	0.7016	0.1595
(3)	3	ξ_B	0	−0.7016	0.7016	0.5895
(3)	3	ξ_W	−0.7016	0	0.7016	−0.1815
(61)	1	ξ_B	0	−0.7016	0.7016	0.6161
(61)	1	ξ_W	−0.7016	0	0.7016	−0.2101
(61)	2	ξ_B	0	−0.7016	0.7016	0.1330
(61)	2	ξ_W	−0.7016	0	0.7016	0.4216
(61)	3	ξ_B	0	−0.7016	0.7016	0.4648
(61)	3	ξ_W	−0.7016	0	0.7016	−0.0390
(65)	1–3	ξ_B	0	−0.7016	0.7016	−0.3935
(65)	1	ξ_W	−0.7016	0	0.7016	−0.1999
(65)	2	ξ_B	0	−0.7016	0.7016	−0.3128
(65)	2	ξ_W	−0.7016	0	0.7016	−0.2840
(65)	3	ξ_B	0	−0.7016	0.7016	0.1583
(65)	3	ξ_W	−0.7016	0	0.7016	−0.6891
(6)	1–3	ξ_B	0	−1	0.3333	−0.25
(6)	1–3	ξ_W	−1	0	1	1

The last MEE is also the same as the initial one. Although multiplications of the third elements of the first pair and second pair of vectors are the same, i.e. $2 \times 2 = 1 \times 4 = 4$, and the other elements remain the same, their ξ^* values are different. On the other hand, for the third pairs, swapping the number between the same positions of both vectors does not change the error results. For the details, the Edge Error Matrices for the above three paired vectors are shown in Tables 2, 3, and 4. The maximal values of the elements in the matrices are their corresponding MEE, respectively.

Example 5. Given $A_B = (1, 8, 2)$ and $A_W = (8, 1, 2)$. Models (3), (4) and (63)–(68) produce the same result, i.e. (0.663, 0.091, 0.246). If $n = 3$, the weights are unique due to the reason explained by applying Eq. (69) as below:

$$\widehat{\xi} = \begin{bmatrix} \xi_B \\ \xi_W \end{bmatrix} = \begin{bmatrix} 0 & -0.7016 & 0.7016 \\ -0.7016 & 0 & 0.7016 \end{bmatrix}.$$

The absolute weight-ratio-to-edge-error for each pair of different criteria is ξ^*. By solving the equations below, we can get the unique weights.

$$\frac{w_B}{w_2} - a_{B2} = \frac{w_1}{w_W} - a_{1W} = -0.7016,$$

$\frac{w_B}{w_3} - a_{B3} = \frac{w_3}{w_W} - a_{3W} = 0.7016.$

However, if the matrix is inconsistent and $n > 3$, multiple versions of weights with the same ξ^* are generated. For $A_B = (1, 8, 2, 2)$ and $A_W = (8, 1, 2, 3)$ in Example 3, the same model produces multiple solutions with respect to the same ξ^*, in which three different sample solution sets for each selected model are shown in Table 5. The reason can be explained by Eq. (69), in which the $\widehat{\xi}$ results are presented in Table 6. According to Example 3 using Algorithm 1, ξ^* is induced by the first other criterion, that is $i = 3$ shown in Table 6. However, there is some flexibility to choose the error for a criterion from the others, e.g. $i = 4$ in ξ_B and ξ_W. Another reason to produce the multiple solutions is the flexibility to change w_B and w_W which is shown in Eq. (46), (49), (52) and (55). For example, by applying Eq. (47) to the Test Case 1 in Models (3), (61) and (65) shown in Tables 4 and 5 with more significant digits for actual calculation, the results leading to the same value of ξ^* are shown as below:

$$\frac{0.5233927}{0.07171297} = \frac{0.5292615}{0.07251709} = \frac{0.5288694}{0.07246336} = (8 - 0.7015621) = 7.298438.$$

It is observed that ξ^* and weights produced in Model (3) or (4) do not have direct relationship to the ξ^* and weights produced in Model (5) or (6), which have the unique solution for their own objective value and weights. If ξ^* of Model (3) solved by a closed form is assigned to Model (67), which is transformed from Model (6), multiple weight solutions are also unavoidable.

If Model (6) is used, a unique weight set presented in Table 5 is produced no matter how many times are executed. If Eqs. (69) and (70) are applied to the unique weights, the results are presented in Table 6. The $\overline{\xi}$ or ξ^* derived from Model 6 is not the same as the ξ^* derived from Model (3). Therefore, objective value and weights derived from Model (5) or (6) have no direct observed relationship to Model (3) or (4). The consistency ratio from Model (3) or (4) is independent of the weights from Model (5) or (6).

Example 6. Refer to Fig. 1, $A_B = (1, 8, 1, 2, 6, 2, 2, 3, 2, 4)$ and$A_W = (8, 1, 4, 2, 1, 3, 6, 4, 8, 4)$. The edge error matrix is presented in Table 7. By Theorem 4, MEE is $\xi^* = \mathrm{Max}(\mathrm{EEM}) = 1$. By Algorithm 1, $A\prime_B = (1, 2, 6, 2, 2, 3, 2, 4)$ and$A\prime_W = (4, 2, 1, 3, 6, 4, 8, 4)$,$D = (-1, -1.33, -0.33, -0.5, 0.5714, 0.667, 0.8889, 1.1429)$. Case 3 is satisfied. As $S^+ = \{8\}$ and$S^- = \{2\}$, which are located at c_4 and c_{10}, $\xi^* = \max(\{0.7016, 1\}, \{1\}) = 1$.

9.2 Simulations and verifications

To further verify the reliability of closed-form solutions from the Max of Edge Error Matrix method (Theorem 4) and the Minmax Edge Error Determinant Method (Algorithm 1), especially to investigate if any missing cases are not uncovered for the algorithm design and implementation, the pseudo code of a test algorithm is proposed in Algorithm 3. Let ξ^{minMax}, ξ^{maxEEM} and ξ^{detMEE} be the MEE values obtained by Model (3) or (4), Eq. (43) of Theorem 4, and MEED method of Algorithm 1, respectively. In principle, the following relationship holds.

$$\mathrm{error} = \begin{cases} 0, \xi^{minMax} = \xi^{maxEEM} = \xi^{detMEE} \\ 1, \mathrm{Otherwise} \end{cases}. \tag{72}$$

In practice, the relationship above is implemented as three conditions stated in Step 2c of Algorithm 3. As the numerical solvers may only achieve a good enough solution that approximates to the exact solution, approximation error can be produced. In addition, if ξ^{maxEEM} or ξ^{detMEE} value is the unique and exact closed-form solution, the ξ^{minMax} value is not possible less than them, i.e. $\xi^{minMax} \geq \xi^{maxEEM} = \xi^{detMEE}$. However, as floating-point

Table 7 The Edge Error Matrix for $A_B = (1, 8, 1, 2, 6, 2, 2, 3, 2, 4)$ and $A_W = (8, 1, 4, 2, 1, 3, 6, 4, 8, 4)$ for Example 6

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}
c_1	0	0	0.2857	0.3077	0.1250	0.1429	0.2353	0.2500	0.4211	0.4706
c_2	0	0	0.2857	0.3077	0.1250	0.1429	0.2353	0.2500	0.4211	0.4706
c_3	0.2857	0.2857	0.6056	0	0.1667	0.2000	0.6154	0.6667	0.8000	0.9231
c_4	0.3077	0.3077	0	0.7016	0.1818	0.2222	0.6667	0.7273	0.8571	1
c_5	0.1250	0.1250	0.1667	0.1818	0.2426	0	0.4000	0.4286	0.5882	0.6667
c_6	0.1429	0.1429	0.2000	0.2222	0	0.3166	0.4615	0.5000	0.6667	0.7692
c_7	0.2353	0.2353	0.6154	0.6667	0.4000	0.4615	0.4689	0	0.2222	0.2500
c_8	0.2500	0.2500	0.6667	0.7273	0.4286	0.5000	0	0.5359	0.2353	0.2667
c_9	0.4211	0.4211	0.8000	0.8571	0.5882	0.6667	0.2222	0.2353	0.7830	0
c_{10}	0.4706	0.4706	0.9231	1	0.6667	0.7692	0.2500	0.2667	0	1

operations may induce rounded and truncated errors, a very small negative error for $\xi^{minMax} - \xi^{maxEEM}$ is still possible. Therefore, the conditions of Eq. (72) are changed and defined for simulations as below:

$$-10^{-6} \leq \left(\xi^{minMax} - \xi^{maxEMM}\right) \leq 10^{-4} \text{ and } \xi^{maxEEM} = \xi^{detMEE}. \quad (73)$$

If the difference between the numerical solution from the solver and the closed-form solution is more than 10^{-4} or less than -10^{-6}, the potential error from Theorem 4 or Algorithm 1 for ξ^{maxEEM} or ξ^{detMEE} will be reported by the algorithm. According to the simulations in this study, the error flag normally resulted from the optimizers being unable to find the optimal values. If the optimization reruns several times and/or enhance the research cost, the ξ^{maxEEM} finally reaches to the defined range.

To obtain the better accuracy of the optimization solver, firstly, the maximum number of function evaluations for the NLOPT_GN_ISRES solver algorithm of the nloptr package (Johnson 2023) to implement Algorithm 2 is set to 200,000 with the fractional tolerance of 10^{-10}, and secondly, the weights obtained by Algorithm 2 are used as the initial values for Model (3) implemented by the same optimization algorithm solver of the same settings for the function evaluations and fractional tolerance. If only Model (3) is implemented, an error from the solver could be more than 10^{-4}.

Algorithm 3 (Testing for MEE derived from solver, max(EEM) and detMEE method)

Input: N: Testing sample size; n^{max}: maximum number of criteria; a^{max}: maximum value for a_{BW},

Step 1: Initialize the values

times = 1 # case number or counter

error = 0 # error checking

flag = TRUE # flag to continue or exist the while loop

Step 2: Perform comparisons for N samples

While (flag):

Step 2a: Generate (A_B, A_W)

$a_{BW} = random(2, a^{max})$ # generate a random integer between 2 and a^{max} for a_{BW}

$n = random(3, n^{max})$ # generate a random integer between 3 and n^{max} for n

$(A_B, A_W) = generateBW(a_{BW}, n)$ # generate a pair of random vectors with a^{max} and n

Step 2b: Obtain MEE's with Model (3) with solver, Theorem 4 and Algorithm 1

$\xi^{minMax} = minMaxOpt(A_B, A_W)$ # Obtain MEE with solver for Model (3)

$\xi^{maxEEM} = maxEEM(A_B, A_W)$ # Obtain MEE with MEEM of Theorem 4

$the\xi^{detMEE} = detMEE(A_B, A_W)$ # Obtain MEE with determinant method of Algorithm 1

Step 2c: comparing ξ^{minMax}, ξ^{maxEEM}, and ξ^{detMEE}

Condition 1: check if the difference between ξ^{minMax} and ξ^{maxEMM} is very small

$$con1 = \begin{cases} 1, \left(\xi^{minMax} - \xi^{maxEEM}\right) \leq 10^{-4} \\ 0, otherwise \end{cases}$$

Condition 2: check if $\xi^{Minmax} \geq \xi^{maxEMM}$ within floating-point error allowance

$$con2 = \begin{cases} 1, \left(\xi^{minMax} - \xi^{maxEEM}\right) \geq -10^{-6} \\ 0, otherwise \end{cases}$$

Condition 3: check if Theorem 4 and Algorithm 1 produce the same results including floating-point error

$$con3 = \begin{cases} 1, \xi^{detMEE} = \xi^{maxEEM} \\ 0, otherwise \end{cases}$$

If all conditions above are met, no error is found

$$error = \begin{cases} 0, con1 \times con2 \times con3 = 1 \\ 1, otherwise \end{cases}$$

Step 3c: determine the flag for the while loop

$times + = 1$ # increment by 1

If N instances are performed or an error is found, exit the while loop. Otherwise, continue

$$flag = \begin{cases} FALSE, error \neq 0 or times > N \\ TRUE, otherwise \end{cases}$$

End While

If an error is found, return the last (A_B, A_W) inducing potential error for further study. Otherwise, no error is found, i.e. error = 0

Return: error, (A_B, A_W) if $error \neq 0$

Algorithm 3 presents the pseudo code for the core procedure. For implementation, the print functions, for example, are added in Algorithm 3 where appropriate for tracking, debugging and analysing. Two simulations have been performed. Both the maximum number of criteria, denoted by n^{max}, and the maximum value for a_{BW}, denoted by a^{max}, are set to 9. When $a_{BW} > 9$, Eq. (59) may not be suitable, which will be discussed in the future study. As the rating is on nine-point scale, Eq. (59) can be used.

For the first simulation, 20,000 random instances have been performed by taking 94,838 s (26.34 h). The file of the simulation data is available in Supplementary 2 or Yuen (2023). According to the simulation results, no error or unexpected case is found from all generated instances. It can be concluded that Theorem 4 or Algorithm 1 produces the exact closed-form solution for Model (3) or (4), and the results can be used to verify the solutions of the optimization solver.

The second simulation is to extensively test the efficiency of the proposed closed-form solutions and verify that the Theorem 4 and Algorithm 1 always produce the same MEE. A new algorithm (called Algorithm 4) is created by removing calculations of ξ^{minMax}, Conditions 1 and 2 from Algorithm 3, and with only using Condition 3 to check if any potential error occurs. One million random instances have been performed by taking only 4600 s (1.27 h). The file of the simulation data is available in Supplementary 3 or Yuen (2023). Again, no error is found from the one million random instances. For very rough estimation based on two simulations, the computation time of the proposed closed forms could be $((94838 - 4600/50)50 - (4600/2))/(4600/2) = 2058$ times faster than the Model (3) or (4) subject to implementation configuration for the solver. It can be concluded that it is quite computationally expensive for the solver to obtain better precision of MEE. The determination method of Algorithm 1 can be used to quickly determine the MEE with slightly less computational effort, whilst Theorem 4 can provide the details of EEM for reference.

10 Discussion and conclusions

The major contribution of this study is the provision of closed-form solutions for the exact Consistency Ratio in BWM Minmax Optimization Model, which currently must use optimization solvers such as Excel Solver in (https://bestworstmethod.com/software/) or MATLAB (Wu et al. 2022). CR is ratio of MEE (or ξ^*) to the Consistency Index. Whilst the CI is computed by Eq. (42), the MEE is obtained by the Max of Edge Error Matrix Method (Eq. (44) of Theorem 4) and the Minmax Edge Error Determinant Method (Algorithm 1). In Theorem 4, the upper triangular part of the edge error matrix is first computed, and the maximum value of the triangular matrix is the MEE. As EEM is the symmetric matrix, the max of either the upper or lower part is MEE. For the MEED of Algorithm 1, the determinant values are first computed to determine the location(s) and the case of the MEE, and then the appropriate function in the case is used to compute the MEE. If the best-over-worst score is more than nine, MEEM method should be used, as the constraint condition in Eq. (59) of MEED is indicated.

Several essential properties of the consistency ratio toward or extended from the closed-form solutions are discussed. Firstly, although Rezaei (2015, 2016) claimed that the comparisons ideally up to 9 criteria, according to Theorem 4, the inconsistency based on MEE is independent of number of criteria as MEE is determined by the worst case determined by the largest value of IEE (Theorem 1) or CEE (Theorem 2). The aggregate edge errors are not considered, as the optimal value is independent of the number of criteria but determined by the most discrepant scores for some criteria.

With the simple algebraic form of EEM, the individual values over the threshold defined by the decision maker can be identified. If the paired vectors are inconsistent, decision makers can find sufficient high values in EEM to identify the inconsistency locations and then revise the judgement ratings. The examples of EEM are shown in Tables 2, 3, 4, and 7.

As the exact MEE can be obtained by the algebraic forms, the optimization models can be revised with fewer variables and constraints, e.g. Algorithm 2 of Model (63) and its alternative Models (64)–(68). All objective values of the models must be one, except for Model (65), which is zero. If these conditions are satisfied, the weights generated from the models can preserve the MEE calculated by Eq. (70). However, the same MEE can lead to different versions of non-reproducible weights due to the reasons discussed in Sect. 8, and Example 5 also demonstrated these issues. According to the testing in this study, weights generated from Models (5) and (6) are more reliable than Models (3) and (4) since Models (5) and (6) produce the unique exact solution of weights. However, according to the closed-form solution formula, the weights generated from Models (5) and (6) are independent of or have no obvious relationship to the EEM generated from Models (3) and (4). Since the CR optimization model must calculate weights to obtain the MEE, the new methods do not need to compute weights to have the MEE. The significance of this paper is that all BWM applications can efficiently use the proposed closed forms for the exact CR values, instead of using an expensive optimization solver to find the approximate values with potential error.

Appendix: Summary of notations and abbreviations

a_{Bi}	A score of the best criterion over the criterion i
a_{BW}	The best-over-worst score
a_{iW}	A score of the criterion I over the worst criterion
A_B and A_W	A pair of best and worst vectors; paired vectors
$\ddot{\xi}$	Edge Error Matrix (EEM)
$\ddot{\xi}^+$	Upper triangular Edge Error Matrix
$\ddot{\xi}^-$	Lower triangular Edge Error Matrix
ξ_{ij}	Co-Edge Error (CEE)
ξ_i	Individual Edge Error (IEE) for c_j in comparisons with c_B and c_W
ξ^*	Minmax Edge Error (MEE) based on non-approximate solutions of Models (3) and (4)
$\hat{\xi}$	Weight-Ratio-To-Edge-Error Matrix (WRTEEM)

ξ_B	A vector of the best-over-all edge errors
ξ_W	A vector of the all-over-worst edge errors
$\overline{\xi}$	$\overline{\xi} = max\left(\left\|\widehat{\xi}\right\|\right)$
$CI_{a_{BW}}$	Consistency Index of a_{BW}
CR	Consistency Ratio
D_k	Determinant value for criteria k with respect to the best and worst criteria
B	Index/location of the Best Criterion
W	Index/location of the Worst Criterion
w_i	Weight of criterion i
MEEM	Max of Edge Error Matrix method, or MEE = max(EEM)
MEED	Minmax Edge Error Determinant method, or detMEE

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s41066-024-00459-5.

Acknowledgements Sincere thanks are extended to the editors and anonymous referees for their time and effort to improve the work.

Author contribution This research is solely conducted by K.K.F. Yuen.

Funding Open access funding provided by The Hong Kong Polytechnic University.

Data availability The data are available in the Supplementary and available in Yuen (2023). The first preprint of this work has been available at https://doi.org/10.2139/ssrn.4551188 since 24 Aug 2023.

Declarations

Conflict of interest The author states that there is no conflict of interest.

References

Ali A, Rashid T (2021) Best–worst method for robot selection. Soft Comput 25(1):563–583

Chen Z, Ming X (2020) A rough–fuzzy approach integrating best–worst method and data envelopment analysis to multi-criteria selection of smart product service module. Appl Soft Comput 94:106479

Chen Z-h, Wan S-p, Dong J-y (2023) An integrated interval-valued intuitionistic fuzzy technique for resumption risk assessment amid COVID-19 prevention. Inf Sci 619:695–721

Dong J-Y, Wan S-P (2024) Interval-valued intuitionistic fuzzy best-worst method with additive consistency. Expert Syst Appl 236:121213

Dong J, Wan S, Chen S-M (2021) Fuzzy best-worst method based on triangular fuzzy numbers for multi-criteria decision-making. Inf Sci 547:1080–1104

Faizi S, Sałabun W, Nawaz S, Rehman AU, Wątróbski J (2021) Best-Worst method and Hamacher aggregation operations for intuitionistic 2-tuple linguistic sets. Exp Syst Appl 181:115088

Faliszewski P, Hemaspaandra E, Hemaspaandra LA (2010) Using complexity to protect elections. Commun ACM 53(11):74–82

Guan C, Yuen KKF, Coenen F (2019) Particle swarm optimized density-based clustering and classification: supervised and unsupervised learning approaches. Swarm Evol Comput 44:876–896

Hafezalkotob A, Hafezalkotob A, Liao H, Herrera F (2020) Interval MULTIMOORA method integrating interval Borda Rule and interval best–worst-method-based weighting model: case study on hybrid vehicle engine selection. IEEE Trans Cybern 50(3):1157–1169

Jafarzadeh Ghoushchi S, Dorosti S, Khazaeili M, Mardani A (2021) Extended approach by using best–worst method on the basis of importance–necessity concept and its application. Appl Intell 51(11):8030–8044

Johnson SG (2023) The NLopt nonlinear-optimization package. https://nlopt.readthedocs.io/en/latest/. Accessed Dec 2023

Kheybari S, Ishizaka A (2022) The behavioural best-worst method. Expert Syst Appl 209:118265

Koczkodaj WW, Mikhailov L, Redlarski G, Soltys M, Szybowski J, Tamazian G et al (2016) Important facts and observations about pairwise comparisons. Fund Inform 144:1–17

Liang F, Brunelli M, Rezaei J (2020) Consistency issues in the best worst method: measurements and thresholds. Omega 96:102175

Liang F, Brunelli M, Rezaei J (2022) Best-worst tradeoff method. Inf Sci 610:957–976

Ma X, Qin J, Martínez L, Pedrycz W (2023) A linguistic information granulation model based on best-worst method in decision making problems. Inf Fusion 89:210–227

Malakoutikhah M, Alimohammadlou M, Jahangiri M, Rabiei H, Faghihi SA, Kamalinia M (2022) Modeling the factors affecting unsafe behaviors using the fuzzy best–worst method and fuzzy cognitive map. Appl Soft Comput 114:108119

Mazurek J, Perzina R, Ramík J, Bartl D (2021) A numerical comparison of the sensitivity of the geometric mean method, eigenvalue method, and best-worst method. Mathematics 9(5):554

Mi X, Tang M, Liao H, Shen W, Lev B (2019) The state-of-the-art survey on integrations and applications of the best worst method in decision making: why, what, what for and what's next? Omega 87:205–225

Oztas GZ, Erdem S (2021) Framework selection for developing optimization algorithms: assessing preferences by conjoint analysis and best–worst method. Soft Comput 25(5):3831–3848

Qin J, Ma X, Liang Y (2023) Building a consensus for the best-worst method in group decision-making with an optimal allocation of information granularity. Inf Sci 619:630–653

Rezaei J (2015) Best-worst multi-criteria decision-making method. Omega 53:49–57

Rezaei J (2016) Best-worst multi-criteria decision-making method: Some properties and a linear model. Omega 64:126–130

Saaty TL (1977) A scaling method for priorities in hierarchical structures. J Math Psychol 15(3):234–281
Saaty TL (1980) Analytic hierarchy process: planning, priority, setting, resource allocation. McGraw-Hill, New York
Saaty TL (1990) How to make a decision: the analytic hierarchy process. Eur J Oper Res 48(1):9–26
Thurstone L (1927) A law of comparative judgment. Psychol Rev 34(4):273–286
Wan S, Dong J (2022) A novel extension of best-worst method with intuitionistic fuzzy reference comparisons. IEEE Trans Fuzzy Syst 30(6):1698–1711
Wan S, Dong J, Chen S-M (2021) Fuzzy best-worst method based on generalized interval-valued trapezoidal fuzzy numbers for multi-criteria decision-making. Inf Sci 573:493–518
Wu S, He B, Liu X (2022) Improving the best-worst method based on optimal completion of incomplete pairwise comparison matrix. IEEE Access 10:127284–127296
Yucesan M, Gul M (2021) Failure prioritization and control using the neutrosophic best and worst method. Granul Comput 6(2):435–449
Yuen KKF (2010) Analytic hierarchy prioritization process in the AHP application development: a prioritization operator selection approach. Appl Soft Comput J 10(4):975–989
Yuen KKF (2012) Pairwise opposite matrix and its cognitive prioritization operators: comparisons with pairwise reciprocal matrix and analytic prioritization operators. J Oper Res Soc 63(3):322–338
Yuen KKF (2014) The primitive cognitive network process in healthcare and medical decision making: comparisons with the analytic hierarchy process. Appl Soft Comput 14 Part A:109–119
Yuen KKF (2022) Decision models for information systems planning using primitive cognitive network process: comparisons with analytic hierarchy process. Oper Res Int J 22(3):1759–1785
Yuen KKF (2009) Cognitive network process with fuzzy soft computing technique for collective decision aiding (Ph.D. thesis). The Hong Kong Polytechnic University, Ph.D. thesis
Yuen KKF (2023) 'Simulation Dataset for "Closed-form Solution of Consistency Ratio in Best Worst Method Minmax Optimization Model"'. https://github.com/kkfyuen/BWRC/tree/main/CR