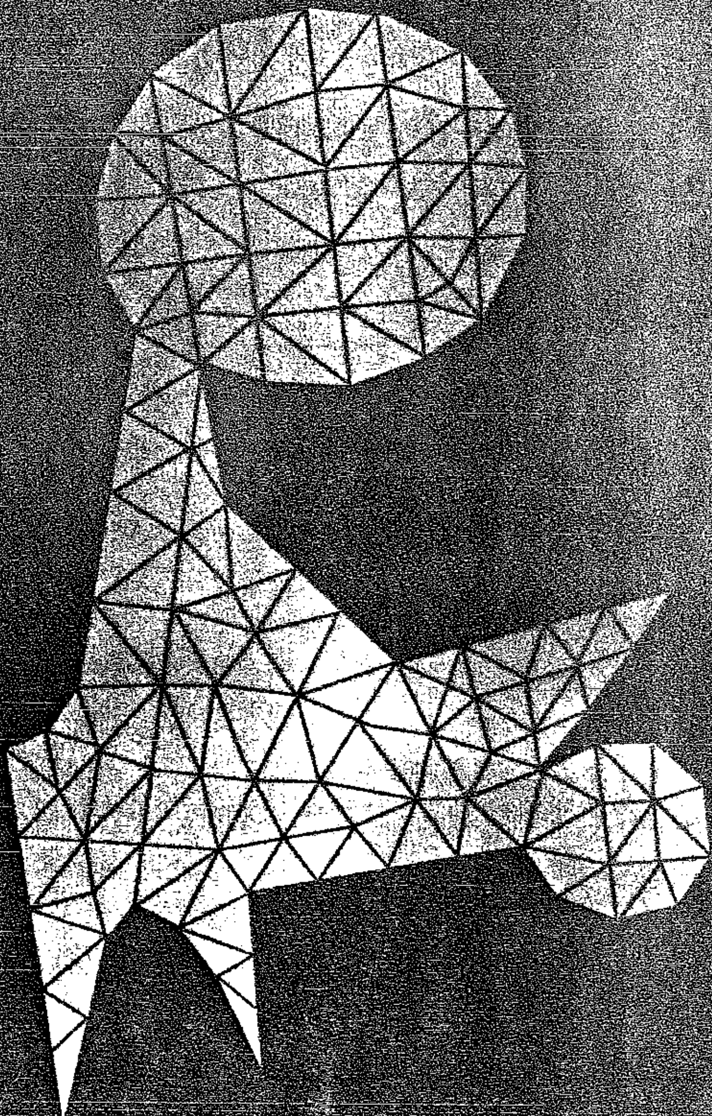
Abraham Kandel

FUZZY EXPERT SYSTEMS



Chapter 15

MILORD: A FUZZY EXPERT SYSTEMS SHELL

R. López de Mántaras, J. Agusti, E. Plaza, and C. Sierra

TABLE OF CONTENTS

I. Introduction 214

II. The Knowledge Representation 214

III. Forward, Backward, and Their Combination 214
- A. The Lookahead Prospection Technique 215
- B. The Rule Selection Criteria 215

IV. The Management of Uncertain Reasoning 216
- A. The Calculus of Uncertainty 216

V. The Linguistic Certainty Values 217
- A. A Default Term Set and Its Representation 218
- B. The Linguistic Approximation 219

VI. The Knowledge Elicitation Module of Expertise Transfer 219
- A. Elicitation 221
- B. Inference Analysis 221
- C. Inference Validation and Refining Environment 222

VII. Concluding Remarks 222

References 223

I. INTRODUCTION

This paper describes MILORD, an expert systems building tool containing a knowledge elicitation module and two inference engines (forward and backward) with uncertain reasoning capabilities based on fuzzy logic. MILORD allows the user to express the degree of certainty by means of expert-defined linguistic statements and provides the possibility to choose among three different calculi of uncertainty corresponding to three different models of the *and*, *or* and *implication* connectives.

The switching between the two engines is transparent to the user. MILORD has two types of control strategies: one consists of a lookahead technique that allows the user to detect, in advance, whether or not the linguistic certainty value of a conclusion will reach a minimal threshold acceptance value. The other concerns the selection of rules according to several criteria. MILORD also contains a limited, but useful, explanation module as well as a rule editor, not described in this chapter.

II. THE KNOWLEDGE REPRESENTATION

The knowledge base consists of facts and rules. The facts are LISP atoms associated with a linguistic certainty value. A nonevaluated fact will have the value *nil* and, therefore, is very fast to check if a given fact is known, i.e., if a certainty value has been assigned to it.

Every rule has a set of conditions which, when evaluated with a certain degree of linguistic certainty, leads to a conclusion whose degree of linguistic certainty depends on the degrees of the conditions. The rules are externally represented as follows:

(Rule rule — number (If conditions) [vc] (Then conclusions))

where [vc] is the linguistic certainty value of the rule.

In order to enable a fast access to the rules, MILORD translates the preceding list into the following internal representation that uses the LISP property lists:

$$\text{Rule-N} \rightarrow \text{VAL[vc] IF } (p_1,\ldots,p_N) \text{ THEN } (c_1,\ldots,c_M)$$

where *VAL*, *IF*, and *THEN* are properties of the atom *rule*. The access to the conditions and conclusions of a rule is then an access to the properties of an atom.

The internal representation of the rules builds, for each conclusion, a property list which is the list of rules that deduce this conclusion, together with the linguistic certainty value of each rule, i.e.,

$$\text{Conclusion} \rightarrow \text{Rules } ((\text{rule}_1\ \text{vc}_1)\ldots(\text{rule}_k\ \text{vc}_k))$$

where the rules in this list are listed in decreasing order of their linguistic certainty values. This ordering will be used by the lookahead control strategy that will be described later.

III. FORWARD, BACKWARD, AND THEIR COMBINATION

The forward reasoning starts with a set of given facts and its goal is to deduce a hypothesis whose linguistic certainty value reaches a given acceptance threshold. If the forward reasoning gets to a hypothesis whose certainty value is below the threshold, the backward reasoning is called in order to try to increase this certainty value by considering, through a lookahead process, other rule-paths that would conclude the same hypothesis with a higher certainty.

A. THE LOOKAHEAD PROSPECTION TECHNIQUE

MILORD applies a prospection process from the hypothesis toward the external (non-deducible) facts in such a way that at any time it checks if the certainty value of the hypothesis can reach the acceptance threshold value. If not, it will consider a new hypothesis. Let us now briefly describe such a process with the following default operators, and for the *and*, *or* and → connectives, to perform the calculus of uncertainty (although the lookahead process is independent of the operators used):

$$v(A \text{ and } B) = \min(v(A), v(B))$$

$$v(C_{R1} \text{ or } C_{R2}) = \max(v(C_{R1}), v(C_{R2}))$$

$$v(C) = \min(v(R), v(P))$$

where A and B are conditions of a same premise, C_{R1} and C_{R2} represent the same conclusion deduced by the two rules R1 and R2, and C is the conclusion of rule R whose premise is P.

The preceding operators are used, respectively, in the evaluation of the satisfaction of the premise, in the combination of several rules with the same conclusion, and in the propagation of the uncertainty from the premise to the conclusion of a rule.

The lookahead process in the backward reasoning starts assuming that all the nonevaluated conditions of the rules leading to the same conclusion, have the highest linguistic certainty value among the ordered set of linguistic values defined by the expert. This allows to compute the highest possible certainty value that this conclusion could reach. If this value is higher than the acceptance threshold, the backward reasoning proceeds asking the user to assign a linguistic certainty value to the nonevaluated, nondeducibile conditions one by one. Each time a condition gets its value, it is propagated to the conclusion using the preceding formula, and if its certainty value is still higher than the threshold, the process proceeds asking for the value of the next nondeducible condition and so on until either the certainty value of the conclusion falls below the threshold (in which case MILORD calls back the forward reasoning mode to deduce another hypothesis), or all the nondeducible conditions have been assigned a certainty value. As far as the deducible conditions are concerned, the lookahead process is applied recursively to each one of them, as described, and its certainty value is also propagated toward the conclusion in order to keep checking if its certainty value is higher than the threshold, in which case the process resumes. If not, the forward reasoning mode will try to deduce a new hypothesis.

If the user initially gives a set of hypotheses, instead of a set of facts, MILORD calls the backward reasoning mode with one of the hypotheses and tries to validate it with a linguistic certainty value higher than the threshold, using exactly the same process described previously. If it fails, it tries another hypothesis, and so on until either one of them succeeds or all of them fail.

B. THE RULE SELECTION CRITERIA

The set of criteria to select rules has to be easily modifiable because the efficiency of any criterion depends on each particular application. In MILORD it is very easy for the user to modify or introduce criteria. The selection among a given set of criteria can, in some cases, be done automatically. For example, if a knowledge base only contains rules which have a single conclusion, any criterion based on the number of conclusions would not be considered. The criteria that, in addition to metarules, are available in MILORD are

1. The order of the rules
2. The linguistic certainty values

3. The number of conditions
4. The number of conclusions
5. The rule most recently used
6. The rule containing the most recently deduced fact in its premise

Furthermore, the user can combine several criteria according to a given priority. For example:

R1: Condition_1, $\text{condition}_2 \Rightarrow$ [absolutely-true] conclusion_1
R2: Condition_2, $\text{condition}_3 \Rightarrow$ [almost-true] conclusion_2
R3: $\text{Condition}_4 \Rightarrow$ [quite-true] conclusion_1

The extreme values corresponding to the following ordered criteria are

1. Maximum certainty value: absolutely-true
2. Maximum number of conclusions: 1
3. Minimum number of conditions: 1

In this case the system will try to select a rule, among the applicable ones, having a certainty value equal to "absolutely-true", and having one condition and one conclusion. If there is no rule satisfying these criteria, it will drop the last one (number of conditions) and so on until one or more rules are obtained. If several rules have been obtained, the user can use the rest of the criteria to end up with only one rule. In our example, after dropping the last criterion, the selected rule is R1.

IV. THE MANAGEMENT OF UNCERTAIN REASONING

The numerical approaches to the representation of uncertainty imply hypotheses of independence, mutual exclusiveness, etc. about the information they deal with. On the other hand, they oblige the expert and the user to be unrealistically precise and consistent in the assignment of such numerical values to rules and facts. Furthermore, these approaches are computationally expensive.

Our approach is based on a linguistic characterization of the uncertainty and follows the work of Bonissone.[3] The linguistic certainty values are terms defined by the expert. The internal representation of each term is a fuzzy number on the interval [0,1] characterized by a parametric representation for computational reasons.

MILORD has been parametrized in order to perform three different calculi of uncertainty operating on the expert defined term set of linguistic certainty values.

A. THE CALCULUS OF UNCERTAINTY

It can be shown[5] that triangular norms (t-norms) and triangular conorms (t-conorms) are the most general families of two-place functions from $[0,1] \times [0,1]$ to $[0,1]$, that satisfy the requirements of conjunction and disjunction operators, respectively.

A t-norm $T(p,q)$ performs a conjunction operator, on the degrees of certainty of two or more conditions in the same premise, satisfying the following properties:

$$T(0,0) = 0$$
$$T(p,1) = T(1,p) = p$$
$$T(p,q) = T(q,p)$$
$$T(p,q) \leq T(r,s) \text{ if } p \leq r \text{ and } q \leq s$$
$$T(p,T(q,r)) = T(T(p,q),r)$$

A t-conorm S(p,q) computes the degree of certainty of a conclusion derived from two or more rules. It is a disjunction operator satisfying the following properties:

$$S(1,1) = 1$$
$$S(0,p) = S(p,0) = p$$
$$S(p,q) = S(q,p)$$
$$S(p,q) \leq S(r,s) \text{ if } p \leq r \text{ and } q \leq s$$
$$S(p,S(q,r)) = S(S(p,q),r)$$

The propagation function P(p,r), giving the certainty value of the conclusion of a rule as a function of the certainty value of the premise and the certainty value of the rule itself, satisfies the properties of a t-norm.

For suitable negation operators N(x)[13], t-norms and t-conorms are dual in the sense of DeMorgan's law.

Some usual pairs of dual t-norms and t-conorms are

$T_0(x,y) = \begin{cases} \min(x,y) \text{ if...etc.} \\ 0 \end{cases}$ $\qquad S_0(x,y) = \begin{cases} \max(x,y) \text{ if...etc.} \\ 1 \end{cases}$

$T_1(x,y) = \max(0,x + y - 1)$ $\qquad S_1(x,y) = \min(1,x + y)$ (Luckasiewicz)

$T_{1.5}(x,y) = x{\cdot}y/[2 - (x + y - xy)]$ $\qquad S_{1.5}(x,y) = (x + y)/(1 + xy)$

$T_2(x,y) = x{\cdot}y$ $\qquad S_2(x,y) = x + y - xy$ (Probabilistic)

$T_{2.5}(x,y) = x{\cdot}y/(x + y - xy)$ $\qquad S_{2.5}(x,y) = (x + y - 2xy)/(1 - xy)$

$T_3(x,y) = \min(x,y)$ $\qquad S_3(x,y) = \max(x,y)$ (Zadeh)

It can be shown that they are ordered as follows:

$$T_0 \leq T_{1.5} \leq T_2 \leq T_{2.5} \leq T_3$$

$$S_3 \leq S_{2.5} \leq S_2 \leq S_{1.5} \leq S_1 \leq S_0$$

In MILORD we have implemented the pairs (T_1,S_1), (T_2,S_2), and (T_3,S_3), following the experimental results obtained by Bonissone,[3] which consisted of applying nine t-norms to three different term sets. Bonissone analyzed the sensitivity of each operator with respect to the granularity (number of elements) in the term sets and concluded that only the t-norms T_1, T_2, and T_3 generated sufficiently different results for term sets that do not have more than nine elements. On the other hand, according to the results of Miller[9] concerning the span of absolute judgment, it is unlikely that any expert or user would consistently qualify uncertainty using more than nine different terms.

V. THE LINGUISTIC CERTAINTY VALUES

MILORD allows the expert to define the term set of linguistic certainty values which constitutes the verbal scale that he and the users will use to express their degree of confidence in the rules and facts, respectively. Recent psychological studies[1] have shown the feasibility of such verbal scales. " . . . A verbal scale of probability expressions is a compromise between people's resistance to the use of numbers and the necessity to have a common numerical scale," according to Beyth-Marom.[1] " . . . People asked to give numerical estimations on a common-day situation err most of the time and in a nonconsistent way. Furthermore, they are unable to appreciate their judgment imprecision (errors are by far bigger than the maximum error accepted as possible by the subjects themselves). Nevertheless, judgments embodied in linguistic descriptors appear consistent in this same situation."[6]

Each linguistic value is represented internally by a fuzzy interval (fuzzy number), i.e., the membership function of a fuzzy set on the real line, or, more precisely, on the truth

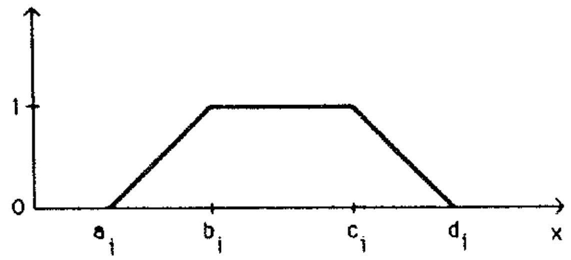


FIGURE 1. The trapezoidal function.

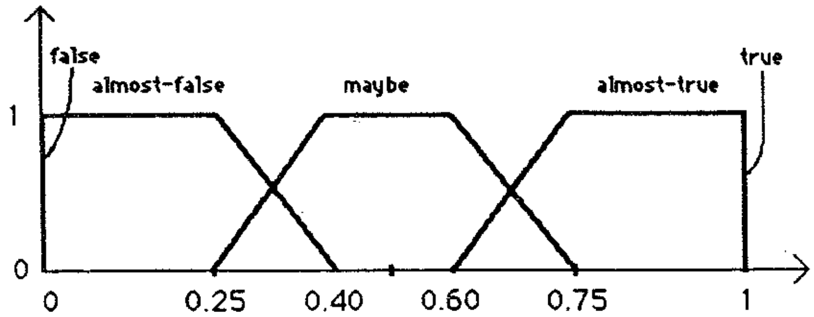


FIGURE 2. Five elements representation.

space represented by the interval [0,1]. These membership functions can be interpreted as the meanings of the terms in the term set. The conjunction and disjunction operators applied to these functions will produce another membership function, as a result that will have to be matched to a term in the term set, in order to keep the term set closed. This can be done by a linguistic approximation process that will be described later (see Bonissone[2] for an extensive study of the linguistic approximation process).

A. A DEFAULT TERM SET AND ITS REPRESENTATION

Although the expert can define its own term set together with its internal representation, MILORD provides the following default term set:

{False, almost-false, maybe, almost-true, true}

Each term T_i is represented by a membership function $\mu_i(x)$, for x in the interval [0,1].

For computational reasons, each membership function is represented by four parameters $T_i = (a_i,b_i,c_i,d_i)$, corresponding to the following trapezoidal function:

The five element default term set has the following representation:

False = (0,0,0,0)
Almost-false = (0,0,.25,.40)
Maybe = (.25,.40,.60,.75)
Almost-true = (.60,.75,1,1)
True = (1,1,1,1)

corresponding to the following functions in Figure 2.

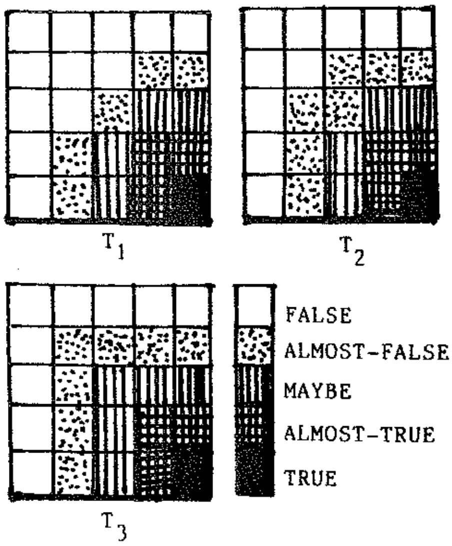
In order to be able to evaluate the t-norms T_1, T_2, T_3 and the t-conorms S_1, S_2, S_3 on the elements of the term set, we have applied the following formulas according to the arithmetic rules on fuzzy numbers.

Given two fuzzy intervals $I = (a,b,c,d)$ and $I' = (a',b',c',d')$, we have the following:

$$\begin{aligned} I+I' &= (a+a',b+b',c+c',d+d') \\ I-I' &= (a-d',b-c',c-b',d-a') \\ I*I' &= (aa',bb',cc',dd') \\ \mathrm{Min}(I,I') &= (\min(a,a'),\min(b,b'),\min(c,c'),\min(d,d')) \\ \mathrm{Max}(I,I') &= (\max(a,a'),\max(b,b'),\max(c,c'),\max(d,d')) \end{aligned}$$

B. THE LINGUISTIC APPROXIMATION

A linguistic approximation process is performed in order to find a term (linguistic value) in the term set whose "meaning" (membership function) is the closest (according to a given metric) to the "meaning" (membership function) of the result of the conjunction or disjunction operation performed on any two linguistic values of the term set. This allows us to maintain, closed, the operations for any t-norm and t-conorm. The problem is, therefore, that of computing a distance between two trapezoidal membership functions. In order to do so, we have adopted a simple solution consisting of the computation of a weighted Euclidean distance of two features of the functions: the first moment and the area under the function. The next figure shows the results obtained with the selected t-norms T_1, T_2, and T_3 on the default term set of Figure 2.



VI. THE KNOWLEDGE ELICITATION MODULE OF EXPERTISE TRANSFER

This section describes elicit-analyze-refine (EAR), a system and ancillary methodology for aiding knowledge engineers in the early phases of knowledge base design. That is to say, we focus on the top half of Figure 3 because we are convinced that much of the difficulty in knowledge acquisition lies in the fact that the expert cannot easily describe how he views a problem, because he may not distinguish between the facts or beliefs and the factors which influence his decision making. Much of the expertise lies in the way an experienced person views the problem, and this is a psychological issue that can be dealt in terms of the personal

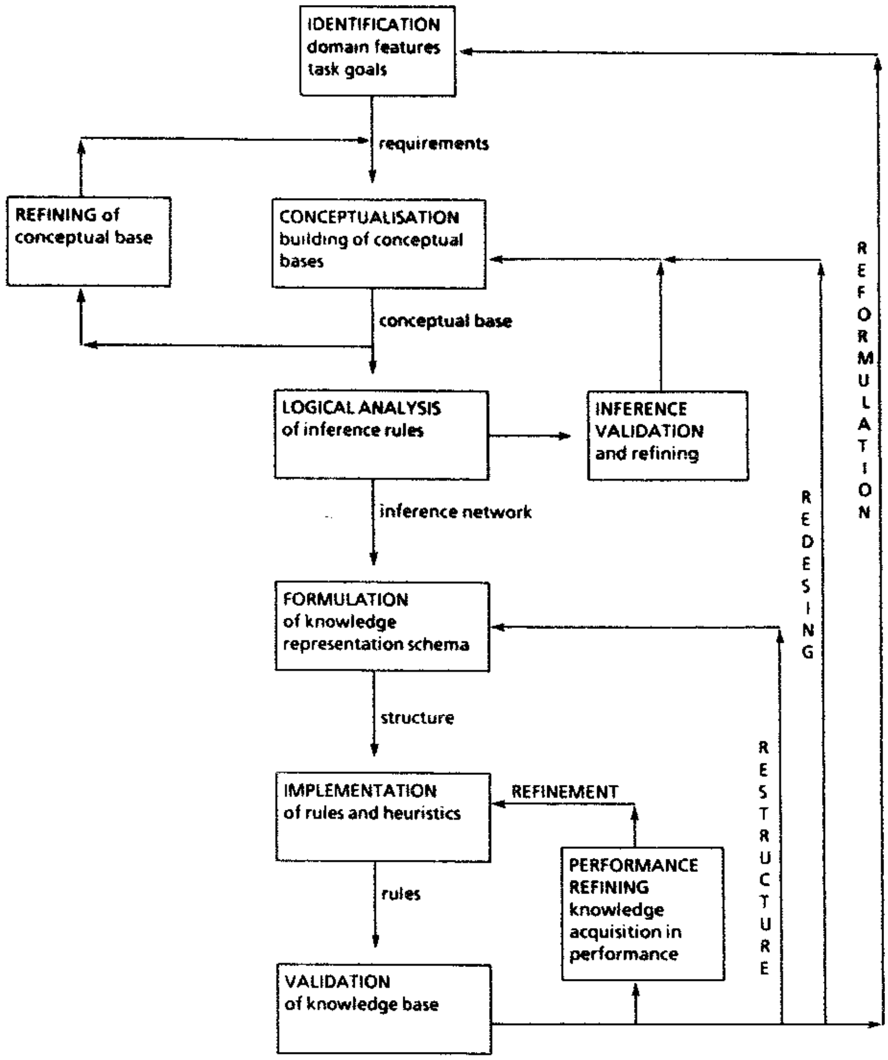


FIGURE 3. Knowledge acquisition stages.

construct psychology. The psychology of Kelly[7] views a human as a scientist classifying and theorizing about his world and basing his theory on the hypothesis that everybody has his own model of the world made up of personal constructs.

Based on this claim, the system conducts a dialogue with a domain expert eliciting relevant constructs and interactively detecting constructs poorly or ambiguously defined. Such constructs are fed back to the dialoguing expert for further refining. In this way, the expert is forced to investigate how he thinks about the problem at hand. This process builds up a repertory grid relating domain constructs with domain elements to which they apply. These relations are expressed by a contrastive set of linguistic labels, and are represented by possibility distributions, e.g., are of the form: E_i is Q^kj C_i, where E_i is an element, C_j is a construct pole, and Q^kj is the linguistic label relating them.

In the second stage, a logical analysis of the repertory grid shows the implication strength between the constructs and this allows us to generate an initial set of fuzzy rules, forming an inference network. This network is then presented to the expert who points out his disagreements and enters a refining stage that uses several techniques of the personal construct theory (PCT). The elicitation dialogue is also based on the PCT and on recent implementations of psychological analysis systems.[4,10,12]

EAR produces a validated rule set for knowledge base building through the following three-stage cycle:

1. Interactive elicitation and analysis (EINA) is a program to assist in the conceptualization by building a repertory grid through a guided dialogue.
2. Subjective inference logical analysis (ALIS) is a program that generates a tentative rule-set represented as an inference network.
3. Inference validation and refining environment (EVR) is a program containing several techniques for disagreement resolution (laddering, concrete explanations, etc.). At this point if disagreement remains, it is possible to go back to point 1. If no disagreement remains, the cyclic process ends.

A. ELICITATION

To enter the elicitation stage the expert must characterize the context with a minimal set of elements (i.e., cases, examples, diagnostics, etc.) pertaining to the domain of expertise. Next, the elicitation process builds up groups of elements according to their similarities and dissimilarities with respect to the constructs already present.

The refining mode carries out two indistinguishability analyses: one over the domain elements and another over the already elicited constructs. Its main feature is that the interactive analysis is fed back to the expert in such a way that the incremental building of the repertory grid, and the validation/refinement of the repertory grid are the same process. Construct analysis shows the expert the most similar constructs, and he may point out his disagreements. If two constructs are similar, it means that they structure similarly the domain elements, and if they had to be more different than the domain context, they should be enlarged with new elements that are still missing. Therefore, the expert is asked to supply a concrete explanation of his disagreement, i.e., a counterexample embodied in a new element that will increase the construct distinction, as well as the representativeness of the domain context with regard to the real expertise domain.

Element analysis shows the most similar elements and, if disagreement arises, the expert is requested to supply a new concrete explanation, i.e., a new construct that distinguishes these too-similar elements. Undue indistinguishability between elements reveals a poor discrimination power of the set of elicited constructs.

By defining a new construct, the expert introduces a new distinction over the context elements in a process to build an opposite characterization of the set of domain elements for the task at hand. Both interaction modes form the incremental constructing process of the repertory grid. The set of fuzzy relations between domain elements and constructs constitutes the fuzzy repertory grid and the set of fuzzy predicates applying to a construct; for example, a type 2 fuzzy set constitutes the representation of the construct.[11]

B. INFERENCE ANALYSIS

In the second stage, ALIS elucidates the implicational relationship holding between constructs. As constructs are represented by type 2 fuzzy sets, we apply a type 2 semantic entailment (an extension of ordinary fuzzy set entailment) to model subjective inferences. The analysis outcome is an inference network, a digraph where nodes stand for construct poles and weighted arcs stand for implicational strength. The inference network is fed back to the expert for validation in the EVRI stage.

C. INFERENCE VALIDATION AND REFINING ENVIRONMENT

This stage implements a set of techniques and aids for validation and refining founded in PCT. The EVRI environment is the turnover of the developmental EAR cycle, for it focuses on disagreement resolution. Expert disagreements about the inference network may arise for different reasons (ambiguous or polysemic constructs, insufficient elements characterization, domain context incompleteness), and they are handled in different ways:

Counterexample proposal — Disagreement with a rule is justified by the expert stating a counterexample that is incorporated in the repertory grid as a new concrete explanation. As before, concrete explanation has a global repercussion and may modify other rules in addition to the intended one. Counterexamples may be new elements, in the case of lack of representativeness of the current context, or new constructs, in the case of insufficient element characterization.

Revision of assignment values — The expert may have used different criteria in estimating the linguistic evaluations, applying constructs to elements, in a nonconsistent way. This is solved by editing the repertory grid to revise the assignment linguistic values.

Revision of contrastive sets — Disagreements may also arise for inappropriate or poorly discriminating contrastive sets. Contrastive sets can then be drastically changed or augmented with new linguistic labels in order to achieve a finer discrimination over the domain. A revision of the linguistic values of the associated constructs is finally conducted.

Laddering techniques — Concept ambiguity is resolved splitting a construct into two or more constructs by asking *how* and *why* questions. These new constructs are added to the repertory grid. *Why* questions lead to superordinate, more abstract constructs whereas *how* questions lead to subordinate, more concrete constructs.

VII. CONCLUDING REMARKS

We have described some aspects of the MILORD system and, in particular, its management of uncertainty. The most relevant features of our approach are the representation of uncertainty by means of expert-defined linguistic statements and the use of the certainty values to guide the search tree by means of a lookahead prospection technique.

The main advantage of this approach is that once the linguistic values have been defined by the expert, the system computes and stores the matrices corresponding to the different conjunction and disjunction operations on all the pairs of terms in the term set. Later, when MILORD is run on a particular application, the propagation and combination of uncertainty is performed by simply accessing these precomputed matrices.

The gain in speed, with respect to the most common numerical approaches, is remarkable; for example, a rule with N conditions in its premise will need N-1 accesses to a matrix to obtain the linguistic certainty value of the premise, and one additional access to combine this value with that of the rule itself in order to obtain the linguistic certainty value of the conclusion.

The easiness for the expert and the user in expressing linguistically their confidence in the rules and facts is also a remarkable feature.

The EAR cycle facilitates the knowledge engineering process, decoupling knowledge acquisition from implementation, and sticks to systematic refinement, requiring concrete explanations for the resolution of disagreements. The decoupling is obtained creating a refinement cycle prior to knowledge base implementation and prototype testing. This decoupling also allows a structured way in which a group of experts may develop their individual perspectives and, furthermore, using several content-free conversational procedures, engage in a process of discussion and negotiation for reaching a meaningful consensus. The system is being used for designing a knowledge base medical diagnosis system. In this experience, we have cited the insight provided by the different perspectives of the elicited data, the easiness in eliciting the individual conceptual repertoires, and generating tentative inference networks.

In the near future, the resulting knowledge base will be submitted to other experts for a final validation, and research has to be done in order to be able to implement a consensus process on a second stage over the inference network.

REFERENCES

1. **Beyth-Marom, R.,** How probable is probable? A numerical taxonomy translation of verbal probability expressions, *J. Forecasting,* 1, 257, 1982.
2. **Bonissone, P. P.,** The Problem of Linguistic Approximation in System Analysis, Ph.D. dissertation, EECS Dept., University of California, Berkeley, in University Microfilms International Publications 80-14, Ann Arbor, MI, 1979, 618.
3. **Bonissone, P. P. and Decker, K. S.,** Selecting uncertainty and granularity: an experiment in trading-off precision and complexity, *KBS Working Paper,* General Electric Corp. Res. Develop. Center, Schenectady, NY, 1985.
4. **Boose, J. H.,** *Personal construct theory and the transfer of human expertise,* in Advances in Artificial Intelligence, O'Sheea, F., Ed., Elsevier/North-Holland, Amsterdam, 1984.
5. **Dubois, D. and Prade, H.,** Criteria aggregation and ranking of alternatives in the framework of fuzzy set theory, in *TIMS/Studies in the Management of Science,* Vol. 20, Elsevier, New York, 1984, 209.
6. **Freksa, C. and Lopez de Mantaras, R.,** A learning system for linguistic categorization of "soft" observations, *Actes Colloq. Assoc. Rec. Cognit.,* Université de Paris-Sud, Orsay, 1984, 331.
7. **Kelly, G. A.,** *The Psychology of Personal Constructs,* W. W. Norton, London, 1955.
8. **Lopez de Mantaras, R., Agusti, J., Cortes, U., and Plaza, E.,** *Fuzzy Knowledge Engineering Techniques in Scientific Document Classification,* Int. Symp. Methodol. Intell. Syst., Knoxville, TN, October 1986.
9. **Miller, G. A.,** The magical number seven plus or minus two: some limits on our capacity for processing information, in *The Psychology of Communication,* Penguin Books, New York, 1967.
10. **Plaza, E.,** *Sistema interactivy d'explicitaciö de constructes personals usant semäntica difusa,* Master's thesis, Facultat d'Informática de Barcelona, 1984.
11. **Plaza, E. and López de Mántaras, R.,** *Knowledge Acquisition and Refinement Using a Fuzzy Conceptual Base,* Proc. NAFIPS Workshop Fuzzy Expert Syst. Decision Support, Georgia State University, Atlanta, October 1985.
12. **Shaw, M. and Gaines, B. R.,** New directions in the analysis and interactive elicitation of personal construct systems, *Int. J. Man-Mach. Stud.,* 13, 86, 1980.
13. **Trillas, E.,** Sobre funciones de negacion en la teoria de conjuntos difusos, *Stochastica* (Journal of the Universitat Politecnica de Barcelona, Spain), 3(1), 47, 1979.