

# A three-level-similarity measuring method of participant opinions in multiple-criteria group decision supports

Jun Ma<sup>a,b,\*</sup>, Jie Lu<sup>b,\*</sup>, Guangquan Zhang<sup>b</sup>

<sup>a</sup>*SMART Infrastructure Facility, University of Wollongong, Wollongong, NSW 2522, Australia*

<sup>b</sup>*DeSI Lab, Centre for QCIS, School of Software, Faculty of Engineering and Information Technology  
University of Technology, Sydney (UTS), NSW 2007, Australia*

---

## Abstract

Group decision making is a commonly-used method to develop decision support applications for use in counter-terrorism, government management and business intelligence development. An appropriate decision often needs extensive debate among participants in a decision committee; overly similar opinions to a large extent may mislead to a final decision. Measuring opinion similarity between participants (MOSP) in advance is an important strategy to reduce the chance of making and applying inappropriate decisions and is also an important concern when developing a reliable decision support system. Due to the lack of opinion data for a focal topic and the varieties of opinion representations, measuring the similarity is difficult and has not been well-studied in developing decision support. Noting that the similarities gradually alter from time to time with the number and order of considered criteria, this paper develops a gradual aggregation algorithm and establishes a method based on it, called the three-level-similarity measuring (TLSM) method, to measure the opinion similarity at three similarity levels, i.e. the Assessment-Level, the Criterion-Level and the Problem-Level. Two applications of the TLSM method on social policy selection and energy policy evaluation are conducted. The study indicates that the TLSM method can effectively measure the similarity between opinions in small-size or possibly missing opinion data and simulate the generation of a decision dynamically.

*Keywords:* group decision making, opinion similarity, measuring method, aggregation operator, opinion analysis

---

## 1. Introduction

Multiple-criteria group decision making (MCGDM) is recognised as an efficient strategy in many organisational decision problems [1–3], where a final decision is made based on the opinions of individual participants of a decision committee on candidate options. Overly similar opinions increase the chance of putting an inappropriate decision into effect. In practice, making an appropriate strategic decision is a time-consuming and costly task; however, tuning an inappropriate decision will cost even more. To reduce this risk, measuring opinion similarity between participants (MOSP) in advance is, therefore, an important issue in developing decision support for essential decision problems relating to such issues as counter-terrorism, business intelligence, nuclear inspection, government management and others.

Opinion similarity has wide applications in various fields, for example, online recommender systems [4–8]. However, the MOSP problem is still an unsolved and challenging issue. Difficulties in solving the MOSP problem include the effective processing of small-size opinion data and of varied opinion representations. Due to the restrictions on funding, time, cost, private policies, and other issues, a decision is often made by a limited number of participants. The total amount of usable opinions for measuring similarity is small sized, even though all participants would like to express their opinions thoroughly. The small-size opinion data makes it is very hard to apply methods for large-size data to the MOSP problem. Varied opinion representation is another difficulty in solving the MOSP problem. Participants prefer to express their opinions in their own ways based on their understandings of and experiences in a given decision topic. However, this is bound to difficulties for measuring the similarity between their opinions. A strategy commonly used to regulate opinion representation is providing a fixed number

---

\*Corresponding author

Email addresses: jma@uow.edu.au (Jun Ma),  
jie.lu@uts.edu.au (Jie Lu),  
guangquan.zhang@uts.edu.au (Guangquan Zhang)

of choices, for example, some predefined linguistic terms or a set of ordinal numbers [2, 9]. However, this cannot completely avoid varied opinion representations because the pre-defined choices may have different semantics for different evaluation criteria. A third difficulty in solving the MOSP problem is the lack of a fixed reference point for the measuring task. A person's opinion is a type of subjective information, which varies from one object to another. Two participants may have similar opinions on some options, but completely different opinions on the others. Hence, the reference point should be able to fit these changes. The MOSP problem needs to find the similarity between two participants on a decision problem as a whole; so the reference point should be used at different levels. Moreover, the majority of research on MCGDM focus on the issue of conducting trade-off between participants to reach a consensus based on their opinions but ignore the MOSP issue which is the basis of that trade-off.

Keeping the aforementioned difficulties in mind, this paper presents a measuring method to solve the MOSP problem. The method is based on three assumptions: 1) Given a criterion, if the opinions of two participant are similar for the majority of testing benchmarks, it is rational to presume that they are similar; 2) Given a set of criteria, if the opinions of two participants are similar for the majority of important criteria, it is rational to presume that they are similar; and 3) Given a decision problem, if the opinions of two participants produce a similar decision, it is rational to presume that they are similar. Because these three assumptions are presented from three similarity levels, this paper refers the method presented to a three-level-similarity measuring (TLSM) method.

The rest of the paper is organized as follows. Section 2 reviews related work in opinion analysis, similarity measurement and aggregation operations. Section 3 develops a gradual aggregation algorithm which is used to generate an overall opinion similarity. In Section 4, we introduce the TLSM method in detail. Section 5 illustrates two case studies for applying the TLSM method to social policy selection and energy policy evaluation problems. Section 6 summarises the main contributions of the work and our future study plans.

## 2. Related works

Opinion analysis is extensively studied in social psychology fields [10, 11]; recently, requirements for effectively extracting, summarizing, and segmenting opinions of general or specific users boosted the growing research on opinion mining and sentiment analysis [12–15]. Com-

monly recognized, the opinion mining research belongs to the field of text analysis [14]; therefore, currently reported methods are mainly conducted on how to efficiently extract and summarize opinions from texts distributed among web Blogs posts [15], BBS [16], online feedback [17, 18], and web forums [12]. Many opinion mining systems have been developed and applied [12, 14, 16, 18, 19]. However, these methods are not suitable for the MOSP problem because of the difficulties mentioned above. In the MCGDM field, study of opinion analysis is conducted in two main areas. Qualitative studies analyse and simulate the behaviour patterns of users based on their opinions of a considered affair [20, 21]. Quantitative research focuses on how to represent and process opinions in a computational framework to support decision making [9, 22–24]. For instance, fuzzy sets and fuzzy logic are widely used as opinion representation and process facilities [25–27] because they can effectively interpret and model the subjective information with uncertainties. Noted that a participant's opinion itself is a kind of subjective expression with uncertainties, these computation-based techniques provide support to develop solutions for the MOSP problem.

Similarity measurement is widely studied in human knowledge representation, behaviour analysis, and real-world problem solving [28]. A similarities measurement can be established on various theories (e.g., classical and/or fuzzy set theories, classical and/or non-classical logics[29]) and applied to image processing [28, 30], natural language understanding [31], recommender systems [32], and other applications. In MCGDM, similarity measures defined on a fuzzy set are particularly useful. Because many similarity measures can be sourced from their counterparts defined on ordinary sets, research on their relationships is conducted. For example, Wang, et al. [33] compared commonly-used similarity measures on the ordinary set. De Baets, et al. [34] discussed systematically a way of generating a similarity measure for ordinary sets and compared it with other 28 similarity measures. Recently, Bosteels and Kerre presented a family of cardinality-based fuzzy similarity measures which is specified by three parameters [35] and De Baets, et al. [36] studied the transitivity of cardinality-based similarity measures. Generally speaking, a similarity measure can be induced from a distance measure. Therefore, investigating the relationship between them is very important [37]. The majority of existing similarity measures are defined on the Euclidean space and the ultimate measurement is a crisp value. Noting that a crisp value cannot sufficiently depict the fuzziness in real cases, Chakraborty

and Chakraborty [22] defined a similarity, whose measurement is a fuzzy set. Using this distance, they implemented a clustering algorithm to solve a group decision making problem.

Integrating evaluations of multiple participants is an important step to develop a solution for an MCGDM problem, where an aggregation operator plays a crucial role. According to whether or not an aggregation operator explicitly considers the relevant importance (weights) of the evaluation criteria, there are three main types of aggregation operators used in MCGDM research. The first type treats all evaluation criteria equally. Typical examples include the arithmetic mean, the geometric mean, and the  $t$ -norms (or  $t$ -conorms) [38–40]. The second type explicitly distinguishes the weights of the evaluation criteria either by their impacts on the decision problem, or by their processing order. The weighted mean and the ordered weighted aggregation (OWA) [41], as well as their extensions [42–46] belong to this type. A third type is defined by certain integrals, such as the Segno and Choquet integrals [23, 47–49]. Currently existing aggregation operators often require that a participant provides a complete evaluation report; in other words, they do not consider and process the cases with missing evaluations. However, a real decision problem more or less faces missing values. How to process missing values is, therefore, a key concern when applying an aggregation operator; but this issue has not yet been solved. Although so many powerful aggregation operators have been presented, little is known about how to select an appropriate one in real applications. Beliakov [50] reported a solution for this problem by using a mathematical programming technique. In his solution, an aggregation operator's form is fixed but its parameters are tunable.

### 3. A gradual aggregation algorithm

#### 3.1. Motivations and implementations

In this section, a gradual aggregation algorithm (GAA) is developed. The GAA is motivated by two practical issues when developing a decision support system. One issue concerns processing missing values; and the other is about generating a decision dynamically.

Evaluation aggregation used in an MCGDM problem is conventionally conducted as a one-off procedure. For effectiveness reason, the inputs are required to be significant values in a given value set (eg. terms or numbers) and cannot have missing values. However, real evaluations cannot avoid missing values. Hence, how to process missing values is a regularly faced issue when making a deci-

sion, which has not been well solved. Another motivation for presenting the GAA is the phenomenon of generating a decision dynamically, which refers to the procedure of making a decision where a final decision is sketched based on a few numbers of criteria at the initial stage, and then amended in the following stages by considering more criteria added gradually. A typical example of generating decision dynamically is booking a flight ticket. At the beginning, a passenger has some preliminary requirements for a ticket such as the airline provider, the departure and/or arrival times. If these requirements cannot be fully met, the passenger may consider extra requirements of price, stops, etc, on top of those preliminary requirements until find the most satisfactory ticket. In order to model this procedure quantitatively, the GAA is therefore presented.

We implemented the GAA in two ways, i.e., the ordinary gradual aggregation (OGA) and the weighted gradual aggregation (WGA). The difference between them is that OGA does not explicitly process the weights of criteria but leaves it to the aggregation itself; while the WGA does. We define the OGA and WGA in Definition 3.1 and Definition 3.2 and illustrate the procedure of GAA in Figure 1.

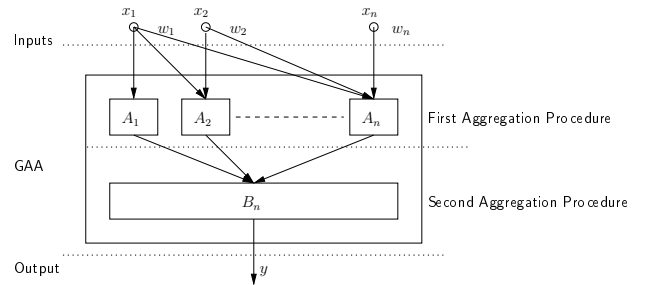


Figure 1: The typical GAA procedure.

Following the formal notations in [40], an aggregation operator  $\mathcal{A}$  over a closed set  $X$  is denoted by  $\mathcal{A} : \bigcup_{i \in \mathbb{N}^+} \{A_i : X^i \rightarrow X\}$  where  $A_i$  is a mapping from  $X^i$  to  $X$  and is called the  $i$ -ary aggregation operator in  $\mathcal{A}$ . By this notation, an aggregation operator  $\mathcal{A}$  refers to a family of operators with the same computational form but variable inputs number. Particularly, the unitary aggregation operator  $A_1$  is the identity mapping. For convenience of discussion and practical demands, let  $X$  be a subset of  $\mathbb{R}$ .

**Definition 3.1.** Let  $\mathcal{A}$  and  $\mathcal{B}$  be two aggregation operators. A mapping  $G_n$  from  $X^n$  to  $X$  is called an  $n$ -ary ordinary gradual aggregation (OGA) with respect to  $\mathcal{A}$  and  $\mathcal{B}$ :

$$\begin{aligned} G_n(x_1, \dots, x_n) \\ = B_n(\{\alpha_i = A_i(x_1, \dots, x_i), i = 1, \dots, n\}). \end{aligned} \quad (1)$$

**Definition 3.2.** Let  $\mathcal{A}$  and  $\mathcal{B}$  be two aggregation operators;  $w_i$  the weight of input  $x_i$ ,  $i = 1, \dots, n$ . A mapping  $G_n$  from  $X^n$  to  $X$  is called an  $n$ -ary weighted gradual aggregation (WGA) with respect to  $\mathcal{A}$  and  $\mathcal{B}$ :

$$\begin{aligned} G_n(x_1, \dots, x_n; w_1, \dots, w_n) \\ = B_n(\{\alpha_i = A_i(x_1, \dots, x_i; w_1, \dots, w_i), \\ i = 1, \dots, n\}). \end{aligned} \quad (2)$$

Because OGA and WGA are defined by the aggregation operators  $\mathcal{A}$  and  $\mathcal{B}$ , they inherit some properties of  $\mathcal{A}$  and  $\mathcal{B}$ . Some examples are given below. These properties indicate that the OGA and WGA exactly can be used to implement aggregation procedure.

**Proposition 3.1.** If both  $\mathcal{A}$  and  $\mathcal{B}$  are idempotent, i.e.,

$$A_i(x, \dots, x) = x, \quad B_j(x, \dots, x) = x;$$

so do OGA and WGA.  $\square$

**Proposition 3.2.** If both  $\mathcal{A}$  and  $\mathcal{B}$  are monotonic, i.e.,

$$\begin{aligned} A_i(x_1, \dots, x_i) \leq A_i(y_1, \dots, y_i) \text{ if } x_k \leq y_k, k = 1, \dots, i \\ B_j(x_1, \dots, x_j) \leq B_j(y_1, \dots, y_j) \text{ if } x_k \leq y_k, k = 1, \dots, j; \end{aligned}$$

so do OGA and WGA.  $\square$

**Proposition 3.3.** If both  $\mathcal{A}$  and  $\mathcal{B}$  are bounded, i.e.,

$$A_i(x_1, \dots, x_i) \in X, \quad B_j(x_1, \dots, x_j) \in X;$$

So do OGA and WGA.  $\square$

### 3.2. Weights assignment and adjustment

Weights of criteria are important parameters in an evaluation aggregation; but assigning and/or adjusting weights is not an easy task [51]. The GAA can implement weights assignment and adjustment by itself when the used aggregation operators are the arithmetic mean and weighted mean. Moreover, it can preserve the impacts of important criteria in the assignment and adjustment procedure.

The OGA does not explicitly process the weights of criteria. However, when both  $\mathcal{A}$  and  $\mathcal{B}$  are arithmetic means, the OGA assigns a set of weights to its inputs implicitly based on their processing order. Suppose a set of inputs  $x_1, x_2, \dots, x_n$  are indexed by their processing order, whose weights are not known. Then by the OGA, we have

$$\begin{aligned} A_1(x_1) &= x_1; \\ A_2(x_1, x_2) &= \frac{x_1 + x_2}{2}; \\ &\vdots \\ A_n(x_1, x_2, \dots, x_n) &= \frac{x_1 + x_2 + \dots + x_n}{n} \end{aligned}$$

and

$$\begin{aligned} G_n(x_1, \dots, x_n) &= \frac{\sum_{i=1}^n A_i(x_1, \dots, x_i)}{n} \\ &= \sum_{i=1}^n x_i \left( \frac{1}{n} \sum_{j=i}^n \frac{1}{j} \right). \end{aligned} \quad (3)$$

Let  $\beta_i$  be the coefficient of  $x_i$  in Eq. (3), i.e.,

$$\beta_i = \frac{1}{n} \sum_{j=i}^n \frac{1}{j}, \quad i = 1, 2, \dots, n. \quad (4)$$

The sum of  $\beta_i$ s is

$$\beta_1 + \beta_2 + \dots + \beta_n = 1, \quad (5)$$

and the order of them is

$$\beta_1 > \beta_2 > \dots > \beta_n > 0. \quad (6)$$

Eq. (5) shows that  $\beta_1, \dots, \beta_n$  form a set of weights and are assigned to the inputs implicitly. Eq. (6) indicates that the an input processed earlier gains a higher weight. Intuitively, this weight assignment result is consistent with a real decision procedure where the most important criteria are often processed ahead.

Furthermore, let us check the changes of these assigned weights with respect to the number  $n$  of inputs. Figure 2 illustrates the first five assigned weights when  $n \leq 18$ . It shows that each  $\beta_i$  is convergent with the increase of  $n$ . A conclusion is drawn from this observation that, given a large enough  $n$ , the newly added inputs will exert little affect on a sketchy decision. Since the parameter  $n$  in a real problem cannot be too large, the impacts of the most important criteria underlying the inputs—which are processed ahead—are therefore strengthened.

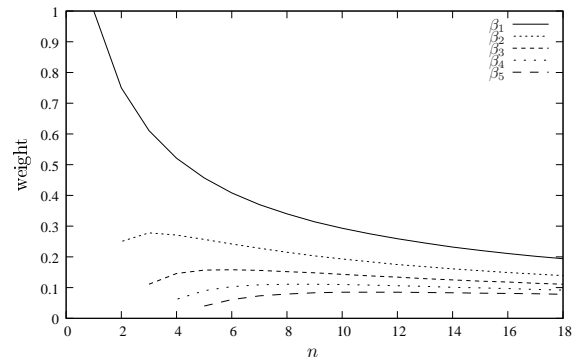


Figure 2: Changing weights with the number of inputs.

The WGA explicitly processes the weights of criteria in its aggregation procedure. By replacing  $A_i$  with the

weighted mean, and supposing the initial weight of input  $x_i$  is  $w_i$ , we noted that the WGA can adjust the initially assigned weights of the inputs. By the WGA, we have

$$\begin{aligned} A_1(x_1; w_1) &= x_1 \\ A_2(x_1, x_2; w_1, w_2) &= \frac{w_1}{w_1 + w_2}x_1 + \frac{w_2}{w_1 + w_2}x_2 \\ &\vdots \\ A_n(x_1, \dots, x_n; w_1, \dots, w_n) &= \frac{w_1}{\sum_{j=1}^n w_j}x_1 + \dots + \frac{w_n}{\sum_{j=1}^n w_j}x_n \end{aligned}$$

and

$$\begin{aligned} G_n(x_1, \dots, x_n; w_1, \dots, w_n) &= \frac{x_1 + \frac{w_1 x_1 + w_2 x_2}{w_1 + w_2} + \dots + \sum_{i=1}^n \frac{w_i x_i}{\sum_{j=1}^n w_j}}{n} \\ &= \frac{1}{n} \sum_{i=1}^n x_i w_i \left( \sum_{k=i}^n \frac{1}{\sum_{j=1}^k w_j} \right). \end{aligned}$$

Let  $\beta_i$  be the coefficient of  $x_i$ , i.e.,

$$\beta_i = \frac{w_i}{n} \sum_{k=i}^n \frac{1}{\sum_{j=1}^k w_j}, \quad i = 1, \dots, n \quad (7)$$

Then we have

$$\beta_1 + \beta_2 + \dots + \beta_n = 1, \quad (8)$$

i.e.,  $\beta_1, \beta_2, \dots, \beta_n$  form a set of weights and the inputs are re-weighted by them. Comparing  $\beta_i$  and  $w_i$ , we have a loose inequity that

$$\beta_i \geq \frac{n - (i - 1)}{n} w_i, \quad i = 1, \dots, n. \quad (9)$$

Further analysis indicates that  $\beta_1 \geq w_1$  and if  $n$  is large enough and  $i$  is smaller, the first several  $\beta_i$ s are very near to, even greater than, the initial  $w_i$ s. This means the impacts of those criteria are still preserved by the WGA.

The above algorithm and discussions indicate that the GAA can effectively maintain the impacts of important criteria. This feature is very important for our next discussion regarding making decisions dynamically and processing missing values.

### 3.3. Dynamic decision and missing values

Definitions of OGA and WGA indicate that the two implementations of GAA are closely related to the processing order of the inputs. The GAA emphasises the processing order of inputs because it is closely related to the

dynamic generation of a decision and process of missing values.

In the course of making a decision, the most important criteria are often considered preferentially, then the secondary important criteria, and finally the not so important criteria. Hence, there is a natural processing order between those criteria. Similarly, there is an ordering among the inputs if they are treated as evaluations of those criteria, as shown in Section 3.2 where the GAA implementation assigns (reassigns) a set of decreasingly changed weights to its inputs according to their processing orders. In this sense, the GAA implementations are models of the generation of a dynamic decision.

Missing values are inevitable in real applications. Two intuitive strategies to handle missing values are: 1) completely discard them; or 2) try to impute them. The GAA implementations can partially combine these. When the parameter  $n$  in GAA takes a value which is smaller than the total number of inputs, some inputs will then not be considered naturally. Obviously, if there are missing values in the unprocessed inputs, these missing values will not affect the obtained aggregation result. However, if the missing values are not avoided, it means some evaluations about key criteria are not presented. In this situation, the GAA repeatedly use the aggregation operator  $\mathcal{A}$  to calculate a set of candidate results by slightly assigning or adjusting the weights of those inputs; and then the use of aggregation operator  $\mathcal{B}$  to generate a final aggregation result which can partially impute the missing values. To illustrate this more clearly, consider the example below.

**Example 3.1.** There are two common ways to process the missed value, i.e. 1) completely ignore it; or 2) treat it as a fixed value (such as 0 or mean for numeric values). For illustrative purpose, suppose there are 10 inputs are given in the second column in Table 1 and the aggregation algorithm is the arithmetic means. We will compare three scenarios:

- scenario 1 (S1): no missing value;
- scenario 2 (S2): ignore missing value;
- scenario 3 (S3): replace the missing value as 0 and mean of others.

For (S1), the aggregation result without using OGA is 0.572 (column ‘‘Input’’), i.e.  $0.572 = 1/10 \times \sum_{i=1}^{10} a_i$ . The aggregation result with OGA, where  $\mathcal{A}$  and  $\mathcal{B}$  are both the arithmetic means, is 0.683 (column ‘‘OGA’’) and the intermediate results are shown in the third column. For (S2), the aggregation result without using OGA is 0.549

Table 1: An example for processing a missing value.

No.	Input	S1			S2		S3		
		OGA	DM	OGA-DM	IM-0	OGA-0	IM-M	OGA-M	
1	0.840	0.840	0.840	0.840	0.840	0.840	0.840	0.840	
2	<b>0.783</b>	0.812	0.912	0.876	<u>0.000</u>	0.420	<u>0.549</u>	0.694	
3	0.912	0.845	0.335	0.696	0.912	0.584	0.912	0.767	
4	0.335	0.718	0.278	0.591	0.335	0.522	0.335	0.659	
5	0.278	0.630	0.477	0.568	0.278	0.473	0.278	0.583	
6	0.477	0.604	0.365	0.535	0.477	0.474	0.477	0.565	
7	0.365	0.570	0.952	0.594	0.365	0.458	0.365	0.537	
8	0.952	0.618	0.636	0.599	0.952	0.520	0.952	0.588	
9	0.636	0.620	0.142	0.549	0.636	0.533	0.636	0.594	
10	0.142	0.572			0.142	0.494	0.142	0.549	
result	<b>0.572</b>	<b>0.683</b>	<b>0.549</b>	<b>0.650</b>	<b>0.494</b>	<b>0.532</b>	<b>0.549</b>	<b>0.638</b>	
rate to benchmark	1.000	1.000	0.959	0.951	0.863	0.778	0.959	0.933	

(column “DM”); while it is 0.650 by using OGA (column “OGA-DM”). For (S3), the aggregation results without using OGA are 0.494 and 0.549 for replacing the missing value by 0 (column “IM-0”) and the average of the others (column “IM-M”), respectively; where the aggregation results by using OGA are 0.532 (column “OGA-0”) and 0.638 (column “OGA-M”), respectively.

If taking the scenario 1 as benchmark and check the changes of aggregation results in the other two scenarios, we noted that the OGA generates a result with bigger difference from the benchmarks than the other methods. This fact indicates that the OGA pays more attention on the missing value.

#### 4. A three-level-similarity measuring method for the MOSP problem

In this section, the TLSM method for solving the MOSP problem is presented. Section 4.1 addresses the MOSP problem briefly. Section 4.2 overviews the main steps of the TLSM method. Details of those steps are introduced in Sections 4.3 and Section 4.5.

##### 4.1. The MOSP problem

An MOSP problem is briefly addressed as follows. Given an MCGDM problem with some candidate options, the participants evaluate these options in terms of a set of evaluation criteria and everyone completes an evaluation report. Each participant’s evaluations are summarised in linguistic terms. After collecting these evaluation reports, a question arises: can we identify which two participants have similar opinions on this kind of decision problem, based on the collected evaluation reports.

For convenience of discussion, the candidate options are denoted by  $O = \{o_i | i \in I\}$ ; the evaluation criteria

are denoted by  $C = \{c_j | j \in J\}$ ; and the participants are denoted by  $E = \{e_k | k \in K\}$ . An evaluation report is denoted by a 2-D matrix  $V_k = (v_{ij})_{I \times J}$ , where  $k$  is the index of participant  $e_k$  and  $v_{ij}$  is the evaluation (i.e., opinion) on option  $o_i$  about criterion  $c_j$ .  $v_{ij}$  is either an element in  $T_j$ , which is the collection of all linguistic terms used by the participants for criterion  $c_j$ , or a blank for “not available” or “no answer”. Without loss of generality, we suppose that each participant provides only one term for each option about each criterion.

##### 4.2. Overview of the TLSM method

The outline of the TLSM method is shown in Table 2. By the TLSM method, the similarity of the opinions of two participants will be measured at three sequential levels, i.e., the Assessment-Level, the Criterion-Level, and the Problem-Level.

At the Assessment-Level, the evaluations of two participants are compared option by option in terms of a given criterion. The comparison is conducted based on the assumption that two participants should have higher similar opinions if the number of candidate options on which they have similar evaluations is bigger. To judge whether or not two evaluations are similar, the term set  $T_j$  is firstly divided into several semantic-equal groups by pari-wise comparison on the semantics of terms used; two terms are, then, said to be similar (or have similar semantics) if they are in the same group. By the comparison conducted option by option on the two participants’ evaluations, it is known to what extent the two participants have similar opinions on a given criterion from the viewpoint of a single criterion. The similarity should be proportional to the ratio of the number of options with similar evaluations against the total number of options.

Table 2: Outline of main processes in the TLSM method.

Process level	Main steps
Assessment-Level	<i>Input: two experts' evaluation reports; evaluation term set <math>T_j</math></i> <i>Output: the similarity about criterion <math>c_j</math></i> <hr/> Step 1.1 determine a similarity matrix for evaluation terms for criterion $c_j$ ; Step 1.2 determine a clustering algorithm; Step 1.3 generate semantic-equal groups by the clustering algorithm; Step 1.4 calculate similarity between two opinions for criterion.
Criterion-Level	<i>Input: the similarity at the assessment level and weight <math>w_{c_j}</math> of criterion <math>c_j, j \in J</math></i> <i>Output: similarity with respect to criterion <math>c_j</math> against the criteria set <math>c_j, j \in J</math></i> <hr/> Step 2.1 identify a similarity utility function $u_j$ of criterion $c_j$ for each $j \in J$ ; Step 2.2 calculate similarity with respect to criterion $c_j$ by $u_j$ .
Problem-Level	<i>Input: similarities obtained at the criterion level</i> <i>Output: similarity between two opinions</i> <hr/> Step 3.1 construct the GAA from a pair of aggregation operators; Step 3.2 calculate the similarity between opinions using the GAA.

At the Criterion-Level, the differences in the weights of evaluation criteria are taken into account. The TLSM method defines for each criterion a similarity utility function based on its weight against those of other criteria. A similarity utility function meets two requirements: 1) it is proportional to similarity obtained at the Assessment-Level; and 2) it is proportional inversely to the weight for the same similarity at the Assessment-Level. The two requirements reflect the demand in practice that requirements on similarity measures of more important criteria are stricter than those of less important criteria. Based on these similarity utility functions, it is known that to what extent the two participants have similar opinions on a given criterion against a set of criteria.

At the Problem-Level, the similarity is measured using the GAA developed in Section 3. The GAA takes the similarities obtained at the Criterion-Level as inputs and re-orders them according to the decreasing-ordered weights of the corresponding criteria. The aggregation algorithm will generate a set of candidate values of the overall similarity of two participants' opinions at the first stage, and then derives the overall similarity from them at the second stage. The overall similarity obtained indicates to what extent the two participants have similar opinions from the viewpoint of a decision problem.

The details of the TLSM method are described in the following sections.

#### 4.3. Measuring similarity at the Assessment-Level

The main task at this level is to segment the term set  $T_j$  of a given criterion  $c_j$  into several semantic-equal groups. To do so, the TLSM method uses pair-wise comparison on

the semantics of each pair of terms in  $T_j$  to obtain a similarity matrix; then applies a clustering algorithm, such as the Hierarchical Clustering for Fuzzy Similarity Matrix (HCFSM) [52], to the similarity matrix to generate semantic-equal groups.

Each element of the similarity matrix is the similarity between a pair of terms in  $T_j$  obtained by direct comparison. For a given criterion  $c_j$ , the similarity matrix  $S_j$  for the terms in  $T_j$  is denoted by

$$S_j = \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1p_j} \\ s_{21} & s_{22} & \cdots & s_{2p_j} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p_j 1} & s_{p_j 2} & \cdots & s_{p_j p_j} \end{pmatrix}, \quad (10)$$

where

- $s_{pr} \in [0, 1]$  for any  $p, r \in \{1, \dots, p_j\}$ ;
- $s_{rr} = 1$  for any  $r \in \{1, \dots, p_j\}$ ;
- $s_{pr} = s_{rp}$  for any  $p, r \in \{1, \dots, p_j\}$ .

Pair-wise comparison is used here for some practical considerations. First, linguistic terms are often represented by fuzzy sets or fuzzy numbers. The semantic interpretation of these terms varies person to person and case by case. Pair-wise comparison can avoid difficulties in the course of defining a term's semantics. Secondly, some linguistic terms are incomparable, such as colour. It is hard to define an appropriate and rational similarity measurement for these types of terms. Thirdly, similarity between some terms may be changeable. In one context, two terms may be distinguishable; however, in the other context, they are identical. Pair-wise comparison has been

proved an effective strategy to analyse relationships between a set of factors; for instance, the Analytic Hierarchy Process (AHP) technique extensively uses pair-wise comparison to obtain local-priority and global-priority. Using it can better fit an application's specific setting and avoid potential heavy and complicated calculations. Of course, we do not reject other methods to determine the semantic similarity matrix.

After obtaining the similarity matrix, the TLSM method will segment the term set by a clustering algorithm based on it. There are lots of clustering algorithms for this purpose. However, noting that the total number of terms in the term set is often between 5 and 9, i.e., it is relatively small-size, the TLSM method uses the HCFSM to implement segmenting:

- derive the transitive closure  $\hat{S}_j$  from  $S_j$  by

$$\hat{S}_j = S_j \cup S_j^2 \cup S_j^4 \cup \dots,$$

where  $S_j^{2^k}$  is the max-min composition of  $S_j^k$  (see Appendix B for more details of the calculation of  $\hat{S}_j$  and an illustrative example);

- decompose  $\hat{S}_j$  into a set of  $\alpha$ -level equivalence class  $(\hat{S}_j)_\alpha$  by

$$\hat{S}_j = \bigcup_{\alpha \in [0,1]} \alpha(\hat{S}_j)_\alpha;$$

and

- terms in  $T_j$  whose similarities belong to the same  $(\hat{S}_j)_\alpha$  form a semantic-equal term group  $TG_j^\alpha$  and are treated with similar semantic.

After segmenting  $T_j$ , the opinions of two participants on criterion  $c_j$  are compared option by option. Based on the comparison result, a similarity can be defined according to the number of options on which the two opinions are treated similarly and the total number of candidate options; as a simple illustrative example, the TLSM let the similarity be the ratio of them.

The segmentation of  $T_j$  is not unique. It is influenced by the parameter  $\alpha$ , i.e., for different  $\alpha$ , the semantic-equal groups may not be identical. The adjustable parameter  $\alpha$  meets the real demands in applications where different parameters should be used for different criteria.

The example below illustrates the processes in this step.

**Example 4.1.** Suppose the similarity matrix between terms in an assessment set  $T$  is given by

$$S = \begin{pmatrix} 1.00 & 0.89 & 0.14 & 0.29 & 0.15 & 0.34 & 0.09 \\ 0.89 & 1.00 & 0.04 & 0.24 & 0.23 & 0.55 & 0.87 \\ 0.14 & 0.04 & 1.00 & 0.09 & 0.34 & 0.20 & 0.80 \\ 0.29 & 0.24 & 0.09 & 1.00 & 0.16 & 0.08 & 0.15 \\ 0.15 & 0.23 & 0.34 & 0.16 & 1.00 & 0.31 & 0.04 \\ 0.34 & 0.55 & 0.20 & 0.08 & 0.31 & 1.00 & 0.33 \\ 0.09 & 0.87 & 0.80 & 0.15 & 0.04 & 0.33 & 1.00 \end{pmatrix},$$

By the HCFSM, the first step is to obtain the transitive closure  $\hat{S}$  of  $S$ , which is

$$\hat{S} = \begin{pmatrix} 1.0 & 0.89 & 0.8 & 0.29 & 0.34 & 0.55 & 0.87 \\ 0.89 & 1.0 & 0.8 & 0.29 & 0.34 & 0.55 & 0.87 \\ 0.8 & 0.8 & 1.0 & 0.29 & 0.34 & 0.55 & 0.8 \\ 0.29 & 0.29 & 0.29 & 1.0 & 0.29 & 0.29 & 0.29 \\ 0.34 & 0.34 & 0.34 & 0.29 & 1.0 & 0.34 & 0.34 \\ 0.55 & 0.55 & 0.55 & 0.29 & 0.34 & 1.0 & 0.55 \\ 0.87 & 0.87 & 0.8 & 0.29 & 0.34 & 0.55 & 1.0 \end{pmatrix}.$$

Based on  $\hat{S}$ , a dendrogram is then obtained as shown in Figure 3. The dendrogram indicates that there are seven possible segmentation results. The 1.0 level is the strictest segmentation, by which no two terms are treated as similar, except that they are identical. The 0.87 level is looser than the 1.0 and 0.89 levels, by which the terms  $t_1$ ,  $t_2$  and  $t_7$  can be treated as similar.

#### 4.4. Measuring similarity at the Criterion-Level

The main task in this step is identifying an appropriate similarity utility function for each criterion. To achieve this goal, two requirements are used to design a similarity utility function: 1) the function is proportional to similarity at the Assessment-Level (PSA); and 2) it is proportional inversely to the weight of a criterion (PRW).

Formally, a similarity utility function is defined below.

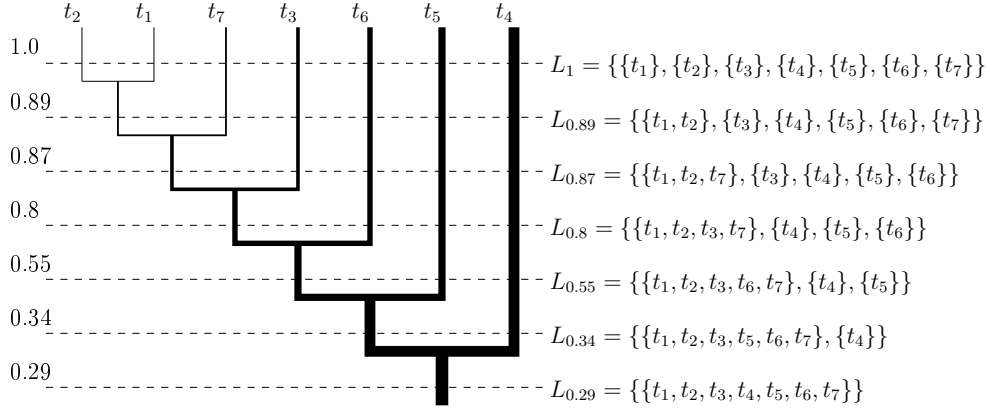
**Definition 4.1.** A similarity utility function  $u(nsp, w)$  of a given criterion  $c$  is a mapping from  $\mathbb{N} \times W$  to  $[0, 1]$  if  $u$  satisfies the PSA and PRW requirements, where  $\mathbb{N}$  is the set of natural numbers and  $W$  is the range of weight.

Functions satisfying Definition 4.1 are numerous. For simplicity, this study uses the following monotone and continuous function to illustrate the TLSM method:

$$u_j(nsp_j, wc_j) = \left( \frac{nsp_j}{n} \right)^{f(wc_j)} \quad (11)$$

where  $nsp_j$  is the number of options on which two opinions are treated as similar,  $n$  is the total number of candidate options, and  $f(wc_j)$  is a parameter determined by



Figure 3: The dendrogram obtained by the HCFSM on  $S$ .

$wc_j$ . To finalize the similarity utility function described in Eq. (11), it needs to determine parameter  $f(wc_j)$ .

Because non-negative real numbers and linguistic terms are commonly used as weights of criteria in an MCGDM problem, we will illustrate how to finalize a similarity utility function for the two requirements.

#### 4.4.1. Weights are non-negative real numbers

Suppose  $wc_1, \dots, wc_m$  is a set of normalized weights and  $wc_j \geq 0$ ,  $\sum_{j=1}^m wc_j = 1$ ,  $m = |C|$ . Without loss of generality, suppose  $wc_1 \leq wc_2 \leq \dots \leq wc_m$ . In this situation, we determine the parameter  $f(wc_j)$  as follows:

- determine a reference value  $wc_{j_0}$  and set  $f(wc_{j_0}) = 1$ ;
- for each  $wc_j$ , set  $f(wc_j) = wc_j/wc_{j_0}$ .

To find a  $wc_{j_0}$  from  $wc_1, \dots, wc_m$ , the following illustrative method is used: if  $m$  is odd, then set  $wc_{j_0} = wc_{(m+1)/2}$ ; if  $m$  is even, then set  $wc_{j_0} = (wc_{m/2} + wc_{m/2+1})/2$ . Based on this  $wc_{j_0}$ , all  $wc_j$ s are then mapped to  $[0, \infty)$  by

$$f(wc_{j_0}) = 1, f(wc_j) = \frac{wc_j}{wc_{j_0}}, \quad j = 1, \dots, m. \quad (12)$$

The  $f$  used in Eq. (12) is just used for illustration purpose. In fact, they can be in other forms in real applications accordingly.

To summarize the above process, let us consider a numeric example. Suppose seven criteria are considered and their weights are shown in column 2 in Table 3. Under this setting, the weights are listed in an increasing order as outlined below:

$$0.01 \leq 0.03 \leq 0.08 \leq 0.09 \leq 0.15 \leq 0.31 \leq 0.33$$

Because 7 (the number of criteria weights) is an odd number, the  $wc_{j_0}$  is therefore set as 0.09. Then let  $f$  be of form shown in Eq. (12), the parameters  $f(wc_j)$ s of the similarity utility function for the seven criteria can be obtained as shown in column 3 in Table 3.

Table 3: Weights and their corresponding parameter  $f(wc_j)$  of criteria.

criteria index $j$	weight $wc_j$	parameter $f(wc_j)$
1	0.09	1.00
2	0.01	0.11
3	0.03	0.33
4	0.31	3.44
5	0.08	0.89
6	0.33	3.67
7	0.15	1.67

#### 4.4.2. Weights are linguistic terms

Linguistic weights are often represented by fuzzy numbers (or fuzzy sets). Specific numeric features of a fuzzy number (set), such as its centre of gravity (COG) or its generalized integral, can be used to determine the parameter  $f(wc_j)$ . A brief outline for determining this parameter is given below.

- select a numeric feature  $NF$  of fuzzy numbers and calculate  $NF_j$  of the linguistic term (i.e., a fuzzy number)  $wc_j$ ;
- determine  $f(NF_j)$  following steps for  $f(wc_j)$  in Section 4.4.1;
- set  $f(wc_j) = f(NF_j)$  in Eq. (12).

Following this outline, let us consider an illustrative example. Suppose the linguistic weights are ‘‘Very High (VH)’’, ‘‘Fairly High (FH)’’, ‘‘Medium (M)’’, ‘‘Rather Low (RL)’’, and ‘‘Very Low (VL)’’; and their corresponding

fuzzy numbers are shown in Figure 4(b) and the selected numeric feature is the horizontal coordinate of COG of a fuzzy number, i.e.,

$$NF_j = \frac{\int x\mu(x)dx}{\int \mu(x)dx} \quad (13)$$

where  $\mu(x)$  is the membership function of the fuzzy number. By Eq. (13), the numeric features of these linguistic weights are calculated as shown in Table 4.

Without loss of generality, we calculate the numeric feature of the linguistic weight “Medium (M)” as follows. The membership function of “Medium” is

$$\mu_M(x) = \begin{cases} 0, & 0 \leq x < 0.2 \\ (x - 0.2)/(0.5 - 0.2), & 0.2 \leq x < 0.5 \\ (x - 0.8)/(0.5 - 0.8), & 0.5 \leq x \leq 0.8 \\ 0, & 0.8 < x \leq 1.0. \end{cases}$$

Hence,  $NF_M$  is calculated as

$$\begin{aligned} NF_M &= \frac{\int_0^1 x\mu_M(x)dx}{\int_0^1 \mu_M(x)dx} \\ &= \frac{\int_{0.2}^{0.5} x \frac{(x-0.2)}{0.5-0.2} dx + \int_{0.5}^{0.8} x \frac{(x-0.8)}{0.5-0.8} dx}{\int_{0.2}^{0.5} \frac{(x-0.2)}{0.5-0.2} dx + \int_{0.5}^{0.8} \frac{(x-0.8)}{0.5-0.8} dx} \\ &= \frac{0.06 + 0.09}{0.15 + 0.15} = 0.5 \end{aligned}$$

Following steps in Section 4.4.1, the  $f(NF_j)$  is calculated and shown in Table 4. Replacing the  $f(wc_j)$  in Eq. (12) by  $f(NF_j)$ , we obtain the similarity utility functions for the five linguistic weights, which can then be applied to calculate the similarity at the criterion level.

Table 4: Linguistic weight, numeric feature, parameter of similarity utility function of criteria.

$wc_j$	VH	FH	M	RL	VL
$NF$	0.9	0.767	0.5	0.233	0.1
$f(NF)$	1.800	1.534	1	0.466	0.200

After determining the similarity utility function for each given criterion, we apply them to measure the similarity of the opinions of two participants at the Criterion-Level. Suppose a referential criterion is weighted “FH” and the evaluations of two participants are treated similarly for seven options against a total nine options, then the similarity of the opinions of two participants with respect to this criterion is 0.680 ( $= (7/9)^{1.534}$ ) by Eq. (11).

#### 4.5. Measuring similarity at the Problem-Level

Section 4.4 details how to measure similarity of two opinions about each individual criterion from the viewpoint of a set of criteria. An individual criterion provides a single perspective by which we observe the similarity of two opinions. A set of criteria provides multiple observations. The main task in this step is to integrate those observations to form a comprehensive one. We will use the GAA developed in Section 3 to generate the comprehensive similarity.

The following two examples illustrate how to use the GAA. Suppose the similarities with respect to 10 criteria are obtained at the criterion level, which are those in the second column of Table 1.

**Example 4.1.** This example illustrates the usage of OGA. Assume that both  $\mathcal{A}$  and  $\mathcal{B}$  are the arithmetic mean. For the 10 inputs, the GAA first generates 10 candidate similarities for the final similarity  $\bar{s}$  by using  $A_i$ , where  $i = 1, \dots, 10$ :

$$\begin{aligned} \bar{s}_1 &= 0.840, & \bar{s}_2 &= 0.812, & \bar{s}_3 &= 0.845, & \bar{s}_4 &= 0.718, \\ \bar{s}_5 &= 0.630, & \bar{s}_6 &= 0.604, & \bar{s}_7 &= 0.570, & \bar{s}_8 &= 0.617, \\ \bar{s}_9 &= 0.619, & \bar{s}_{10} &= 0.572. \end{aligned}$$

Then the GAA applies  $B_{10}$  to the 10 candidate similarities  $\bar{s}_1, \dots, \bar{s}_{10}$  to generate  $\bar{s}$  which is  $\bar{s} = 0.683$ , i.e., the similarity of the two experts’ opinions is 0.683.

**Example 4.2.** This example illustrates the usage of WGA. Assume that  $\mathcal{A}$  is the OWA aggregation (see Appendix A for a brief definition, more details about OWA please refer [41]) and  $\mathcal{B}$  is the arithmetic mean. Because an OWA aggregation needs the weights of inputs, we randomly generate 10 weights for them as:

$$\begin{aligned} w_1 &= 0.394, & w_2 &= 0.798, & w_3 &= 0.198, & w_4 &= 0.768, \\ w_5 &= 0.554, & w_6 &= 0.629, & w_7 &= 0.513, & w_8 &= 0.916, \\ w_9 &= 0.717, & w_{10} &= 0.607. \end{aligned}$$

Then, GAA calculates the candidate values of  $\bar{s}_i$ s following OWA:

$$\begin{aligned} \bar{s}_1 &= 0.952, & \bar{s}_2 &= 0.925, & \bar{s}_3 &= 0.913, & \bar{s}_4 &= 0.866, \\ \bar{s}_5 &= 0.819, & \bar{s}_6 &= 0.755, & \bar{s}_7 &= 0.703, & \bar{s}_8 &= 0.632, \\ \bar{s}_9 &= 0.586, & \bar{s}_{10} &= 0.541. \end{aligned}$$

Finally, GAA applies the  $B_{10}$  to  $\bar{s}_1, \dots, \bar{s}_{10}$  to find the overall similarity, which is 0.769.

Based on the similarity measurement at the three levels, an overall similarity between the opinions of two participants is generated, which can be used as the answer of the MOSP problem. In the next section, we will apply the TLSM method to two real-world problems.

## 5. Applications in policy selection and evaluation

This section applies the TLSM method to an social policy selection application and an energy policy evaluation application.

### 5.1. Case 1: Do similarities exist between social actors?

This example is quoted from [2, 52]. In a social policy selection problem, six social actors (i.e., participants) have presented their assessments for seven possible policies (i.e., options). The social impact matrix (i.e., evaluation report) is given in Table 5 and the semantics of the used linguistic assessments are given in Figure 4(a). The problem is to answer whether or not similarities exist between these social actors.

Firstly, we recited the solution in [52] as a comparison with the TLSM method. The Munda's method includes three main steps.

- Generate a similarity matrix between the social actors by a similarity measurement  $s(b_i, b_j)$  of linguistic assessments:

$$\frac{1}{1 + \left[ \sum_{k=1}^7 \left( \iint_{x,y} |x - y| f_k(x) g_k(y) dy dx \right)^2 \right]^{1/2}}$$

where  $\mu_1(x)$  and  $\mu_2(y)$  are membership functions of two linguistic terms (as assessments)  $x$  and  $y$ , respectively; and  $\iint_{x,y} |x - y| f(x) g(y) dy dx$  is the semantic distance between  $x$  and  $y$ .

By this measurement, the similarity matrix  $S$  of the six social actors is obtained, and is presented in Table 6.

- Generate hierarchical clustering. By the HCFSM clustering algorithm, a dendrogram is given in Figure 5(a).
- Analyze clustering result. By the clustering result, the social actors  $b_1$  and  $b_2$  have higher similarity.

We now apply the presented TLSM method to resolve this problem. For convenience, we take the social actors  $b_1$  and  $b_4$  as examples to illustrate the experiment. Moreover, because the problem setting does not provide any

Table 6: Similarity matrix between six social actors.

	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$
$b_1$	1	0.729	0.426	0.399	0.403	0.403
$b_2$	0.729	1	0.410	0.386	0.390	0.390
$b_3$	0.426	0.410	1	0.675	0.584	0.569
$b_4$	0.399	0.386	0.675	1	0.729	0.672
$b_5$	0.403	0.390	0.584	0.729	1	0.595
$b_6$	0.403	0.390	0.569	0.672	0.595	1

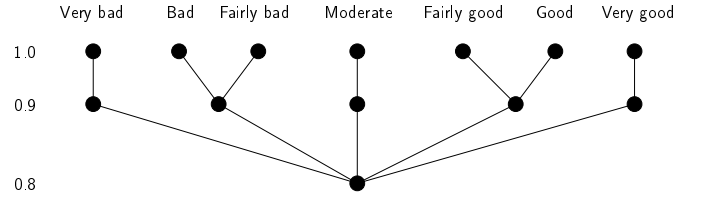


Figure 6: Dendrogram of linguistic assessments (terms).

information about evaluation criteria, we can assume that it only concerns one criterion.

Step 1: Measuring similarity at the Assessment-Level.

Firstly, we use the following distance measure between two terms  $t_i$  and  $t_j$  to obtain the similarity matrix of all linguistic assessments:

$$d(t_i, t_j) = |x_i - x_j|, \quad (14)$$

where  $x_i$  and  $x_j$  are the points whose membership degrees are equal to 1 with respect to  $t_i$  and  $t_j$  respectively. Based on this distance, the similarity between  $t_i$  and  $t_j$  is defined by

$$s_{ij} = 1 - d(t_i, t_j). \quad (15)$$

Therefore the similarity matrix for linguistic assessments is obtained and shown in Table 7. Hence, the dendrogram for the seven linguistic assessments by the HCFSM clustering algorithm is obtained and presented in Figure 6.

We next take 0.9-level equivalence-class in Figure 6 to segment the seven terms and compare the evaluations from actors  $b_1$  and  $b_4$ . It is noted that these two social actors have a similar opinion on policy  $a_5$  only. Table 8 lists the number of options on which participants have similar opinions by pair-wise comparison.

Step 2: Measuring similarity at the Criterion-Level. Because this problem involves only one criterion, it is enough to determine a unique parameter  $f(wc)$ .

For simplicity, suppose the similarity utility function used is of the form in Eq. (11). Setting  $f(wc)$  to be less than, equal to, or greater than 1.0 obtains three typical utilities of a criterion. The three utilities are illustrated below respectively.

The first situation is setting  $f(wc) = 1$ . The similarity utility function is, therefore, a linear function, by which

Table 5: An illustrative example of social impact matrix

Social actors	Policy options						
	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$
$b_1$	Very good	Good	Moderate	bad	Fairly good	Fairly bad	Very bad
$b_2$	Very good	Good	Moderate	Bad	Fairly good	Very bad	Very bad
$b_3$	Very bad	Fairly bad	Moderate	Good	Very good	Good	Moderate
$b_4$	Very bad	Fairly bad	Fairly bad	Good	Fairly good	Good	Very good
$b_5$	Very bad	Bad	Fairly bad	Moderate	Fairly good	Good	Very good
$b_6$	Very bad	Good	Bad	Good	Good	Good	Very good

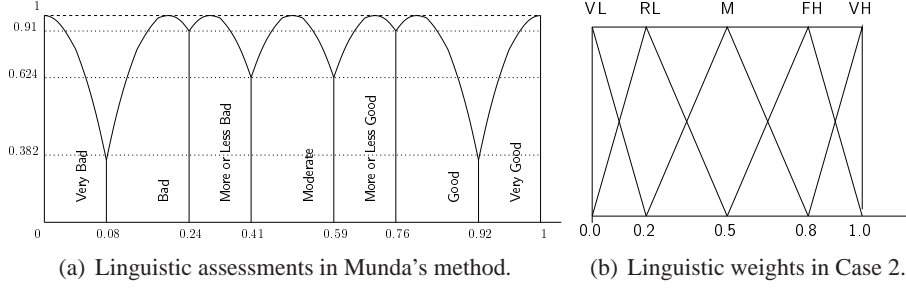


Figure 4: Semantic of linguistic terms.

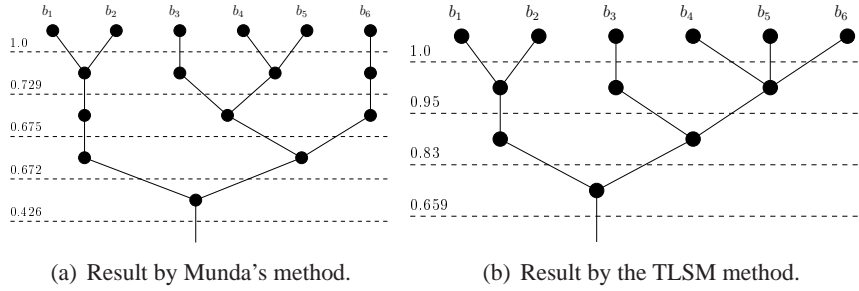


Figure 5: Dendrogram of similarities between experts.

Table 8: Number of options with similar opinions by pairwise comparison.

$nsp$	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$
$b_1$	7	6	1	1	1	2
$b_2$	6	7	1	1	1	2
$b_3$	1	1	7	4	3	3
$b_4$	1	1	4	7	6	6
$b_5$	1	1	3	6	7	5
$b_6$	2	2	3	6	5	7

Table 9: Pair-wise comparison of similarity at the Criterion-Level ( $f(wc) = 1$ ).

$f(n(v))$	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$
$b_1$	1	0.857	0.143	0.143	0.143	0.286
$b_2$	0.857	1	0.143	0.143	0.143	0.286
$b_3$	0.143	0.143	1	0.571	0.429	0.429
$b_4$	0.143	0.143	0.571	1	0.857	0.857
$b_5$	0.143	0.143	0.429	0.857	1	0.714
$b_6$	0.286	0.286	0.429	0.857	0.714	1

the similarity between  $b_1$  and  $b_4$  is 0.143. Table 9 illustrates the pair-wise similarity of all actors under this setting.

The second situation is setting  $f(wc) > 1$ . The obtained similarity utility function increases slowly with a smaller similarity at the Assessment-Level and then increases quickly with a larger one. Suppose  $f(wc) = 2$ , then the pair-wise similarities of the six actors are shown

in Table 10.

The third situation is  $f(wc) < 1$ . Under this setting, the obtained similarity utility function increases quickly with a smaller similarity at the Assessment-Level and then increases slowly with a bigger one. When setting  $f(wc) = 1/3$ , the pair-wise similarities are shown in Table 11.

Based on the identified similarity utility function, the

Table 7: Similarity matrix for linguistic assessments.

Term	Very bad	Bad	Fairly bad	Moderate	Fairly good	good	Very good
Very bad	1.0	0.8	0.7	0.5	0.3	0.2	0.0
Bad	0.8	1.0	0.9	0.7	0.5	0.4	0.2
Fairly bad	0.7	0.9	1.0	0.8	0.6	0.5	0.3
Moderate	0.5	0.7	0.8	1.0	0.8	0.7	0.5
Fairly good	0.3	0.5	0.6	0.8	1.0	0.9	0.7
good	0.2	0.4	0.5	0.7	0.9	1.0	0.8
Very good	0.0	0.2	0.3	0.5	0.7	0.8	1.0

Table 10: Pairwise comparison of similarity at the Criterion-Level ( $f(wc) = 2$ ).

$f(n(v))$	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$
$b_1$	1	0.735	0.020	0.020	0.020	0.082
$b_2$	0.735	1	0.020	0.020	0.020	0.082
$b_3$	0.020	0.020	1	0.327	0.184	0.184
$b_4$	0.020	0.020	0.327	1	0.735	0.735
$b_5$	0.020	0.020	0.184	0.735	1	0.510
$b_6$	0.082	0.082	0.184	0.735	0.510	1

Table 11: Pairwise comparison of similarity at the Criterion-Level ( $\alpha = 1/3$ ).

$f(wc)$	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$	$b_6$
$b_1$	1	0.950	0.523	0.523	0.523	0.659
$b_2$	0.950	1	0.523	0.523	0.523	0.659
$b_3$	0.523	0.523	1	0.830	0.754	0.754
$b_4$	0.523	0.523	0.830	1	0.950	0.950
$b_5$	0.523	0.523	0.754	0.950	1	0.894
$b_6$	0.659	0.659	0.754	0.950	0.894	1

similarity between  $b_1$  and  $b_4$  is obtained at the Criterion-Level.

Step 3: Measuring similarity at the Problem-Level. Because the example only involves a unique criterion, this step is redundant, i.e., the similarity at the Criterion-Level can be used at the subject level. Therefore, the similarity between  $b_1$  and  $b_4$  has already been obtained, i.e. 0.020.

Noting that Table 11 is a similarity matrix of the six social actors, we can use the HCFSM to obtain a similar dendrogram (Figure 5(b)). Comparing these two dendrograms, we recognized two minor differences: 1) social actor  $b_6$  will join the group of  $b_4$  and  $b_5$  earlier than social actor  $b_3$ ; and 2) the parameter  $\alpha$  is slightly different.

## 5.2. Case 2: energy policy selection for sustainable development with missing assessments

A governmental consultant committee has designed some energy policies for a nation's sustainable development in the future. Three of them are sent to six domain experts for evaluation in terms of eight primary criteria. Each primary criterion is composed of a few sec-

ondary criteria and the total number of criteria really evaluated is 16. An expert's evaluation report includes two components: 1) the assessments on the importance of all primary criteria concerned for sustainable development; and 2) the assessments on the impacts of the three alternative policies on sustainable development according to those criteria. All assessments are selected from a set of provided linguistic terms, or left blank for "unavailable", or with a question mark for "uncertain assessments (unknown or unsure)". After collecting the evaluation reports from these experts, the committee wants to know which two experts have similar opinions.

Without loss of generality, this study assumes that the weights of those 16 evaluation criteria have been determined in advance and the only task is to measure the similarity of the six experts. To illustrate our process more clearly, let  $O_1, O_2, O_3$  be the three alternative policies;  $c_1, \dots, c_{16}$  be the 16 evaluation criteria; and  $e_1, \dots, e_6$  be the six experts. The collected evaluation reports (only the assessments section in a real report) are listed in Table 12. The linguistic terms used in Table 12 for weights of criteria and evaluations on policies are summarised in Table 13.

Table 13: Abbreviations and semantics of linguistic terms used in evaluation reports.

Abbreviation.	Names	Semantics
VH	Very high	(0.7, 1.0, 1.0)
FH	Fairly high	(0.5, 0.8, 1.0)
M	Medium	(0.2, 0.5, 0.8)
RL	Rather low	(0.0, 0.2, 0.5)
VL	Very low	(0.0, 0.0, 0.3)
AC	Almost certain	(0.7, 1.0, 1.0)
VL	Very likely	(0.5, 0.8, 1.0)
L	Likely	(0.2, 0.5, 0.8)
UL	Unlikely	(0.0, 0.2, 0.5)
HUL	Highly Unlikely	(0.0, 0.0, 0.3)
NA	No answer	

Noting that all weights and assessments of the six experts are expressed by linguistic terms, this study uses

Table 12: Evaluation reports of six experts

$c_i$	$w_i$	$O_1$	$O_2$	$O_3$	$O_1$	$O_2$	$O_3$	$O_1$	$O_2$	$O_3$	$O_1$	$O_2$	$O_3$	$O_1$	$O_2$	$O_3$	$O_1$	$O_2$	$O_3$
		Expert 1			Expert 2			Expert 3			Expert 4			Expert 5			Expert 6		
1	VH	UL	L	AC	VL	VL	L	HUL	L	VL	VL	L	UL	VL	UL	HUL	L	UL	HUL
2	FH	L	L	AC	UL	L	L	UL	UL	L	VL	L	VL	VL	UL	HUL	VL	L	UL
3	FH	UL	L	VL	UL	HUL	L	HUL	L	VL	AC	UL	VL	VL	UL	HUL	L	HUL	HUL
4	FH	HUL	VL	AC	UL	UL	L	HUL	UL	HUL	L	L	HUL	L	HUL	HUL	UL	UL	HUL
5	FH	L	L	VL	L	VL	L	UL	VL	VL	AC	L	UL	AC	L	HUL	VL	VL	L
6	FH	AC	VL	AC	VL	VL	UL	L	VL	AC	UL	UL	HUL	AC	UL	HUL	L	UL	HUL
7	FH	L	UL	VL	UL	HUL	L	HUL	L	VL	UL	HUL	HUL	UL	L	HUL	HUL	HUL	HUL
8	FH	VL	L	VL	AC	AC	AC	UL	VL	VL	AC	VL	L	AC	UL	HUL	AC	AC	AC
9	FH	AC	VL	L	AC	AC	AC	UL	VL	AC	VL	AC	L	AC	UL	HUL	AC	AC	AC
10	FH	L	UL	L	VL	L	L	VL	VL	UL	VL	L	AC	VL	UL	HUL	HUL	HUL	HUL
11	FH	UL	UL	?	L	L	VL	VL	VL	HUL	HUL	HUL	L	L	UL	HUL			
12	FH	HUL	UL	L	HUL	HUL	VL	AC	AC	L	AC	AC	VL	VL	UL	HUL	L	UL	HUL
13	VH							UL	VL	UL	VL	L	UL	AC	L	UL	L	HUL	HUL
14	VH	VL	VL	VL	VL	VL	VL		VL	UL	VL	VL	UL	VL	L	UL			
15	FH	UL	HUL	VL	HUL	HUL	UL	L	HUL	HUL			UL	VL	UL	HUL	VL	UL	HUL
16	FH	UL	UL	L	HUL	HUL	L	L	VL	L	UL	UL	HUL	VL	L	UL	L	UL	HUL

triangular normal fuzzy numbers to represent linguistic terms. The semantic definitions of those linguistic terms are shown in the fourth column in Table 13 and in Figure 4(b). Based on this pre-process, the TLSM method is applied to this case and detailed steps are illustrated below.

Step 1: Measuring similarity at the Assessment-Level. To determine a similarity matrix for assessment terms, this study uses the same method shown in case 1 to define similarity between linguistic terms. The obtained similarity matrix  $S$  is

$s_{ij}$	AC	VL	L	UL	HUL
AC	1.0	0.8	0.5	0.2	0.0
VL	0.8	1.0	0.7	0.4	0.2
L	0.5	0.7	1.0	0.7	0.5
UL	0.2	0.4	0.7	1.0	0.8
HUL	0.0	0.2	0.5	0.8	1.0

By applying the HCSFM algorithm to  $S$ , we obtain three possible segments :

segment level	Segments
1.0	{AC}, {VL}, {L}, {UL}, {HUL}
0.8	{AC, VL}, {L}, {UL, HUL}
0.7	{AC, VL, L, UL, HUL}

It is noted that the only two possible weights are used for the 16 criteria, i.e., “VH” and “FH”, the segments with 1.0-level is used for criteria with weight “VH”; and the segments with 0.8-level is used for criteria with weight “FH”. (Note. The segments with 0.7-level will not be used in this study because it lacks capability to distinguish different terms.) Therefore, we can compare experts’ opinions at the assessments level. The following illustration will take experts  $e_1$  and  $e_2$  as an example.

For criterion  $c_1$ : Because the weight of  $c_1$  is “VH”, two assessments are similar if and only if they are the same

one. Hence, the number of assessments with similar semantics between  $(UL, L, AC)$  (of  $e_1$ ) and  $(VL, VL, L)$  (of  $e_2$ ) about this criterion is 0.

For criterion  $c_2$ : Because the weight of  $c_2$  is “FH”, the assessment “AC” is treated the same as “VL”; so do “UL” and “HUL”. Hence, the number of assessments with similar semantics between  $(L, L, AC)$  (of  $e_1$ ) and  $(UL, L, L)$  (of  $e_2$ ) about this criterion is 1 because the two opinions have the same assessment on policy  $O_2$  only.

Similarly, we can compare these two experts on the remaining 14 criteria one by one. Table 14 lists the number of options with similar opinion for all 16 criteria.

Table 14: Number of options with similar opinion for 16 criteria with respect to  $e_1$  and  $e_2$ .

$c_i$	1	2	3	4	5	6	7	8
no. of similar ass.	0	1	1	1	1	2	1	2
$c_i$	9	10	11	12	13	14	15	16
no. of similar ass.	2	1	0	2	0	3	2	3

It is noted that criteria  $c_{11}$  and  $c_{13}$  are different from other criteria because of the missing or uncertain assessments. To deal with these missing assessments, this study treats them as dissimilar.

Step 2: Measuring similarity at the Criterion-Level. For simplicity, this study uses the similarity utility function defined in Eq. (11). The parameter  $f(wc_j)$  is determined by the same method as used in case 1. The numeric feature of these five linguistic terms are:

$$\begin{aligned}
 NF_{VH} &= 0.9, & NF_{FH} &= 0.767, & NF_M &= 0.5 \\
 NF_{RL} &= 0.233, & NF_{VL} &= 0.1.
 \end{aligned} \tag{16}$$

The study sets  $f(NF_M) = 1.0$  and calculates the param-

eters for the other four weights accordingly:

$$\begin{aligned} f(NF_{VH}) &= 0.9/0.5 = 1.8, & f(NF_{FH}) &= 1.534, \\ f(NF_{RL}) &= 0.466, & f(NF_{VL}) &= 0.2. \end{aligned} \quad (17)$$

Once similarity utility functions of all evaluation criteria are finalized, they can be used to obtain similarity at the Criterion-Level. For instance, consider the criteria  $c_1$  and  $c_6$ . The weight of  $c_1$  is ‘‘VH’’ and  $f(NF_{VH}) = 1.8$ ; hence the similarity with respect to  $c_1$  is 0.134. Because the weight of  $c_6$  is ‘‘FH’’ and the  $f(NF_{FH}) = 1.534$ , then the similarity with respect to  $c_6$  is 0.537. For the other 14 criteria, the calculation is similar. The similarities at the Criterion-Level between  $e_1$  and  $e_2$  are summarized below.

$$\begin{aligned} s_1 &= 0.000, & s_2 &= 0.185, & s_3 &= 0.185, & s_4 &= 0.185, \\ s_5 &= 0.185, & s_6 &= 0.537, & s_7 &= 0.185, & s_8 &= 0.537, \\ s_9 &= 0.537, & s_{10} &= 0.185, & s_{11} &= 0.000, & s_{12} &= 0.537, \\ s_{13} &= 0.000, & s_{14} &= 1, & s_{15} &= 0.537, & s_{16} &= 1. \end{aligned}$$

Step 3: Measuring similarity at the Problem-Level. The GAA is implemented as follows:

- re-order the criteria by their weights in descending order;
- set  $A_i$  to be the arithmetic mean,  $i = 1, \dots, 16$ ;
- set  $B_{16}$  to be the  $t$ -conorm maximum max.

To re-order the criteria, this study used the  $NF$  values obtained at the Criterion-Level as the ordering reference. Then following the order of criteria, the  $i$ -ary aggregation operator  $A_i$  is applied to those similarities to obtain possible similarities between the two experts:

$$0.000, 0.000, 0.333, 0.296, 0.274, 0.259, 0.249, 0.285, \\ 0.274, 0.300, 0.322, 0.310, 0.304, 0.320, 0.362, 0.362$$

From them the biggest is selected by  $B_{16}$ , which is 0.362. Therefore, the similarity between the experts  $e_1$  and  $e_2$  is 0.362.

Table 15 gives the pair-wise similarity of the six experts. Based on the pair-wise similarity measurement, the experts can be grouped again based on a clustering method. For instance, Figure 7 is the dendrogram that uses the HCFSM algorithm. Further observation indicates that experts  $e_4$ ,  $e_5$ , and  $e_6$  have higher similarities in their opinions.

Table 15: Pair-wise similarities of all six experts.

	$e_1$	$e_2$	$e_3$	$e_4$	$e_5$	$e_6$
$e_1$	1	0.362	0.273	0.289	0.108	0.151
$e_2$	0.362	1	0.275	0.277	0.189	0.379
$e_3$	0.273	0.275	1	0.253	0.199	0.239
$e_4$	0.289	0.277	0.253	1	0.493	0.337
$e_5$	0.108	0.189	0.199	0.493	1	0.482
$e_6$	0.151	0.379	0.239	0.337	0.482	1

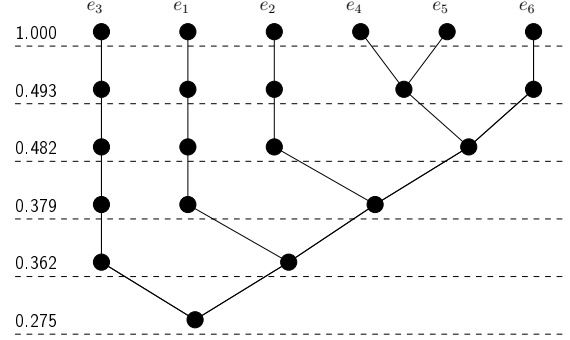


Figure 7: Dendrogram of experts using the HCFSM.

## 6. Conclusions and future works

MCGDM is an efficient strategy to support decision making in many applications. However, overly similar opinions of participants may lead to an inappropriate decision. To reduce the potential risk of putting an inappropriate decision into practice, measuring opinion similarity between participants (MOSP) is an important issue, which has not been solved. To solve the MOSP problem, our research develops a gradual aggregation algorithm to model the dynamic generation of a decision and to process the missing value in it. Based on the gradual aggregation algorithm, a three-level similarity measuring (TLSM) method for the MOSP problem is presented which measures the similarity between two opinions at the assessment level. Applying the TLSM method, two applications in social policy selection and energy policy evaluation are conducted.

The main contributions of this research are summarised below. Firstly, the TLSM method provides a processing framework for the MOSP problem. The MOSP problem is a significant but easily neglected practical topic in many applications. Existing opinion similarity measuring methods can tackle a part of the MOSP problem; however, they do not present a whole solution for it. Secondly, the small size of relevant opinion samples is a primary obstacle that prevents existing statistical learning techniques from being applied to the MOSP problem. The TLSM method can resolve these problem partially. Moreover, the TLSM

method combines an opinion with its provider in its entire processing. This helps to develop more effective opinion similarity measuring and analysis techniques to overcome difficulties resulting from separation of opinions and their providers in real applications. Finally, the experiments indicate that the TLSM method effectively handle missing data, unclear information, and linguistic assessments by adjusting the developed gradual aggregation algorithm. Highly satisfactory results have been obtained from the experiments.

Based on these two case-based experiments, some issues will be further studied. Firstly, the GAA is a novel technique to integrate information according to a group of inputs. The process order of the inputs has special meaning and impact on the final result. This study rearranges the inputs according to the descent order of the weights of criteria and a satisfactory result is obtained, however, the GAA is still need to amend. For example, we can extend it to and implement it at multiple levels. Secondly, missing data and unclear answers are very common in real applications. The TSLM method treats them as distinct without distinguishing their real meanings and utilities further. This is an intuitive and simple processing strategy. Whether there is a better strategy is a further area requiring investigation. Moreover, we will pay more attention on how to select a clustering algorithm for the TSLM method. For simplicity and illustrating purpose, this paper mainly used the HCFM method. Although the experiment results are consistent with our expectation, it is by no means that the HCFM is the best one. We recognised that selecting an appropriate clustering method should base on real applications. Thirdly, the MOSP problem is a special case of the user opinion analysis and behaviour modelling problem. Due to a variety in the natures of different application contexts, effective techniques for solving the user opinion analysis and behaviour modelling problem have not yet been found. Our next step is to extend the TLSM method and develop new techniques to provide applicable solutions for both the MOSP problem and the user opinion analysis and behaviour modelling problem. Finally, the application of the proposed TSLM method involves heavy computational burden for large size decision making problems, which requires to develop a corresponding decision support system. We currently implemented the presented method using the C++ and Java programming languages in a Linux distribution. We aim to amend and integrate the method into a decision support system which is being designed and developed.

## Appendix A: Ordered weighted aggregation (OWA)

The ordered weighted aggregation was presented by Yager in [41]. It becomes an important type of aggregation operators and is widely used in MCGDM research.

**Definition 6.1.** Let  $\text{Agg}$  be a mapping from  $\mathbb{R}^n$  to  $\mathbb{R}$ . Suppose  $a_1, a_2, \dots, a_n$  is a set of inputs of  $\text{Agg}$ , and  $w_1, w_2, \dots, w_n$  is a set of weights such that  $w_i \geq 0$  for  $i = 1, \dots, n$  and  $\sum_{i=1}^n w_i = 1$ . Then the output of  $\text{Agg}$  with respect to the inputs  $a_1, \dots, a_n$  is

$$\text{Agg}(a_1, \dots, a_n) = \sum_{i=1}^n w_i b_i, \quad (18)$$

where  $b_i$  is the  $i$ -th largest element in  $\{a_1, a_2, \dots, a_n\}$ .

From Definition 6.1, it is noted that the weight values are closely associated with the places where the inputs will be assigned to and an input's value determines which place it should go. Compared with the GGA in the presented work, the Yager's OWA aggregation orders the inputs; while the GGA potentially orders the places.

## Appendix B: Calculation of transitive closure $\hat{S}$

Suppose  $S$  is a similarity matrix and, for convenience, described as follows

$$S = S^1 = \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1p} \\ s_{21} & s_{22} & \cdots & s_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \cdots & s_{pp} \end{pmatrix}. \quad (19)$$

Then the max-min composition  $S^{2^k}$  of  $S^{2^{k-1}}$  is a similarity matrix

$$S^{2^k} = S^{2^{k-1}} \circ S^{2^{k-1}} = \begin{pmatrix} s_{11}^{(2^k)} & s_{12}^{(2^k)} & \cdots & s_{1p}^{(2^k)} \\ s_{21}^{(2^k)} & s_{22}^{(2^k)} & \cdots & s_{2p}^{(2^k)} \\ \vdots & \vdots & \cdots & \vdots \\ s_{p1}^{(2^k)} & s_{p2}^{(2^k)} & \cdots & s_{pp}^{(2^k)} \end{pmatrix}, \quad (20)$$

where  $s_{ij}^{(2^k)}$  is calculated as

$$s_{ij}^{(2^k)} = \max_{l=1}^p \min(s_{il}^{(2^{k-1})}, s_{lj}^{(2^{k-1})}) \quad (21)$$

where  $k = 0, 1, \dots, i, j = 1, \dots, p$ . Finally, the transitive closure  $\hat{S}$  of  $S$  is defined as

$$\hat{S} = S \cup S^2 \cup S^4 \cup S^8 \cup \dots \quad (22)$$



where  $\hat{s}_{ij} = \max(s_{ij}, s_{ij}^{(2)}, s_{ij}^{(4)}, s_{ij}^{(8)}, \dots)$ .

Reconsidering the similarity matrix  $S$  in Example 4.1, we will show how to calculate the composition  $S^2$  from  $S^1$ . Suppose  $i = 2$  and  $j = 4$ , then the element  $s_{24}^{(2)}$  in  $S^2$  is calculated by

$$\begin{aligned} s_{24}^{(2)} &= \max(\min(0.89, 0.29), \min(1.0, 0.24), \\ &\quad \min(0.04, 0.09), \min(0.24, 1.00), \min(0.23, 0.16), \\ &\quad \min(0.55, 0.08), \min(0.87, 0.15)) \\ &= \max(0.29, 0.24, 0.04, 0.24, 0.16, 0.08, 0.15) \\ &= 0.29 \end{aligned} \tag{23}$$

The other elements in  $S^2$  are calculated in the similar way. After obtaining the matrixes  $S, S^2, S^4, \dots$ , we can obtain the transitive closure  $\hat{S}$  by Eq. (22).

### Acknowledgements

This work is supported by Australian Research Council under Discovery Project DP0880739. We sincerely appreciate professor Da Ruan, who past away during the review procedure, from Belgian Nuclear Research Centre (SCK·CEN) for providing the experiment data. May he rest in peace in heaven!

We sincerely appreciate the anonymous reviewers for their suggestions and comments to improve the presented work.

### References

- [1] G. Kou, Y. Shi, S. Wang, Multiple criteria decision making and decision support system – guest editor’s introduction, *Decision Support Systems* 51 (2).
- [2] G. Munda, *Social Multi-Criteria Evaluation for a Sustainable Economy*, Springer-Verlag: Berlin Heidelberg, 2008.
- [3] E. Triantaphyllou, *Multi-Criteria Decision Making Methods: A Comparative Study*, Kluwer Academic Publishers, Dordrecht/Boston/London, 2000.
- [4] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, *IEEE Transactions on Knowledge and Data Engineering* 17 (6) (2005) 737–749.
- [5] J. Hakanen, K. Miettinen, K. Sahlstedt, Wastewater treatment: new insight provided by interactive multiobjective optimization, *Decision Support Systems* 51 (2011) 328–337.
- [6] P. Symeonidis, A. Nanopoulos, Y. Manolopoulos, A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis, *IEEE Transactions on Knowledge and Data Engineering* 22 (2) (2010) 179–192.
- [7] M. Vojnović, J. Cruise, D. Gunawardena, P. Marbach, Ranking and suggesting popular items, *IEEE Transactions on Knowledge and Data Engineering* 21 (8) (2009) 1133–1146.
- [8] L. Yu, K. K. Lai, A distance-based group decision-making methodology for multi-person multi-criteria emergency decision support, *Decision Support Systems* 51 (2011) 307–315.
- [9] F. Herrera, E. Herrera-Viedma, F. Chiclana, Multiperson decision-making based on multiplicative preference relations, *European J. Operational Research* 129 (2001) 372–385.
- [10] S. Banisch, T. Araújo, On the empirical relevance of the transient in opinion models, *Physics Letters A* 374 (2010) 3197–3200.
- [11] H. Sigall, E. Aronson, Opinion change and the gain-loss model of interpersonal attraction, *Journal of Experimental Social Psychology* 3 (1967) 178–188.
- [12] K. Ganesan, C. Zhai, J. Han, Opinosis: a graph based approach to abstractive summarization of highly redundant opinions, in: *Proceedings of the 23rd International Conference on Computational Linguistics*, Beijing, China, 2010, pp. 340–348.
- [13] B. Liu, Sentiment analysis: a multifaceted problem, *IEEE Intelligent Systems* 25 (3) (2010) 76–80.
- [14] B. Pang, L. Lee, Opinion mining and sentiment analysis, *Foundations and Trend in Information Retrieval* 2 (1-2) (2008) 1–135.
- [15] O. Vechtomova, Facet-based opinion retrieval from blogs, *Information Processing and Management* 46 (2010) 71–88.
- [16] W. Huang, Y. Zhao, S. Yang, Y. Lu, Analysis of the user behavior and opinion classification based on the BBS, *Applied Mathematics and Computation* 205 (2008) 668–676.
- [17] M. Hu, B. Liu, Mining and summarizing customer reviews, in: *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD 2004, ACM, New York, NY, USA, 2004, pp. 168–177.
- [18] Y. Wu, F. Wei, S. Liu, N. Au, W. Cui, H. Zhou, H. Qu, Opinion-See: interactive visualization of hotel customer feedback, *IEEE Transactions on Visualization and Computer Graphics* 16 (6) (2010) 1109–1118.
- [19] H. Chen, D. Zimbra, AI and opinion mining, *IEEE Intelligent Systems* 26 (2010) 74–76.
- [20] D. T. Miller, K. R. Morrison, Expressing deviant opinions: believing you are in the majority helps, *Journal of Experimental Social Psychology* 45 (2009) 740–747.
- [21] M. Pandelaere, B. Briers, S. Dewitte, L. Warlop, Better think before agreeing twice mere agreement: a similarity-based persuasion mechanism, *International Journal of Research in Marketing* 27 (2010) 133–141.
- [22] C. Chakraborty, D. Chakraborty, A fuzzy clustering methodology for linguistic opinions in group decision making, *Applied Soft Computing* 7 (2010) 858–869.
- [23] P. Meyer, M. Roubens, On the use of the Choquet integral with fuzzy numbers in multiple criteria decision support, *Fuzzy Sets and Systems* 157 (2006) 927–938.
- [24] G. Pasi, R. R. Yager, Modeling the concept of majority opinion in group decision making, *Information Sciences* 176 (2006) 390–414.
- [25] E. Herrera-Viedma, F. Herrera, L. Martínez, J. C. Herrera, A. G. López, Incorporating filtering techniques in a fuzzy linguistic multi-agent model for information gathering on the web, *Fuzzy Sets and Systems* 148 (2004) 61–83.
- [26] E. Herrera-Viedma, F. Chiclana, F. Herrera, S. Alonso, Group decision-making model with incomplete fuzzy preference relations based on additive consistency, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics* 37 (1) (2007) 176–189.
- [27] V.-N. Huynh, Y. Nakamori, A satisfactory-oriented approach to multiexpert decision-making with linguistic assessments, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cy-*

<sup>1</sup>The calculation of  $S^{2^k}$  from  $S^{2^{k-1}}$  is similar for  $k = 0, 1, \dots$

- bernetics 35 (2) (2005) 184–196.
- [28] J. Yu, J. Amores, N. Sebe, P. Radeva, Q. Tian, Distance learning for similarity estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (3) (2008) 451–462.
- [29] V. V. Cross, T. A. Sudkamp, *Similarity and Compatibility in Fuzzy Set Theory*, Physica-Verlag, Heidelberg, 2002.
- [30] J. Inglada, G. Mercier, A new statistical similarity measure for change detection in multitemporal SAR images and its extension to multiscale change analysis, *IEEE Transactions on Geoscience and Remote Sensing* 45 (5) (2007) 1432–1445.
- [31] E. Iosif, A. Potamianos, Unsupervised semantic similarity computation between terms using web documents, *IEEE Transactions on Knowledge and Data Engineering* 22 (11) (2010) 1637–1647.
- [32] A. Nanopoulos, D. Rafailidis, P. Symeonidis, Y. Manolopoulos, MusicBox: personalized music recommendation based on cubic analysis of social tags, *IEEE Transactions on Audio, Speech, and Language Processing* 18 (2) (2010) 407–412.
- [33] X. Wang, B. De Baets, E. E. Kerre, A comparative study of similarity measures, *Fuzzy Sets and Systems* 73 (1995) 259–268.
- [34] B. De Baets, H. De Meyer, H. Naessens, A class of rational cardinality-based similarity measures, *Journal of Computational and Applied Mathematics* 132 (2001) 51–69.
- [35] K. Bosteels, E. E. Kerre, A triparametric family of cardinality-based fuzzy similarity measures, *Fuzzy Sets and Systems* 158 (2007) 2466–2479.
- [36] B. De Baets, S. Janssens, H. De Meyer, On the transitivity of a parametric family of cardinality-based similarity measures, *International Journal of Approximate Reasoning* 50 (2009) 104–116.
- [37] V. Balopoulos, A. G. Hatzimichailidis, B. K. Papadopoulos, Distance and similarity measures for fuzzy operators, *Information Sciences* 177 (2007) 2336–2348.
- [38] B. Bouchon-Meunier, *Aggregation and Fusion of Imperfect Information*, Physica-Verlag, 1998.
- [39] H. Bustince, F. Herrera, J. Montero (Eds.), *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models*, Vol. 220 of *Studies in Fuzziness and Soft Computing*, Springer-Verlag, Berlin Heidelberg, 2008.
- [40] T. Calvo, A. Kolesárová, M. Komorníková, R. Mesiar, *Aggregation Operators: Properties, Classes and Construction Methods*, Physica-Verlag: Springer, Heidelberg, Germany, Germany, 2002, Ch. 1, pp. 3–104.
- [41] R. R. Yager, On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Transactions on Systems, Man, and Cybernetics* 18 (1) (1988) 183–190.
- [42] R. Mesiar, A. Kolesárová, T. Calvo, M. Komorníková, A review of aggregation functions, in: H. Bustince, F. Herrera, J. Montero (Eds.), *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models*, Vol. 220 of *Studies in Fuzziness and Soft Computing*, Springer-Verlag: Berlin Heidelberg, 2008, pp. 121–144.
- [43] R. Mesiar, J. Špirková, L. Vavříková, Weighted aggregation operators based on minimization, *Information Sciences* 178 (2008) 1133–1140.
- [44] R. R. Yager, OWA aggregation over a continuous interval argument with applications to decision making, *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics* 34 (5) (2004) 1952–1963.
- [45] R. R. Yager, Time series smoothing and OWA aggregation, *IEEE Transactions on Fuzzy Systems* 16 (4) (2008) 994–1007.
- [46] R. R. Yager, A. Rybalov, Uninorm aggregation operators, *Fuzzy Sets and Systems* 80 (1) (1996) 111–120.
- [47] J.-L. Marichal, An axiomatic approach of the discrete Choquet integral as a tool to aggregate interacting criteria, *IEEE Transactions on Fuzzy Systems* 8 (6) (2000) 800–807.
- [48] J.-L. Marichal, An axiomatic approach of the discrete Sugeno integral as a tool to aggregate interacting criteria in a qualitative framework, *IEEE Transactions on Fuzzy Systems* 9 (1) (2001) 164–172.
- [49] P. Meyer, M. Roubens, On the use of the Choquet integral with fuzzy numbers in multiple criteria decision aiding, in: *Proceedings of EUSFLAT-LFA 2005 conference*, 2005, pp. 210–215.
- [50] G. Beliakov, J. Warren, Appropriate choice of aggregation operators in fuzzy decision support systems, *IEEE Transactions on Fuzzy Systems* 9 (6) (2001) 773–784.
- [51] Q. Zhang, J. C. Chen, P. P. Chong, Decision consolidation: criteria weight determination using multiple preference formats, *Decision Support Systems* 38 (2004) 247–258.
- [52] G. Munda, A conflict analysis approach for illuminating distributional issues in sustainability policy, *European Journal of Operational Research* 194 (1) (2009) 307–322.