Polska
Akademia
Nauk
Instytut
Badań
Systemowych

Methodology and applications of decision support systems

Proceedings of the 3-rd
Polish-Finnish Symposium
Gdańsk-Sobieszewo, September 26-29, 1988

edited by
Roman Kulikowski

Polska Akademia Nauk Instytut Badań Systemowych

Methodology and applications of decision support systems

Proceedings of the 3-rd
Polish-Finnish Symposium
Gdańsk-Sobieszewo, September 26-29, 1988

edited by
Roman Kulikowski

Warszawa 1989

Secretary of the Conference
dr. Andrzej Stachurski

Wykonano z gotowych oryginałów tekstowych dostarczonych przez autorów

ISBN 83-00-02543-X

A DECISION SUPPORT SYSTEM FOR ANALYSING AND AGGREGATING FUZZY ORDERINGS

Jan W. Owsiński, Sławomir Zadrożny

Systems Research Institute
Polish Academy of Sciences
Newelska 6, 01-447 Warszawa, Poland

Summary. The paper presents a practical method for implementing a software for the in-session processing and aggregation of precedence data obtained from a number, m, of judges and concerning a number, n, of items. Precedence data can be given in a variety of ways ranging from a unique ordering to $1/2 \cdot n(n-1)$ fuzzy precedence indices. First, the principles of processing of such data are presented, and then problems and suggested solutions. The latter refer to two notions: "expert agreement" as to the final precedence coefficients on the one hand and "resulting ordering" of the items on the other.

Keywords: ordering, fuzzy precedences, aggregation, agreement measures, consensus.

1. INTRODUCTION

In the development of practicable computer-based data analysis tools for decision aiding in multi-person, multi-item situations it is necessary to find means for acquiring, processing, aggregating and assessing data reflecting various opinions. This paper deals with a specific, but still sufficiently general, situation in which a number, m, of judges express opinion as to the precedence (based upon the comparison of value, of importance, of temporal sequence,...) of a number, n, of certain items.

Two fundamental problems appear: that of moving towards the consensus among experts, and of determining the resulting ordering of the items considered. When data obtained from judges are expressed in the form of orderings then the formulation proposed by Marcotorchino and Michaud (1979) yields simultaneous solution of the two problems, whether by application of the method developed by those authors or by the heuristic approach of Owsiński and Zadrożny (1986). Thus, in the space of orderings there exists such an ordering that minimizes the aggregate distance from the orderings given by judges.

Judges may, however, be unwilling or incapable of giving orderings. The requirement of providing orderings is a limitation which for some purposes can be treated as advantageous, since thereby judges are forced to provide in fact more information. Still, in many cases this requirement cannot be kept. Thus, precedence coefficients may appear with values greater than 0 and smaller than 1, indicating a vagueness of precedence. When this broadened space of data is accepted as solution space then the problems previously mentioned arise. Namely, it is obvious that within this space solutions can be found that are much nearer to all of the opinions than any of the orderings is.

The method described here is not aimed at direct resolution of this dilemma. Its purpose is more modest, namely to, provide the basis for a software that would accept, process and aggregate the data in a variety of ways, so as to highlight the contents of this dilemma and a number of other aspects of a decision-preparation session.

2. BASIC NOTATIONS

This section contains the basic notions used in the paper:

n - the number of items considered, $n=\text{card } I$, $i,j,l \in I$,

m - the number of judges assessing the precedence of items, $m=\text{card } K$, $k \in K$,

d^k_{ij} - degree of precedence of item "i" before item "j" as defined by judge "k" when considering "i" and "j" apart from other items, with $d^k_{ij} \in [0,1]$ where

$$d^k_{ij} = \begin{cases} 1, & \text{when i strictly precedes j,} \\ 0, & \text{when j strictly precedes i,} \end{cases}$$

(when $d^k_{ij} \in (0,1)$ then coefficients shall be referred to as "fuzzy", while for $d^k_{ij} \in \{0,1\}$ - as "crisp"),

$o^k(i)$ -serial number of item i in the ordering defined by precedences d^k_{ij}, whenever determination of such ordering is feasible and desired, the ordering itself being denoted o^k,

I^{kf}_{α} - subsets of I, numbered α, defined by judge k, for which precedences are given in different ways, f denoting type of data, and

$$I = \bigcup_{\alpha=1}^{a_k} I_\alpha^{kf} \qquad (1)$$

d_{ij} - aggregate precedence indices, i.e. variables forming the solutions to aggregation problems, with $\{d_{ij}\}_{i,j} = D$,

Q - proper objective functions.

More detailed notions shall be explained in the further course of this paper.

3. PURPOSE OF THE SOFTWARE

The purposes of this work are as follows: to secure a number of input management (data acquisition and processing) functions, to aggregate (fuzzy or crisp) precedence indices for particular judges, to calculate other indices, such as e.g. "degree of consensus", and to perform such session management functions, as determination of the "outlying" individual orderings, all of these in an interactive environment. The goal is to provide guidance for the multi-item-oriented (ordering-type) voting-based attempt at defining a common opinion (ordering), with a "sufficient" degree of agreement (consensus).

Thus, precedence aggregation would be but one of the functions performed. Before proceeding to proper aggregation the software system has therefore to acquire and process data obtained from individual judges.

4. INPUT FORMS

There are two fundamental forms of data input:

1. ordering of items, from which d_{ij}^k can be deduced, and
2. explicit values of d_{ij}^k,

but these two fundamental forms can be presented and mixed in a variety of ways.

Each judge, k, starts by dividing the set I into nonoverlapping nonempty subsets, each subset denoted I_α^{kf}. Index f is composed of two subindices, f', f", where f' indicates whether items in I_α^{kf} shall be ordered (f'=1), their preference coefficients given (f'=2) or "unknown precedence" (f'=3) flagged. The second index specifies choices within these three categories. At the stage of division of I into subsets, the sequence of items within subsets is of no importance.

Once I_α^{kf} specified, data concerning them, starting with $\alpha=1$, are given. The number of subsets for a judge may range from $a_k=1$ to entier(n/2)+1, with entier defined via strong inequality.

Within the subsets I_α^{kf} for particular f′ the following further choices can be made, indicated by values of f″:

1. f′=1, f″=1: items contained in $I_\alpha^{kf'f''}$ are simply unequivocally ordered, forming $0_\alpha^k(I_\alpha^k)$, so that for all $i,j \in I_\alpha^{k11}$ there is $d_{ij}^k \in \{0,1\}$,

   f″=2: 0_α^k shall contain an indifferent subsequence (only one), denoted 0_α^{kI}, so that for all $i,j \in 0_\alpha^{kI}$ there is $d_{ij}^k=0.5$, and for all the other $i,j \in I_\alpha^{k12}$ there is $d_{ij}^k \in \{0,1\}$,

   f″=3: within 0_α^k a subsequence shall be indicated (only one, as before) for which one or more alternative suborderings shall be given, $0_{\alpha\beta}^{kA}$, where β is the serial number of alternative subordering, and for the $i,j \in 0_{\alpha\beta}^{kA}$ there is

$$d_{ij}^k = \frac{1}{B^{k\alpha}} \sum_{\beta=1}^{B^{k\alpha}} d_{ij}^{k\beta}$$

where $B^{k\alpha}$ is the total number of alternative suborderings within the index subset α of the judge k, while for all the other $i,j \in I_\alpha^{k13}$ there is $d_{ij}^k \in \{0,1\}$.

2. f′=2, f″=1: all the d_{ij}^k are specified, one for each pair (i,j), i.e. 1/2n(n−1) numbers $\in [0,1]$,

   f″=2: only some d_{ij}^k are given, and all those which are not specified, are deduced from the given ones.

Besides the f index combinations shown there may be a situation, accepted by the system, in which, during the I_α^{kf} elicitation phase, not all $i \in I$ are considered, which means that for all the $i \in I - I_a^k$, where

$$I_a^k = \bigcup_\alpha I_\alpha^{kf} \qquad (2)$$

"unknown terminal precedence" is assumed, so that, by default,

for all $i,j \in I - I_a^k$ there is $d_{ij}^k = 0.5$, and

for all $i \in I_a^k$ and $j \in I - I_a^k$ there is $d_{ij}^k = 1$.

It is also possible to set $f^- = 3$, meaning "unknown precedence", taken to be

$f^- = 3$ ($f^+ = 1$): for items contained in I_α^{k3}, $i \in I_\alpha^{k3}$, there is $d_{ij}^k = 0.5$, for all $j \in I_a^k$.

5. FORMULATIONS OF THE AGGREGATION PROBLEM

Assume that all the d_{ij}^k, i.e. for all $(i,j) \in I \times I$ and $k \in K$, are given, specified and calculated in whichever manner.

The problem of aggregation of precedences takes on two forms, depending on whether the aggregate is to have the form of ordering, $d_{ij} \in \{0,1\}$, or can allow fuzzy precedences, d_{ij} $[0,1]$.

In the "crisp" case, the one of aggregate orderings the problem can be formulated along the lines set by Marcotorchino and Michaud (1979), i.e. in the form of the binary LP problem:

$$\max_D \{Q_1(D) = \sum_{i<j} (\hat{d}_{ij} d_{ij} + \hat{d}_{ji} d_{ji})\} \qquad (3.a)$$

subject to

$$d_{ij} \in \{0,1\} \quad \forall i,j \in I \qquad (3.b)$$

$$d_{ij} + d_{ji} = 1 \quad \forall i,j \in I \qquad (3.c)$$

$$d_{ij} + d_{ji} - d_{il} \leq 1 \quad \forall i,j,l \in I \qquad (3.d)$$

where

$D = \{d_{ij}\}_{n \times n-1}$, and

$$\hat{d}_{ij} = \frac{1}{m} \sum_{k=1}^{m} d_{ij}^k$$

The binary LP problem can be, in fact, solved as a usual continuous one via standard routines, insofar as substitution of (3.b) by

$$d_{ij} \in [0,1]$$

yields solutions satisfying (3.b).

The problem (3) consists in finding of an ordering, defined by d_{ij}, which is the closest to the common score given by $\hat{d}_{ij}$. The constraints (3.c) and (3.d) require antisymmetry (in the continuous case - completeness) and transitivity in the ordering relation.

In the second case, when aggregate precedence coefficients are allowed to be fuzzy, the problem can take the following form:

$$\min_{D}\{Q_2(D) = \sum_{i,j} |\hat{d}_{ij} - d_{ij}|^2\} \qquad (4.a)$$

subject to

$$d_{ij} \in [0,1] \qquad \forall i,j \in I \qquad (4.b)$$

$$d_{ij} + d_{ji} = 1 \qquad \forall i,j \in I \qquad (4.c)$$

$$d_{ij} + d_{jl} - d_{il} \leq 1 \qquad \forall i,j,l \in I \qquad (4.d)$$

which means approximation of $\{\hat{d}_{ij}\}$ by antisymmetric and transitive $\{d_{ij}\}$ within the same space.

6. FUZZY DATA CONSISTENCY

In terms of chronological description of the in-session proceeding, this section should precede the previous one, since it refers to functions performed by the software directly after the initial data acquistion and processing, described in section 4. Since, however, several notions to be used here belong to problem formulation, therefore this order was taken.

There are three questions related to consistency of fuzzy data on precedence coefficients as provided by the judges:

* completeness,
* transitivity,
* determination of the unspecified precedence coefficients (for $f'=2$, $f''=2$).

6.1. Completeness

The first of these questions is solved throughout the system by assuming that

$$d^k_{ji} = 1 - d^k_{ij} \tag{5}$$

with d^k_{ij} previously specified. Whenever a judge gives a d^k_{ji} after d^k_{ij} has been given, and (5) is not satisfied, the system displays both values and asks for the proper ones. This arbitrary assumption is justified not only by - at least- halving of the time and effort necessary to complete $\{d^k_{ij}\}$, but also by the need to avoid unduly and counterintuitional complexity of further operations if completeness is not kept to.

6.2. Transitivity

Transitivity requirement may be applied both to input data and to aggregation results, and it may be relaxed on both these ends. These relaxations have, of course, different meanings. Thus:

Remark 1: If (5) holds for all $k \in K$ and $i,j \in I$, and all $\{d^k_{ij}\}$ for $k \in K$ are transitive, i.e.

$$d^k_{ij} + d^k_{jl} - d^k_{il} \leq 1 \qquad \forall i,j,l \in I \tag{6}$$

then problem (4) can be solved without constraints (4.c) and (4.d) to yield $d_{ij} = \hat{d}_{ij}$.

Proof: There is

$$\hat{d}_{ij} + \hat{d}_{ji} = \frac{1}{m} \sum_{k=1}^{m} d^k_{ij} + \frac{1}{m} \sum_{k=1}^{m} d^k_{ji} = \frac{1}{m}\left(\sum_{k=1}^{m} d^k_{ij} + \sum_{k=1}^{m} d^k_{ji}\right) =$$

$$= \frac{1}{m} \sum_{k=1}^{m} (d^k_{ij} + d^k_{ji})$$

which, in view of (5),

$$= \frac{1}{m} \sum_{k=1}^{m} 1 = 1$$

Similarly

$$\hat{d}_{ij} + \hat{d}_{jl} - \hat{d}_{il} = \frac{1}{m} \sum_{k=1}^{m} d^k_{ij} + \frac{1}{m} \sum_{k=1}^{m} d^k_{jl} - \frac{1}{m} \sum_{k=1}^{m} d^k_{il} =$$

$$= \frac{1}{m} \sum_{k=1}^{m} (d^k_{ij} + d^k_{jl} - d^k_{il})$$

which, in view of (6),

$$\leq \frac{1}{m} \sum_{k=1}^{m} 1 = 1$$

Thus, completeness and transitivity hold with respect to $\hat{D}=\{\hat{d}_{ij}\}$ if they hold for all $D^k=\{d^k_{ij}\}$, $k\in K$. Since, additionally, $\hat{d}_{ij}\in [0,1]$ $\forall i,j\in I$, therefore $\hat{D}$ belongs to the feasible set for D, Ω_D, determined by all the constraints (4.b), (4.c) and (4.d). Hence, there exists $D=\hat{D}\in\Omega_D$, for which $Q_2(D)=0$, and therefore application of (4.b), (4.c) and (4.d) is not necessary. QED.

Remark 1 diminishes the computational effort necessary for aggregation. The advantage is however, not very great, since all $\{d^k_{ij}\}$, $k\in K$ still have to be transitive. That is why it is more advisable to secure transitivity in (4) while relaxing this requirement for the judge-provided data. Not only is that way of proceeding much easier computationally, but also it may justly be believed that averaging over a greater number m of judges would yield results violating the transitivity constraint to only a low degree, if at all (the similar would happen to the completeness constraint, were it necessary). Thus, though the system provides heuristic procedures for "transitivization", it is not advised to use them for each D^k, since this may make the session too lengthy and cumbersome.

Whether, however, transitivity constraints are relaxed or not at the D^k definition stage, it is important for the fuzzy context to consider the properties given below.

Note, first, that

<u>Remark 2</u>: $d^k_{ij}+d^k_{jl}-d^k_{il}\leq 2$ for all $d^k_{ij}, d^k_{jl}, d^k_{il}\in [0,1]$.

Thus, it may be useful to introduce the notion of <u>ε-transitivity</u>: a $D^k=\{d^k_{ij}\}_{i,j}$ shall be referred to as ε-transitive if

$$d^k_{ij}+d^k_{jl}-d^k_{il}\leq 2-\varepsilon \quad \forall i,j,l\in I \tag{7}$$

where, obviously $\varepsilon\in[0,1]$, with 0-transitivity characterizing all possible D^k's (Remark 2), and 1-transitivity corresponding to (6), i.e. usual transitivity.

<u>Remark 3</u>: If all D^k's, $k\in K$, are complete and ε-transitive then the solution D of problem (4), obtained in the absence of constraint (4.c) and (4.d) shall also be complete and ε-transitive.

Proof follows the one for Remark 1.

Now, two properties related to ε-tansitivity shall be introduced, of importance for algorithmic transitivization.

Remark 4: If $d^k_{ij} \in [\varepsilon, 1-\varepsilon]$ then (7) holds for these i,j and $l \in I-(i \cup j)$.

Remark 5: If $d^k_{ij} + d^k_{jl} \in [\varepsilon, 2-\varepsilon]$ then (7) holds for these i,j and l.

Validity of these two remarks is obvious. In fact, checking of (7) for all the permutations of (i,j,l) reduces to

$$d^k_{ij}+d^k_{jl} \le 2-\varepsilon \quad \text{for } (i,j,l),\ (j,l,i) \text{ and } (l,i,j) \text{ and} \tag{8}$$
$$d^k_{il}-d^k_{jl}-d^k_{ij} \le 1-\varepsilon \quad \text{for } (i,l,j),\ (j,i,l) \text{ and } (l,j,i)$$

from where Remarks 4 and 5 follow immediately. The first of them can be applied, though, only to $\varepsilon \le 0.5$, i.e. to quite low ε values, since violations of (6) beyond 0.5 would be counter-intuitive, though cannot be ruled out a priori.

6.3. Completion of a fuzzy D^k

This subsection presents a few remarks on calculation of implicit d^k_{ij}, when $f'=2$, $f''=2$. Such an option is chosen whenever specification of all the fuzzy d^k_{ij} for $i,j \in I$ is not feasible for some reasons. It should be mentioned, however, that first - a high degree of arbitrariness is introduced, and secondly - that with this option, if arbitrariness is not to go beyond acceptability, one must still specify quite a large share of the $1/2n(n-1)$ values of d^k_{ij}.

Assume, for instance, that d^k_{ij} and d^k_{jl} are given. Then, it seems plausible to calculate d^k_{il} according to

$$d^k_{il} = F(d^k_{ij}, d^k_{jl}) \tag{9}$$

where F is of the nature shown in Fig.1. Not only is such F intuitively, and also formally, acceptable, but it also ensures satisfaction of (6), even for fuzzy coefficients. In view of (5) an F can be used to determine any remaining single coefficient when the coefficients for two other index pairs are given, for any index triplet $(i,j,l) \in I^3$. Obviously, F brings a degree of

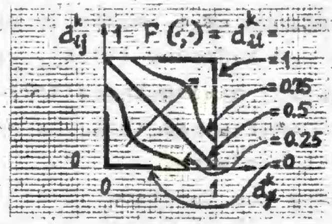

Fig. 1. Shape of a function F serving for determination of implicit precedence coefficients: $d^k_{il} = F(d^k_{ij}, d^k_{jl})$.

arbitrariness, which seems acceptable when both its arguments are specified by the judge. The question arises, however, whether it is acceptable to superimpose F over itself:

$$d^k_{ii'} = F(F(d^k_{ij}, d^k_{jl}), F(d^k_{lj}, d^k_{ji'}))$$

thereby importantly increasing arbitrariness of a part of data. The system would work in such a case, but certainly this suboption should not be abused. Even when unlimited superposition of F is allowed, at least n-1 values of d^k_{ij} must be specified.

7. AGGREGATION SOLUTIONS AND AGREEMENT MEASURES

7.1. Aggregation solutions

Solutions to the ordering problem (3), whether exact or suboptimizing, are obtained via methods described elsewhere - see Marcotorchino and Michaud (1979), Michaud (1981), and Owsiński and Zadrożny (1986). That is why only a few general remarks shall be forwarded here.

According to (3) the aggregation problem in its strict-crisp-ordering-oriented version is being solved through application of standard LP techniques. Because of the dimensions of the LP problem, $O(n^3)$, direct application of standard techniques for higher n encounters difficulties. The authors of the method suggest the use of simplified algorithms, but this does not yet solve the question of the in-session use of the software with simple, all-accessible hardware of the PC type. That is why the present authors introduced a heuristic suboptimization procedure, see Owsiński and Zadrożny (1986), referring to the objective function (3.a), but having the simplicity of classical progressive merger procedures, so that the software can be used in-session with modest hardware.

The simplified method uses the function (3.a) in its modified form:

$$Q_1(D,r) = r \sum_{i<j} \hat{d}_{ij} d_{ij} + (1-r) \sum_{i<j} \hat{d}_{ji} d_{ji} \qquad (10)$$

where $r \in [0,1]$ is a weighting coefficient, whose changes accompany the working of this algorithm. There is, obviously,

$$\arg\max_D Q_1(D,1) = \{1,2,\ldots,n\} = D^{opt}(1) = D_I \qquad (11)$$

and if proper d_{ij} are equal zero then there are more maximizing D's than D_I. On the other hand, there is

$$\arg\max_D Q_1(D,0) = \{n,n-1,\ldots,2,1\} = D^{opt}(0) = \bar{D}_I \qquad (12)$$

$\max_D Q_1(D,1/2)$ corresponds to the optimum for (3). By changing r from 1 to 0 one gets a finite series of different $D^{opt}(r)$ going from D_I to $\bar{D}_I$. Each such $D^{opt}(r)$ would be valid for a segment Δr^t, where t denotes the step number, with

$$\bigcup_t \Delta r^t = [0,1] \text{ and } \Delta r^{t-1} \cap \Delta r^t = r^t, \quad r^0 = 1.$$

The simplified method does not solve this series of LP problems equal to parametrization of the LP problem (3), but proceeds in an approximative way: starting with the ordering D_I it applies a sequence of transformations taken from a limited class, so that $\bar{D}_I$ is finally reached. At every step $\hat{r}^t$ and $Q_1(D^t,\hat{r}^t)$ are calculated.

As far as problem (4) is concerned its solution reduces to mere transitivization of $\hat{D}$, whenever deemed necessary.

7.2. Agreement measures

Certainly, the values of the very objective functions Q_1 and Q_2 provide some measures of agreement: Q_1 - agreement as to the aggregate ordering and Q_2 - agreement among the experts with no reference. Without, however, some knowledge about the scales of their values, actual assessment of the degree of agreement is doubtful.

Thus, with respect to Q_1, when completeness holds and X are subject to usual constraints:

<u>Remark 6</u>:

$$\arg\min_{\hat{D}} \max_{X} Q_1(\hat{D},X) = \{\tfrac{1}{2}\}_{i,j}, \text{ and}$$
$$\min_{\hat{D}} \max_{X} Q_1(\hat{D},X) = \frac{1}{4}n(n-1) \tag{13}$$

while

$$\max_{\hat{D}} \max_{X} Q_1(\hat{D},X) = \frac{1}{2}n(n-1), \tag{14}$$

corresponding to any argument $\hat{D}$ representing an ordering.

Using the results (13) and (14) the following relative measure can be proposed:

$$M_1^1(\hat{D}) = \frac{Q_1^{opt}(\hat{D}) - \frac{1}{4}n(n-1)}{\frac{1}{4}n(n-1)} = \frac{Q_1^{opt}(\hat{D})}{\min_{\hat{D}} Q_1^{opt}(\hat{D})} - 1 \tag{15}$$

with $M_1^1(\hat{D}) \in [0,1]$, reaching zero for $\hat{D} = \{\frac{1}{2}\}_{i,j}$ and 1 for $\hat{D}$ representing an ordering. Note that $M_1(\hat{D})$ does not reflect that much the agreement <u>among</u> judges as the agreement with respect to <u>ordering</u> of items. Thus, $\hat{D} = \{\frac{1}{2}\}_{i,j}$ can be obtained if $d_{ij}^k = \frac{1}{2}$ for all $i,j \in I$ and <u>all</u> $k \in K$. On the other hand, $M_1(\hat{D})$ reaches its maximum when <u>all</u> the judges $k \in K$ give d_{ij}^k (explicitly or implicitly) corresponding to the <u>same</u>, consensory, ordering.

Another measure takes the form of

$$M_1^2(\hat{D}) = \frac{\max_X Q_1(\hat{D},X) - \min_X Q_1(\hat{D},X)}{\max_X Q_1(\hat{D},X)} \tag{16}$$

with $M_1^2(\hat{D}) \in [0,1]$ reaching zero when $\max_X Q_1(\hat{D},X) = \min_X Q_1(\hat{D},X)$, i.e. when $\hat{D} = \{\frac{1}{2}\}_{i,j}$, and 1 when $\hat{D} \in E_0$, E_0 being the space of orderings. The latter results from:

<u>Remark 7</u>:

$$\min_X Q_1(\hat{D},X)\Big|_{\hat{D} \in E_0} = 0 \tag{17}$$

Both measures are easy to calculate, since the minimum appearing in (16) can be obtained from X maximizing $Q_1(\hat{D},X)$ by reversing the ordering obtained thereby.

With regard to the agreement among judges without any "outer" reference, the proper absolute measure can be based upon the objective function (4.1) yielding:

$$M_2^A(\{D_k\}) = \frac{2}{n(n-1)} \frac{1}{m} \sum_k \sum_{i<j} |d_{ij}^k - \hat{d}_{ij}^k| \tag{18}$$

Again, this absolute measure has to be scaled against the scope of its values:

$$M_2(\{D_k\}) = \frac{\max\limits_{\{D_k\}} M_2^A(\{D_k\}) - M_2^A(\{D_k\})}{\max\limits_{\{D_k\}} M_2^A(\{D_k\})} \tag{19}$$

so that the relative values of the measure M_2 range from 0 to 1, reaching 0 for the maximum diversity of opinions and 1 when all the precedence coefficients given by all the judges are identical: $d_{ij}^k = \hat{d}_{ij}$ $\forall k \in K$, $i,j \in I$. For the sake of simplicity an approximate measure can be used, namely

$$\tilde{M}_2(\{D_k\}) = 1 - 2 \cdot M_2(\{D_k\}) \tag{20}$$

resulting from

<u>Remark 8</u>: for even m: $\max\limits_{\{D_k\}} M_2(\{D_k\}) = \frac{1}{2}$

while for odd m: $\max\limits_{\{D_k\}} M_2(\{D_k\}) = \frac{1}{2} \frac{(m-1)(m+1)}{m^2}$ (21)

the latter approximating $\frac{1}{2}$ from below sufficiently well for higher m.

8. ALGORITHMS

This section presents the outlines of three procedures, decisive for the working of the system.

8.1. <u>Suboptimal aggregation of orderings</u>

As mentioned previously, this procedure refers to Q_1 of (3.a), but suboptimizes it only, since the class of operations to be preformed on an ordering is strictly limited. The procedure is, roughly, as follows:

1. t, step number, =1, $r^t=1$, $0^t=\{1,2,3,\ldots,n\}$, where n-total number of items ordered, and r is as in (10).

2. $t=t+1$.

3. Calculate, for all $j,l \in I$, $j<l$:

$$r^t_{jl} = \frac{\sum_{i\in 0^{t-1}_{l-}(j,l-1)} d_{li} - \sum_{i\in 0^{t-1}_{l+}(j,l-1)} d_{li}}{\sum_{i\in 0^{t-1}_{l-}(j,l-1)} (d_{li}+d_{il}) - \sum_{i\in 0^{t-1}_{l+}(j,l-1)} (d_{li}+d_{il})}$$

and (22)

$$r^t_{lj} = \frac{\sum_{i\in 0^{t-1}_{j+}(j+1,l)} d_{ij} - \sum_{i\in 0^{t-1}_{j-}(j+1,l)} d_{ij}}{\sum_{i\in 0^{t-1}_{j+}(j+1,l)} (d_{ij}+d_{ji}) - \sum_{i\in 0^{t-1}_{j-}(j+1,l)} (d_{ij}+d_{ji})}$$

where, for instance,

$$0^{t-1}_{j+}(j+1,l) = \{i \mid j+1 >_{0^{t-1}} i >_{0^{t-1}} l,\ i>j\} \quad (23)$$

with $>_{0^{t-1}}$ denoting precedence according to 0^{t-1}. The two r^t series of values correspond, respectively, to the following operations:

$r^t_{jl}: \ldots j i_1 i_2 \ldots \ldots l \ldots \rightarrow \ldots l j i_1 i_2 \ldots \ldots$

$r^t_{lj}: \ldots j \ldots i_2 i_1 l \ldots \rightarrow \ldots i_2 i_1 l j \ldots \ldots$

(Should the class of transformations of 0^{t-1} allow actual optimization, calculation of appropriate r^t would be much more complex, referring to the LP algorithms.)

4. Find $\max_{j,l}\{r^t_{jl}, r^t_{lj}\} = r^t_{j*l*} = r^t$.

5. Execute the operation corresponding to r^t_{j*l*} and thereby form 0^t.

6. If 0^t is the reverse of 0^1 or $r^t=0$ then go to 7., otherwise return to 2.

7. Stop.

It can easily be seen that whatever this procedure loses on optimization, it more than compensates in simplicity, especially so since experience shows that optimal and suboptimal results are either identical or sufficiently close.

8.2. Calculation of implicit d_{ij}^{k}

The first assumptions is that calculation of implicit d_{ij}^{k} is not made at every new value of d_{ij}^{k} given by a judge, but at an explicit request, so as not to unnecessarily multiply the operations. At each such request for every unspecified d_{ij}^{k} a search is made for such an l that in the triplet i,j,l two precedences are given, so that d_{ij}^{k} can be calculated through F. Two problems appear: first, of calculating implicit d_{ij}^{k} on the basis of already calculated implicit "third" coefficients, and second, of situations with more than one such "third" coefficient. With regard to the first problem an option is assured under which once the first search for the "third index", l, is completed for all the pairs corresponding to unspecified d_{ij}^{k}, the subsequent searches are initiated until some search is totally unseccessful. This, however, significantly prolongs the whole procedure. The other problem results from existence of multiple "third indices" which may serve to calculate a specific coefficient. It is assumed that, for a given level of search for the "third indices", once a d_{ij}^{k} was calculated for an l, the subsequent "third indices" l', l",... are treated in the following way:

$$d_{ij}^{k,s} = \frac{1}{s}\left[d_{ij}^{k,s-1}\cdot(s-1)+F(d^{k}_{i,l^{s}},d^{k}_{l^{s},j})\right]$$

where s is the current number of the "third index" for i,j, and, of course, the arguments of F can appear in any other proper combination.

8.3. Transitivization

Again, as with calculation of implicit d_{ij}^{k}, transitivization has to be asked for, and by default is performed after all d_{ij}^{k} had been calculated. The transitivity level ε^{T} has to be specified, together with an option choice referring to the way of proceeding and to the information provided. The option choice points out whether the procedure should go at once to the final ε^T level specified or whether it should pass through intermediate levels with a predetermined step, indicating first the transitivity level, ε^0, of the raw data. The second approach, although slower, does not only provide additional information, but also may give more plausible results. The working, in general, is as follows: for every triplet of indices $\in I$, for which it is feasible, ε-transitivity is checked using Remarks 4 and 5. When ε-transitivity is violated, this precedence index which is the closest to 0 or to 1 (see Remark 4) is altered so as to secure ε-transitivity. It is obviously sufficient to alter just one value, but when such alteration exceeds a certain threshold, specified in terms of ε, most often $\frac{1}{2}\varepsilon$, then the second-in-rank coefficient is also altered.

Another choice problem results from coexistence of explicitly and implicitly given d^k_{ij}. No assumptions as to that are made as of now, however, since there exist two contradictory arguments:

* most of the intuitively obvious functions F preserve transitivity, so that if violation of this condition occurs, it is due to the explicit d^k_{ij}, but
* the session is oriented at the judge-generated data, and not at the computer-generated ones.

Transitivization is always preceded by completion of $\{d^k_{ij}\}$ or by calculation of the implicit d^k_{ij}'s.

The question of indication of the way of increasing agreement is dealt with quite easily through determination of the "closest" and "farthest" opinions among the D^k's.

9. CONCLUSIONS

The paper presents a complete framework for the group decision / decision aiding session software. From the outline presented one may easily conclude that management of such a session, though requiring some preparation for the session manager, is not only quite feasible, both hardware- and software-wise, but may lead to valuable results, which could not be obtained with the methods to date.

REFERENCES

Marcotorchino, J-.F. and P. Michaud (1979): Optimisation en Analyse Ordinale des Données. Masson, Paris.

Michaud, P. (1981): Agrégation des préferences. Ph. D. Thesis. University Paris VI.

Owsiński, J.W. and Sł. Zadrożny (1986): Structuring a Regional Problem: Aggregation and Clustering in Orderings. Applied Stochastic Models and Data Analysis, 2, 1 and 2, pp.83-95.

Owsiński, J.W. and Sł. Zadrożny (1986): Clustering for ordinal data: a linear programming formulation. Control & Cybernetics, 15, 183-194.