4. H. J. Bernardin, J. S. Kane, S. Ross, J. D. Spina and D. L. Johnson. P...

AN INTERACTIVE SUPPORT SYSTEM TO AID EXPERTS TO EXPRESS CONSISTENT PREFERENCES

S. ALONSO, E. HERRERA-VIEDMA, F. HERRERA AND F.J. CABRERIZO

Department of Computer Science and Artificial Intelligence
University of Granada, 18071, Granada, Spain
E-mail: {salonso, viedma, herrera}@decsai.ugr.es, cabre@correo.ugr.es

F. CHICLANA

Centre for Computational Intelligence, School of Computing
De Montfort University, Leicester LE1 9BH, UK.
E-mail: chiclana@dmu.ac.uk

In Group Decision Making, the expression of preferences is often a very difficult task for the experts, specially in decision problems with a high number of alternatives. The problem is increased when they are asked to give their preferences in the form of preference relations: although preference relations have a very high level of expressivity and they present good properties that allow to operate with them easily, the amount of preference values that the experts are required to give increases exponentially. This usually leads to situations where the expert is not capable of properly express all his/her preferences in a consistent way (that is, without contradiction), so finally the information provided can easily be either inconsistent or incomplete (when the expert prefers not to give some particular preference values). In this paper we develop a transitivity based support system to aid experts to express their preferences (in the form of preference relations) in a more consistent way. The system works interactively with the expert making recommendations for the preference values that the expert have not yet expressed. Those recommendations are computed trying to maintain the consistency level of the expert as high as possible.

1. Introduction

One of the key issues when solving Group Decision Making (GDM) problems is to obtain the preferences of the different experts in order to lately combine them and find which solution x_j among the feasible set of alternatives $X = \{x_1, ..., x_n\}$ is the best. There exist several different representation formats in which experts can express their preferences but, among others, Fuzzy Preference Relations (FPR) [5,6,8] have been widely used because they are a very expressive format and also they present good properties that allow to operate with them easily [6,8].

Preference relations may also present some disadvantages. As it is required to express a preference degree among all possible pairs of different alternatives, the amount of information that the experts have to provide increases exponentially. Clearly, when the cardinality of the problem is high then we may find situations where the experts do not provide good (consistent and complete) preference relations. In this cases, an expert might choose not to provide all the preference values that he is required to, or the expert might provide his/her preferences in an inconsistent way, i.e., his/her preferences might be contradictory. In a previous paper [1] a procedure to compute the missing values of an incomplete FPR taking into account the expert consistency level has been developed. Nevertheless, that procedure could not deal with the initial contradiction that the expert could have introduced in his/her preferences, and what could be worse, the expert might not accept the estimated values (even if they increase the overall consistency level).

Thus, when designing a computer driven model to deal with GDM problems where the information is given in the form of FPR, software tools to aid the experts to express their preferences avoiding the mentioned problems should be implemented. As experts might not be familiar with preference relations, the aiding tools should be easy enough to use and they should follow the general principles of interface design [4].

In this paper we present an *interactive support system* to aid experts to express their preferences using fuzzy preference relations. The system will give recommendations to the expert while he/she is providing the preference values in order to maintain a high level of consistency in the preferences, as well as trying to avoid missing information. Also, the system will provide measures of the current level of consistency and completeness that the expert has achieved, which can be used to avoid situations of self contradiction. The system has been programmed using Java technologies, which allows its integration in web-based applications which are increasingly being used in GDM and Decision Support environments [3,10].

The rest of the paper is set as follows: In Section 2 we present our preliminaries. In Section 3 we describe in detail our support system. Finally in Section 4 we point out our conclusions and future improvements.

2. Preliminaries

In this section we present the preliminaries concepts needed for the rest of the paper: the notion of Incomplete Linguistic Preference Relation, the Additive Transitivity Property and how this transitivity property can be used to estimate missing values in a fuzzy preference relation.

2.1. *Incomplete Fuzzy Preference Relations*

One of the most frequently used formats to represent preferences are Fuzzy Preference Relations [5,6,8]. They present a very high level of expressivity and good properties that allow to operate with them easily [6,8].

Definition 1: A fuzzy preference relation P on a set of alternatives X is a fuzzy set on the product set $X \times X$, i.e., it is characterized by a membership function $\mu_P : X \times X \longrightarrow [0, 1]$.

When cardinality of X is small, the preference relation may be conveniently represented by the $n \times n$ matrix $P = (p_{ik})$, being $p_{ik} = \mu_P(x_i, x_k)$ $(\forall i, k \in \{1, \ldots, n\})$ interpreted as the preference degree or intensity of the alternative x_i over x_k: $p_{ik} = 1/2$ indicates indifference between x_i and x_k $(x_i \sim x_k)$, $p_{ik} = 1$ indicates that x_i is absolutely preferred to x_k, and $p_{ik} > 1/2$ indicates that x_i is preferred to x_k $(x_i \succ x_k)$. Based on this interpretation we have that $p_{ii} = 1/2$ $\forall i \in \{1, \ldots, n\}$ $(x_i \sim x_i)$.

Usual models to solve GDM problems assume that experts are always able to provide all the preferences required, that is, to provide all p_{ik} values. This situation is not always possible to achieve. Experts could have some difficulties in giving all their preferences due to lack of knowledge about part of the problem, or simply because they may not be able to quantify some of their degree of preference. In order to model such situations, we define the concept of an *incomplete fuzzy preference relation* [7].

Definition 2 A function $f : X \longrightarrow Y$ is partial when not every element in the set X necessarily maps onto an element in the set Y. When every element from the set X maps onto one element of the set Y then we have a total function.

Definition 3 An incomplete fuzzy preference relation P on a set of alternatives X is a fuzzy set on the product set $X \times X$ that is characterized by a partial membership function.

2.2. *Additive Transitivity Property*

For GDM problems where the preferences are given as fuzzy preference relations, some properties about the preferences expressed by the experts are usually assumed desirable to avoid contradictions in their opinions, that is, to avoid inconsistent opinions. One of them is the *additive transitiv-*

*ity*property [6,9]:

$$(p_{ij} - 0.5) + (p_{jk} - 0.5) = (p_{ik} - 0.5) \quad \forall i, j, k \in \{1, \ldots, n\} \tag{1}$$

2.3. *Estimating Missing Values Using Additive Transitivity*

Expression (1) can be used to calculate an estimated value of a preference degree using other preference degrees in a fuzzy preference relation. Indeed, the preference value p_{ik} $(i \neq k)$ can be estimated using an intermediate alternative x_j in three different ways:

- From $p_{ik} = p_{ij} + p_{jk} - 0.5$ we obtain the estimate
$$(cp_{ik})^{j1} = p_{ij} + p_{jk} - 0.5 \tag{2}$$
- From $p_{jk} = p_{ji} + p_{ik} - 0.5$ we obtain the estimate
$$(cp_{ik})^{j2} = p_{jk} - p_{ji} + 0.5 \tag{3}$$
- From $p_{ij} = p_{ik} + p_{kj} - 0.5$ we obtain the estimate
$$(cp_{ik})^{j3} = p_{ij} - p_{kj} + 0.5 \tag{4}$$

As we have already said, and expert can choose to not provide complete preference relations, thus, the above equations may not be possible to be applied for every alternative x_i, x_k, x_j. If expert e_h provides an incomplete fuzzy preference relation P^h, the following sets are defined [7]:

$$A = \{(i,j) \mid i,j \in \{1,\ldots,n\} \wedge i \neq j\} \quad ; H_{ik}^{h1} = \left\{j \neq i,k \mid (i,j),(j,k) \in EV^h\right\}$$
$$MV^h = \left\{(i,j) \in A \mid p_{ij}^h \textit{ is unknown}\right\} ; H_{ik}^{h2} = \left\{j \neq i,k \mid (j,i),(j,k) \in EV^h\right\}$$
$$EV^h = A \setminus MV^h \quad ; H_{ik}^{h3} = \left\{j \neq i,k \mid (i,j),(k,j) \in EV^h\right\}$$

MV^h is the set of pairs of alternatives whose preference degrees are not given by expert e_h, EV^h is the set of pairs of alternatives whose preference degrees are given by the expert e_h; H_{ik}^{h1}, H_{ik}^{h2}, H_{ik}^{h3} are the sets of intermediate alternative $x_j (j \neq i, k)$ that can be used to estimate the preference value p_{ik}^h $(i \neq k)$ using equations (2), (3), (4) respectively.

The *final* estimated value of a particular preference degree p_{ik}^h $((i,k) \in EV^h)$ can be calculated only when $\#(H_{ik}^{h1} + H_{ik}^{h2} + H_{ik}^{h3}) \neq 0$:

$$cp_{ik}^h = \frac{\sum_{j \in H_{ik}^{h1}} (cp_{ik}^h)^{j1} + \sum_{j \in H_{ik}^{h2}} (cp_{ik}^h)^{j2} + \sum_{j \in H_{ik}^{h3}} (cp_{ik}^h)^{j3}}{(\#H_{ik}^{h1} + \#H_{ik}^{h2} + \#H_{ik}^{h3})} \tag{5}$$

In the case of being $(\#H_{ik}^{h1} + \#H_{ik}^{h2} + \#H_{ik}^{h3}) = 0$ then the preference value p_{ik}^h $((i,k) \in EV^h)$ cannot be estimated using the rest of known values.

3. Interactive Support System to Aid Experts to Express Consistent Preferences

In this section we describe in detail our interactive support system to aid experts to express their fuzzy preference relations in a consistent way. Firstly we will enumerate all the design goals and requirements that we have taken into account and secondly we will describe the actual implementation of every requirement in the system.

3.1. *Design Goals and Requirements*

Our design goals and requirements could be split in two different parts: *Interface Requirements*, and *Logical Goals*.

Interface Requirements: These requirements deal with the visual representation of the information and the different controls in the system. We want our system to comply the so called *"Eight Golden Rules"*[4] for interface design:

- **GR 1.** Strive for consistency.
- **GR 2.** Enable frequent users to use shortcuts.
- **GR 3.** Offer informative feedback.
- **GR 4.** Design dialogues to yield closure.
- **GR 5.** Offer simple error handling.
- **GR 6.** Permit easy reversal of actions (undo action).
- **GR 7.** Support internal focus of control (user is in charge).
- **GR 8.** Reduce short-term memory load of the user.

Logical Goals:

- **Goal 1.** Offer recommendations to the expert to guide him toward a highly consistent and complete fuzzy preference relation.
- **Goal 2.** Recommendations must be given interactively.
- **Goal 3.** Recommendations must be simple to understand and to apply.
- **Goal 4.** The user must be able to refuse recommendations.
- **Goal 5.** The system must provide indicators of the consistency and completeness level achieved in every step.
- **Goal 6.** The system should be easy to adapt to other types of preference relations.
- **Goal 7.** The system should be easy to incorporate to Web-based GDM models and decision support systems[3,10]

3.2. *Actual Implementation*

We will now detail how we have dealt with every requirement and goal that we have presented in the previous section. To do so we will make use of a snapshot of the system (*figure 1*) where we will point out every implementation solution. **Implementation of the Interface Requirements:**

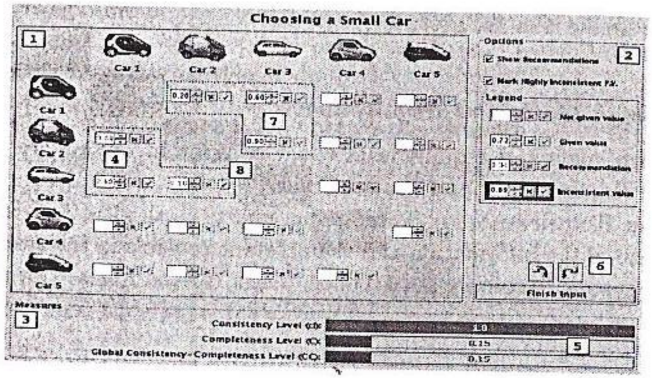

Figure 1. Snapshot of the Support System

- **GR 1.** The interface has been homogenised in order to present a easy to understand view of the process which is being carried. We have introduced 3 main areas: In area number **(1)** we present the *fuzzy preference relation* that the expert is introducing, as well as a brief description of every alternative. Area number **(2)** contains several global controls to activate/deactivate certain functions, as well as to finish the input process. Area number **(3)** contains different measures that show the overall progress (see below).
- **GR 2.** Shortcuts have been added to the most frequent options and the input text areas for the preference values have been ordered to access to them easily using the keyboard.
- **GR 3.** Our systems provides recommendations **(4)** and consistency and completeness measures **(5)** (see below). All controls have tooltips.
- **GR 4.** With every change that the user makes to his/her preferences the system provides new recommendations and measures.
- **GR 5.** Incorrect inputs are prompt with error messages.
- **GR 6.** We have introduced *undo* and *redo* buttons **(6)**.
- **GR 7.** The user can choose at every moment which preference value wants to give or update, as well as enabling/disabling options.
- **GR 8.** All information is presented in a single screen.

Logical Goals:

- **Goal 1.** To offer recommendations, the system computes all the missing values that could be estimated by using *equation 5* and it presents them in area **(1)**. As the values are computed taking into account the additive transitivity property, the recommendations should tend to increment the overall consistency level. They are presented in a different color (gray) **(4)** to be easily distinguishable from the proper expert values **(7)**.
- **Goal 2.** When the expert introduces or updates a preference value all possible recommendations are recomputed and presented.
- **Goal 3.** Recommendations are given in the same manner as the user inputs his/her preferences. There is also a button that enables the user to accept or validate a given recommendation **(8)**.
- **Goal 4.** A user can choose any value for a particular preference degree ignoring all the recommendations.
- **Goal 5.** In previous works[2] we provided some measures of the consistency and completeness of fuzzy preference relations **(5)**. The consistency measure for a particular FPR P^h (called cl^h) is based on the error that can be computed between the p^h_{ik} values that the expert e_h provides and the cp^h_{ik} values that can be estimated using *expression 5*. The completeness measure (C^h) is obtained as a ratio between the number of values given by the expert ($\#EV^h$) and the total number of values that the expert should give to have a complete FPR. In our system we also combine these two measures into a global consistency/completeness measure that informs the expert with his/her current degree of consistency and completeness:

$$CC^h = cl^h \cdot C^h \qquad (6)$$

- **Goal 6.** As the system is programmed following the principles of Object Oriented Programming, to adapt it to new kinds of preference relations is an easy task.
- **Goal 7.** As the system is Java based, it is easy to incorporate it into a web-based environment.

4. Conclusions and Future Improvements

In this paper we have presented an interactive support system which aids experts to provide consistent preferences and to help them to avoid incomplete information situations in GDM environments where the opinions must are provided as fuzzy preference relations. The system works providing easy recommendations while the expert gives his/her preference values, always trying to maximize the consistency of the expert's opinions.

In the future we will extend the system to allow the use of different preference relations (linguistic, interval-valued and multiplicative preference relations, for example) and we will integrate it into a complete consensus reaching process to enrich the preference acquisition step in the process.

Acknowledgments

This work has been supported by the Research Project TIC2003-07977, and the EPSRC research project "EP/C542215/1".

References

1. Alonso, S., Chiclana, F., Herrera, F., Herrera-Viedma, E.: A consistency based procedure to estimate missing pairwise preference values. *International Journal of Intelligent Systems*, in press.
2. Alonso, S., Herrera-Viedma, E., Chiclana, F., Herrera, F., Managing Incomplete Information in Consensus Processes. *Proc. of Simposio sobre Lógica Fuzzy & Soft Computing (LFSC2005)*, Granada (Spain) (2005) 175–182.
3. Bhargava, H.K., Power, D.J., Sun, D.: Progress in Web-based decision support technologies. *Decision Support Systems*, In Press.
4. Chen, Z.: Interacting with Software Components. *Decision Support Systems* **14** (1995) 349–357.
5. Chiclana, F., Herrera, F., Herrera-Viedma, E.: Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations. *Fuzzy Sets and Systems* **97** (1998) 33–48.
6. Herrera-Viedma, E., Herrera, F., Chiclana, F., Luque, M.: Some issues on consistency of fuzzy preference relations. *European Journal of Operational Research* **154** (2004) 98–109.
7. Herrera-Viedma, E., Chiclana, F., Herrera, F., Alonso, S.: A group decision making model with incomplete fuzzy preference relations based on additive consistency. *Technical Report #SCI2S-2004-11*. University of Granada (2004). *http://sci2s.ugr.es/publications/ficheros/TechnicalReportSCI2S-2004-11.pdf*
8. Kacprzyk, J.: Group decision making with a fuzzy linguistic majority. *Fuzzy Sets and Systems* **18** (1986) 105–118.
9. Tanino, T.: Fuzzy preference orderings in group decision making. *Fuzzy Sets and Systems* **12** (1984) 117–131.
10. Zhang S., Goddard, S.: A software architecture and framework for Web-based distributed Decision Support Systems. *Decision Support Systems*, In Press.

Applied Artificial Intelligence

Proceedings of the 7th International FLINS Conference

Edited by

Da Ruan
Pierre D'hondt
Paolo F. Fantoni
Martine De Cock
Mike Nachtegael
Etienne E. Kerre