

R. 93
Sig 1203 Lop

1400008574
M-REPORT/371

**ALIS
+
MILORD**

**Ramon LÓPEZ DE MANTARAS
Ulises CORTÉS (*)
Enric PLAZA
Carlos SIERRA
Alfredo VILLAR**

RR85/12

Julio 1985

**Amb el suport de la CIRIT de la Generalitat
de Catalunya i l'ajut 1503/82 de la CAICYT**

(*) Amb l'ajut 45157 de CONACYT de México.

ABSTRACT

This work develops the principal lines on knowledge engineering: first a methodology for expertise acquisition and, second, the design and implementation of several inference engines. In the first part we report the methodology for knowledge acquisition based on subjective inferences logical analysis (ALIS) and the validation and refinement techniques of knowledge bases. In the second part we explain the development of a set of inference engines based on fuzzy logic reasoning (MILORD). This set contains an engine with forward chaining, one with backward chaining, an other with a control based an four selection criteria. Also we describe a prospection technique into a graf formed by the set of rules, and a communication modul which includes the explanations of the followed reasoning paths to the user.

RESUMEN

Aquest report desenvolupa les dues vessants principals en el camp de l'enginyeria de coneixements i una metodologia per a l'adquisició de l'expertesa i el disseny i implementació de diversos motors d'interferència. La primera part reporta la metodologia d'adquisició de coneixements basada en l'anàlisi lògica d'inferències subjectives (ALIS) i les tècniques de validació i refinament de bases de coneixements. En la segona part s'exposa el desenvolupament d'un paquet de tres motors d'inferència basats en lògiques de raonament difús (MILORD). Aquest paquet consta d'un motor d'encadenament endavant, un d'encadenament enrera, i un motor amb el control basat en quatre criteris de selecció. Així mateix es descriu una tècnica de prospecció sobre el graf de regles i el mòdul de comunicació amb l'usuari d'explicació del raonament.

TAULA

PART I: METODOLOGIA D'ADQUISICIO DE CONEIXEMENTS

- 1 ADQUISICIO DE CONEIXEMENTS
- 2 EXPLICITACIO DE CONCEPTES
- 3 ANALISI LOGICA D'INFERENCIES SUBJECTIVES
- 4 IMPLEMENTACIO I VALIDACIO
- 5 CONSTRUCCIO D'UN PROTOTIPUS
- 6 CONCLUSIONS

PART II: DESENVOLUPAMENT I IMPLANTACIO DE MILORD

- 1 BASES DE CONEIXEMENTS
- 2 MOTORS D'INFERENCIA
- 3 CONCLUSIONS

LINEES FUTURES DE TREBALL

PROEMI

Aquest report descriu la tasca realitzada durant l'últim any pel grup de treball en Enginyeria de Coneixements del Departament de Matemàtiques de la Facultat d'Informàtica de la Universitat Politècnica de Catalunya.

Ramon Lopez de Mantaras. (Coordinador)

Estiu de 1985

PART I: METODOLOGIA D'ADQUISICIO DE CONEIXEMENTS

1. ADQUISICIO DE CONEIXEMENTS.

Un dels problemes centrals en Intel·ligència Artificial és l'adquisició dels coneixements d'un domini per tal de transformar-los, dins una estructura de representació, en una base de coneixements que els dongui suport. En particular, en la construcció de sistemes basats en coneixements (sistemes experts), el disseny de la base de coneixements es realitza mitjançant la metodologia de l'enginyeria de coneixements, basada en el diàleg entre l'expert en el domini i l'enginyer de coneixements. Un sistema expert consta, succintament, d'un motor d'inferències, que encarna els coneixements i les heurístiques sobre la resolució de problemes, i una base de coneixements formada amb la següent sintaxi:

```
[reglai si (PREMISA1,...,PREMISAn) llavors CONCLUSIOreglai] (1)
```

Aquest procés poc formalitzat, ha de fer explícits els conceptes i mètodes de resolució de problemes que l'expert utilitza en el seu domini i expressar-los en un formalisme adient de representació de la base de coneixements. Hi ha diverses etapes en el procés d'adquisició de coneixements [HAYES83], que poden agrupar-se en dues fases:

FASE I

1. Identificació : de característiques i conceptes clau del problema. (On es formulen els requeriments del problema).
2. Conceptualització: explicitació dels conceptes (i llurs relacions) en el domini de coneixements.

FASE II

3. Formalització : elecció o disseny d'una estructura de representació de coneixements adient i del mètode d'inferència emprat.
4. Implementació : formulació concreta de les regles particulars que constitueixen la base de coneixements així com la formulació de les meta-regles i heurístiques per a la resolució de problemes.
5. Validació : de les regles i les heurístiques del prototipus implementat.

A partir d'aquesta validació es detecten les modificacions necessàries en el conjunt de les regles (refinament de l'Etapa 4), o bé en l'estructura de representació (redisseny de l'Etapa 3), fins i tot canvis al nivell de conceptualització i identificació del problema (reformulació de la Fase I).

Actualment existeixen eines per a la construcció de sistemes experts que cobreixen el que hem anomenat Fase II. L'elecció d'una eina de construcció comporta l'elecció de l'estructura de representació de coneixements que aquesta eina incorpora. Algunes eines són senzillament un llenguatge de definició de les regles de producció dins un motor d'inferències existent, mentre que d'altres inclouen ajuts a la validació i refinament de la base de coneixements. Tanmateix, aquests sistemes de construcció deixen fora del seu abast la Fase I, en especial una etapa molt costosa, en temps i esforç, del procés d'enginyeria de coneixements com és la conceptualització. En aquest treball presentem un sistema d'ajut a l'explicitació dels conceptes i llurs relacions, juntament amb les tècniques adjacents per a la

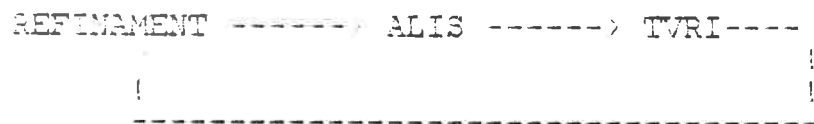
validació i refinament de la base de coneixements en aquesta primera fase.

El present metode d'adquisició de coneixements consta de tres parts:

1. Explicitació dels conceptes amb el programa EINA del sistema d'ajut a l'explicitació.
 - 1.1 Explicitació dels conceptes clau.
 - 1.2 Refinament interactiu de la base de coneixements.
2. Anàlisi Lògica d'Inferències Subjectives, programa ALIS.
3. Validació de les Regles d'Inferència (i retorn al refinament del pas 1.2)
4. Generació de les Regles.

Els passos 1 i 2 utilitzen el conjunt de programes del sistema d'ajut a l'explicitació conceptual (secció 2), mentre que el pas 3 el constitueixen les tècniques de validació i refinament de les inferències (TVRI) de la base de coneixements. Val a remarcar:

i) l'existència d'un mecanisme per al refinament de la base de coneixements en la Fase I (previ al disseny de l'estructura de coneixements) constituït per la iteració:



ii) l'ús combinat de l'enfocament clàssic de l'enginyeria de coneixements i el mètode ALIS. En el mètode clàssic es demana a l'expert de formular les regles i justificar-les, mentre que ALIS l'hi presenta les regles generades automàticament i es demana a l'expert d'acceptar-les o rebutjar-les, i justificar-ne la decisió usant les tècniques VRI

2. EXPLICITACIÓ DE CONCEPTES

El sistema d'ajut a l'explicitació de conceptes es basa en la metodologia de la teoria dels constructes personals del psicòleg G. A. KELLY [KELLY55] realitzat per un conjunt de programes d'explicitació i anàlisi de constructes personals [PLAZA84] i que permeten d'una manera interactiva que aquest procés (en aquest cas, el d'adquisició de coneixements) sigui dirigit pel mateix subjecte (en aquest cas, l'expert).

Per a la identificació del problema únicament cal designar els objectes del domini d'expertesa (els objectius de la tasca que ha de realitzar el sistema basat en coneixements), per exemple les enfermetats en un sistema de diagnòstic clínic, o, en el cas que il·lustrarem, en un sistema d'orientació per bibliografia especialitzada, els llibres d'una biblioteca. Opcionalment, també poden incloure-se d'entrada els conceptes claus per bé que aquests

i els altres s'explicitaran en el transcurs de la conversa home-maquina. El procés d'explicitació, típicament, consisteix en una sèrie de preguntes a l'expert com les següents:

QUINS LLIBRES A, B, C S'ASSEMBLEN MES? A i C.

EN QUE S'ASSEMBLEN A i C? Estudis Teòrics

EN QUE ES DIFERENCIA B? Estudi Aplicat

(denotem en majúscules les qüestions del programa subratllades les respostes de l'entrevistat i en negreta les variables).

Aquest parell de conceptes antònims constitueix els dos pols d'un constructe (i.e. "teòric/aplicat" o bé "tracta de robòtica/no tracta de robòtica") definint-se d'aquesta manera les "dimensions" subjectives que l'expert utilitza en el seu raonament sobre el domini. Ara, els objectes poden avenir-se a un concepte completament o no, i en aquest cas, cal determinar-ne el grau. El grau d'aplicació d'un objecte a un constructe s'explicita amb preguntes com:

EN QUINA MESURA A ES UN ESTUDI TEÒRIC? Molt

Aquest lligam d'un objecte amb un concepte mitjançant un descriptor lingüístic ve expressat per un predicat difus, el qual es representa semanticament en el sistema per una distribució de possibilitat sobre un univers de discurs

estandard que és l'interval $U=[0,1]$ [ZADEH78]. De manera que, com que el lligam entre un objecte i un concepte és un conjunt difús, la representació semàntica d'un concepte ve donada per un conjunt de predicats difusos que apliquen aquest a tots els objectes del domini i que serà denotat per un conjunt de conjunts difusos, és a dir, un conjunt difús d'ordre 2. Actualment el sistema implementat empra una col·lecció de cinc descriptors difusos (MOLT, FORÇA, MIG, POC, GENS), representats per cinc distribucions de possibilitat que formen un recobriment difús de U , per bé que es pot definir qualsevol col·lecció (mentre formi un recobriment de U i tota distribució de possibilitat de la col·lecció tingui el seu antònim dins aquesta mateixa col·lecció), segons sigui avinent [PLAZA84].

2.1. Refinament de la Base Conceptual

El principal avantatge de l'explicitació automàtica de conceptes davant de la conversa humana expert/enginyer de coneixements, es la possibilitat d'efectuar en el procés d'explicitació una anàlisi interactiva de la base conceptual existent en cada moment i, presentant-ne els resultats a l'expert, donar-li un mecanisme d'autoavaluació que el permeti de refinar aquesta base conceptual.

El sistema realitza una anàlisi interactiva de la similitud (indistingibilitat) entre els conceptes i una altra dels objectes del domini. La similitud s'estima a partir d'una distància euclídiana entre els conjunts difusos d'ordre 2 que són la representació dels conceptes i també dels objectes (en la taula d'assignacions concepte/objecte, cada cel.la és un conjunt difús, i si una fila denota un concepte, una columna denota un objecte [PLAZA84]).

2.1.1 Anàlisi dels Conceptes

El sistema presenta com a resultat de l'anàlisi de conceptes els que son mes semblants. Si l'expert pensa que no haurien de ser-ho, i, essent la causa que les seves característiques (assignacions als objectes) son molt semblants, l'expert ha de fornir una explicació concreta de per que no és el cas, és a dir, ha de donar exemples (nous objectes que hi mancaven) en que els conceptes es comporten diferentment. La declaració de contraexemples augmenta la distinció dels conceptes i per tant també la seva caracterització específica.

2.1.2 Anàlisi dels Objectes

El sistema presenta els objectes més similars. Si el expert no hi es d'acord, es que manquen característiques que defineixen (i distingeixen) els objectes, en aquest cas

l'explicació concreta que ha de fornir l'expert son nous conceptes que millorin la definició dels objectes i per tant els diferenciïn.

2.1.3 Sistematicitat i explicació concreta

Ja s'ha esmentat l'avantatge que representa tenir, en un estadi primer del procés d'adquisició de coneixements, un mecanisme de refinament de l'entrallat conceptual. Cal remarcar també la sistematicitat que procura el requeriment d'explicació concreta: las modificacions que les explicacions de l'expert exigeixen s'efectuen puntualment sobre la base conceptual (essent obligat a definir un cas concret com a justificació), però la repercussió d'aquesta novetat s'extén a tot l'entrallat conceptual, com explanarem a la secció 4.1.

3. ANALISI LOGICA D'INFERENCIES SUBJECTIVES

Per tal d'esbrinar les regles emprades per l'expert en el seu domini de coneixements, el programa d'anàlisi lògica d'inferències subjectives ALIS busca les relacions d'implicació entre els constructes explicitats en la base conceptual. En la segona secció desenvolupem l'ús que s'en fa del mapa d'inferències resultant, en la primera definirem la relació d'implicació.

3.1 CONJUNTS DIFUSOS D'ORDRE 2

Primerament, definirem un conjunt difús d'ordre 2 com aquell conjunt difús que per cada element x de l'univers de discurs la seva funció característica $\mu_A(x) \subseteq [0,1]$, dona, per cada element, un nou conjunt difús anomenat nivell difús o grau difús. En aquest cas, si $L=[0,1]$ aleshores les funcions característiques que defineixen conjunts difusos d'ordre 2 són de la forma: $\mu_A: U \rightarrow L^L$. Aquesta noció, proposada per Zadeh, fou desenvolupada per Kauffman [KAUF83] que denota un conjunt difús d'ordre 2 A de la manera següent:

$$A = \{ (x, \mu_A(x)) \}, \text{ on } \forall x \in U, \mu_A(x) = \{ (\mu / \mu(x)) \}$$

Per exemple, un constructe com teòric/aplicat en l'univers $U = \{11libre1, 11libre2, 11libre3\}$ pot representar-se pel conjunt difús d'ordre 2 següent:

$$A = \text{"teòric"} = \left\{ \begin{array}{l} 11libre1 | \text{MOLT}, \quad 11libre2 | \text{POC}, \\ 11libre3 | \text{FORÇA} \end{array} \right\}$$

mentre que el oposat tindria implícitament una configuració antònima:

$$A^{\sim} = \text{"aplicat"} = \left\{ \begin{array}{l} 11libre1 | \text{GENS}, \quad 11libre2 | \text{FORÇA}, \\ 11libre3 | \text{POC} \end{array} \right\}$$

on cada objecte té la distribució de possibilitat antònima a la del pol oposat i que s'obté definint la funció característica antònima com:

$$\mu_{A^{\sim}}(x) = \mu_A(1-x), \quad \forall x \in U = [0, 1]$$

(Cf. PLAZA84, YAGER83).

3.2 IMPLICACIÓ SEMÀNTICA D'ORDRE 2

Per tal de definir una relació d'implicació entre conjunts difusos d'ordre 2 introduïrem primer la max-composició o unió d'ordre 2 [KAUF83]. Siguin A i B conjunts difusos d'ordre 2, la max-composició es defineix com:

$$A \underset{2}{\cup} B = \underset{2}{A \cup B}(W|x) =$$

$$\left(\max_{\substack{u \in [0, w] \\ v = w}} (\mu_A(u|x) \wedge \mu_B(v|x)) \right) \vee \left(\max_{\substack{u = w \\ v \in [0, w]}} (\mu_A(u|x) \wedge \mu_B(v|x)) \right)$$

Per mitjà de la qual es pot definir una inclusió d'ordre 2:

$$A \underset{2}{\subset} B \text{ ssi } A \underset{2}{\cup} B = B$$

Com a extensió de la implicació semàntica definida entre conjunts difusos [ZADEH78] emprarem la implicació semàntica d'ordre 2 entre conjunt difusos d'ordre 2 de la manera següent:

$$A \text{ ---} \rightarrow B \text{ ssi } A \underset{2}{\subset} B$$

El programa ALIS usa també un operador d'implicació difusa considerant la força de la implicació segons es dongui en tots els objectes de l'univers de discurs, cas de la implicació semàntica, o es dongui en un nombre menor d'objectes. El subunivers de discurs on es dona la implicació determina la força d'aquesta, cosa que permet crear un mapa d'inferències amb implicacions estrictes i amb inferències aproximades que es formalitzaran en regles del sistema expert.

4. IMPLEMENTACIO I VALIDACIO

El mapa d'inferències esmentat es un graf orientat i etiquetat, on els nodes representen els conceptes explicitats, els arcs representen les implicacions entre conceptes i els pesos dels arcs la força d'aquestes implicacions. Aquesta presentació gràfica permet l'expert

de revisar els resultats de l'anàlisi lògica automàtica i acceptar o rebutjar les inferències. La modificació d'aquest mapa es realitza usant les tècniques de validació explicades a la secció 4.2 que usa el procediment de les explicacions concretes per reconfigurar la base conceptual. Primer tractarem dels tipus de regles i llur generació a partir del mapa d'inferències i la base conceptual.

4.1 Generació de les Regles

La base de coneixements d'un sistema expert es compon bàsicament d'un conjunt de regles sobre el domini d'expertesa donat. La sintaxi d'aquestes regles la denotarem:

```
<regla> ::= < NomRegla SI (Premissa1, ..., PremissaN) ALESHORES
           (FactorCertesa = ValorCertesa) ConclusióK >
```

Hi ha dos tipus de regles [BOOSE84] segons el criteri utilitzat en llur construcció: regles d'inferència, generades a partir de l'anàlisi lògica que determina el mapa d'inferències, i les regles de conclusió, generades a partir de la base conceptual.

Regles d'Inferència. L'anàlisi lògica produeix les implicacions que existeixen entre els conceptes explicitats i llur força, les quals constitueixen el mapa d'inferències i que s'implementen mitjançant regles. La implementació d'una regla d'inferència consta de: a) donar un

identificador a la regla, b) donar el concepte implicador com a premissa, c) assignar el grau d'implicació al factor de certesa, i d) indicar el concepte implicat com a conclusió. Remarquem que, si un concepte ve implicat independentment per dos conceptes diferents, això s'implementa en dues regles separades, car la forma normal de les regles és que les conjuncions de premisses van en una sola regla mentre que considera com a disjuncions les diferents regles amb la mateixa conclusió.

Regles de Conclusió. El principi que seguim per determinar l'estructura de regles que tenen com a conclusió els objectes del domini (els objectius de la tasca) és que ha de venir donada per les assignacions d'aquest objecte als conceptes. Dins del marc d'aquest principi, són possibles diversos criteris de construcció de regles, entre els quals, hem seleccionat el següent:

- a) el factor de certesa és de valor 1
- b) les premisses s'apleguen en dos grups conjuntivament, els quals són
 - b.1) la unió conjuntiva dels conceptes amb assignacions més altes ("MOLT").
 - b.2) la disjunció dels conceptes d'assignació amb força immediatament inferior ("FORÇA").

Per exemple, una regla de conclusió que caracteritza, en el cas d'una biblioteca, un llibre tindrà la forma:

(REGLA1 (si (C1 & C2 & C3)) aleshores (FC=1) LLIBRE1)

4.2 Validació de la Base de Coneixements

Les tècniques de validació i refinament d'inferències (VRI) constitueixen un primer mecanisme de reformulació de la base de coneixements, anterior a la validació a partir del prototipus del sistema implementat i independent del motor d'inferències usat, la qual cosa agilitza en gran mesura tot el procés d'adquisició de coneixements.

Les tècniques VRI funcionen en cicle dins l'algorisme del mètode d'ajut a l'explicitació i adquisició de coneixements següent:

- 1) Explicitació de conceptes rellevants
- 2) Refinament de la base de conceptes mitjançant explicacions concretes
- 3) Anàlisi lògica d'inferències
- 4) Validació i refinament de la base de coneixements
 - 4.1) Si es correcte, Pas 5.
 - 4.2) Si no, aplicar tècniques VRI i tornar al Pas 2.
- 5) Implementació del prototipus.
- 6) Validació del prototipus.

Remarquem que els resultats d'aplicar les tècniques requereixen, en el refinament de la base de coneixements, l'ús del procediment de les explicacions concretes presentat a la Secció 2.1.

4.2.1 Resolució de Conflictos

Comunament, l'expert serà en desacord amb alguns dels resultats de l'anàlisi lògica efectuada a partir de la base conceptual per ell construïda. Entre les raons que originen aquests conflictes podem destacar l'existència de conceptes ambigus o polisèmics, la caracterització insuficient dels objectes i la incompletesa de l'univers de discurs sobre el que s'apliquen els conceptes. El present mètode disposa de diverses tècniques per a la resolució de conflictes.

A) Proposició de Contraexemples. Quan l'expert és en desacord amb una inferència (o amb el seu grau de certesa), ha de justificar el seu raonament proposant-hi un o més contraexemples : un nou concepte en cas de caracterització insuficient dels objectes del domini o un nou objecte en el cas d'incompletesa o poca representativitat de la col·lecció d'objectes respecte del domini que es vol representar. Un contraexemple, en ser incorporat a la base conceptual com explicació concreta possiblement donarà lloc a canvis també en d'altres implicacions, mantenint-se d'aquesta manera la sistematicitat en la modificació del mapa d'inferències. La raó és que el conjunt d'inferències no es modifica localment sino globalment en derivar-se amb l'anàlisi lògica del conjunt d'assignacions que componen la base conceptual.

B) Revisió dels valors de les assignacions. Atesos els resultats, l'expert pot decidir un nou criteri per a les

assignacions de conceptes als objectes, o bé adonar-se que n'ha usat de diferents prèviament i cal unificar les assignacions sota un de sol. La revisió dels valors d'assignació s'efectua simplement editant la base conceptual. La base ara modificada generarà un nou conjunt de regles on s'hauran transformat no sols les regles que l'expert considerava errònies sino d'altres afectades pel nou criteri d'assignació, mantenint-ne així la consistència dins el conjunt de regles sota les transformacions efectuades en la matriu d'assignacions.

C) Tècniques de "laddering". Alguns desacords de l'expert poden ser deguts al cas en què un dels conceptes d'una regla donada sigui ambigu o polisèmic, és a dir, quan l'ambigüitat o els diversos significats són rellevants a la tasca que es vol resoldre (els sentits diversos que pugui tenir en altres contextos al domini considerat no hi intervenen). Quan es dona aquesta situació es poden usar les tècniques de "laddering" dissenyades originalment per a la psicologia dels constructes personals [KELLY55] en què es fonamenta el sistema d'explicitació de la Secció 2 i que bàsicament consisteixen en trencar un concepte en dos o més conceptes nous que s'afegiran a la base conceptual. Després es reelabora l'anàlisi lògica per tal de veure si el conflicte senyalat per l'expert s'ha resolt. Aquest trencament s'efectua amb les respostes de l'expert a les preguntes "PER QUÈ" i preguntes "COM" generades per la tècnica de

laddering.

C.1) Preguntes "PER QUE". Per exemple, preguntes com "Per què és important el concepte C?". La resposta genera un nou concepte o més, els quals solen estar en una relació de superordinació (més generals o més abstractes) respecte del concepte C. La pregunta pot aplicar-se tot seguit al nou concepte C1 per generar-ne un altre, C2, i així recursivament explicitar una sèrie de nous conceptes.

C.2) Preguntes "COM" i "QUE". Són preguntes del tipus "Què vol dir el concepte C?" o "Com explicaries C?". En aquest cas les respostes solen ser conceptes subordinats als concepte C (més específics o més concrets).

En cas que certs punts de conflicte quedessin sense resoldre amb les tècniques VRI, aquests s'apleguen en una llista de conflictes que serà el temari a discutir en una sessió de treball entre l'expert i l'enginyer de coneixements. Aquests conflictes poden deure's a una vertadera confusió o inconsistència per part de l'expert o per altres raons i s'han de resoldre per altres mètodes, més informals, conversacionals, de l'enginyeria de coneixements (Cf. HAYES83), per be que cal remarcar que el temari de discussió ha estat explicitat per la metodologia emprada.

4.2.2 Revisió del mapa d'inferències

Si en l'apartat anterior hem tractat dels casos de conflicte més puntuals, ara enfocarem els que poden sorgir del mapa d'inferències en general i que poden exigir una revisió global de la base conceptual. Podem senyalar dues tasques:

- i) Cerca d'inconsistències conceptuais, que pot portar, si es donen, a la reformulació de la base de conceptes.
- ii) Regles intuïtives inexistent. Aquestes regles es poden afegir directament o usant les tècniques de l'apartat anterior.
- iii) Reavaluació dels factors de certesa de les regles. Un cop implementat el sistema es poden estudiar les diferents performances que resulten de modificar-ne els factors de certesa generats per l'anàlisi lògica per uns altres (cosa que equival a modificar el nombre d'objectes assignats als conceptes fins a obtenir un univers de discurs constituït per un conjunt d'objectes representatiu).

5. CONSTRUCCIÓ D'UN PROTOTIPUS

En l'exemple esmentat, per a l'elecció d'un llibre entre un conjunt especialitzat d'obres d'Intel·ligència Artificial, s'han prosseguit els estadis d'implementació i validació d'un prototipus que consta de 41 regles d'inferència i 30 regles de conclusió (que es converteixen en 89 regles en el format normalitzat). En el procés de refinament de la base de coneixements emprant les tècniques VRI cal remarcar:

- a) s'ha usat preferentment la generació i inclusió de contraexemples i explicacions concretes.
- b) s'ha realitzat la modificació de les assignacions en alguns conceptes.
- c) únicament s'ha afegit una regla intuïtiva no explicitada automàticament.

El sistema expert implementat funciona amb els motors d'inferències desenvolupats a MILORD (Cf. LOPEZ85).

6. CONCLUSIONS

La utilitat del mètode d'ajut a l'explicitació i refinament en el procés d'adquisició de coneixements és remarcable atesa la dificultat en la formulació de les regles que formen part dels sistemes basats en coneixements i podem senyalar com a principals aportacions:

- facilitat i rapidesa en l'explicitació de conceptes rellevants
- facilitat i rapidesa en la generació automàtica de conjunts temptatius de regles d'inferència
- facilitat i sistematicitat en el refinament de la base de coneixements mitjançant les anàlisis interactives i les tècniques VRI
- rapidesa en la creació d'un prototipus per a validar-ne la performance.

També cal remarcar la utilitat de les tècniques VRI en la resolució dels conflictes entre l'expert i les regles generades així com la principal característica d'aquestes: la sistematicitat en la modificació de la base conceptual prèvia per al refinament de les regles mitjançant les explicacions concretes i la repercussió global que té cada modificació particular.

PART II: DESENVOLUPAMENT I IMPLANTACIO DE MILORD

1 BASES DE CONEIXEMENTS

1.1 Introducció

Els sistemes experts tracten de modelitzar el coneixement humà en aquells camps en que l'experiència de l'individu és fonamental en els resultats per ell obtinguts en la diagnosi, la classificació, la interpretació, etc. del món que li envolta, moltes vegades aquest coneixement no és expressable com a un receptari que s'adequa a situacions estereotipades sinó que dona respostes noves a situacions canviants. Tanmateix, el procés de diagnosi té una funció de realimentació en el coneixement que l'individu posseeix sobre aquest camp d'aplicació, de manera que les respostes donades amb anterioritat a situacions similars a la que s'està tractant a cada moment influeixen en la resposta que es donarà.

Es a dir, podriem modelitzar un expert com a un conjunt de regles d'inferència a partir d'uns fets observats en cada cas concret, que han estat construïdes per relacions de causalitat a partir de l'experiència obtinguda en l'anàlisi de situacions semblants al llarg del temps, i un sistema capaç d'inferir fets nous a partir de la situació en curs.

El conjunt de regles conforma el que s'anomena Base de Regles o Base de Coneixements, si bé aquest segon terme inclou també el conjunt de fets observats i deduïts en l'anàlisi de cada situació concreta.

Es mitjançant aquestes regles que hom intenta formalitzar el coneixement dels experts. Això té des del punt de vista informàtic dues grans avantatges: l'una és la modularitat: el coneixement no es barreja amb l'estructura de control; i l'altre, fortament depenent d'aquesta, és la facilitat en l'actualització, que es redueix a la modificació, inserció o esborrat de regles.

Resulta evident que la adequada representació d'aquestes regles és un punt crucial en l'eficiència del sistema en el seu conjunt. Tanmateix el contingut semàntic de la representació farà de les regles una eina útil o no.

La major part de sistemes fan servir com a llenguatge formal alguna variant de la lògica de predicats, d'altres es conformen amb l'ús de la lògica proposicional. En ambdós casos però, s'utilitza un llenguatge formal com a eina de treball, encara que s'ofereixi la possibilitat de diàleg en llenguatge natural amb certes restriccions.

Per altra banda molts dels prototipus actualment en funcionament ofereixen possibilitats d'actualització directe de la Base de Coneixements mitjançant el diàleg amb

l'usuari.[rosie](veure ampliacions)

1.2 Base de Coneixements a MILORD

Com ja s'ha comentat a la part introductoria, les bases de coneixements es componen de dues parts fonamentals: el conjunt de regles i els fets relatius al cas en tractament. Passem a comentar, doncs, com hem implantat aquestes dues parts al nostre sistema.

Fets: Els fets són per a nosaltres atòms de LISP; per exemple: `animal_es_tigre`, `animal_menja_carn`, `bolet_peu_fi`, `intel.ligencia_artificial`, etc. que tenen associat el seu valor de certesa com a propietat APVAL. Inicialment tots els fets són inicialitzats a NIL. Amb això aconseguim que l'accés al valor d'un fet sigui ràpid. Tanmateix saber si un fet és conegut per la Base de Coneixements és ben senzill, només cal preguntar si el fet es nul o no. També el motor manté una llista dels fets demostrats per facilitar posteriorment l'explicació del procés de raonament següent.

Amb aquesta representació aconseguim un guany en velocitat respecte a la major part de sistemes, que fan servir una llista de fets on van a parar totes les consultes sobre els fets de la Base de Coneixements, amb la inevitable cerca seqüencial dins la llista. Aquest guany ve donat pel fet que les propietats atòmiques són accessibles directament

a partir de l'àtom.

Regles: Tota regla té unes premisses que de complir-se (amb un cert grau) dedueixen unes conclusions (amb uns certs graus). Generalment les regles es representen mitjançant una llista del tipus:

(regla numero (si premisses) [vc] (llavors conclusions))

El procés de verificació de diagnòstics és seqüencial, agafant una per una les regles, verificant-ne les premisses i deduint-ne les conclusions. Aixó és un procés lent, i si bé quan el nombre de regles no és gaire elevat no representa un problema greu, quan el nombre n'és el d'una aplicació real (100 a 500 regles) el problema pot arribar a ser realment molt seriós.

Nosaltres hem volgut representar les regles de manera que la velocitat d'inferència no depengui del tamany de la B.C. Per tal de fer-ho mantenim dues representacions de les regles: una externa, llegible, amb una sintaxi propera a la de la logica i una altra interna, fent servir les propietats dels atóms, més operativa.

La representació externa es la següent:

(regla-index (si (p1...pn)) [vr] (llavors (c1...cm)))

La representació interna és:

regla ==> APVAL [vr] SI (pl...pn) LLAVORS (cl...cm)

essent APVAL, SI, LLAVORS propietats de l'àtom regla. L'accés a les premisses i les conclusions d'una regla serà, doncs, un accés a les propietats de la regla. Un tipus de consulta molt usual és aquell que ens diu quines regles dedueixen un fet (en l'encadenament enrera); per accelerar aquesta consulta a cada conclusió li assignem com a propietat REGLES les regles que la dedueixen:

Concl ==> REGLES ((reglal vr1)...(reglak vrk))

a més a més aquestes regles seràn ordenades en ordre decreixent de VRi, per tal de fer més ràpida l'anàlisi prospectiva en l'encadenament enrera (veure 2.4).

2 MOTORS D'INFERENCIA

2.1 Objectius

Es preten el disseny d'un paquet de motors d'inferència fent servir lògica difusa, amb una interacció dinàmica entre ells de forma que la utilització d'un motor o un altre romanguí invisible a l'usuari. Aquests motors han de ser el més ràpids possible, per la qual cosa s'utilitza una estructura de representació del coneixement, de la qual ja se n'ha parlat, que permet un accés ràpid a la informació de la Base de Coneixements. Tanmateix, fem servir regles heurístiques en dues vessants diferents: una en el tall de la cerca dins el graf que conformen les regles, de manera que es pogui veure de bell antuvi quan una conclusió no podrà assolir un valor de certesa que nosaltres considerem mínim per considerar acceptable la conclusió; i una altra en quant a la selecció de la regla a aplicar a cada moment segons uns criteris que comentarem en el moment adient (motor conjunt). Pel que toca a l'arquitectura ha d'esser el més modular possible (fer-ho en LISP ja suposa un gran avantatge en aquest sentit) per tal de fer servir diferents funcions de propagació a través dels connectors lògics "&", "v", "=>", sense que l'estructura principal del paquet pateixi unes modificacions massa importants.

Per fi, una utilitat gairebé indispensable en tot motor que tingui unes mínimes pretensions de ser útil, és que permeti l'explicació del procés de raonament que l'ha dut a la verificació d'una hipòtesi. En aquest sentit implantem un petit joc converçacional entre el paquet i l'usuari.

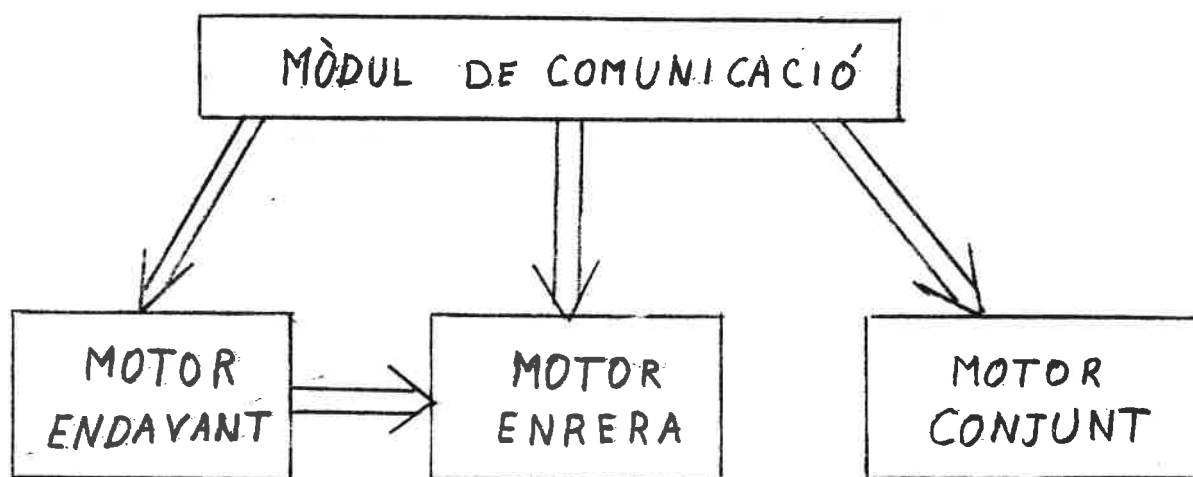
2.2 Arquitectura

El paquet és compost bàsicament de dos motors, un d'ells fent servir encadenament endavant i l'altre fent servir encadenament enrera, encara que com s'ha comentat als objectius la possibilitat de lligar ambdós motors d'una manera

dinàmica obre les portes a un altre motor que podríem anomenar "mixt". Aquests motors junt amb el mòdul d'explicació del raonament (comunicació amb l'usuari) i el mòdul de transformació d'estructura externa a interna formen l'esquelet bàsic d'aquest paquet. Existeix un tercer motor (motor conjunt) amb característiques peculiars que permet una quarta forma de treballar, a més de les tres ja vistes (endavant, enrera i mixt). El gràfic del següent full mostra l'estructura del sistema.

El motor endavant fa una validació d'hipòtesis dirigida per les dades contingudes a la base de coneixements. Inicialment l'usuari informa al sistema dels fets que ell observa en el cas objecte d'estudi. Aquests fets són rebuts pel mòdul de comunicació amb l'usuari, que realitza una sèrie d'accions sobre d'ells: primer els introdueix a la base de coneixements amb l'estructura de representació adient; seguidament comprova si dintre dels fets que l'usuari observa es troba algun dels diagnòstics previstos

pel sistema dins aquella area de coneixements. De trobar-ne algun, aquest modul fa una crida al motor amb encadenament enrera per tal de veure si aquest diagnostic es cert (veure 2.4), sino crida al motor amb encadenament endavant.



2.3 Motor endavant

Aquest motor parteix, doncs, d'uns fets sobre la situació a diagnosticar degudament representats a la base de coneixements. L'objectiu d'aquest motor és aconseguir la deducció d'una hipòtesi amb un valor de certesa per sobre d'un mínim predefinit, sense interaccionar amb l'usuari i fent servir només les dades proporcionades per aquest. En cas de no poder inferir cap hipòtesi que superi aquest valor mínim de certesa el motor fa una crida al motor enrera per tal de veure si ell, mitjançant la interacció amb l'usuari, pot inferir-ne alguna. Aquest és el primer cas de funcionament mixt dels nostres motors.

En cas que el motor endavant arribi a verificar una hipòtesi amb un valor de certesa no superior al mínim preestablert, però proper en un cert grau, fem servir la regla heurística de cridar al motor enrera per tal de pujar el valor d'aquesta hipòtesi resseguint el graf de regles que porten a aquesta conclusió, per poder aconseguir augmentar el seu valor, mitjançant l'aplicació de la propagació tipus "0" de les diferents regles que dedueixen la hipòtesi.

La selecció de la regla a aplicar en cada moment depèn de molts factors: nombre de premisses, nombre de conclusions, etc. En aquesta aplicació no hem fet cap estudi en quant a la selecció de les regles i ho hem deixat per la segona

aplicació en la que parlarem més extensament d'aquest tema. L'elecció que fem es seqüencial, de totes les regles que dedueixen un fet anem triant una a una fins a exhaurir-les.

A continuació fem una síntesi de les funcions més significatives del motor endavant, on es veuen reflectides les crides a les funcions diferenciadores respecte als algorismes més clàssics de motors endavant [winston84].

La següent funció comença aplicant el motor endavant. Si després d'intentar l'aplicació del metode endavant veiem que no és possible la verificació de cap hipòtesi cridem al motor enrera per veure si ell pot arribar a demostrar-ne alguna mitjancant la interacció amb l'usuari.

```
(defun endavant ()
  (prog ()
    bucle (cond (fi (return t))
                ((passaendavant))
                (t (return (enrera diag max fita 0.4))))
    )
  (go bucle)))
```

La següent funció tria una regla segons un criteri concret i intenta aplicarla. Si no pot ser-ho tria una altra fins que o bé les exhaureix o bé troba una que si pot ser aplicada.

```

(defun passaendavant ()
  (prog (regaplicab)
    (SETQ regaplicab (diferencia llistaregles reglesusades))
  bucle (SETQ regla (triar-regla regaplicab))

    (cond ((null regaplicab) (return nil))
          ((assaigregla regla) (SETQ reglesusades
                                   (cons regla reglesusades))
                                   (return t)))

    (SETQ regaplicab (diferencia regaplicab (list regla)))
    (go bucle)))

```

Una vegada s'ha triat una regla per ser aplicada la següent funció comprova si la aplicació pot concloure quelcom o no.

```

(defun assaigregla (regla)
  (and (provar-si regla) (concloure regla)))

```

Funció de reforçament del valor d'una hipòtesi mitjançant un encadenament enrera.

```

(defun reforçar (hipo)
  (SETQ reforçades (cons hipo reforçades))
  (enrera hipo maxim fita (calc-tall hipo))
  (cond ((greaterp (eval hipo) maxim)
         (SETQ fi t)
         (SETQ hipotesi hipo))))

```

2.4 Motor enrera

El motor enrera fa una validació de les hipòtesis recorrent el graf des d'aquestes fins als fets no deduïbles pel conjunt de regles de la base de coneixements. Practicament tots els motors que avui en dia es poden trobar en la literatura fan servir el mateix tipus d'algorismes: agafar les hipòtesis una per una, recórrer el graf enrera fins als fets no deduïbles, i interaccionar amb l'usuari per demanar-li el seu valor de certesa; propagant després aquest valor endavant en el graf. S'introdueixen generalment unes certes regles heurístiques com ara son: valors de tall de les premisses d'una regla, metaregles, ordenació de les regles, etc.

Nosaltres hem desenvolupat un algorisme que fa servir una regla heurística original que explicarem a continuació, i d'altres ja existents en altres sistemes, com és el cas dels valors de tall en la validació de les premisses de les regles.

Farem ara a exposar la innovació que hem realitzat en quant a les regles heurístiques que fan que el proces de raonament guanyi en velocitat. Basicament aquestes regles heurístiques consisteixen en fer una prospecció des de les hipòtesis cap als fets externs (no deduïbles), de manera que en el moment en que preveiem que el valor final que es propagara a la hipòtesi en curs no assolira el mínim que per la hipòtesi com a vàlida, aturem el proces de tractament d'aquesta hipòtesi per

passar a continuació a considerar-ne una altra.

L'estudi que aquí presentem es específic pel cas de fer servir com a regla de combinació i propagació dels connectors lògics "V" "&" els següents:

$$V(A \vee B) = V(A) + V(B) - V(A) V(B)$$

$V(A \& B) = \text{MIN}(V(A), V(B))$ encara que és fàcil fer-ho per altres modelitzacions d'aquests connectius.

En el que segueix utilitzarem la següent notació:

$V(F)$ = "valor de certesa del fet F"

$VR(i)$ = "valor de certesa en l'aplicació de la regla i-èsima"

$VP(i)$ = "valor mínim que han d'assolir les premisses de la regla i-èsima"

$VC(C,F)$ = "valor màxim propagable per les regles que dedueixen el fet F sense considerar el subconjunt de regles C"

$\text{MIN}(D)$ = "valor mínim d'acceptació d'una conclusió"

$\text{VTM}(C)$ = "valor teòric màxim assolible per la conclusió comu a les regles C"

Veiem ara quines condicions ha de cumplir el valor d'una conclusió per esser acceptat:

$$V(D) = VC(\langle i \rangle, D) + VR(i) * VP(i) - VC(\langle i \rangle, D) * VR(i) * VP(i)$$

$$V(D) > MIN(D)$$

d'aquí podem deduir que:

$$VP(i) > \frac{(MIN(D) - VC(\langle i \rangle, D))}{((1 - VC(\langle i \rangle, D)) * VR(i))}$$

Veiem ara com podem calcular la funció VC. Primer calculem el valor màxim assolible per una conclusió determinada; aixó ho farem suposant que el valor de totes les premisses que la dedueixen es 1.

Suposarem les regles que dedueixen D ordenades de 1 a N, pertanyents a un conjunt que anomenarem C.

$$VTM(C) = \text{SI } C = \text{buit LLAVORS } 0$$

$$\text{SINO } VR(i) + VTM(C-i) - VR(i) * VTM(C-i)$$

$$[i \text{ pertanyent a } C]$$

Per la qual cosa :

$$VC(\langle i \rangle, D) = VTM(C-i) \quad \text{si } VR(i) = 1.0$$

$$VC(\langle i \rangle, D) = \frac{VTM(C) - VR(i)}{1 - VR(i)} \quad \text{si } VR(i) < 1.0$$

en aquest cas unint les dues expressions tindrem:

$$VP(i) > \frac{MIN(D)*(1-VR(i))-VTM(C)+VR(i)}{(1-VTM(C))*VR(i)} \quad \text{si } VR(i) = 1.0$$

$$VP(i) > \frac{MIN(D)-VTM(C-i)}{(1-VTM(C-i))*VR(i)} \quad \text{si } VR(i) < 1.0$$

Un cop considerades aquestes formules indiquem l'algorisme general d'avaluació d'un diagnòstic seguint aquesta filosofia:

```

procediment  avaluar  (D,MIN(D));

regles := <obtenir regles que dedueixen D>;
si regles = 0 llavors retornar nil fsi;
continuar := cert; valord := 0; VC := VTM(regles);
mentre  continuar fer
    vmaxd := valord + VC - valord * VC;
    si vmaxd < min llavors
        continuar := fals
    sino
        regla := <triar regla>;
        regles := regles - regla;
        VC := <calcular nou VC>;
        VP(regla) := <calcular VP(regla)>;
        si VP(regla) > 1.0 llavors
            retornar nil
        fsi;
        vaux := avaluar-premises(regla,VP(regla));
        valord := valord + vaux*VR(regla) -
            valord*vaux*vr(regla)
    fsi;
si regles = 0 llavors
    continuar := fals
fsi
fmentre;
si valord = MIN(D) llavors retornar 0
sino  retornar nil
fsi
fprocediment

```

2.5 Funcionament mixt

Diem funcionament mixt i no motor, perquè no es tracta d'un motor amb entitat pròpia, sinó que es tracta d'un sistema de crides entre els motors endavant i enrera, implantant d'aquesta manera una nova regla heurística que ens permet d'augmentar la velocitat de treball del nostre paquet. Aquest funcionament el fem servir en dos cassos que passem a exposar:

A) Quan l'usuari dona informació inicial sobre la situació que observa, es crida al motor amb encadenament endavant, però si aquest motor no aconsegueix inferir cap hipòtesi amb un valor de certesa superior al mínim establert, crida al motor amb encadenament enrera per tal de veure si aquest es capaç de verificar alguna hipòtesi mitjançant la interacció amb l'usuari.

B) Quan el motor amb encadenament endavant arriba a la validació d'una hipòtesi amb un valor de certesa no superior al mínim acceptable però proper en un cert grau a aquest, es fa una crida al motor enrera per que tracti d'apujar aquest valor per sobre del mínim reforçant la hipòtesi per altres camins del graf. Tenint com a regla heurística el fet de que si hem arribat a una conclusió amb un grau de certesa proper al mínim val la pena intentar reforçar la hipòtesi mirant si hi ha algun altre camí en el graf que conclou la mateixa amb un grau de certesa suficientment alt com per assolir el valor de certesa mínim d'acceptació combinant els dos valors de certesa dels dos camins convergents.

2.6 Motor conjunt

2.6.1 Objectiu

Despres de les anteriors descripcions de motors, es suposa que l'algorisme basic d'un motor endavant o d'un enrera estan clars. Partint d'aquest coneixement és interessant plantejar diverses variacions respecte dels esquemes anteriors que vagin aprofundint i aclarint i a la vegada obrint nous punts de vista, sobre els modus de treballar amb una Base de Coneixements, expressada mitjançant regles.

No ens allunyarem de l'essència del tipus de regles utilitzades fins ara, quedant properes ampliacions (algunes molt evidents com és la introducció de la lògica de predicats i d'altres menys evidents com és l'extensió de la lògica difusa del motor o el tractament del llenguatge natural) esbossades a l'apartat de properes linees d'investigació i estudiades en un futur pròxim.

Tractarem aquests nous aspectes mitjançant la descripció i desenvolupament d'un nou motor, el motor conjunt, que només s'ha insinuat fins ara i que apareix en aquest punt de la trama.

2.6.2 Introducció de criteris de selecció de regles a aplicar

El primer aspecte d'aquest motor que convé remarcar és que serveix per a corregir una perspectiva errònia que ens poden haver provocat els anteriors algorismes de motors d'inferència.

Degut a la tecnologia actual el nostre raonament aplica uns esquemes a la resolució de problemes, dins els quals el tractament seqüencial es una opció clara, sovint única.

Els llenguatges funcionals com LISP ofereixen a qui treballa en Intel·ligència Artificial una amplitud de mires que permet de crear algorismes amb un estil més lliure que el que imposen altres llenguatges de tipus procedural incapaces, malgrat la seva evolució, de trencar els seus rígids esquemes.

Fins ara la nostra forma de treball amb la Base de Regles es pot resumir segons el següent esquema: amb el motor endavant obtenir la primera regla de la B.C. i intentar aplicarla, i amb el motor enrera la primera d'aquelles regles de les quals es pugui deduir l'objectiu que volem demostrar. Fora millor, de ser possible, intentar aplicar totes les regles de la B.C. i seguir els camins que semblin ser els més encoratjadors. En aquest motor intentem veure la B.C. com un conjunt del qual podem seleccionar i intentar aplicar aquelles regles que segons certs criteris considerem més adients per a obtenir el resultat amb valor de certesa més gran i de la manera més ràpida possible.

Per seleccionar les regles a aplicar els criteris han de poder esser modificables, doncs segons la base de regles concreta amb la qual treballem en un moment donat, uns criteris seran més efectius que altres d'acord amb l'objectiu anterior. Hem introduït la possibilitat de seleccionar alguns criteris evidents per a triar regles de la B.C. tot estructurant el motor de

manera que la introducció de nous criteris sigui senzilla. La selecció dels criteris a seguir d'entre els previstos es decidida per l'usuari al començament de l'actuació del sistema. No es difícil fer que aquesta decisió pugui ser presa pel propi sistema d'acord amb el tipus de regles amb les que treballi mitjançant un algorisme que examini les característiques de les regles que tracta en un moment donat. Per exemple: en una B.C. en que totes les regles tenen només una conclusió el sistema ja pot deduir que utilitzar un criteri de selecció de regles que tingui en compte el nombre de conclusions és inoperant.

Els criteris amb els que pot treballar actualment el motor són (llevat de la utilització de meta-regles):

A) L'ordenació de la Base de Regles. Es el criteri seguit fins ara i es necessari que el poguem continuar mantenint ja que algunes Bases de Regles són dissenyades tot considerant que s'intentara aplicar per ordre les regles existents al utilitzarlas. Aquest es el criteri que segueix FROLOG a l'hora de seleccionar el predicat a aplicar.

B) Les de més força d'implicació. També ho hem fet servir abans amb el sistema de prospecció. Es un criteri lògic per a intentar obtenir conclusions amb un valor de certesa més fort.

C) Les de menys premises sense instanciar.
Per a obtenir conclusions més ràpidament.

D) Les de més conclusions. Fins ara hem fet servir regles que només tenien una conclusió, però és interessant tenir aquest criteri ja que el sistema permet treballar amb regles que contenen més d'una conclusió.

El sistema ha estat pensat per a treballar amb més d'un criteri dels esmentats a dalt, simultaniament. En aquest cas cal que establím una jerarquia dins dels criteris a aplicar. Més endavant es dona un exemple concret.

2.6.3 Preelaboració de regles

L'algorisme de gestió del sistema de selecció de regles anteriorment descrit pot ser molt lent doncs requereix una cerca al llarg de tota la Base de Regles comprovant quina és la que posseeix les millors característiques d'acord amb els criteris seleccionats, en un moment donat. Per a agilitzar en certa mesura aquest procés utilitzarem una altra tècnica que convé resaltar. Al descriure la B.C. que utilitza el motor exposavem la transformació que, a partir de la notació externa de les regles, obtenia una representació interna orientada a una utilització més eficient per part del motor. El nostre objectiu era alleugerar la feina del motor carregant amb més feina el moment d'acceptació de regles. El procés de seleccionar la regla

optima d'acord amb un cert criteri, per exemple la regla de més força d'implicació, ens porta a una recerca entre totes les regles que és possible aplicar, en un moment donat, accedint a la seva força d'implicació. Si en el moment d'acceptar la B.C. obtenim una llista de les forces d'implicació de totes les regles i podem calcular el màxim d'aquesta llista en cada moment, quan volguem seleccionar la regla òptima, des del punt de vista de la força d'implicació, només tindrem que mirar a la llista de regles que és possible aplicar la primera que coincideixi amb aquest màxim. Seguim doncs els passos següents:

+ OBTENIR CRITERIS DE SELECCIO DE REGLES A APLICAR

- Solicitar-los a l'usuari.
- Els decideix el propi sistema segons les característiques de les regles sobre les que treballa.

+ D'ACORD AMB EL CRITERI, IDENTIFICAR QUIN ES EL VALOR OPTIM QUE ES POSSIBLE TROBAR EN UN MOMENT DONAT

+ SELECCIONAR COM A REGLA OPTIMA LA PRIMERA QUE TINGUI AQUEST VALOR OPTIM

La combinació de diversos criteris de selecció de regles consisteix en aplicar primerament el criteri de més importància al total de regles i a continuació aplicar sobre el conjunt de regles seleccionades d'acord amb aquest primer criteri, els

altres criteris. S'ha de tenir en compte que és molt probable que existeixi més d'una regla que es trobi en el màxim possible d'un criteri; per exemple, amb respecte al criteri de major nombre de conclusions, les regles presenten una gran uniformitat ja que la majoria d'elles només posseeixen una conclusió o dues. Amb un exemple es veurà tot això més clar.

Suposem que tenim la següent B.C. amb les regles en l'ordre en que apareixen

R1 PREMISA-1 PREMISA-2 => 1.0 CONCLUSIO-1

R2 PREMISA-2 PREMISA-3 => 0.9 CONCLUSIO-2

R3 PREMISA-4 => 0.8 CONCLUSIO-1

Els òptims possibles respecte als criteris poden tenir-se a priori,

MAXIM DE FORÇA D'IMPLICACIO	1
MAXIM EN MAJOR NOMBRE DE CONCLUSIONS	1
MENOR NOMBRE DE PREMISES NO INSTANCIADES	1

I suposem finalment que els criteris d'acord amb els quals volem seleccionar la regla òptima son:

MES FORÇA D'IMPLICACIO

MES NOMBRE DE CONCLUSIONS

MENOR NOMBRE DE PREMISES NO INSTANCIADES

Amb aquest ordre d'importància, el nostre algorisme intentaria trobar una regla amb una força d'implicació 1.0, amb una sola premisa sense instanciar i amb una conclusió. Si d'entre les regles existents en la nostra B.C. no n'hi hagués cap que complís aquestes tres condicions, tractariem aquest cas el.liminant el criteri de menys importància el.liminant el criteri de menys importància. En el nostre cas el.liminem el darrer criteri: menor nombre de premisses no instanciades, i obtenim, d'acord amb els dos primers criteris la primera de les regles de la B.C.,R1.

2.6.4 Diferències motors endavant y enrera

Existeix un cinquè criteri, a més dels seleccionables directament que hem vist fins ara que actua a l'hora de seleccionar una regla a aplicar. Es lògicament el criteri de triar regles que tinguin com a conclusió l'objectiu que es preten demostrar. Aquest criteri s'aplica quan utilitzem el sistema enrera, i és una de les diferències de funcionament entre els motors endavant i enrera. Això fa que es plantegin ràpidament quines son les diferències entre ambdos tipus de motors, que fins ara hem presentat com dos algorismes independents llevat del que pertoca a la utilització d'una B.C. amb una estructura comu. Les diferències substancials son:

- El motor amb raonament endavant no intenta demostrar un cert objectiu, sino que aplica regles comprobant després de tota deducció si s'ha aconseguit satisfer un

dels objectius que li han estat presentats com a tals a la llista de diagnostics o resultats finals. El motor amb raonament enrera intenta verificar un cert objectiu, en general hi ha un nivell superior que utilitza aquest motor per anar comprovant si algun dels diagnostics de la llista de diagnostics es demostrable. Existeix doncs, en primer lloc, una diferència externa al motor, el motor endavant es posa en marxa fins que dedueix un resultat mentre que el motor enrera es posa en marxa per cada un dels diagnostics possibles. En segon terme la condició d'acabament no és exactament la mateixa, doncs mentre un motor endavant acaba la seva búsqueda tot fent un recorregut complet de la B.C. intentant trobar una regla que sigui aplicable, sense aconseguir-ho; un motor enrera acaba quan ha intentat comprovar tots els diagnostics possibles.

- El motor amb raonament endavant selecciona la regla a aplicar d'entre el total de regles existents a la B.C. que no hagin estat eliminades com a conseqüència de haber estat utilitzades abans o d'haber-se comprovat que no són utilitzables per que falla alguna de les premisses. El motor amb raonament enrera selecciona la regla a aplicar d'entre totes aquelles que estan a la B.C. siguin utilitzables, amb la condició suplementaria de que tinguin l'objectiu que es preten demostrar com a conclusió.

- A la fi, respecte al moment d'intentar aplicar una regla, el motor amb raonament endavant comprova si les premisses estan instanciades. De no ser així considera que la regla no és aplicable en aquest moment. El motor amb raonament enrera, intenta verificar les premisses i si alguna d'elles no està instanciada es crida a si mateix, amb aquesta premissa no instanciada com a objectiu, per a intentar verificar-la.

En la secció anterior proposavem dos motors que, mostrant-se davant l'usuari com un sistema únic, aprofitaven els avantatges de l'aplicació d'un mètode de resolució endavant o enrera segons les característiques de la Base de Regles o de l'aplicació concreta en que fos utilitzada dita Base de Regles. Donades les diferències anteriors entre ambdós motors es comprèn la facilitat de fer que no només sigui un sistema compost de dos motors, sinó que es pot fer que sigui o bé un sol motor el qui gestioni ambdós sistemes de resolució, o bé també de forma mixta, segons es desitgi. En aquest algorisme es barregen conceptes d'ambdós motors amb la qual cosa perdem una mica de legibilitat, essent, però, un bloc únic. Esbossadament l'algorisme fora:

PER totes les regles FER

seleccionar la regla aplicable segons criteris
(Si estem en modus enrera hi haurà com a
criteri prioritari el que la regla seleccionada
tingui com a conclusió l'objectiu cercat).

Intentar aplicar la regla seleccionada:

SI els objectius no estan instanciats

LLAVORS

SI estem en modus enrera

LLAVORS

cridar-se recursivament a si
mateix amb les premisses no
instanciades.

SINO la regla no és aplicable en
aquest moment.

SINO aplicar la regla i obtenir la seva
conclusió.

2.6.5 Identificació d'objectius externs i interns

Al parlar del modus de transformació de les regles del format extern a l'intern es va dir que es disposava, per cada objectiu les regles que el deduïan. Aquesta estructura es de molta utilitat a l'hora de seleccionar les regles a aplicar en el funcionament enrera; existeix, però, la possibilitat d'utilitzar aquesta particularitat de la B.C. per tal de diferenciar els objectius en dos grups: externs i interns. Els interns són aquells que poden ser deduïts per la utilització d'alguna regla de la B.C., els externs són aquells que no es poden deduir per la B.C. En aquest darrer cas la única forma de validació' és mitjançant una consulta a l'usuari. Quan volem demostrar un cert fet, ja sigui un diagnòstic, ja sigui un objectiu intermig, amb la comprovació de que aquest fet és extern aconseguim dos objectius principals:

- 1) Estalviar temps, ja que sabem que no és necessari que intentem demostrar fets que no són deduïbles.
- 2) Aumentar l'autonomia i la intel·ligència del motor, ja que anul·lem la possibilitat de preguntar a l'usuari cap fet fins no estar segurs de que amb la informació actual no el podem deduir. Amb això evitem moltes preguntes que faria el sistema en seguir camins erronis.

2.6.6 Interval d'instanciació

Per fer aquest motor conjunt d'acord amb un criteri que permeti a l'usuari treballar de la forma que li resulti més natural, i considerant que en determinades aplicacions el rang en el qual valorem els fets influeix en la naturalitat amb que l'usuari pot donar la seva ponderació, és interessant que el motor d'inferències permeti treballar sobre més d'un interval. Per tot això, i degut a la independència de l'interval utilitzat (sempre que sigui simètric respecte a l'origen, per a basarse en els criteris de la lògica multivaluada), fora bó que l'usuari pogues definir aquest interval. El nostre sistema, com hem vist fins ara, permet treballar dins l'interval $[0,1]$, però existeix també la possibilitat de treballar dins l'interval $[-1,1]$ quan així es volgui.

2.7 Comunicació amb l'usuari

El sistema disposa d'una sèrie de Bases de Coneixement, com ja se n'ha parlat, amb les quals l'usuari pot fer proves. La primera cosa que el sistema demana és amb quina d'aquestes Bases l'usuari vol treballar. Seguidament es demana si es vol donar informació sobre la situació. De ser així començarà a treballar el motor endavant, sinó serà el motor enrera qui comenci a funcionar.

Després del funcionament normal i de les preguntes que el sistema faci a l'usuari, s'arriba eventualment a un diagnòstic. En qualsevol cas el sistema dona resposta sobre el procés de raonament seguit; podent-se fer les següents preguntes tipus (el mòdul de L.N. no s'ha implantat encara):

(premises? r)

==> Ens diu les premises de la regla r.

(conclusio? r)

==> Ens diu les conclusion de la regla r.

(regles-que-dedueixen? C)

==> Ens diu les regles que dedueixen el fet C.

(fets_demostrats?)

==> Ens diu quins fets a aconseguir demostrar.

(regla_utilitzada? r)

==> Ens diu si ha fet servir la regla r.

(regles_utilitzades?)

=> Ens diu quines regles ha fet servir.

(per-que? f)

=> Ens explica perque ha deduit el fet f.

(regla? r)

=> Ens escriu la regla r.

(es_deducible? f)

=> Ens diu per quines regles pot ser deduit el fet f.

(pot_servir? f)

=> Ens diu si el fet f es útil a la nostra B.C.

(regles?)

=> Ens diu les regles que conté la B.C.

D'aquestes preguntes s'en poden fer tantes com l'usuari vulgui. Un cop acabada aquesta interacció el sistema permet de continuar treballant amb la mateixa B.C. o amb una altra d'entre les disponibles.

3 CONCLUSIONS

S'han plantetjat uns algorismes generals de resolució en Bases de Regles que constitueixen el nucli sobre el qual s'afegeixen les diverses utilitats del sistema MILORD. Aquests enfocaments presenten millores - si més no visions alternatives- als algorismes bàsics de sistemes experts accessibles de diverses fonts (EMYCIN, EPROSPECTOR, WINSTON, etc.) i serviran com a base per anar desenvolupant els punts que s'especifiquen a l'apartat de Futures Línies d'Investigació del present treball.

Bé que el sistema MILORD no posseeix, encara, alguna de les estructures de dades més complexes que presenten altres sistemes (càlcul de predicats, Frames, etc.) si posseeix davant d'ells alguns avantatges a nivell d'algorisme. Podem citar les següents:

- Una flexibilitat més gran en la utilització del motor en una aplicació concreta; el sistema ofereix un motor endavant, un enrera, un mixt i un conjuntat endavant i enrera. Podent-se resoldre el problema de la manera més adient.

- La possibilitat de triar d'entre els criteris previstos aquells que utilitzarem per seleccionar les regles a aplicar en cada moment, completant la possible utilització de meta-regles.

- Un sistema de prospecció, no existent en la bibliografia consultada, que permet tallar l'arbre de regles utilitzables per a deduir un fet concret.

- L'estructuració de la Base de Coneixements i la utilització dels fets que tracta el sistema a nivell d'àtoms amb propietats i no com a llista de valors associats amb el consegüent guany en temps de tractament.

- Una gran modularitat que permet fer més adient el sistema a aplicacions concretes permeten la còmoda modificació de les funcions de propagació i combinació, l'interval d'instanciació dels fets, o el sistema de prospecció.

I, en general, un acurat procés de disseny que permet introduir amb facilitat les modificacions que sobre de MILORD s'aniran afegint en un futur immediat, amb l'intenció d'implantar un entorn propi de desenvolupament de Sistemes Experts.

4 LINEES FUTURES DE TREBALL

El camp de la adquisició i validació de coneixements esdevindrà encara més un dels focus cabdals de recerca en l'enginyeria de coneixements en els propers anys i del seu reeiximent dependrà en gran part la difusió que assoliran els sistemes experts en el futur. El futur desenvolupament del mòdul ALIS comportarà dues vessants. D'una banda els resultats empírics de la seva aplicació en la construcció d'un sistema expert en medicina determinarà en gran part la seva evolució. D'altra banda cal el desenvolupament teòric d'un "càlcul de restriccions difuses" que permeti d'efectuar una anàlisi lògica de les inferències a un nivell més fi que l'actual per tal de generar les regles amb funcions difuses que implementarà MILORD.

En quant a MILORD es continuarà treballant en el camp de la lògica difusa tractant la imprecisió dels fets i de les regles amb funcions difuses; es faran servir etiquetes lingüístiques per qualificar la incertesa i la imprecisió. També la Representació del Coneixement es combinarà amb altres sistemes com Frames o Xarxes Semàntiques per explicitar les relacions entre els fets. Tanmateix s'introduïren millores tècniques en els motors per tal de fer un entorn de programació de desenvolupament de sistemes experts, incloent aquest entorn un Editor de Regles i un sistema de comunicació en Llenguatge Natural molt més acurat que el que hem utilitzat fins ara. Utilitzarem lògica de predicats fent d'aquesta manera més potent

i flexible el llenguatge de representació.

Finalment direm que MILORD s'utilitzarà com a entorn en la realització d'una Tesi doctoral en Medicina i una Tesina en Psicologia.

REFERENCIES

- BOOSE J H Personal construct theory and the transfer of human expertise. In F O'Shea Ed., Advances in Artificial Intelligence. Elsevier North-Holland (1984).
- HAYES-ROTH et al.
Building Expert Systems. Addison-Wesley (1983).
- KAUMANNF A Complements sur les Concepts Flous. Recherches et applications. Tome II. Mimeo no publicat. (1983)
- KELLY G A A Theory of Personality. The Psychology of Personal Constructs. W W Norton & Co (1955).
- LOPEZ DE MANTARAS R, ULISES CORTES & CARLOS SIERRA
MILORD : Motors d'infereia basats en logiques del raonament difus. Actes del IV Congres Català de Logica (1985).
- PLAZA E Sistema interactiu d'explicitacio de constructes personals usant semantica difusa. Tesi de Licenciatura. Facultat d'Informatica de Barcelona (1984).

YAGER R Y Opposites and measures of extremism in concepts and
 constructs. Internationl. J. Man-Machine Studies,
 18, 249-291.(1983)

EADEN L A PRUF: A meaning representation language for natural
 language. International J. Man-Machine Studies, 10,
 395-460.(1978)