THE EXPERT SYSTEM SHELL EQUANT-PC: BRIEF INFORMATION

PETR HÁJEK, MARIE HÁJKOVÁ, TOMÁŠ HAVRÁNEK, MILAN DANIEL

An experimental expert system shell is described. The system allows the user to choose the operation for combination of various information sources. Theoretically it relies on the algebraic theory of uncertainty processing in rule based systems due to Hájek and Valdes. Comparative approach is stressed.

1. INTRODUCTION

EQUANT is a rule based expert system shell which is MYCIN-like (extensional) and in many respects rather standard; but its specific characteristic is in the fact that the user (consulting the system filled by a knowledge base) can choose the combining function determining how uncertain information is propagated. The combining function in question defines how joint effect of two rules with the same conclusions ($A \Rightarrow C$ and $B \Rightarrow C$, say) is computed: if the contribution of the two rules are x, y respectively then the joint contribution of these two rules is $x \oplus y$. Pseudoprobabilistic derivations of a formula for $x \oplus y$ (under unrealistic conditional independence assumptions) can have only a heuristic value; this is why in [2], [6] probability was disregarded and instead some natural algebraic axioms for $\oplus$ were formulated. One says "weight" instead of "belief". There are two extremal weights: *certainly yes* (maximal) and *certainly no* (minimal). It turns out that *non-extremal* weights together with the operation $\oplus$ form an *ordered Abelian group.* Under some assumptions this group determines all remaining combining functions necessary for the inference engine. More generally, we may work with *intervals* of weights and have a kind of "interval arithmetics" for weights based on the (algebraized) Dempster's rule. All this is reflected in EQUANT.

The first generation of EQUANT (see [4], [5]) relied theoretically on [2], was implemented in PL/1 and run on IBM 370 and compatible mainframes (in batch mode or interactively using TSO). It just offered the user the choice from ten specific operations presented as examples in [2]; for some experiments see [8].

The new version relies theoretically on [6] and offers the user, in a sense specified below, choice from the full offer of all possibilities for combining functions. Comparative approach is stressed: not the weights (degrees of belief) as numbers are important but only the ordering of goals according to their weights (i.e. which diagnosis is the best, which the next etc.). For formalization of this see [6] and [2]. The new generation is being implemented for IBM-PC and compatible personal computers.

2. THEORETICAL BACKGROUND

2.1. A *knowledge basis* consists of *propositions* (numbered by natural numbers $1, 2, \ldots, N$) and *rules*. Each rule has the form $A \Rightarrow C(w)$ where A (antecedent) is an elementary conjunction of (some) propositions (e.g. empty conjunction, 1, -2, $1 \,\&\, -3$, $1 \,\&\, 2 \,\&\, 4 \,\&\, -5$ etc.), C (succedent) is a proposition not occurring in A and w is a *designated value* – an integer such that $-10 \leqq w \leqq 10$. Here 10 means "certainly yes", -10 "certainly no", and integers $-9, \ldots, 9$ are intermediate comparative degrees of belief. The system of rules must be loop-free (see [6] or [2]. During a consultation the user is asked to ascribe its degree of belief to some propositions (questions); he may answer by a designated value w or, more generally by an interval, i.e. pair (w_1, w_2) of designated values such that $w_1 \leqq w_2$. (w may be identified with the pair (w, w).) By answering questions the user defines a *questionnaire*, i.e. a partial mapping of propositions into designated values of a generalized questionnaire (a mapping from propositions to intervals).

2.2. A *realization* of designated values is a one-to-one monotone mapping r of designated values into the closed interval $[-1, 1]$ of reals such that $r(10) = 1$ and $r(-w) = -r(w)$. $[-1, 1]$ is endowed with the operation $\oplus$ defined as follows:

$$x \oplus y = \frac{x + y}{1 + xy}.$$

$P = [-1, 1]$ with this operation, the unary operation $-x$ and the usual ordering $\leqq$ of reals is the *standard* (*PROSPECTOR*) *combining structure*. The set $D = \{(u, v) \mid u, v \in [-1, 1] \text{ and } u \leqq v\}$ is endowed with the operation $\oplus$ defined as follows:

$$(a, b) \oplus (c, d) = \left(\frac{a + b + c + d + ad + bc - 2ac}{(b - a) + (d - c) + ad + bc + 2}, \frac{a + b + c + d - ad - bc + 2bd}{(b - a) + (d - c) + ad + bc + 2}\right).$$

Further put $h(a, b) = (a + b)/(b - a + 2) - (a, b) = (-b, -a), (a, b) \leqq (c, d)$ iff $h(a, b) < h(c, d)$ or $(h(a, b) = h(c, d)$ and $a \leqq c)$. The set D with the operation $\oplus$, $-$ and the ordering $\leqq$ defined above is the *standard Dempster combining structure* (see [6]; here we transform pairs (x, y) such that $0 \leqq x, y, x + y \leqq 1$ into intervals (a, b) where $a = 2x - 1$, $b = 1 - 2y$.) Observe that up to the above identification, P is a substructure of D; moreover, h maps D homomorphically to P.

2.3. The answers of the user *plus* the realization of designated values determine for each proposition i which has beed a goal of the consulation a unique element of D (which belongs to P if the user does not use intervals) called the *numerical weight* of i. The computation may be done by the usual backward chaining method, using $\oplus$ for joining contributions of rules with the same succedent. The *comparative result* of the consultation is the quasiordering of goals according to their numerical weights. (This can be refined by telling for each goal that its weight lies strictly between realizations of the successive designated values w, w' or that it coincides with the realization of w.)

2.4. Our approach is *general* in the following sense (here we disregard intervals for simplicity): One can use realizations of designated values in each combining structure G that results from an ordered Abelian group G by adding a top element $\top$ and a bottom element $\bot$ and putting $\top \oplus x = \top$, $\bot \oplus x = \bot$ (where $\oplus$ is the group operation), see [6]. Call (G, r) and (G', r') *equivalent* for the knowledge base Θ if for each questionnaire q, the comparative result of the run given by (Θ, G, r, q) is the same as that given by (Θ, G', r', d). (This is to say that if the user is told the comparative result he is unable to distinguish whether the inference machine computes with (G, r) or with (G', r').)

2.5. Main Theorem. Let Θ be given. For each realization r of designated values in a combining structure G there is a realization r' in P such that (G, r) and (P, r') are equivalent for Θ (for the proof see [6]).

3. THE SHELL

3.1. General design choices. The *aim* is to create a system leading the user to the comparative (non-numerical) understanding of the results: to the mistrust to particular numbers and particular realization of designated values. This is because numerically the MYCIN-like systems give results incoherent as probabilities except if a special care is made (cf [3]). Thus the system must allow the user to *choose* a realization and to compare results given by the same answers but different realizations of designated values. The system may also serve for experiments with uncertainty processing.

The system is being implemented on a personal computer (IBM-PC compatible), in Turbo-PROLOG. As usual, it consists of two big parts, EQUANT-F (filling), accepting a knowledge base, and EQUANT-R (run), conducting a consultation using a given knowledge base.

3.2. Filling. Tasks: Input of control parameters and of the knowledge base, elaboration of an interval realization of the knowledge base, checkings, printed information on request, possibly transformation of the knowledge base understood as a belief system (cf. [3]) into a knowledge base generating a belief system compatible with the former one (cf. also [1]).

3.3 Run. Basic components (modules) are the same as in the first generation EQUANT:
- input module
- core module
- final module

– module of dynamic control
– module communicating with the base of runs.

The module of dynamic control yields standard explanations (HOW, WHY etc.) and possibilities of change of control, e.g. immediate stop.

There is a base of runs made in the past (using the given knowledge base; a run made in the past may be used in a present run (see below).

3.4. Input module. The following is determined:
(1) name of the run,
(2) *goals* of the consultation (propositions whose weight is to be compared),
(3) details of use of the *base of runs* (whether, how and which past run will be used; answers of the user in the past run may be either accepted for the present run or only suggested for information),
(4) *volunteered information*,
(5) *verbosity degree*,
(6) *realization(s) of designated values* (this is most important!) how many and which ones (equidistant, i.e. PROSPECTOR, a realization equivalent to EMYCIN's group with equidistant realization, user's own realization(s), randomly generated realization(s)).

3.5. Core module. Rule processing and proposition processing call each other recursively: standard backward chaining with optimalization (unnecessary questions are not asked). The combining operation for joining contributions of rules is just the Dempster operation as presented above. Each user's answer is either an interval (w_1, w_2) or a single designated value w (understood as (w, w)).

3.6. Final module. Output of various overviews of results, saving the run in the base if requested, start of a run or stop. The user may want to see most details on results given by a chosen realization or, on the other hand, condensed overview of results given by all used realizations. On a special request the user may see numerical results; but he is warned not to take them too seriously. Various other kinds of information are possible.

3.7. Example. For experiments, a knowledge base for cluster analysis was used. In a run, one can consider 10 goals:

No:	Goal
2	bmdp 1M, measure is corr, linkage is complete
1	bmdp 1M, measure is corr, linkage is single
6	bmdp 1M, measure is abscorr, linkage is average
3	bmdp 1M, measure is corr, linkage is average
5	bmdp 1M, measure is abscorr, linkage is complete
4	bmdp 1M, measure is abscorr, linkage is single
11	bmdp 1M, measure is absang, linkage is complete
9	bmdp 1M, measure is angular, linkage is average
17	bmdp 2M, euclid, linkage is centroid, no standardization
26	bmdp 3M, number is 10

These goals refer to the usage of BMDP programs for cluster analysis as described in [8], where the above ten goals from the total of 48 goals were chosen as goals with highest weights in an experimental run.

The user can choose, for example, the Prospector equidistant realisation of the designated values (see 2.1) and four additional random realisations. After answering questions, the following overview can be obtained using the services of the final module:

Realization	Goals
1	9 > 17 = 3 = 1 = 2 > 11 > 26 > 4 = 5 = 6
2	3 = 1 = 2 = 17 > 9 > 11 > 4 = 5 = 6 > 26
3	9 > 3 = 1 = 2 = 17 > 11 > 26 = 4 = 5 = 6
4	9 > 3 = 1 = 2 > 11 > 26 = 17 > 4 = 5 = 6
5	3 = 1 = 2 > 4 = 5 = 6 > 9 > 17 > 11 > 26

For the ordering of goals, the ordering defined in 2.2 above was used. The user can see that for all realisations, e.g., 9 > 11 and 9 > 26 etc., but on the other hand the results are dependent on realisations of designated values. For each single realisation, a more specific pattern can be obtained displaying the position of goals w.r.t. the designated values:

The 1 realisation

Designated values	10	9	8	7	6	5	4	3
Goals				>9	>17 = 3 = 1 = 2		>11	>26
Designated values			2	1	0	−1	−2	...
Goals	>4 = 5 = 6							

3.8. Concluding remarks. Practically, EQUANT-PC should help the user by encovering "white places" in a MYCIN-like inference engine where a choice is possible and may influence the results. Theoretically, it is hoped to be useful for experiments concerning interval weights and the comparative approach. Note that, as stated in [6], algebraisation is not the ultimate aim; rather, it is a means for an attempt to pose and discuss the question of probabilistic relevance of MYCIN-like systems on an appropriate theoretical level (see [3] on the last topic*).

4. PRESENT STATE OF IMPLEMENTATION

The program for EQUANT-F is under development (M. Daniel). The program for a truncated version of EQUANT-R has been written (M. Hájková); there is no base of past runs and no dynamic control, but a finished run may be used in the immediately following run. This version will not be developed further; a full version will be programmed. The present version may be used for experiments and is a useful preliminary result.

* P. Hájek had a talk on the content of [3] at WUPES.

REFERENCES

[1] P. Hájek: Combining functions in consulting systems and dependence of premisses — remark. In: Artificial Intelligence and Information-Control Systems of Robots (I. Plander, ed.), North-Holland, Amsterdam 1984.

[2] P. Hájek: Combining functions for certainty factors in consulting systems. Internat. J. Man-Machine Studies *22* (1985), 59—67.

[3] P. Hájek: Towards a probabilistic analysis of MYCIN-like expert systems (working paper). In: COMPSTAT 1988, Physica-Verlag, Heidelberg 1988, pp. 117—122.

[4] P. Hájek and M. Hájková: The consulting system EQUANT — brief description and user's manual. In: New Enhancements in GUHA Software (P. Hájek, ed.), Preprint No. 8, Math. Inst. ČSAV Prague 1984.

[5] P. Hájek, M. Hájková and T. Havránek: EQUANT-87, Apples '88, Dům techniky, České Budějovice 1988, pp. 20—26.

[6] P. Hájek and J. Valdés: Algebraic foundations of uncertainty processing in rule-based expert systems, submitted (also: Preprint No. 28, Math. Inst. ČSAV, 1987).

[7] T. Havránek: An interpretation of Hájek-Valdés results on the Dempster's semigroup. In: AI '88 (V. Mařík, ed.), Ústav pro informační systémy v kultuře, Praha 1988.

[8] T. Havránek and O. Soudský: Using an expert system shell for setting statistical package parameters. Computat. Statist. Quart. *4* (1988), 3, 159—171.

RNDr. Petr Hájek, CSc., Matematický ústav ČSAV (Mathematical Institute — Czechoslovak Academy of Sciences), Žitná 25, 115 67 Praha 1. Czechoslovakia.

RNDr. Marie Hájková, CSc., RNDr. Tomáš Havránek, CSc., RNDr. Milan Daniel, Středisko výpočetní techniky ČSAV (Institute of Computer Science — Czechoslovak Academy of Sciences), Pod vodárenskou věží 2, 182 07 Praha 8. Czechoslovakia.