

CEJOR (2015) 23:187–203
DOI 10.1007/s10100-013-0311-x

ORIGINAL PAPER

A heuristic rating estimation algorithm for the pairwise comparisons method

Konrad Kułakowski

Published online: 20 June 2013

© The Author(s) 2013. This article is published with open access at Springerlink.com

Abstract The pairwise comparisons method is a powerful tool used for establishing the relative order between different concepts in situations in which it is difficult (or sometimes even impossible) to provide explicit rating. Appropriate ratings are determined by solving eigenvalue problem for the pairwise comparisons matrix. This study presents a new iterative heuristic rating estimation algorithm that tries to deal with the situation when exact estimations for some concepts (stimulus) C_K are a priori known and fixed, whilst the estimates for the others (unknown concepts C_U) need to be computed. The relationship between the local estimation error, understood as the average absolute error $E(c)$ over all direct estimates for the concept $c \in C_U$ and the pairwise comparisons matrix inconsistency index is shown. The problem of convergence of subsequent intermediate results is discussed and the convergence conditions are given.

Keywords Decision analysis · Pairwise comparisons · Iterative algorithms · Data inconsistency

1 Introduction

The *pairwise comparisons (PC) method* was introduced in its early form by [Fechner \(1966\)](#), then it was popularized and developed by [Thurstone \(1994\)](#). The introduction of hierarchical structures by [Saaty \(2008\)](#) was another important contribution to the *PC* method, providing the methodology and practical ways to deal with the large amounts of criterion parameters. Initially the *PC* method was used in the scientific study of psychometrics and psychophysics ([Thurstone 1994](#)), however, it then came to be used in other areas of applications, such as complex decision theory ([Saaty 2008](#)),

K. Kułakowski (✉)

Department of Applied Computer Science, AGH University of Science and Technology,
Al. Mickiewicza 30, 30-059 Cracow, Poland
e-mail: konrad.kulakowski@agh.edu.pl

economics (Peterson and Brown 1998), voting systems (Tideman 1987) and others. Since the data, which are input to the *PC* method are the result of human judgment, it is very easy for inaccuracy to occur. Hence, the input data set is frequently ambiguous and does not allow users to draw firm conclusions. There are several indexes of the data inconsistency (Bozóki and Rapcsak 2008), including the best known *Saaty's eigenvector method* (Saaty 1980), *Least Squares Method*, *Chi Squares Method*, *Koczkodaj's distance based inconsistency index* (Koczkodaj 1993), and others. Using these indexes, the reliability of the data can be assessed, hence, the confidence in the result can be evaluated. The answer to the question of how much input data must be consistent to ensure the result reliability is the subject of empirical research. For instance, according to *Saaty's* recommendation every occurrence of the consistency ratio greater than or equal to 0.1 should be the subject of re-examination of the pairwise judgments until the inconsistency becomes less than or equal to the desired value (Bozóki and Rapcsak 2008; Triantaphyllou et al. 1990). The main criticism of this approach relates to its separation from the data and lack of localizing the most problematic matrix elements (Bozóki and Rapcsak 2008; Koczkodaj 1993; Triantaphyllou et al. 1990). In contrast, *Koczkodaj's* inconsistency index has a meaningful interpretation and provides information about the inconsistency location, but it does not provide an exact answer to the question of how good the average data sample is. Inconsistency identified as too high, must be reduced to an acceptable level (ideally to zero). Since the ratio coefficients, which are the input to the *PC* method, frequently represent experts' judgements, thus a natural way of inconsistency reduction is to call the expert panel once again and ask the professionals gathered to agree on the opinion (Gomes 1993). Because such a solution is usually time-consuming and expensive, heuristic algorithms of inconsistency reduction have been proposed (Koczkodaj and Orłowski 1999; Koczkodaj and Szarek 2010; Gomes 1993; Xu and Wei 1999; Bozóki 2008; Temesi 2006; Cao et al. 2008). The result of these algorithms is a new set of data, which preserves most of the decision maker's original judgment structure and significantly reduces the data inconsistency.

The proposed innovative solution approaches the problem differently. It does not attempt to minimize inconsistency in the data, but rather proposes a way of using the data, which takes into account their inconsistency. Hence, knowing the exact values for a few concepts¹ and some inconsistent set of ratios between them, the data analyst is able to estimate values of all other concepts with errors depending on the degree of data inconsistency. The presented approach is comparable to the inconsistency reduction methods mentioned above, since the set of concepts for which the estimates are known can be easily transformed into a consistent set of data (as addressed in Sect. 8).

The first and second sections of the article focus on the presentation of the necessary facts and definitions concerning the pairwise comparisons methods. Section three formulates formally the problem considered in the work. The fourth and fifth, sections provide the *Heuristic Rating Estimation (HRE)* algorithm together with a numerical example demonstrating the algorithm application in practice. The sixth section shows the relationship between errors of estimations obtained by using the *HRE* algorithm and the data inconsistency. The next, seventh section addresses the case of which

¹ Thurstone (1994) originally called them stimulus.

the subsequent estimation sets converge to some limit. Finally, the last two sections contain the closing discussion and summary.

2 A pairwise comparisons method

A crucial part of the *PC* method is $M = (m_{ij}) \wedge m_{i,j} \in \mathbb{R}_+ \wedge i, j \in \{1, \dots, n\}$ a *PC* matrix that expresses some quantitative relation R over the finite set of concepts $C \stackrel{df}{=} \{c_i \in \mathcal{C} \wedge i \in \{1, \dots, n\}\}$ where \mathcal{C} is a non empty universe of concepts and $R(c_i, c_j) = m_{ij}$, $R(c_j, c_i) = m_{ji}$. Traditionally, these concepts are interpreted as subjective stimuli (Thurstone 1994), whilst the values m_{ij} and m_{ji} are considered as the relative importance indicators (stimulus intensities), so that according to the best knowledge of an expert the significance of c_i equals $m_{ij}c_j$.

Definition 1 A matrix M is said to be reciprocal if $\forall i, j \in \{1, \dots, n\} : m_{ij} = \frac{1}{m_{ji}}$, and M is said to be consistent if $\forall i, j, k \in \{1, \dots, n\} : m_{ij} \cdot m_{jk} \cdot m_{ki} = 1$.

Since the knowledge stored in the *PC* matrix usually comes from different professionals in the field of relation R , it often results in inaccuracy. In such a case, reasoning using the data gathered in M may give ambiguous results. This observation gave rise to the research on the concept of data consistency. In the ideal case, M is consistent, and there is no doubt as regards the value assigned to the concept c_j if the value assigned to c_i and m_{ji} is known. Unfortunately, in practice the knowledge in M is inconsistent, and professionals using the *PC* method have to deal with this incoherence. Thus, it is important to answer the question of how inconsistent the knowledge in the *PC* matrix is. There are a number of inconsistency indexes, including *Eigenvecor Method* (Saaty 2008), *Least Squares Method*, *Chi Squares Method* (Bozóki and Rapcsak 2008), *Koczkodaj’s distance based inconsistency index* (Koczkodaj 1993) and others. For the purpose of this article, the Koczkodaj’s distance based inconsistency index has been adopted since it is the only localizing index amongst the above mentioned indexes.

Definition 2 Koczkodaj’s distance based inconsistency index \mathcal{K} of $n \times n$ and $(n > 2)$ reciprocal matrix M is equal to

$$\mathcal{K}(M) = \max \left\{ \min \left\{ \left| 1 - \frac{m_{ij}}{m_{ik}m_{kj}} \right|, \left| 1 - \frac{m_{ik}m_{kj}}{m_{ij}} \right| \right\} \right\} \tag{1}$$

where $i, j, k = 1, \dots, n$ and $i \neq j \wedge j \neq k \wedge i \neq k$.

Intuitively speaking, since in an “ideal” matrix $\forall i, j, k \in \{1, \dots, n\} : m_{ij} \cdot m_{jk} \cdot m_{ki} = 1$ the *Koczkodaj’s* index localizes the worse triad (by the Euclidean distance) from this ideal.

PC matrices may be over-complete, i.e. there is more than one estimation describing one and the same relation between v_i and v_j , but they can be also incomplete, i.e. not all values m_{ji} are defined. While the first situation can be addressed in many ways, for instance the estimates related to the same pair of concepts can be averaged, the other is not desirable. It indicates that the model is lacking of knowledge, although

the missing estimates can be somehow compensated (for example, by employing the properties of reciprocity, consistency or transitivity) (Koczkodaj et al. 1999). Hence, for further consideration, it will be assumed that the $n \times n$ PC matrix M is complete in the sense specified below.

Definition 3 A matrix M ($n \times n$) is said to be complete if $\forall i, j = 1, \dots, n : m_{ij}$ is defined.

Due to the further consideration it is useful to define a graph structure over the set C and matrix M .

Definition 4 Let a pairwise comparisons graph $G = (C, E, M)$ be a weighted directed graph over the matrix M , where

- C , is a set of vertices
- $E \subseteq C \times C$ so that $(v_i, v_j) = e \in E \Leftrightarrow m_{ji}$ exists, is a set of edges
- $M : E \rightarrow \mathbb{R}$ so that $M(e) \stackrel{df}{=} m_{ji} \wedge e = (v_i, v_j)$ is a function of experts' assessments

Wherever it does not raise doubts, instead of $M(e) \wedge e = (u, v)$ the function M will be written with two arguments i.e. $M(u, v)$.

3 Problem formulation

Since the concepts are linked to each other by a quantitative relation R , then assuming that the exact values of some concepts are known, the values of the others should be proposed. Thus, let $\mu : C \rightarrow \mathbb{R}_+$ be a partial function that assigns to some concepts from $C \subset \mathcal{C}$ positive values from \mathbb{R}_+ . Hence, the concepts for which the actual value μ is known are denoted by $C_K \subset C$ and called known concepts, whilst concepts for which μ need to be determined are denoted $C_U = C \setminus C_K$ and called unknown concepts. The relation between different concepts in terms of the function μ is represented in the form of the PC matrix M , so that $m_{ji}\mu(v_i) = \mu(v_j)$. Since m_{ji} usually aims to express how much greater or smaller v_i is than v_j with respect to μ by convention it is assumed that elements of the $n \times n$ PC matrix are positive real numbers, i.e. $m_{ji} \in \mathbb{R}_+$ where $i, j \in 1, \dots, n$.

The presented method aims to provide an iterative heuristic estimation algorithm that for all $v \in C_U$ proposes the appropriate value of $\mu(v)$. In this approach $m_{ji}\mu(v_i)$ is treated as a sample of $\mu(v_j)$, hence the expected value of $\mu(v_j)$ is the arithmetic mean of values $m_{ji}\mu(v_i)$. The algorithm is iterative and sets the new expected values based on the ones previously determined. It stops either when it reaches a fixed number of iterations or (if convergent) when calculations reach the desired accuracy. Although during the course of the algorithm the new values for $\mu(v)$ and $v \in C_U$ are calculated, M remains unchanged and serves as reference data.

4 Heuristic rating estimation algorithm

The principle of operation of the rating estimation algorithm (Listing 1) is to iteratively assign the value $\mu(u)$ to every unknown vertex $u \in C_U$ by calculating the mean of

```

1 HRE( $G, \epsilon, level$ )
2  $L \leftarrow \{v : (u, v) \in E \wedge u \in C_K\} \cap C_U$ 
3  $l \leftarrow 0$ 
4  $tmp \leftarrow \emptyset$ 
5  $Est \leftarrow \emptyset$ 
6 while  $stop(\mu, \mu_{old}, \epsilon, l, level)$  do
7    $\mu_{old} \leftarrow \mu$ 
8   while  $L \neq \emptyset$  do
9      $u \leftarrow poll(L)$ 
10     $T \leftarrow \{e = (v, u) \in E \mid \mu(v) \text{ is defined}\}$ 
11     $tmp(u) \leftarrow \frac{1}{\#T} \sum_{(v,u) \in T} \mu(v) \cdot M(v, u)$ 
12  end while
13  foreach  $u \in Dom(tmp)$ 
14     $\mu(u) \leftarrow tmp(u)$ 
15    foreach  $v \in \{v : (u, v) \in E\} \cap C_U$ 
16       $L \leftarrow L \cup \{v\}$ 
17     $Est \leftarrow Est \cup \{\mu(u) : u \in C_U\}$ 
18     $l \leftarrow l + 1$ 
19     $tmp \leftarrow \emptyset$ 
20  end while
21  return  $chose\_optimal\_est(Est)$ 

```

Listing 1: Heuristic ratio estimation algorithm

its samples, and then choosing from among all the calculated estimations the one which is optimal. The idea of the procedure comes from the BFS algorithm (Cormen et al. 2009), which, layer by layer, traverses the graph of interest. In the presented approach, the estimates for the next layer are computed on the base of the previous layer, assuming that vertex repetition in different steps is allowed. The algorithm stops when either the appropriate layer is reached or if the algorithm converges for the given G , then it stops when the distance in the sense of the chosen metrics (see Eq. 47) between the subsequent estimates for elements in C_U is smaller than some desired ϵ . If the algorithm does not converge (when G is fixed) then the number of steps that need to be made is explicitly set at the beginning of the estimation procedure. In the case of standard complete *PC matrices* the graph is a directed clique, where every single vertex is connected to each other. In such a case it appears that even in the first step all the vertices are visited, whilst in the second step the computed estimations may take into account all possible values² gathered in M except those describing ratios between elements in C_K . Thus, when the algorithm is not convergent, a few iterations of the *HRE* procedure seem to be a reasonable choice.

The main principle of the algorithm (Listing 1) seems to be quite straightforward. It starts from assigning followers (in sense E) of all elements of C_K to the set L (Listing: 1, line: 2). For each u , for which $\mu(u)$ is unknown, all its predecessors in E are scanned. If $\mu(v)$ is already known then v becomes an element taken into account during the mean computation (Listing: 1, line: 10). Then, within the two loops *while* the current

² Following the reciprocity principle all the M values in the form $M(c_i, c_j)$ where $c_i \in C_U \wedge c_j \in C_K$ can be used to create expressions $\frac{\mu(c_j)}{M(c_i, c_j)}$ are treated as just one specimen of $\mu(c_i)$.

level is traversed (the outer loop Listing: 1, lines: 6–20), and appropriate estimates are computed (the inner loop Listing: 1, lines: 8–12). The predicate *stop* becomes false when either the auxiliary variable *l* reaches the assumed number of levels or it holds that $l > 1$ and $\rho(x_l, x_{l+1}) < \epsilon$, where $x_l = (\mu_{old}(c_{u_1}), \dots, \mu_{old}(c_{u_k}))$ and $x_{l+1} = (\mu(c_{u_1}), \dots, \mu(c_{u_k}))$ where the concepts $c_{u_i} \in C_U$ and ρ is one of the metrics defined in (Eq. 47). Otherwise it is true, and the outer loop continues traversing *G*. As long as *L* is not empty the inner loop proceeds as follows: removes one element from the set *L* and assigns it to the variable *u* (Listing: 1, line: 9), forms the auxiliary set *T* containing input edges for *u* so that their beginnings are predecessors of *u* in *E* for which μ is already known, and for all $u \in L$ adds to the auxiliary mapping $tmp : C_U \rightarrow \mathbb{R}_+$ a pair $(u, \mu(v)M(v, u))$ (Listing: 1, line: 11). When all the elements of *L* are processed then the inner loop ends. Next the outer loop rewrites the auxiliary mapping *tmp* to the result map μ (Listing: 1, lines: 13–14). Since for all the concepts that were previously in *L* the function μ is known, the next level is devoted to μ calculation for their followers in *L*, hence the set *L* is filled back (Listing: 1, lines: 15–16). At the end of the outer loop the step counter is incremented and the auxiliary mapping *tmp* is emptied. When the outer loop completes its operation, the set *Est* contains a sequence of subsequent sets of estimates for concepts from C_U . Then, at the end of the procedure (Listing: 1, line: 21) an optimal set of estimates needs to be chosen. For the purposes of this work it was assumed that the optimal set of estimates is one for which the average of the absolute mean estimation errors is minimal (Eq. 4). Let us define the absolute mean error $e(u)$ as an error indicator for some concept *u* and mapping μ as follows:

$$e_\mu(u) = \frac{1}{n} \sum_{i=0}^n |\mu(u) - \mu(v_i) \cdot M(v_i, u)| \tag{2}$$

where $u \in C_U, (v_i, u) \in E$ and $\mu(v_i)$ is already defined. Then the average error for all unknown concepts with respect to μ is defined as:

$$e_\mu(C_U) = \frac{1}{\#C_U} \sum_{c \in C_U} e_\mu(c) \tag{3}$$

hence

$$\text{chose_optimal_est}(Est) = \min_{\mu \in Est} \{e_\mu(C_U)\} \tag{4}$$

In every iteration of the algorithm every weight $M(v, u)$ (Listing: 1, line: 11) is considered exactly once. Hence, the running time of the outer loop is $O(r \cdot |E|) = O(r \cdot n^2)$ where *r* is the number of iterations, and *n* is the size of *C*. Similarly, the calculation of the average error (Eq. 3) requires the consideration of every $M(v, u)$ exactly once, hence computing formulae (Eq. 4) also requires at most $r \cdot n^2$ steps. Thus, the overall running time of the *HRE* algorithm is also $O(r \cdot n^2)$.

5 Numerical example

Let us illustrate the algorithm defined above by a simple numerical example. Some state agency supports five innovative projects u_1, \dots, u_5 . After a while the two of them u_4

and u_5 come to an end and their actual cost becomes known $\mu(u_4) = 6, \mu(u_5) = 2$. As both projects exceeded the initial budget, the agency wants to reestimate the expected costs of the other projects using the already acquired knowledge. For this purpose the agency organizes a panel of experts under which a PC matrix M is formed (Eq. 5) reflecting the predicted cost relations between all the five projects.

$$M = \begin{bmatrix} 1 & \frac{3}{5} & \frac{3}{4} & \frac{1}{2} & \frac{4}{3} \\ \frac{5}{4} & 1 & \frac{5}{4} & \frac{11}{12} & \frac{5}{2} \\ \frac{6}{5} & \frac{4}{5} & 1 & \frac{1}{2} & 2 \\ \frac{3}{2} & \frac{6}{5} & \frac{6}{5} & 1 & 3 \\ \frac{2}{3} & \frac{3}{5} & \frac{4}{7} & \frac{1}{3} & 1 \end{bmatrix} \tag{5}$$

Since experts were assigned to the pairs (u_i, u_j) randomly and did not know each other’s estimates, the matrix M is neither reciprocal nor consistent. Based on this data the agency must calculate adequate estimates of the projects u_1, u_2 and u_3 and take appropriate decisions regarding the future of these projects. The reasoning presented below shows how to calculate the estimated cost of the projects using a HRE procedure (Listing: 1).

The matrix M is used for the generation graph G , the set of known concepts C_K is formed by the projects u_4 and u_5 , and finally the unknown concept set is $C_U = \{u_1, u_2, u_3\}$. Since all concepts are reachable in the first step, and during the second step the μ mapping is computed for every $c \in C_U$ using all other concepts in C , then for the practical demonstration of the algorithm the number of levels traversed is limited to 2. In fact, for the matrix M as given (Eq. 5) the subsequent estimations calculated by HRE converge. Thus, instead of limiting the number of steps, an appropriately small ϵ can be chosen. This case will be discussed later, after the convergence criterion for HRE is defined.

Let us assume that the first vertex considered during the first step of the algorithm is u_1 . Then, according to the presented procedure $\mu(u_1)$ is computed as follows:

$$\mu(u_1) = \frac{1}{2} \cdot (\mu(u_4) \cdot M(u_4, u_1) + \mu(u_5) \cdot M(u_5, u_1)) \tag{6}$$

hence,

$$\mu(u_1) = \frac{1}{2} \cdot \left(6 \cdot \frac{1}{2} + 2 \cdot \frac{4}{3} \right) = \frac{17}{6} \approx 2.83 \tag{7}$$

and further, in the same way:

$$\mu(u_2) = \frac{1}{2} \cdot \left(6 \cdot \frac{11}{12} + 2 \cdot \frac{5}{2} \right) = \frac{21}{4} = 5.25 \tag{8}$$

$$\mu(u_3) = \frac{1}{2} \cdot \left(6 \cdot \frac{1}{2} + 2 \cdot 2 \right) = \frac{7}{2} = 3.5 \tag{9}$$

The first turn of the outer loop (traversing the first level of G) brings estimations for all vertices in C_U . Thus, although the estimation process can be terminated, the agency concludes that it is better to take into account more data (for instance, due to the excluding of accidental errors of some experts), then decides to perform the second iteration of the algorithm. The new subsequent values $\mu(u_1)$, $\mu(u_2)$ and $\mu(u_3)$ are calculated below.

$$\mu(u_1) = \frac{1}{4} \cdot \left(\frac{21}{4} \cdot \frac{3}{5} + \frac{7}{2} \cdot \frac{3}{4} + 6 \cdot \frac{1}{2} \cdot \frac{1}{4} + 2 \cdot \frac{4}{3} \right) = \frac{1373}{480} \approx 2.86 \quad (10)$$

$$\mu(u_2) = \frac{1}{4} \cdot \left(\frac{17}{6} \cdot \frac{5}{4} + \frac{7}{2} \cdot \frac{5}{4} + 6 \cdot \frac{11}{12} + 2 \cdot \frac{5}{2} \right) = \frac{221}{48} \approx 4.6 \quad (11)$$

$$\mu(u_3) = \frac{1}{4} \cdot \left(\frac{17}{6} \cdot \frac{6}{5} + \frac{21}{4} \cdot \frac{4}{5} + 6 \cdot \frac{1}{2} + 2 \cdot 2 \right) = \frac{73}{20} \approx 3.65 \quad (12)$$

Recognizing the achieved result as optimal, the agency finishes the algorithm. It is worth noting that in the matrix M (Eq. 5) the two estimates do not come from the experts and have been introduced just for the matrix completeness. Namely, since the values $\mu(v_4)$ and $\mu(v_5)$ were previously known, the values $m_{4,5}$ and $m_{5,4}$ were calculated as ratios $\mu(v_4)/\mu(v_5)$ and $\mu(v_5)/\mu(v_4)$ respectively.

6 Data consistency and estimation error

For any algorithm, the result of which being inaccurate, a natural questions concern the accuracy of the resulting solution, and what should be done to improve the accuracy. The presented procedure is based on the fundamental sample mean estimation equation (Walpole 2012), where for the purpose of the algorithm the product $\mu(u_i) \cdot M(u_i, u_j)$ is treated as a sample, whilst $\mu(u_j)$ denotes the expected value inferred from samples. In this case the natural measure of the algorithm output accuracy is an estimation error understood as the distance between sample and the mean. Of course, the smaller the error, the more accurate the result. According to the popular adage ‘‘garbage in, garbage out’’ it is expected that even the best algorithm is not able to provide good output if the input data are bad. Hence, it might be expected that the estimation errors of *HRE* depend on data consistency, and are smaller in correlation with data inconsistency. The reasoning below supports this assertion.

Theorem 1 *For a complete PC matrix M , and PC graph G over M it holds that*

$$\mathcal{H}(M) \rightarrow 0 \Rightarrow e_\mu(u) \rightarrow 0 \quad (13)$$

where $\mathcal{H}(M)$ is Koczkodaj’s distance based inconsistency index for M , $e(u)$ is the mean absolute estimation error for $u \in C_U$, and μ is any estimation provided by the *HRE* procedure.

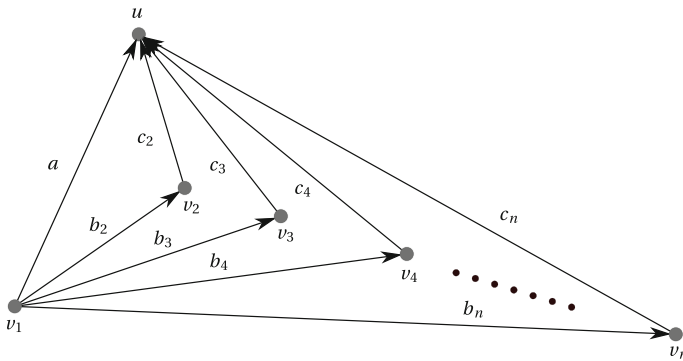


Fig. 1 Triads of vertices with u

Proof Let us consider some element $u \in C$ representing an initially unknown concept i.e. $u \in C_U$. Let v_1, \dots, v_n be its predecessors in G so that $(v_i, u) \in E \wedge i = 1, \dots, n$, for which the value $\mu(v_i)$ is known. Since the matrix M is complete there must exist edges $(v_1, v_2), \dots, (v_1, v_n) \in E$ so that $b_2 = M(v_1, v_2), b_3 = M(v_1, v_3), \dots, b_n = M(v_1, v_n)$. Moreover, let us denote edges between v_i 's and u as $a = M(v_1, u)$, and $c_2 = M(v_2, u), \dots, c_n = M(v_n, u)$ (see Fig. 1). Following the Eq. 1, Koczkodaj's distance inconsistency index $\mathcal{K}(M)$, in short \mathcal{K} , means that the maximal local inconsistency for some maximal triad of three vertices $t_1, t_2, t_3 \in C$ is \mathcal{K} . Thus, in the case of triads composed of the concepts v_1, v_i, u it must hold that:

$$\mathcal{K} \geq \min \left\{ \left| 1 - \frac{a}{b_i c_i} \right|, \left| 1 - \frac{b_i c_i}{a} \right| \right\} \tag{14}$$

for all $i = 2, \dots, n$. This implies that one of the following two statements is true:

$$a \leq b_i c_i \wedge \mathcal{K} \geq 1 - \frac{a}{b_i c_i} \tag{15}$$

$$b_i c_i \leq a \wedge \mathcal{K} \geq 1 - \frac{b_i c_i}{a} \tag{16}$$

Let us denote $\alpha \stackrel{df}{=} 1 - \mathcal{K}$ then the statements above can be written in the form:

$$a \leq b_i c_i \wedge \frac{1}{\alpha} \cdot a \geq b_i c_i \tag{17}$$

$$b_i c_i \leq a \wedge b_i c_i \geq \alpha a \tag{18}$$

Combining these two expressions (Eqs. 17 and 18) we obtain:

$$a \leq b_i c_i \leq \frac{1}{\alpha} \cdot a \vee \alpha a \leq b_i c_i \leq a \tag{19}$$

Let us denote $\beta_1 \stackrel{df}{=} a$ and $\beta_i \stackrel{df}{=} b_i \cdot c_i$, thus

$$a \leq \beta_i \leq \frac{1}{\alpha} \cdot a \vee \alpha a \leq \beta_i \leq a \quad (20)$$

Since $\alpha \leq 1$ (see Eqs. 15 and 16) thus the statement (Eq. 20) implies:

$$\alpha a \leq \beta_i \leq \frac{1}{\alpha} \cdot a \quad (21)$$

and of course

$$\alpha a \mu(v_1) \leq \beta_i \mu(v_1) \leq \frac{1}{\alpha} \cdot a \mu(v_1) \quad (22)$$

During the first step of the algorithm the μ is defined only for known concepts $v \in C_K$. Thus, all the concepts v_1, \dots, v_n are in C_K since only such elements are taken into account when calculating the value of $\mu(u)$. It is assumed that the value of the ratio $M(v_i, v_j) = m_{ji}$ for two a priori known concepts $v_i, v_j \in C_K$, corresponds to the actual fraction $\mu(v_j)/\mu(v_i)$, thus it holds that $\mu(v_1) \cdot b_i = \mu(v_i)$. Therefore the update equation for $\mu(u)$ (Listing: 1, line: 11) can be written in the form:

$$\mu(u) = \frac{1}{n} (\beta_1 + \dots + \beta_2) \mu(v_1) \quad (23)$$

Since every component β_i is bounded (Eq. 21) their mean must also be within the same bounds, which leads to the conclusion that:

$$\alpha \cdot a \cdot \mu(v_1) \leq \mu(u) \leq \frac{1}{\alpha} \cdot a \cdot \mu(v_1) \quad (24)$$

The absolute estimation error for some u with respect to μ at the end of the first step of the algorithm (Listing: 1) can be written and bounded from above as follows:

$$e_1(u) = \frac{1}{n} \sum_{i=0}^n |\mu(u) - \beta_i \mu(v_1)| \leq \max_{j=1, \dots, n} |\mu(u) - \beta_j \cdot \mu(v_1)| = |\mu(u) - \beta_k \cdot \mu(v_1)| \quad (25)$$

where $k \in \{1, \dots, n\}$. Because both components of the absolute difference on the right side of the expression 25 have the same lower and upper bounds (Eqs. 22 and 24), then the maximal possible distance between them is limited by the difference between their upper and lower bounds. Thus,

$$|\mu(u) - \beta_k \cdot \mu(v_1)| \leq \frac{1}{\alpha} a \mu(v_1) - \alpha a \mu(v_1) = a \mu(v_1) \left(\frac{1}{\alpha} - \alpha \right) \quad (26)$$

then the absolute mean error for the purpose of traversing the first level of G is upper bounded by:

$$e_1(u) \leq a\mu(v_1) \left(\frac{1}{\alpha} - \alpha \right) \tag{27}$$

The second level is a bit more complicated. Under the conditions of the algorithm there is $u \in C_U$, and M is complete. Thus, there exists at least one known concept $v \in C_K$ which precedes u in E i.e. $(v, u) \in E$. Let us put $v_1 = v$. This means that during the first step of the algorithm either v_i (for $i = 2, \dots, n$) was in C_K thus obviously $\mu(v_1) \cdot b_i = \mu(v_i)$ or $\mu(v_i)$ can be bounded using Eq. 24 (note that in order to use Eq. 24 a needs to be replaced by b_i). This leads to the following inequality:

$$\alpha \cdot b_i \cdot \mu(v_1) \leq \mu(v_i) \leq \frac{1}{\alpha} \cdot b_i \cdot \mu(v_1) \tag{28}$$

thus,

$$\alpha \cdot \beta_i \cdot \mu(v_1) \leq c_i \cdot \mu(v_i) \leq \frac{1}{\alpha} \cdot \beta_i \cdot \mu(v_1) \tag{29}$$

Since the Eq. 21 is valid for each triad, hence

$$\alpha^2 \cdot a \cdot \mu(v_1) \leq c_i \cdot \mu(v_i) \leq \frac{1}{\alpha^2} \cdot a \cdot \mu(v_1) \tag{30}$$

and then,

$$\alpha^2 \cdot a \cdot \mu(v_1) \leq \frac{(c_1 \cdot \mu(v_1) + \dots + c_n \cdot \mu(v_n))}{n} \leq \frac{1}{\alpha^2} \cdot a \cdot \mu(v_1) \tag{31}$$

which provides the estimation for $\mu(u)$ for the purpose of the second step of the algorithm:

$$\alpha^2 \cdot a \cdot \mu(v_1) \leq \mu(u) \leq \frac{1}{\alpha^2} \cdot a \cdot \mu(v_1) \tag{32}$$

Once again, the absolute error with respect to μ at the end of the second step can be written and upper bounded as follows:

$$e_2(u) = \frac{1}{n} \sum_{i=0}^n |\mu(u) - \mu(v_i) \cdot c_i| \leq \max_{i=1, \dots, n} |\mu(u) - \mu(v_i) \cdot c_i| = |\mu(u) - \mu(v_k) \cdot c_k| \tag{33}$$

where $k \in \{1, \dots, n\}$. Then, similarly as in the first step, both components have the same upper and lower bounds (Eqs. 30 and 32), thus the distance between them must not be greater than the distance between these limits. So,

$$e_2(u) \leq |\mu(u) - \mu(v_k) \cdot c_k| \leq \frac{1}{\alpha^2} a\mu(v_1) - \alpha^2 a\mu(v_1) = a\mu(v_1) \left(\frac{1}{\alpha^2} - \alpha^2 \right) \tag{34}$$

which means that the absolute mean error in the second step of the algorithm is bounded from above as follows:

$$e_2(u) \leq a\mu(v_1) \cdot \left(\frac{1}{\alpha^2} - \alpha^2\right) \tag{35}$$

Let us consider the r 'th step of the algorithm. Similarly as in the second step we consider $u \in C_U \wedge v_1 \in C_K$ where $(v_1, u) \in E$ (See Fig. 1). Let us assume by induction that every v_i for $i = 2, \dots, n$ is bounded as follows:

$$\alpha^{r-1} \cdot b_i \cdot \mu(v_1) \leq \mu(v_i) \leq \frac{1}{\alpha^{r-1}} \cdot b_i \cdot \mu(v_1) \tag{36}$$

(compare with Eq. 28). Then by repeating the same reasoning as for the second step (Eqs. 29–35) we come to the conclusion that:

$$e_r(u) \leq a\mu(v_1) \cdot \left(\frac{1}{\alpha^r} - \alpha^r\right) \tag{37}$$

Due to the principle of induction the above inequality holds for every $r \in \mathbb{N}_+$.

Since every $e_r(u)$ is a sum of absolute values then it cannot be negative i.e.

$$0 \leq e_r(u) \tag{38}$$

Moreover, the definition of α implies

$$\mathcal{K} \rightarrow 0 \Rightarrow \alpha \rightarrow 1 \tag{39}$$

Thus, due to the $(1/\alpha^r - \alpha^r)$ component on the right side of the inequality 37, when $\alpha \rightarrow 1$ then the whole right side of Eq. 37 also approaches to 0. Therefore, it holds that for every step $r > 0$ it is true that:

$$\mathcal{K} \rightarrow 0 \Rightarrow e_r(u) \rightarrow 0 \tag{40}$$

The above assertion, in the light of the arbitrary choice of $u \in C_U$, satisfies the thesis of the theorem. □

7 Convergence of solution

One of the immediate questions to come up is about the optimal number of iterations. The answer is not obvious, since the result of the algorithm depends on the input data. In particular, it is easy to construct simple graphs over the inconsistent PC matrix M in cases where every further step of the algorithm significantly increases the absolute error of estimation. For instance, the graph G such that $G = (C_K \cup C_U, E, M)$ where $C_K = v_1, C_U = v_2, v_3$ and $\{(v_1, v_2), (v_1, v_3), (v_2, v_3), (v_3, v_2)\} \subseteq E$ where the values $M(v_1, v_2), M(v_1, v_3), M(v_2, v_3), M(v_3, v_2)$ or all are less than or greater

than 1. In the first case the subsequent estimates $\mu(v_2)$ and $\mu(v_3)$ tend to 0, and reversely, if all the values are greater than 1, then $\mu(v_2)$ and $\mu(v_3)$ tends to ∞ . In the general case, the inequality 37 (Th. 1) suggests that with every subsequent step errors may increase. This leads to the conclusion that the optimal strategy is “the fewer steps the better”. Therefore, if the algorithm is not convergent, traversing at most one or two levels seems like a good idea.

If there are some edges with weights below 1 and some edges with weights above 1, the behavior of the algorithm is not so obvious. During the conducted experiments, it turned out that very often the subsequent iterations produce series of estimations convergent to some fixed positive results. To explain this phenomenon let us write the algorithm in the form of an appropriate system of equations, describing the on-step calculations (for the second step and the following ones). For this purpose, let us assume that the first iteration of the procedure was performed, hence the values $\mu(v)$ are assigned to all the concepts $v \in C$. For simplicity, let us assume that $C_U = \{c_1, \dots, c_k\}$, $C_K = \{c_{k+1}, \dots, c_n\}$ and denote b_i for all $i = 1, \dots, k$ as

$$b_i = \frac{1}{n-1}M(c_{k+1}, c_i)\mu(c_{k+1}) + \dots + \frac{1}{n-1}M(c_n, c_i)\mu(c_n) \tag{41}$$

Thus, during the second and subsequent iterations the algorithm calculates the new estimation value $\mu(c_i)$ for each unknown concepts $c_i \in C_U$ according to one of the following equations:

$$\begin{aligned} \mu(c_1) &= \frac{1}{n-1}M(c_2, c_1)\mu(c_2) + \dots + \frac{1}{n-1}M(c_k, c_1)\mu(c_k) + b_1 \\ \mu(c_2) &= \frac{1}{n-1}M(c_1, c_2)\mu(c_1) + \frac{1}{n-1}M(c_3, c_2)\mu(c_3) + \dots + \frac{1}{n-1}M(c_k, c_2)\mu(c_k) + b_2 \\ &\dots \\ \mu(c_k) &= \frac{1}{n-1}M(c_1, c_k)\mu(c_1) + \dots + \frac{1}{n-1}M(c_{k-1}, c_k)\mu(c_{k-1}) + b_k \end{aligned} \tag{42}$$

Let us denote:

$$a_{ij} = \frac{1}{n-1}M(c_j, c_i) \wedge i \neq j \text{ and } a_{ii} = M(c_i, c_i) = 1 \text{ and } \mu(c_i) = x_i \tag{43}$$

Then, the equation system takes the form:

$$\begin{aligned} x_1 - (1 - a_{11})x_1 - a_{12}x_2 - \dots - a_{1k}x_k &= b_1 \\ x_2 - a_{21}x_1 - (1 - a_{22})x_2 - \dots - a_{2k}x_k &= b_2 \\ \dots &\dots \\ x_k - a_{k1}x_1 - a_{k2}x_2 - \dots - (1 - a_{kk})x_k &= b_k \end{aligned} \tag{44}$$

Let us define the operator $T : \mathbb{R}^k \rightarrow \mathbb{R}^k$ (see Bronstein et al. 2005) as follows:

$$Tx = \left(x_1 - \sum_{r=1}^k a_{1r}x_r + b_1, \dots, x_k - \sum_{r=1}^k a_{kr}x_r + b_k \right)^T \tag{45}$$

The considered equation system can be written in the form of a fixed point problem in the metric space \mathbb{R}^k :

$$x = Tx \quad (46)$$

Let us assume one of the following metrics:

$$\rho(x, y) = \sqrt{\sum_{i=1}^k |\xi_i - \eta_i|^2}, \quad \rho(x, y) = \max_{1 \leq i \leq n} |\xi_i - \eta_i|, \quad \rho(x, y) = \sum_{i=1}^k |\xi_i - \eta_i| \quad (47)$$

and $x = (\xi_1, \dots, \xi_k), y = (\eta_1, \dots, \eta_k)$. It holds that if one of the following conditions Q_1, Q_2 or Q_3 is less than 1,

$$Q_1 = \sqrt{\sum_{i,j=1}^k |a_{ij}|^2}, \quad Q_2 = \max_{1 \leq i \leq n} \sum_{j=1}^k |a_{ij}|, \quad Q_3 = \max_{1 \leq j \leq n} \sum_{i=1}^k |a_{ij}| \quad (48)$$

then T turns out to be a contracting operator (Bronstein et al. 2005). According to the Banach fixed point theorem, there exists only one such point. Hence, in such a case there is only one set of values $\mu(c_1), \dots, \mu(c_k)$ which is the limit of the sequence of HRE estimations.

Due to the fraction $1/n-1$ the smaller a_{ij} is the larger C is and the smaller $M(c_j, c_i)$ is. Moreover, the conditions (Eq. 48) can be more easily met with fewer concepts in C_U (that is because the summations embedded in the conditions (Eq. 48) include fewer elements). In other words, the estimation procedure has a high chance to be convergent if

1. The set C_U is relatively small (C_K is relatively large),
2. The estimated values $\mu(v)$ are similar.

Both of these conditions are quite intuitive and, in practice, are likely to be satisfied. The first of them reflects the natural desire to provide the experts with rather more than the lower number of known, reference concepts. The second corresponds to the common-sense observation that all the considered concepts should be similar to each other, because then, it is easy to compare them. In other words, the expert estimates are more reliable when the compared projects are more similar.

The convergence of the algorithm implies the convergence of the estimation error. Unfortunately, the limit towards which the estimation error tends is not necessarily the smallest possible error value. Hence, in order to minimize the error, the user needs to choose the best estimation from all the estimations generated during the course of the algorithm (Listing: 1, line: 21).

In the case of the previously considered example (Sect. 5) all the three conditions Q_1, Q_2 and Q_3 are below 1. Thus, the algorithm is convergent and the computed limits for $\mu(v_1), \mu(v_2)$ and $\mu(v_3)$ are 2.758, 4.578 and 3.493, respectively. The absolute average estimation errors converge to $e_\mu(v_1) = 0.12$, $e_\mu(v_2) = 0.631$ and $e_\mu(v_3) = 0.338$, and are minimal with respect to the mean of errors $e_\mu(v_1) + e_\mu(v_2) + e_\mu(v_3)/3$.

8 Discussion

The idea underlying the *HRE* algorithm is the assumption that experts work independently and try to do their job as best as they can. Thus, they may be wrong in a random manner, and if so, it makes sense to accept their estimates as a part of samples, and the expected value of the sample (the arithmetic mean) as the alleged value of the estimated μ for the unknown concept. The absolute average estimation error may indicate how good such an estimation is. Depending on the data, the algorithm may or may not converge. If the algorithm does not converge (subsequent estimates are getting bigger) or converges to 0, selecting from among the first few estimates the one with the smallest error seems to be the best choice. If the algorithm converges to a non 0 value, it is very often useful to generate subsequent estimates until the current estimate does not differ from the limit (fixed point) with some ϵ , then to choose a set of estimations (among those generated) for which the average estimation error is minimal. Of course, this is not the only possible strategy. For instance, if some subset of C_U is particularly important, only errors for its elements may be taken into account while determining the optimal set of estimations.

However, the problem considered in the work relies on determining values of μ for elements from $C_U = \{c_1, \dots, c_k\}$ on the basis of concepts $C_K = \{c_{k+1}, \dots, c_n\}$ using the matrix M , it can be reformulated as computing a consistent approximation of a matrix M where certain elements are fixed. Indeed, at the end of the presented algorithm all the concepts $c \in C$ have assigned some values $\mu(c)$. Thus, defining $m'_{ij} = \frac{\mu(c_i)}{\mu(c_j)}$ allows the construction of a new matrix $M' = [m'_{ij}]$, which is a consistent approximation of M where $m_{ij} = m'_{ij}$ for $c_i, c_j \in C_K$. In the case of the previously considered example, assuming the values of $\mu(v_1)$, $\mu(v_2)$ and $\mu(v_3)$ corresponding to the smallest average absolute mean error $e_\mu(C_U)$, the matrix M' equals:

$$M' = \begin{bmatrix} 1 & 0.602 & 0.789 & 0.46 & 1.379 \\ 1.66 & 1 & 1.311 & 0.763 & 2.289 \\ 1.266 & 0.763 & 1 & 0.582 & 1.746 \\ 2.175 & 1.311 & 1.718 & 1 & 3 \\ 0.725 & 0.437 & 0.572 & 0.333 & 1 \end{bmatrix} \tag{49}$$

In particular it holds that $m_{45} = m'_{45} = 3$ and $m_{54} = m'_{54} = \frac{1}{3}$.

Although the presented considerations in (Sect. 6 and 7) assume that the matrix M is complete (see Def. 3), the *HRE* algorithm seems to work without this assumption. For incomplete matrices, the value $\mu(u)$ for $u \in C_U$ can be determined as long as a path exists in the pairwise comparisons graph G over M (Def. 4) between u and some element $v \in C_K$. Hence, there is a chance that the *HRE* algorithm may support different pairwise comparisons techniques also when the input data are incomplete. In particular, it might be useful for the *AHP* (Saaty 1977) approach. The properties of the presented algorithm for the incomplete sets of input data will be the subject of future research.

9 Summary

The *HRE* algorithm presented here for computing estimations of initially unknown concepts C_U using information about known concepts C_K and the matrix M proposes a new approach to the pairwise comparisons method. It defines the intuitive algorithm of using the pairwise comparisons matrix M for determining the most probable values of unknown concepts (original stimulus) on the basis of the known concepts. The presented procedure iteratively generates sets of estimations, then chooses the set which has the smallest average absolute mean estimation error. According to the proven theorem, the size of the estimation error depends on \mathcal{K} — *Koczkodaj's distance based inconsistency index* shows that the lower the inconsistency index, the lower the estimation errors. The theorem can be particularly useful when the number of iterations of the *HRE* algorithm is small. In such cases, it may in practice be used to estimate the size of the estimation errors.

For some input data sets, the subsequent estimations produced by the *HRE* algorithm converges. If this happens, the estimation errors also converge to some limit, thus the number of estimation sets produced by the *HRE* algorithm does not need to be limited to a few elements. In such a case the sets of estimations can be generated until the limit towards which the *HRE* results converge will not be close enough. Such a situation has also been addressed in the paper. The given conditions of convergence have an intuitive explanation and in many practical situations are likely to be met.

The *HRE* algorithm is suitable for any matrix with positive elements, i.e., even in those situations where the applicability of the classical eigenvector method can be limited (finding the largest absolute eigenvalue of a nonreciprocal matrix may be difficult). The presented algorithm remains open for much more data than can be stored in a single pairwise comparisons matrix. Due to the graph representation of the problem, multiple values defining ratios between the same pairs of concepts can be easily encompassed within the algorithm as multiple arcs between the same pairs of vertices.

Acknowledgments The author would like to thank Prof. W.W. Koczkodaj, Laurentian University, Sudbury, Ontario, Canada, for the inspiration and valuable Skype discussions about the pairwise comparisons method. The author is also grateful to the anonymous reviewers for their comments and suggestions which have helped improve the quality and content of the paper.

Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

- Bozóki S (2008) Solution of the least squares method problem of pairwise comparison matrices. *Central European Journal of Operations Research* 16(4):345–358
- Bozóki S, Rapsak T (2008) On Saaty's and Koczkodaj's inconsistencies of pairwise comparison matrices. *Journal of Global Optimization* 42(2):157–175
- Bronstein IN, Semendjajew KA, Musiol G, Mühlig H (2005) *Handbook of mathematics*, 5th edn. Springer, Frankfurt am Main

- Cao D, Leung LC, Law JS (2008) Modifying inconsistent comparison matrix in analytic hierarchy process: a heuristic approach. *Decis Support Syst* 44(4):944–953
- Cormen TH, Leiserson CE, Rivest RL, Stein C (2009) Introduction to algorithms, 3rd edn. MIT Press, Cambridge
- Fechner GT (1966) Elements of psychophysics, vol 1. Holt, Rinehart and Winston, New York
- Gomes L (1993) Efficient reduction of inconsistency in pairwise comparison matrices. *Syst Anal Model Simul* 11(4):333–335
- Koczkodaj WW (1993) A new definition of consistency of pairwise comparisons. *Math Comput Model* 18(7):79–84
- Koczkodaj WW, Herman MW, Orłowski M (1999) Managing null entries in pairwise comparisons. *Knowl Inf Syst* 1(1):119–125
- Koczkodaj WW, Orłowski M (1999) Computing a consistent approximation to a generalized pairwise comparisons matrix. *Comput Math Appl* 37(3):79–85
- Koczkodaj WW, Szarek SJ (2010) On distance-based inconsistency reduction algorithms for pairwise comparisons. *Log J IGPL* 18(6):859–869
- Peterson GL, Brown TC (1998) Economic valuation by the method of paired comparison, with emphasis on evaluation of the transitivity axiom. *Land Econ* 74:240–261
- Saaty TL (1977) A scaling method for priorities in hierarchical structures. *J Math Psychol* 15(3):234–281
- Saaty TL (1980) The analytic hierarchy process: planning, priority setting, resource allocation. McGraw-Hill International Book, New York
- Saaty TL (2008) Relative measurement and its generalization in decision making. Why pairwise comparisons are central in mathematics for the measurement of intangible factors. The analytic hierarchy/network process. *Stat Oper Res (RACSAM)* 102:251–318
- Temesi J (2006) Consistency of the decision-maker in pair-wise comparisons. *Int J Manag Decis Mak* 7(2):267–274
- Thurstone LL (1994) A law of comparative judgment, reprint of an original work published in 1927. *Psychol Rev* 101:266–270
- Tideman TN (1987) Independence of clones as a criterion for voting rules. *Soc Choice Welf* 4:185–206
- Triantaphyllou E, Pardalos PM, Mann SH (1990) A minimization approach to membership evaluation in fuzzy sets and error analysis. *J Optim Theory Appl* 66(2):275–287
- Walpole RE et al (2012) Probability & statistics for engineers & scientists, 9th edn. Prentice Hall, Englewood Cliffs
- Xu Z, Wei C (1999) A consistency improving method in the analytic hierarchy process. *Eur J Oper Res* 116(2):443–449