Knowledge-Based Systems 58 (2014) 66–74

Contents lists available at SciVerse ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

MENTOR: A graphical monitoring tool of preferences evolution in large-scale group decision making

Iván Palomares [a,*], Luis Martínez [a], Francisco Herrera [b,c]

[a] Department of Computer Science, University of Jaén, 23071 Jaén, Spain
[b] Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain
[c] Faculty of Computing and Information Technology - North Jeddah, King Abdulaziz University, 21589 Jeddah, Saudi Arabia

ARTICLE INFO

Article history:
Available online 16 July 2013

Keywords:
Group decision making
Preferences
Consensus reaching processes
Self-Organizing Maps
Group Decision Support Systems

ABSTRACT

Group decision making problems aim to manage situations in which two or more experts need to achieve a common solution to a decision problem. Different rules and processes can be applied to solve such problems (e.g. majority rule, consensus reaching, and so on), and several models have been proposed to deal with them. Some difficulties may arise in group decisions, being most of them caused by the presence of disagreement positions amongst experts. Given that group decision making problems have classically focused on a few number of experts, such difficulties have been relatively manageable by means of supporting tools based on textual or numerical information. However, such tools are not adequate when a large number of experts take part in the problem, therefore an alternate tool that provides decision makers with more easily interpretable information about the status of the problem becomes necessary. This paper proposes a graphical monitoring tool based on Self-Organizing Maps so-called MENTOR, that provides a 2-D graphical interface whose information is related to experts' preferences and their evolution during group decision making problems, and facilitates the analysis of information about large-scale problems.

1. Introduction

Decision Making is a common process in daily life. In Group Decision Making (GDM) problems, two or more individuals or *experts*, with their own attitudes and opinions, need to achieve a common solution to a decision problem consisting of several alternatives [1–3]. GDM problems are present in diverse application areas that require the participation of multiple experts, such as management and engineering and politics [4–6].

GDM problems can be solved by applying different processes, ranging from the use of classical decision rules (such as the majority or minority rule [7]), to the application of a consensus reaching process, which is a process of negotiation between experts, aimed to achieve a high level of agreement in the group before making a decision [8]. Consensus reaching processes are increasingly necessary in nowadays group decisions [9].

A large number of theoretical models and approaches to facilitate the resolution of GDM problems have been proposed in the literature [3,10–14]. Moreover, several authors have developed some computer-based Group Decision Support Systems (GDSS), to give groups further assistance in such problems [3,15]. Some of these GDSS make use of the Internet to allow groups to solve GDM problems ubiquitously [16,17].

Classically, GDM problems have been solved by a few number of experts. In these cases, when typical difficulties in group decisions arise (such as the presence of disagreement positions), they can be managed with the aid of GDSS and supporting tools that provide numerical or textual information about preferences of experts in the group [2,15,16]. Such tools could be often utilized with analytical purposes by a person who is responsible for making the final decision or *decision maker*. They can also be utilized by the *moderator* of a consensus reaching process [7,8].

However, new paradigms and ways of making group decisions, such as social networks [18] and e-democracy [5], have caused that decisions made by a larger number of experts become more frequent in recent years, therefore *large-scale GDM problems* are attaining greater importance. The resolution of large-scale GDM problems implies new challenges and requirements in terms of the higher cost and time invested to make the decision, and the increasing complexity of the problem. Additionally, in large-scale GDM problems, a considerable amount of information related to the preferences of experts must be managed, therefore a higher complexity appears in those analysis tasks that would be much more manageable in the case of dealing with small groups, for instance: (i) detecting conflicts amongst experts, (ii) determining the closeness between experts' opinions, (iii) identifying the

* Corresponding author. Tel.: +34 953213356.
E-mail addresses: ivanp@ujaen.es (I. Palomares), martin@ujaen.es (L. Martínez), herrera@decsai.ugr.es (F. Herrera).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.07.003

number and identity of experts that agree/disagree with each other, and (iv) finding coalitions or subgroups of interest in the group, etc. Most existing GDSS focus on GDM problems with few experts, in which numerical information about the status of the problem can be easily analyzed by a decision maker interested in it. However, in large-scale GDM problems the amount of information available may become much larger and, consequently, much more complicated to understand.

Different solutions can be proposed to support the previous analysis tasks [16]. In large-scale GDM, it would be particularly interesting to increase knowledge about the problem and make it more accessible to the decision maker interested in it, by means of a graphical 2-D tool that visualizes information about the whole group. In this sense, Self-Organizing Maps (SOMs) [19,20] have previously proved to be an effective means to visualize high dimensional data in a low-dimensional space [21,22]. Therefore, a graphical tool based on two-dimensional SOMs would facilitate the analysis and interpretation of diverse aspects of interest in large-scale GDM problems.

In this paper, we present a SOM-based graphical monitoring tool so-called MENTOR, that supports decision makers in the analysis of information about the status of large-scale GDM problems during their resolution. Such a tool facilitates the obtaining of important information about diverse features in these problems, such as the detection of agreement/disagreement positions within the group, the evolution of experts' preferences, or the level of closeness between experts' opinions achieved during consensus reaching processes in the cases they are carried out. MENTOR is also presented as a tool that can be integrated with different GDSS proposed in the literature, therefore it implies a important step towards the design of new, highly-interpretable GDSS.

The paper is structured as follows: in Section 2, some preliminaries about GDM and SOMs are reviewed. Section 3 presents MENTOR, the graphical monitoring tool based on SOMs, by explaining how it works and describing its main features for analysis and interpretation of graphical information about the GDM problem. Section 4 shows an example of application of MENTOR in a large-scale GDM problem. Finally, some concluding remarks are exposed in Section 5.

2. Preliminaries

Given the paper proposal of a SOM-based graphical monitoring tool to support large-scale GDM problems, in this section we review GDM problems, paying special attention to consensus reaching processes as a means for smoothing group conflicts and finding agreed solutions. Eventually, it is revised some elementary concepts about SOMs, which are the basis for graphical representation of information in the proposed tool.

2.1. Group decision making problems

The need for making decisions in which multiple experts with different viewpoints are involved, is frequent in many complex real-life decision situations and organizational structures. GDM problems, where a group of experts must make a common decision together, are normally utilized in such situations [2,3]. Some examples of application of GDM problems are: political and democratical systems, engineering, management, etc. [4–6].

Formally, GDM problems can be defined as decision situations characterized by the participation of two or more experts, with their own knowledge and attitudes, in a decision problem consisting of a set of alternatives or possible solutions to such a problem [1,3]. The following elements are found in any GDM problem:

- A set $X = \{x_1, \ldots, x_n\}$, $(n \geqslant 2)$ of *alternatives*.
- A set $E = \{e_1, \ldots, e_m\}$, $(m \geqslant 2)$ of *experts*, who express their judgements on the alternatives in X.

Each expert e_i, $i \in \{1, \ldots, m\}$, provides his/her opinions over alternatives in X by means of a preference structure. Some types of preference structures commonly utilized in GDM are: preference relations [23], utility vectors [24] and preference orderings [25]. Preference relations have been specially utilized in many models of GDM under uncertainty. They are defined as follows:

Definition 1 (*[23,26]*). A *preference relation* P_i associated to expert e_i, $i \in \{1, \ldots, m\}$, on a set of alternatives X is a fuzzy set on $X \times X$, represented by a $n \times n$ matrix of assessments $p_i^{lk} = \mu_{P_i}(x_l, x_k)$ as follows:

$$P_i = \begin{pmatrix} - & \ldots & p_i^{1n} \\ \vdots & \ddots & \vdots \\ p_i^{n1} & \ldots & - \end{pmatrix}$$

where each *assessment*, $p_i^{lk} = \mu_{P_i}(x_l, x_k)$, represents the preference degree of alternative x_l over x_k according to e_i. Assessments $p_i^{ll}, l \in \{1, \ldots, n\}$, situated in the diagonal of the matrix, are not defined, since an alternative x_l is not assessed with respect to itself.

Experts' assessments are expressed in a specific information domain. Some information domains widely used in GDM are: numerical, interval-valued and linguistic [11].

The solution to a GDM problem is obtained by using either a *direct approach*, where the solution is directly obtained from experts' preferences, or an *indirect approach*, in which a collective opinion is computed before determining the chosen alternative/s [27]. In both approaches, the selection process to solve GDM problems consists of two phases [28]: (i) an *aggregation phase*, where individual preferences are combined and (ii) an *exploitation phase*, where an alternative or subset of alternatives are obtained as the solution to the problem.

Despite different classic guiding rules, such as the majority rule and minority rule, have been suggested to carry out the selection process in GDM [7], they do not guarantee a high level of agreement amongst experts regarding the decision made: it is possible that some of them may not accept the solution chosen, because they might consider that their opinions have not been considered sufficiently [8]. In such cases that a more agreed decision is necessary, a *negotiation phase* should be introduced as part of the GDM problem resolution process to achieve a high degree of agreement among experts before making a decision. A variety of formal negotiation models based on different theoretical backgrounds can be found in the literature [29,30]. Nevertheless, in the research field of GDM we move in, it is usually applied a *consensus reaching process* to achieve a collective agreement before making a group decision [8]. Consensus has attained a great importance to reach more appreciated solutions in GDM problems, and it has become a major research topic in the last decades [14–17,31,32].

The process to reach a consensus is a dynamic and iterative discussion process, frequently coordinated by a human figure known as moderator [7,8]. A general scheme of consensus reaching process is shown Fig. 1. Its phases are briefly described below:

1. *Gathering preferences*: Each expert provides his/her preferences to the moderator.
2. *Computing the level of agreement*: The moderator determines the level of agreement in the group.
3. *Consensus control*: If the level of agreement is enough, the group moves onto the selection process, otherwise more discussion is required.

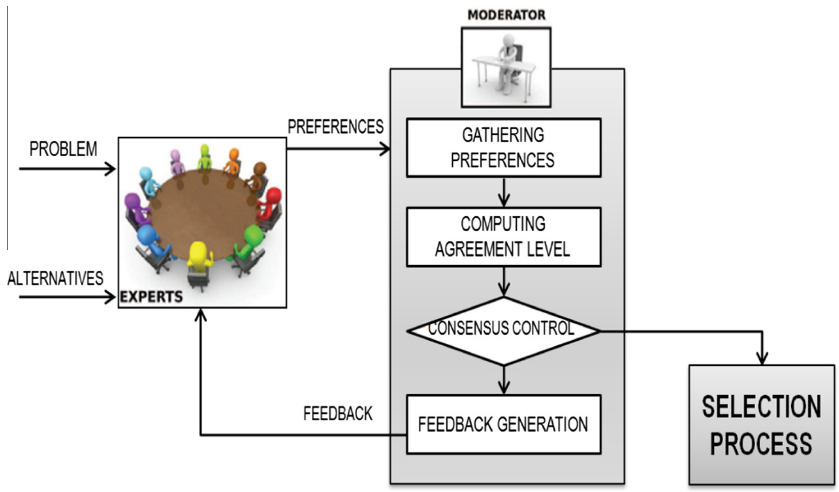

Fig. 1. General consensus reaching scheme in GDM.

4. *Feedback generation*: The moderator gives experts some feedback, suggesting them how to modify their preferences and make them closer to each other, to increase the level of agreement in the group.

In consensus-driven GDM problems, some crucial aspects that should be monitored during the consensus reaching process are the status of experts' preferences across the time, and the evolution of the level of agreement achieved in the group. Besides, in large-scale GDM problems, it is usual that some experts or subgroups of them disagree with each other on their opinions, they do not cooperate to reach a consensus or they try to deviate the collective opinion. A graphical tool that monitors these features both in GDM problems and in consensus reaching processes becomes then necessary, in order to analyze the positions of experts' preferences with respect to the group. The tool proposed in this paper is based on SOMs [19], therefore some basic concepts about this visualization technique will be reviewed in the following subsection.

2.2. Basic concepts on Self-Organizing Maps (SOMs)

Self-organizing Maps (SOMs) are a non-supervised learning technique used in exploratory data mining, introduced by Kohonen [19] and based on neural networks [33]. It is one of the best known methods for the construction of topographic maps, i.e. low-dimensional (usually 2D or 3D) visualizations of high dimensional data [21,22,34].

The SOM algorithm can be regarded as a "nonparametric regression" method, whose goal is fitting a number of discrete reference vectors to a distribution of vectorial input data samples [20]. The reference vectors define the nodes of a kind of *elastic* neural network, where a topologically ordered mapping is formed from the input space onto the neural network, thus obtaining a *feature map*. This adaptive process is biologically inspired by the organizations found in brain structures. If the network is a regular two-dimensional lattice, the feature map can be used to project and visualize high-dimensional data on it.

In the following, the basic SOM algorithm in the euclidean space is briefly reviewed [19,20]. Assume a two-dimensional regular (hexagonal or rectangular) lattice in which the array of nodes (*neurons*) are situated. Each node has associated a reference vector m_i of dimension n, which is defined by $m_i = [\mu_{i1} \ldots \mu_{in}]^T \in \mathbb{R}^n$, being $i \in \mathbb{R}^2$ the position in the lattice of the node associated to m_i. Weights $\mu_{ij} \in \mathbb{R}$ are initialized either randomly or by means of an initializing technique. On the other hand, a training input vector x of dimension n is defined as $x = [\xi_1 \ldots \xi_n]^T \in \mathbb{R}^n$.

At each iteration of the algorithm, an input data sample x is compared with all the m_i, and the location c of the *best matching unit* (BMU), i.e. the reference vector m_c whose weights are closest to values of x, is determined. x is then mapped onto this location. The BMU, denoted by m_c, accomplishes:

$$\|x - m_c\| = \min_i\{\|x - m_i\|\} \quad (1)$$

which is equivalent, in terms of the BMU location c, to:

$$c = \arg\min_i\{\|x - m_i\|\} \quad (2)$$

During learning, those nodes topographically close to the BMU (neighbor nodes), activate each other to learn something from input x. This process causes a smoothing effect on weights of nodes situated within this neighborhood. Fig. 2 illustrates this process [22]. Solid and dashed lines represent the situation before and after updating weights of nodes upon x, respectively.

Given an iteration t of the algorithm, $t = 0, 1, 2, \ldots$, weights in reference vector m_i are updated as follows:

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \quad (3)$$

where $h_{ci}(t) = h(\|r_c - r_i\|, t)$ is the so-called neighborhood function defined over the lattice nodes. $h_{ci}(t) \to 0$ when $t \to \infty$, thus ensuring convergence. $r_c, r_i \in \mathbb{R}^2$ are the locations of vectors m_c, m_i in the lattice. When $\|r_c - r_i\|$ increases, $h_{ci}(t) \to 0$. Let $N_c(t)$ be a

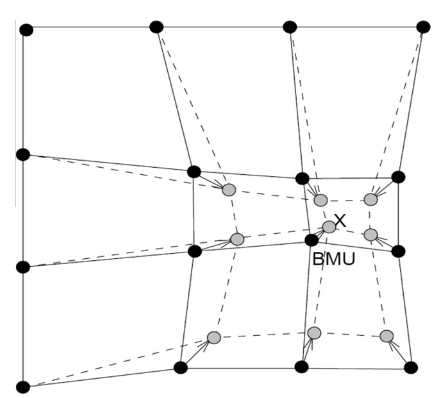

Fig. 2. Update of the BMU and its neighbors upon x (taken from [22]).

neighborhood set of lattice nodes around c. Then, the neighborhood function can be defined as follows:

$$h_{ci}(t) = \alpha(t) \quad \text{if } i \in N_c, \\ h_{ci}(t) = 0 \qquad \text{otherwise} \tag{4}$$

being $\alpha(t) \in (0,1)$ a learning rate that decreases over time (a value commonly taken is $\alpha(t) = 0.9(1 - \frac{t}{100})$). The radius of $N_c(t)$ also decreases over time, thus reducing the neighborhood set of c progressively. Another possible, smoother neighborhood function in terms of the Gaussian function, is:

$$h_{ci} = \alpha(t) = \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right) \tag{5}$$

As a result of applying the above mentioned steps iteratively with a set of input data samples, reference vectors tend to approximate them in an orderly fashion, and the lattice becomes ordered, in the sense that reference vectors in neighboring nodes have similar weights. The training process ends when a sufficient number of input vectors have been processed and the iterative process given by Eq. (3) converges towards stationary values.

Variants of the basic SOM algorithm include the so-called "Dot-Product" SOM, which involves the use of a more *biological* matching criterion, based on dot product operations [19]. In this case, the BMU is determined by:

$$x^T(t) \cdot m_c(t) = \max_i \{x^T(t) \cdot m_i(t)\} \tag{6}$$

Once the SOM has been constructed, we can proceed to locate on the map projections of those data samples that must be interpreted and visually analyzed. There are multiple SOM-based methods to visualize data, such as distance matrices, similarity coloring, data histograms and PCA projections [21,22].

SOMs have been successfully utilized in different descriptive data mining applications, such as full-text and financial data analysis, cluster analysis, and vector quantization and projection [21,34].

3. MENTOR: SOM-based graphical monitoring tool of preferences to support group decision making

As stated in the introduction, large-scale GDM problems are increasingly common in multiple real-life contexts. In these problems, classical tools and GDSS based on numerical or textual information that have been proposed to support GDM problems with small groups, may not be appropriate for a decision maker, when he/she needs to analyze the large amount of information related to experts' preferences to have a deeper knowledge about the current status of the problem.

For these reasons, in this section we present a graphical monitoring tool based on SOMs, so-called MENTOR, that supports decision makers by providing them with easy interpretable information about the status of large-scale GDM problems during their resolution, thus facilitating the analysis of diverse crucial aspects that are common in these problems, such as:

- The closeness between experts' preferences.
- Detection of conflicts amongst experts.
- Identification of subgroups of experts that disagree with the rest of the group.

Firstly, we will show a detailed scheme of the tool operation during the resolution process of GDM problems. We will then describe some examples of GDM situations in which the tool can be utilized to overcome the difficulties stated above.

Fig. 3 shows the architecture of MENTOR. The tool has been conceived as a local application that receives a set of experts' preferences about a GDM problem and generates a 2-D graphical interface with their representation. Although the use of MENTOR is currently proposed as a self-contained tool that is directly used by decision groups, it is also suggested its integration with new or already existing GDSS, to make them more interpretable for decision makers and support them in the overall decision analysis process. Further detail on the use of the technologies used in MENTOR (Java, MATLAB and SOM Toolbox), is given in the following subsection.

3.1. Scheme of the monitoring tool

A scheme of operation of MENTOR is shown in Fig. 4. The procedure it follows to generate a graphical representation about the status of the GDM problem consists of three phases, which are described below:

(1) *Gathering Information about the GDM problem*: Information about the status of the GDM problem that will be graphically represented, is gathered in this phase. Such information usually consists in preferences of all experts in the group. Sometimes it would be also interesting to gather additional information, for example the collective preference of the group.
MENTOR deals with opinions expressed numerically. More specifically, we consider the use of fuzzy preference relations (in which assessments $p_i^{lk} \in [0,1]$), to generate graphical representation of them (as will be shown with more detail in the following phase). Nevertheless, the tool also allows the management of different preference structures [35]. To do so, it is proposed the use of existing approaches to unify them into fuzzy preference relations. For instance, in [35] it is shown the relationship between different representation formats (preference orderings, utility values, multiplicative and fuzzy preference relations), and a set of transformation functions are defined to obtain a fuzzy preference relation from preferences expressed under each of these representation formats.
Regarding preferences expressed under different information domains (such as intervals or linguistic values), some approaches to conduct them into a common information domain can be also found in the literature. For example, in [11] some transformation functions are proposed to unify numerical, interval-valued and linguistic assessments into fuzzy preference relations.
Taking into account the above mentioned approaches, it is shown that MENTOR can be utilized in large-scale GDM problems in which experts can use different preference structures or information domains to express their opinions. Consequently, its integration with existing GDSS that incorporate such approaches is also possible.

(2) *Transforming Information to SOM-based format*: Once information to be visualized has been gathered, it must be transformed into a suitable format for its treatment by MENTOR. Since the tool is based on SOMs, it is necessary to represent preferences as input data samples (vectors) that can be managed by SOM algorithms (see Section 2.2). To do so, a *preference data-set* is generated upon preferences.
The software that generates preference data-sets upon the set of experts' preferences has been implemented with Java[1]

[1] A sample version of the Java application to generate preference data-sets upon a set of preferences is available at our website: http://sinbad2.ujaen.es/cod/mentor.

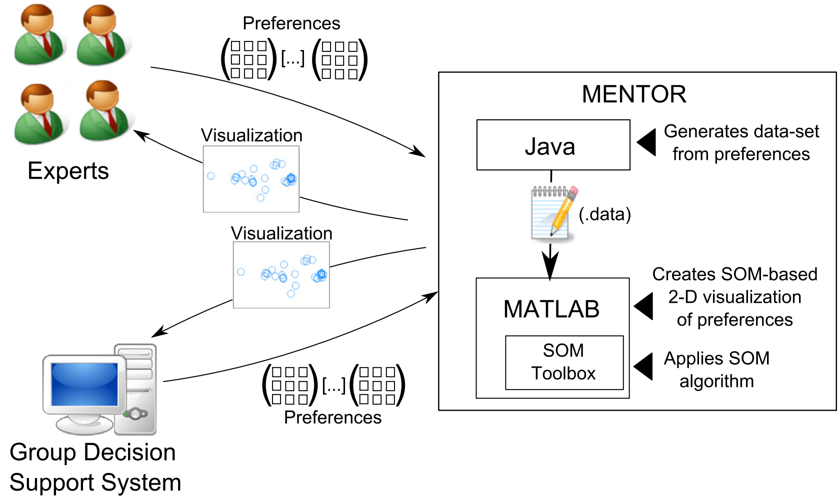

Fig. 3. Architecture of MENTOR.

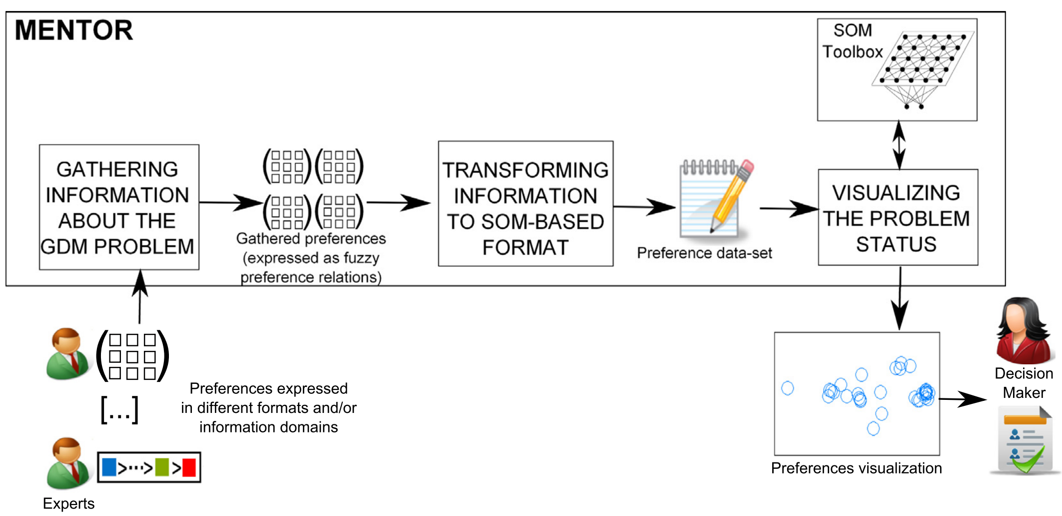

Fig. 4. General scheme of MENTOR.

(see Fig. 3). Preference data-sets are generated as files with extension *.data*. The structure of the preference data-set is as follows: the first row contains an integer value indicating the dimension of data samples, which is equal to the number of assessments each preference consists of. From the second row onwards, each row represents a data sample, corresponding to the preference of a single expert. The input preference format required to build the data-set is a numerical preference relation (e.g. a fuzzy preference relation, whose assessments are values in the unit interval [23]). Therefore, given a GDM problem with n alternatives, the dimension of data obtained from preferences must be equal to $n(n-1)$ (assessments of the type p_i^{ll}, $e_i \in E$, $x_l \in X$, are not considered, as stated in Section 2.1). Assessments are separated by blanks.

Data samples can be optionally tagged with informative purposes, by placing an alphanumerical tag at the end of the corresponding row. Tagging may provide additional information about a specific preference (for example, the name or role of its corresponding expert). Tags are not processed by the underlying SOM algorithm of MENTOR, but their content can be visualized together with the corresponding preference to provide additional knowledge about the problem.

Fig. 5 shows an extract of a preference data-set structure, in which two preferences have been tagged.

The following example illustrates the transformation of an expert's preference relation into an element of the preference data-set:

Example 1. Let P_i be the following fuzzy preference relation provided by an expert e_i, about a GDM problem consisting of $n = 4$ alternatives:

$$P_i = \begin{pmatrix} - & 1 & 0.5 & 0.9 \\ 0 & - & 0.15 & 0.4 \\ 0.5 & 0.85 & - & 1 \\ 0.1 & 0.6 & 0 & - \end{pmatrix}$$

Then, its corresponding data sample in the preference data-set obtained, is represented as follows:

```
12
0.2 0.5 0.2 0.8 0.5 0.5 0.5 0.5 0.0 0.8 0.5 1.0
0.2 0.35 0.2 0.8 0.75 0.3 0.65 0.25 0.0 0.8 0.7 1.0
[...]
0.7 0.5 0.7 0.3 0.4 0.5 0.5 0.6 0.7 0.3 0.5 0.3 X
[...]
0.41 0.45 0.44 0.59 0.56 0.4 0.54 0.43 0.31 0.55 0.6 0.68 P
```

Fig. 5. Example of preference data-set with tags.

1 0.5 0.9 0 0.15 0.4 0.5 0.85 1 0.1 0.6 0

Although data samples in the data-set must be built upon numerical preference relations, some existing GDM models and approaches allow the unification into such format from experts' preferences expressed by means of different structures [35] or heterogeneous information [11], as aforementioned in the previous phase. Similarly, incomplete preferences [13] and preferences expressed in different scales [36] could be also considered, because the underlying SOM algorithm (which is applied in the following phase) can deal with incomplete data, and it also implicitly normalizes data values expressed in different numerical scales.

(3) *Visualizing the problem status*: The preference data-set is used as an input to apply a SOM-based technique that generates a 2-D graphical projection of data contained in it. Such a projection may be utilized by a group member (e.g. a decision maker who coordinates the whole group) for analyzing aspects of interest about the GDM problem.

The application to visualize preferences has been implemented by means of the software suite MATLAB[2] (see Fig. 3), which facilitates the management of data-sets and their graphical representation. MATLAB also offers possibilities to integrate its user-developed applications with a variety of widely used technologies, such as, Java, C++, and .NET, thus offering the possibility to communicate MENTOR with other systems.

Preference data-sets with extension *.data* obtained previously, are directly read by MATLAB, without the need for any further processing. Then, a SOM algorithm must be invoked to create the map on which data will be visualized. To do so, we have utilized the implemented SOM algorithms provided by a third-party MATLAB library so-called SOM Toolbox,[3] which was developed by Vesanto et al. [22] and constitutes a powerful research-oriented library with numerous functions and possibilities for managing SOMs and analyzing/visualizing data with them. By using this library, MENTOR offers the flexibility to apply different SOM algorithms defined by several settings, including: (i) the choice of the map size and shape (rectangular or hexagonal lattice), (ii) a matching criterion (see Eqs. (1) and (6) for instance), (iii) the neighborhood function, $h_{ci}(t)$, or (iv) the learning rate, $\alpha(t)$, amongst others.

Once constructed the map, each preference in the data-set is projected into it. The visualization method considered to show this task is a two-dimensional PCA projection of preferences [21]. Functions to generate and plot a graphical interface tho show PCA projections are also provided by MATLAB and SOM Toolbox.

It is noteworthy that in this phase, instead of obtaining a single graphical projection of experts' preferences solely, it would be sometimes useful to provide further detailed graphical information. For example, visualizing preferences at different levels of detail can be particularly interesting in GDM problems based on preference relations [26], because it would be sometimes convenient to view experts' opinions on each specific alternative (for purposes of disagreement detection, for instance). Then, a visual projection can be generated for each alternative $x_l \in X$ separately.

Tagging data might also be useful for several visualization purposes, some of which are:

- Viewing the collective preference of the group, by including and tagging it in the preference data-set.
- In some cases, it can be interesting to provide each expert with a visual representation of his/her own position with respect to the group. This can be done by generating a personalized graphical projection for each expert, in which his/her own preference is tagged.

Fig. 6 shows the graphical visualization corresponding to the complete data-set whose extract was shown in Fig. 5, in which the expert's self preference and the collective preference have been tagged.

In group decisions under consensus, the graphical visualization of the GDM problem status across the discussion process would be particularly convenient. Given that such processes consist of several rounds in which experts modify their opinions to increase agreement in the group (see Section 2.1), MENTOR can be iteratively used in consensus-based GDM problems, so that graphical information of the problem status is generated at each consensus round (as will be shown in the application example in Section 4). Visualizing the evolution of experts' preferences across the time may provide a better insight on the overall performance of this kind of problems and even a foresight of the future status of such problems in upcoming consensus rounds.

3.2. On the use of MENTOR in large-scale GDM

In the following, we illustrate how the graphical information provided by MENTOR can be used to facilitate the analysis of some important aspects and difficulties found in GDM problems, which are especially frequent in large-scale GDM. Such aspects and difficulties, and the way in which MENTOR facilitates their detection and analysis, are enumerated below:

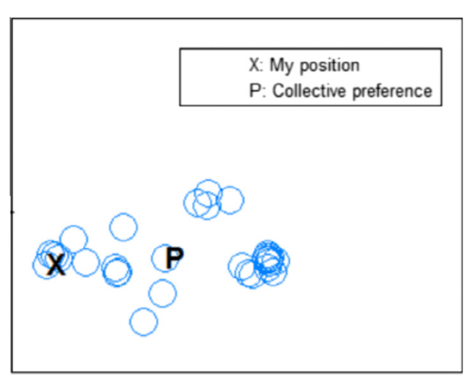

Fig. 6. Example of preferences visualization with tags.

[2] We are currently working on obtaining the necessary MATLAB license to release a sample version of the visualizing application in our website. Meanwhile, readers interested in obtaining a visualization of their preferences can follow the instructions found in: http://sinbad2.ujaen.es/cod/mentor.

[3] http://www.cis.hut.fi/somtoolbox/.

- *Detecting conflicting opinions amongst experts*: Analyzing numerical or textual information about experts' preferences to identify conflicting opinions can be an affordable task if dealing with small groups, but not so adequate when the group size is large. The 2-D representation of preferences generated by MENTOR can provide a visual insight on conflicting opinions (if any) in these cases, because such preferences are visually represented as data points that are allocated far from each other.
- *Identifying disagreing experts*: When conflicting opinions are detected (see above), it would be interesting for the decision maker to view the identity of experts or subgroups of them who disagree with each other. This can be done by tagging the preferences of such experts, so that their names or identifiers can be also represented graphically.
- *Determining the closeness and agreement cardinality graphically*: Although most consensus models to support consensus reaching processes compute a global degree of agreement in the group analytically (usually as a numerical value in the unit interval) [1,14,15], such computations are frequently based on compensative consensus measures, in which case the collective agreement level computed might sometimes not reflect possible disagreement positions between some experts faithfully. In such cases, preferences visualization may help the decision maker to view the closeness between experts' opinions and decide whether the agreement cardinality (i.e. the number of experts who present a high agreement on the collective opinion with respect to the total group size) is enough or not to make a final decision, in situations of hesitancy.
- *Detecting non-cooperative behaviors in consensus reaching*: In GDM problems that require consensus, experts may adopt different types of behavior during the discussion process, regarding their predisposition to modify their initial opinions to make them closer to the collective opinion. Some experts or coalitions of experts with similar interests may not present a cooperative behavior in these problems, in the sense that they might move their preferences strategically trying to deviate the collective opinion [37]. If the necessary mechanisms to detect such behaviors analytically are utilized, then it is possible to tag experts involved in such behaviors and facilitate their graphical detection as well. Additional information about the relative size of the disagreeing subgroup with respect to the total group size would also be useful.

The illustrative example presented in the following section shows some of the above mentioned issues in practice.

4. Application example

In this section, an example of application of MENTOR to a real-life GDM problem is presented to show some of the possibilities such a tool offers, as well as its usefulness in practice. To do so, firstly an example of large-scale GDM problem is proposed. Then, the problem is solved by applying a simple GDM resolution scheme, and preferences in the group are visualized and analyzed by using MENTOR. Finally, a consensus reaching process is also applied to seek a higher degree of agreement, and MENTOR is used to visualize the evolution of experts' preferences across the process of negotiation.

4.1. Definition of the large-scale GDM problem

The GDM problem is formulated as follows: the 2013 graduating class of Computer Science M.Sc. Degree, compound by 46 students, $E = \{e_1, \ldots, e_{46}\}$, needs to decide the destination for their final year trip, amongst four possible choices, $X = \{x_1$: Mediterranean cruise, x_2: Tunisia tour, x_3: Canary Islands, x_4: Prague, Vienna and Budapest$\}$. During a lab session to which all 46 students attended, each one was requested to provide a fuzzy preference relation over the four alternatives.[4]

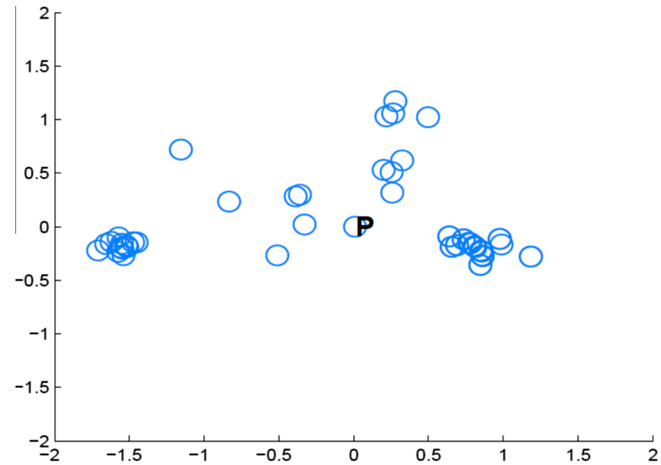

Fig. 7. Graphical representation of the group's preferences.

4.2. Visualization of a simple GDM resolution process

The large-scale GDM problem defined above was solved by applying a direct resolution scheme [27]. MENTOR was used to gather and visualize all experts' preferences and the collective preference obtained in the aggregation phase [28], having the latter been tagged to facilitate its detection.

Fig. 7 shows the graphical projection of preferences generated by MENTOR. The tag "P" indicates the position of the collective preference. As can be seen, some useful information can be easily obtained by analyzing the graphical representation generated: there exist two significant subgroups of students with very similar interests. However, such subgroups present a strong disagreement with each other and with the rest of students, who have diverse preferences that are situated far from the majority opinions.

The graphical representation of preferences provided by MENTOR let us conclude, without the need for analyzing the large amount of numerical information about experts' preferences, that the proposed solution to the GDM problem (given by the collective preference) is supported by a minor number of experts only, therefore it would not be a well-accepted solution by the group.

4.3. Visualization during a consensus reaching process

Given the low level of students' agreement on the initially obtained solution, it would be convenient to apply a consensus reaching process before carrying out the selection process. To do so, the consensus model proposed in [17] has been used, by considering the same initial preferences of students (see Fig. 7).

A total of five consensus rounds were carried out. At the end of each round, MENTOR generated a graphical projection of preferences to facilitate an analysis of their evolution, as well as the detection of possible disagreement positions and patterns of behavior adopted by some students. Fig. 8 shows the projections obtained from the second round onwards. Most students tended to move their preferences closer to the agreement position, which

[4] A large amount of information about preferences has been used in this example, therefore it was omitted for the sake of space. A supplementary material document that contains such preferences is also available at: http://sinbad2.ujaen.es/cod/mentor.

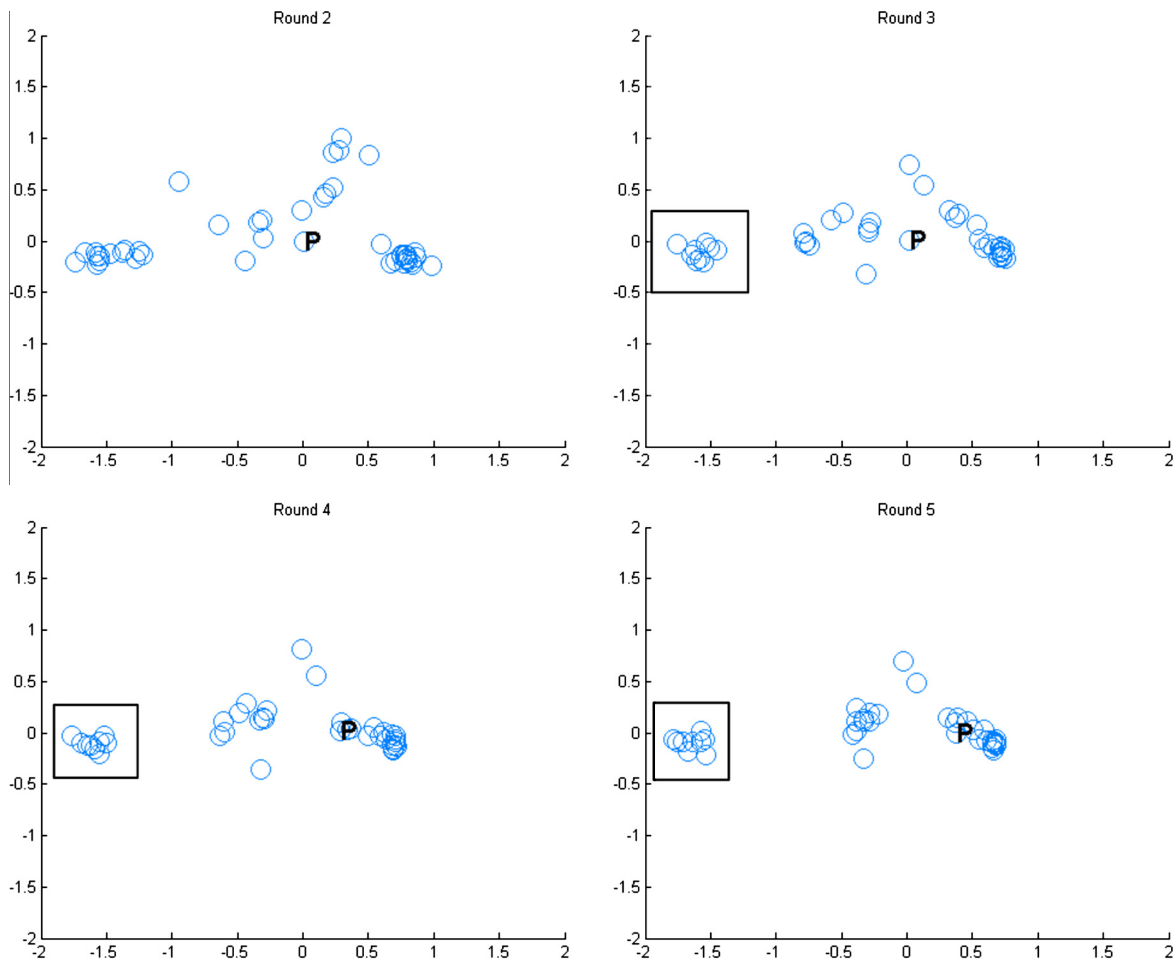

Fig. 8. Graphical representation of preferences during the consensus reaching process.

means they contributed positively to reach a consensus by applying changes suggested by the consensus model considered. However, MENTOR allowed us to notice that one of the two aforementioned subgroups of interest presented a different behavior, as students belonging to it did not move their preferences at all, showing that they were not interested in achieving an agreed decision, but rather in their own preferred options. This fact illustrates how MENTOR facilitates the detection of both disagreement positions and undesired behavioral patterns of experts or coalitions of them.

Based on subgroup behaviors detected, different alternate actions or decisions could be carried out by a decision maker depending of each particular problem circumstances, for example: informing experts involved that they are hindering the achievement of a consensus, moving onto the selection process to make the final decision before such experts can deviate the group opinion excessively, or penalizing experts who do not cooperate with the rest of the group [37].

5. Concluding remarks

This paper has presented MENTOR, a graphical monitoring tool based on Self-Organizing Maps to support large-scale Group Decision Making problems. The main goal of such a tool consists in helping decision makers to obtain and analyze easy interpretable information about the status of these problems during their resolution, as well as letting them analyze visually how different individuals or subgroups of them behave during the problem. MENTOR can also be used to detect and analyze visually a variety of aspects that are especially frequent in large-scale group decisions, such as the presence of subgroups of individuals with similar interests or the existence of agreement or disagreement positions. Additionally, it facilitates the monitoring of the problem status across the time in the cases that a consensus reaching processes is carried out. The visual analysis that MENTOR provides goes beyond the numerical information that Group Decision Support Systems or consensus models usually manage and provide: with MENTOR it is possible to find out, in a more understandable way, what does such numerical information mean, how do experts organize in subgroups, which experts do not contribute to achieve a consensus in the group, etc.

Although the tool is rather oriented towards giving support to a decision maker who is responsible for supervising the problem (e.g. a moderator in a consensus reaching process or a system administrator if the problem is solved with the aid of a Group Decision Support System), it has been shown that for some specific purposes (such as visualizing an expert's self position with respect of the rest of the group) it would be also interesting to provide experts with personalized visual information about the current problem status.

An example of application of the monitoring tool has been also presented, to solve a group decision making problem by applying both a direct selection process and a consensus reaching process. Such an example has illustrated how to analyze the behavior of experts through their preferences, as well as how to detect disagreement positions easily.

Acknowledgment

This work is partially supported by the Research Project TIN-2012-31263 and ERDF.

References

[1] J. Kacprzyk, Group decision making with a fuzzy linguistic majority, Fuzzy Sets and Systems 18 (1986) 105–118.
[2] N. Bryson, Group decision-making and the analytic hierarchy process. exploring the consensus-relevant information content, Computers and Operations Research 23 (1996) 27–35.
[3] J. Lu, G. Zhang, D. Ruan, F. Wu, Multi-Objective Group Decision Making, Imperial College Press., 2006.
[4] C. Kwong, Y. Ye, Y. Chen, K. Choy, A novel fuzzy group decision-making approach to prioritising engineering characteristics in QFD under uncertainties, International Journal of Productions Research 49 (2011) 5801–5820.
[5] J. Kim, A model and case for supporting participatory public decision making in e-democracy, Group Decision and Negotiation 17 (2008) 179–192.
[6] D.C. Morais, A.T. Almeida, Group decision making on water resources based on analysis of individual rankings, Omega 40 (2012) 42–52.
[7] C. Butler, A. Rothstein, On Conflict and Consensus: A Handbook on Formal Consensus Decision Making, Takoma Park, 2006.
[8] S. Saint, J.R. Lawson, Rules for Reaching Consensus A Modern Approach to Decision Making, Jossey-Bass, 1994.
[9] L. Martínez, J. Montero, Challenges for improving consensus reaching process in collective decisions, New Mathematics and Natural Computation 3 (2007) 203–217.
[10] G. Wei, R. Lin, X. Zhao, H. Wang, Models for multiple attribute group decision making with 2-tuple linguistic assessment information, International Journal of Computational Intelligence Systems 3 (2010) 315–324.
[11] F. Herrera, L. Martínez, P. Sánchez, Managing non-homogeneous information in group decision making, European Journal of Operational Research 166 (2005) 115–132.
[12] R. Yager, Non-numeric multi-criteria multi-person decision making, Group Decision and Negotiation 2 (1993) 81–93.
[13] Z. Xu, An integrated model-based interactive approach to FMAGDM with incomplete preference information, Fuzzy Optimization and Decision Making 9 (2010) 333–357.
[14] R. Parreiras, P. Ekel, J. Martini, R. Palhares, A flexible consensus scheme for multicriteria group decision making under linguistic assessments, Information Sciences 180 (2010) 1075–1089.
[15] I. Palomares, L. Martínez, A semi-supervised multi-agent system model to support consensus reaching processes, IEEE Transactions on Fuzzy Systems (2013), (in press), http://dx.doi.org/10.1109/TFUZZ.2013.2272588.
[16] J. Kacprzyk, S. Zadrozny, Soft computing and web intelligence for supporting consensus reaching, Soft Computing 14 (2010) 833–846.
[17] I. Palomares, R. Rodríguez, L. Martínez, An attitude-driven web consensus support system for heterogeneous group decision making, Expert Systems with Applications 40 (2013) 139–149.
[18] C. Sueur, J. Deneubourg, O. Petit, From social network (centralized vs. decentralized) to collective decision-making (unshared vs. shared consensus), PLoS one 7 (2012) 1–10.
[19] T. Kohonen, Self-Organizing Maps, Springer, Heidelberg, 1995.
[20] T. Kohonen, The self-organizing map, Neurocomputing 21 (1998) 19–30.
[21] J. Vesanto, SOM-based data visualization methods, Intelligent Data Analysis 3 (1999) 111–126.
[22] J. Vesanto, J. Himberg, E. Alhoniemi, J. Parhankangas, SOM toolbox for MATLAB 5, Technical Report, Helsinki University of Technology, 2000. <http://www.cis.hut.fi/projects/somtoolbox/>.
[23] S. Orlovsky, Decision-making with a fuzzy preference relation, Fuzzy Sets and Systems 1 (1978) 155–167.
[24] R. Luce, P. Suppes, et al., Preferences utility and subject probability, in: R.D. Luce et al. (Eds.), Handbook of Mathematical Psychology, vol. III, Wiley, New York, 1965, pp. 249–410.
[25] T. Tanino, Fuzzy preference orderings in group decision making, Fuzzy Sets and Systems 12 (1984) 117–131.
[26] T. Tanino, Fuzzy preference relations in group decision making, in: J. Kacprzyk, M. Roubens (Eds.), Non-Conventional Preference Relations in Decision Making, Springer, Berlin, 1988, pp. 54–71.
[27] F. Herrera, E. Herrera-Viedma, J. Verdegay, A sequential selection process in group decision making with linguistic assessments, Information Sciences 85 (1995) 223–239.
[28] M. Roubens, Fuzzy sets and decision analysis, Fuzzy Sets and Systems 90 (1997) 199–206.
[29] P. Gulliver, Disputes and Negotiations, Academic Press, New York, 1995.
[30] J. Sebenius, Negotiation analysis: a characterization and review, Management Sciences 38 (1992) 18–38.
[31] F. Herrera, E. Herrera-Viedma, J. Verdegay, A model of consensus in group decision making under linguistic assessments, Fuzzy Sets and Systems 78 (1996) 73–87.
[32] G. Bordogna, M. Fedrizzi, G. Pasi, A linguistic modeling of consensus in group decision making based on OWA operators, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 27 (1) (1997) 126–133.
[33] M. Van-Hulle, Self-Organizing Maps: Theory, Design, and Application, Kaibundo, Tokyo, 2011.
[34] J. Vesanto, E. Alhoniemi, Clustering of the self-organizing map, IEEE Transactions on Neural Networks 11 (2000) 586–600.
[35] E. Herrera-Viedma, F. Herrera, F. Chiclana, A consensus model for multiperson decision making with different preference structures, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 32 (2002) 394–402.
[36] F. Herrera, L. Martínez, E. Herrera-Viedma, A fusion approach for managing multi-granularity linguistic term sets in decision making, Fuzzy Sets and Systems 114 (2000) 43–58.
[37] R.R. Yager, Penalizing strategic preference manipulation in multi-agent decision making, IEEE Transactions on Fuzzy Systems 9 (2001) 393–403.