

UNIVERSITAT POLITÈCNICA DE CATALUNYA  
DEPARTAMENT DE LLENGUATGES I SISTEMES INFORMÀTICS  
PROGRAMA DE DOCTORAT EN INTEL·LIGÈNCIA ARTIFICIAL

TESI DOCTORAL

**MILORD:  
ARQUITECTURA MULTI-NIVELL PER A  
SISTEMES EXPERTS EN CLASSIFICACIÓ**

Carlos Alberto Sierra García

Maig 1989

1950

1950

1950

1950

1950

1950

UNIVERSITAT POLITECNICA DE CATALUNYA  
PROGRAMA DE DOCTORAT EN INTEL·LIGENCIA ARTIFICIAL

TESI DOCTORAL

**MILORD:  
ARQUITECTURA MULTI-NIVELL PER A  
SISTEMES EXPERTS EN CLASSIFICACIO**

Maig 1989

Memòria presentada per en Carlos Alberto Sierra García per optar al títol de Doctor en Informàtica. El treball que ha conduït a aquesta memòria ha estat realitzat al Centre d'Estudis Avançats de Blanes del CSIC, i ha estat parcialment finançat pel projecte CSIC-CICYT n 836.

Director: Ramon López de Mántaras i Badia



*A mis padres.*



## **Abstract**

In this thesis a design of a shell for the development of Expert Systems in classification, in the framework of rule based systems, is developed. It enhances two important aspects in the process of problem solving: a flexible representation of approximate reasoning and an explicit management of different levels of control. The treatment of uncertainty has been developed around the theory of fuzzy sets. It has a linguistic representation of certainty values, and control is based on a multi-level architecture whose structure comes from a thorough study of the different knowledge and reasoning types to be modeled. Domain and control knowledge are specifically differentiated at each level of the architecture. A modularization technique, which can be reused in different languages is also proposed. This technique allows the incorporation into the language of elements such as inheritance, sharing and typing. Finally a description of four applications which have been successfully developed up till now is provided: two of them apply to medical framework, the remaining two to industrial framework; PNEUMON-IA RENOIR, VLSI AND SOLAR.

## **Keywords**

Expert Systems Development Environments, Problem Solving, Multi-level Architecture, Fuzzy Logic, Meta-knowledge, Knowledge Representation, Classification, Expert Systems in Medicine.





## **Resum**

En aquesta tesi es presenta el disseny d'un entorn de desenvolupament de sistemes experts en tasques de classificació, en el marc dels sistemes basats en regles, que remarca dos aspectes importants del procés de resolució de problemes: una representació flexible del raonament aproximat, i un maneig explícit dels diferents nivells de control. El tractament de la incertesa s'ha desenvolupat al voltant de la teoria dels conjunts difusos amb una representació lingüística dels valors de certesa, i el control en forma d'una arquitectura multi-nivell estructurada a partir d'un estudi en profunditat dels diversos tipus de coneixements i raonaments que es volen modelitzar. Els coneixements de domini i els coneixements de control estan ben diferenciats dins de cada nivell de l'arquitectura. També es proposa una tècnica de modularització del llenguatge que és reutilitzable en altres entorns de desenvolupament. Aquesta tècnica permet incorporar al llenguatge elements com: herència, compartició i tipatge. Finalment es fa una descripció de les quatre aplicacions que s'han desenvolupat amb èxit fins al present: dues en l'entorn mèdic i dues en l'entorn industrial: PNEUMON-IA, RENOIR, VLSI i SOLAR.

## **Paraules clau**

Entorns de disseny de sistemes experts, Resolució de problemes, Arquitectura multi-nivell, Lògica difusa, Metaconeixements, Representació de coneixements, Classificació, Sistemes experts en medicina.



## Prefaci

Aquesta tesi s'emmarca en l'àrea de la resolució de problemes, el raonament aproximat, la representació de coneixements i el meta-control multi-nivell.

El meu treball té les seves arrels en els treballs teòrics sobre la representació de la incertesa desenvolupat al departament de matemàtiques de la Facultat d'Informàtica de Barcelona durant els anys 1982 i 1985, conjuntament amb el departament de matemàtiques de l'Escola Tècnica superior d'Arquitectura de Barcelona. La utilització de la lògica difusa com a base d'un entorn de desenvolupament de Sistemes Experts va ser el primer prototipus del que anys més tard s'ha convertit en la base d'aquesta tesi. En aquells anys va rebre el nom de **MILORD** que ha conservat fins avui dia, encara que les sigles no recullin tot el significat de l'arquitectura actual: **Motors d'Inferència amb LOGica del Raonament Difús**. En el capítol tres es pot veure el fruit d'aquells anys i d'èpoques més recents durant les quals s'ha redefinit la semàntica de la incertesa.

Posteriorment, i ja en el Centre d'Estudis Avançats de Blanes (CEAB), les línies fonamentals de recerca es varen complementar amb la resolució de problemes i la representació de coneixements. Això va ser així degut a la col·laboració amb l'Institut Municipal d'Investigació Mèdica de Barcelona (IMIM) que va concretar-se en el desenvolupament d'una aplicació en el camp del diagnòstic mèdic. Aquesta aplicació va rebre el nom de PNEUMON-IA i és tema de la tesis doctoral en medicina d'Albert Verdaguer [Verdaguer, 1989].

Aquesta col·laboració va fer avançar l'arquitectura del sistema MILORD fins al punt on es troba avui dia, determinant part de les decisions de disseny que es van prendre.

El sistema MILORD va començar a ser dissenyat l'any 1984 sobre UPCLISP, un intèrpret de LISP en el desenvolupament del qual jo vaig participar [Cortés, Sierra i Villar, 1984] i que posteriorment donaria lloc a un llibre realitzat en col·laboració amb l'Ulises Cortés [Cortés i Sierra, 1987]. Més tard, i degut a la necessitat de disposar d'un intèrpret més potent, es va traslladar el desenvolupament a NIL [Burke i Carrette, 1982], per finalment, i a partir de 1986, utilitzar definitivament VAXLISP, versió DEC de l'estàndard Common LISP.

Actualment el sistema ha estat utilitzat per desenvolupar quatre aplicacions i està

essent utilitzat als següents centres de recerca:

- Centre d'Estudis Avançats de Blanes.
- Institut Municipal d'Investigació Mèdica. Barcelona.
- Aliança Mataronina. Mataró.
- Facultat d'Informàtica de la UPC.
- Escola d'enginyers de camins de la UPC.
- Facultat d'informàtica de la UPM.
- Departament d'Informàtica Universitat de Dortmund (RFA). Versió sobre SUN.
- Universitat de Rosario (Argentina).
- CINVESTAV Mèxic D.F. (Mèxic).
- Escola d'Enginyers Industrials. Universitat de Santiago. Vigo.
- Centro Nacional de Dosimetrias. Hospital "La Fe". València.

L'any 1987 un treball sobre el sistema MILORD presentat al congrés Avignon-87 va rebre el premi Digital europeu al millor treball de recerca en intel·ligència Artificial [Godo et al, 1987].

## Agraïments

Les idees exposades en aquesta tesi són el resultat de converses i discussions amb molta gent. Principalment amb el meu director, Ramon López de Mántaras, que em va introduir en l'estudi de la intel·ligència artificial, i que m'ha guiat al llarg d'aquests cinc anys de col·laboració fructífera. Altres companys del Centre d'Estudis Avançats de Blanes m'han proporcionat idees i suggeriments molt importants. Enric Plaza i Lluís Godo han ajudat decisivament en l'estructuració de diferents parts de la tesi. Jaume Agustí m'ha introduït en els sistemes d'especificació formal que han permès la modularització de MILORD i que obren importants línees futures de desenvolupament del sistema.

Agraeixo, molt especialment, a l'Albert Verdaguer els centenars d'hores de discussió, reflexió i treball conjunt sobre els processos de raonament en medicina concretitzats en l'aplicació PNEUMON-IA. Sense la seva col·laboració aquesta tesi no hagués estat possible.

Les converses amb Josep Aguilar i Ton Sales, sempre interessants, han ajudat a definir clarament les limitacions i les utilitats d'eines com la que es presenta.

Agraeixo també les idees proporcionades per tots els experts, a més de l'Albert Verdager, que han participat en el desenvolupament d'aplicacions utilitzant MILORD: Miquel Belmonte, Rudolf Felix, Achim Höffmann, Peter Grabienski i Rafael Sanz.

Per últim, agraeixo a la Carme el seu encoratjament constant i la seva ajuda al llarg d'aquests anys de treball.

Aquesta tesi ha rebut el suport dels següents projectes científics:

+ ACRE projecte número 836 de la CAICYT

+ Projectes número 87/1387 i 88/1382 del FISSS



# Index

1	Introducció .....	1
1.1	Plantejament dels problemes a resoldre.....	1
1.2	Descripció del sistema MILORD .....	2
1.3	Principals contribucions.....	4
1.4	Esquema dels capítols de la tesi .....	7
2	Representació de Coneixements de domini a MILORD.....	13
2.1	Introducció.....	13
2.2	Tipus de Coneixements.....	15
2.2.1	Coneixements teòrics.....	15
2.2.1.1	Models empírico-quantitatius .....	16
2.2.1.2	Models fisiopatològics.....	17
2.2.1.3	Coneixement Definicional.....	17
2.2.2	Coneixements empírics .....	18
2.2.2.1	Coneixements Heurístics.....	19
2.2.2.2	Coneixements Estadístics .....	21
2.3	Elements de representació.....	21
2.3.1	Fets.....	22
2.3.1.1	Sintaxi de la creació de símbols .....	23
2.3.1.2	Predicació sobre els fets.....	28
2.3.2	Regles d'inferència .....	35
2.3.3	Representació modular dels coneixements.....	36
2.3.4	Graf orientat de relacions funcionals .....	38
2.3.5	Us dels GRF per definir jerarquies pures.....	45
2.4	Compilació de les BC.....	48
3	Tractament del coneixement incert i incomplet.....	51
3.1	Introducció.....	51
3.1.1	La teoria probabilista .....	52
3.1.2	La teoria de l'evidència.....	59
3.1.3	La teoria de la possibilitat.....	64
3.2	Lògica difusa i coneixements incerts.....	69

3.2.1. Termes lingüístics com a valors de certesa.....	69
3.2.2. Semàntica dels termes lingüístics.....	70
3.2.2.1. Representació mitjançant intervals difusos .....	70
3.2.2.2. Representació com elements d'una lògica multi-valuada.....	77
3.2.2.3 Modus ponens a MILORD.....	81
3.2.2.3. Definició dels termes lingüístics .....	82
3.2.2.4. Definició d'una lògica difusa concreta .....	83
3.2.3. Anàlisi de sensibilitat .....	85
3.2.3.1. Criteris de cost .....	86
3.2.3.2. Criteris de rellevància .....	87
3.2.3.3. Criteris de rendiment.....	87
3.2.3.4. Criteris de sensibilitat .....	88
3.3. Coneixements incomplets. Subsumpció. ....	89
4 Nivell estructural. Modularització dels coneixements del domini .....	91
4.1. Introducció .....	91
4.2. Sintàxi del llenguatge de mòduls.....	94
4.2.1. Declaracions primitives.....	95
4.2.2. Declaració d'estructures .....	97
4.3. Conjunts .....	97
4.4. Especificacions.....	97
4.5. Mòduls.....	100
4.5.1. Declaracions encapsulades .....	100
4.5.2. Declaracions amb identificadors de mòdul .....	100
4.5.3. Aplicació d'un mòdul genèric.....	102
4.6. Mòduls genèrics .....	103
4.7. Semàntica de la modularització.....	105
4.7.1. Abstracció funcional .....	105
4.7.2. Amagament d'informació .....	106
4.7.3. Declaracions compartides .....	106
5 Control Multinivell a MILORD.....	109
5.1 Introducció .....	109
5.2 Tipus de raonament en Resolució de Problemes.....	111
5.2.1 Raonament Classificadori.....	112
5.2.1.1 Establir/refinar .....	112
5.2.1.2 Raonament hipotètico-deductiu .....	113



5.2.1.3 Raonament causal.....	114
5.2.1.4 Classificació Heurística.....	115
5.2.2 Raonament Generatiu .....	115
5.3 Arquitectura de control a MILORD .....	116
5.3.1 Metaregles .....	120
5.4 Nivell de l'univers de discurs.....	121
5.5 Nivell associatiu i relacional .....	121
5.5.1 Metaregles referents al nivell associatiu i relacional .....	121
5.6 Nivell estructural .....	123
5.6.1 Estructures de control.....	123
5.6.1.1 Nivells de tall .....	123
5.6.1.2 Tipus de cerca .....	124
5.6.1.3 Subsumpció.....	124
5.6.2 Sintaxi de les estructures de control.....	125
5.7 Nivell hipotètic .....	127
5.7.1 Estratègies .....	127
5.7.2 Metaregles referents a estratègies.....	128
5.7.2.1 Jerarquies pures versus jerarquies mixtes.....	130
5.7.3 Metaregles referents a excepcions .....	131
5.8 Nivell heurístic. ....	132
5.8.1 Plans d'actuació.....	132
5.8.2 Generació de Plans d'actuació. Metaregles referents a plans.....	135
5.8.3 Semàntica dels combinadors .....	141
5.8.3.1 Objectes semàntics .....	141
5.8.3.2 Equacions semàntiques.....	142
5.8.4 Condicions d'acabament. Metaregles d'acabament.....	144
5.8.4.1 Resultats del sistema.....	145
5.8.5 Solucions úniques versus solucions múltiples.....	145
5.9 Cicle de control a MILORD.....	146
6 Aplicacions .....	149
6.1 PNEUMON-IA.....	149
6.1.1 Justificació del sistema expert PNEUMON-IA .....	150
6.1.1.1 Establiment d'hipòtesis.....	150
6.1.1.2 Tractament sota incertesa.....	151
6.1.1.3 Conclusió .....	151

6.1.2	Arquitectura.....	152
6.1.2.1	Nivell de l'univers de discurs .....	152
6.1.2.2	Nivell associatiu i relacional.....	153
6.1.2.3	Nivell estructural.....	155
6.1.2.4	Nivell hipotètic.....	158
6.1.2.5	Nivell heurístic.....	162
6.1.3	Situació actual .....	166
6.1.3.1	Validació.....	166
6.2	Renoir.....	171
6.2.1	Objectiu.....	171
6.2.2	Arquitectura.....	173
6.2.3	Situació actual .....	175
6.3	Disseny VLSI.....	175
6.3.1	Objectiu.....	176
6.3.2	El problema de la selecció de l'arquitectura d'un chip .....	176
6.3.3	Arquitectura.....	178
6.3.3.1	Raonament aproximat .....	178
6.3.3.2	Coneixements associatius.....	178
6.3.3.3	Coneixements hipotètics .....	180
6.3.4	Situació actual .....	182
6.4	Control Industrial.....	182
6.4.1	Objectiu.....	183
6.4.2	Arquitectura.....	184
6.4.3	Situació actual .....	185
7.	Conclusions i línies futures .....	187
Apèndix I		
	BNF del llenguatge de MILORD .....	191
I.1	Definició del nivell d'univers de discurs .....	191
I.2	Definició dels nivells associatiu i estructural.....	192
I.2.1	Definició de la lògica per defecte.....	192
I.2.2	Definició dels mòduls no estructurats .....	192
I.2.3	Definició dels mòduls estructurats .....	194
I.3	Definició del nivell hipotètic .....	195
I.4	Definició del nivell heurístic .....	196
Apèndix II		
	Equacions semàntiques del compilador i interpret.....	197

II.1 Compilació del nivell d'univers de discurs .....	197
II.1.1 Objectes i operadors semàntics .....	197
II.1.2 Funcions de valuació .....	198
II.2 Compilació de regles .....	201
II.2.1 Objectes i operadors semàntics .....	201
II.2.2 Funcions de valuació de l'etapa I.....	201
II.2.3 Objectes i operadors semàntics de la interpretació.....	205
II.2.4 Funcions de valuació de la interpretació.....	206
II.3 Compilació del nivell estructural.....	210
II.3.1 Objectes i operacions semàntiques .....	210
II.3.2 Funcions de valuació de l'etapa I.....	210
II.3.3 Semàntica denotacional de la interpretació de les mre.....	212
Apèndix III	
Conjunt consensuat de termes .....	215
III.1 Introducció .....	215
III.2 Enquesta .....	215
III.2.1 Resultats.....	216
III.3 Elecció dels termes lingüístics.....	217
Apèndix IV	
Equacions semàntiques de la modularització .....	221
IV.1. Compilador.....	221
IV.1.1. Objectes i operacions semàntiques .....	221
IV.1.2 Equacions semàntiques.....	225
IV.2. Exemple de compilació.....	233
Apèndix V	
Exemple de modularització .....	241
V.1. Declaració de CONJUNTS.....	242
V.2. Especificacions.....	243
V.3. Mòduls auxiliars .....	243
V.4. Mòduls referents a la CLINICA .....	244
V.5. Mòdul referent a la RADIOLOGIA.....	245
V.6. Mòduls referents a LABORATORI.....	246
V.7. Mòdul referent al diagnòstic de PNEUMOCOC .....	248
Apèndix VI. Exemple d'interacció. Sortides. ....	
VI.1. Interacció amb el sistema .....	249
VI.2 Sortides del sistema .....	258

VI.2.1 Informe d'execució.....	259
VI.2.2 Informe de validació.....	262
Bibliografia.....	265

# 1 Introducció

OMNIS MUNDI CREATURA  
QUASI LIBER ET PICTURA  
NOBIS EST IN SPECULUM

*Alain de Lille*

;

Aquesta introducció està dividida en quatre apartats: un plantejament dels problemes a resoldre, una descripció del sistema proposat, les principals contribucions i un esquema dels capítols de la tesi.

## 1.1 Plantejament dels problemes a resoldre

La utilització dels sistemes basats en coneixements per a la resolució de problemes ha estat freqüent des de bastants anys ençà. Inicialment, i enfrontats a problemes de complexitat reduïda, els sistemes posseïen arquitectures poc estructurades, on el control i els coneixements de domini estaven barrejats de forma difícilment dissoluble [Shortliffe, 1976]. El desenvolupament d'aplicacions de gran mida, però, es fa inviable sinó es dissenyen eines que ajudin a la estructuració tant dels coneixements relatius al domini d'aplicació, com els referents al control. Diferents treballs han incidit en aquesta problemàtica des de diferents angles: aprenentatge [Van de Velde, 1988], arquitectures multi-nivell [Akkermans et al, 1989], aplicacions mèdiques [Pople, 1982].

Aquesta tesi incideix en el problema de l'estructuració i separació dels elements de control i de domini de les aplicacions. Es proposa una arquitectura en diferents nivells de control i metacontrol que permeti la programació modular i ajudi, per tant, a un millor manteniment i actualització de les bases de coneixements.

Aquesta arquitectura multi-nivell de control proposada ha estat orientada a la resolució de tasques de classificació, entenent com a classificació la cerca de solucions

dins d'un espai finit i conegut a priori. S'ha fet així per dues raons fonamentals: la major part de les aplicacions dels sistemes experts es desenvolupen entorn a la classificació, els experts amb els que he col·laborat plantejaven problemes que es podien resoldre des d'aquest angle. Malgrat això, entenem que els criteris definidors de l'arquitectura multi-nivell són fàcilment adaptables a altres tipus de problemes.

Finalment, un dels punts que ha definit els treballs que he realitzat ha estat el desig que les aplicacions desenvolupades fossin totes elles útils en els entorns on s'havien d'aplicar. Un dels problemes que sempre s'esmenten referents als sistemes experts és el seu baix o nul ús per part dels suposats utilitzadors. Aquesta voluntat, més aviat restricció, que els sistemes fossin útils ha marcat moltes de les decisions preses en el disseny de l'arquitectura.

## 1.2 Descripció del sistema MILORD

L'arquitectura proposada ha estat fruit de l'experiència adquirida en la resolució de problemes, sobre tot en el domini mèdic. Aquesta interacció constant amb els problemes a resoldre ha permès eliminar solucions, teòricament interessants d'entrada, que s'allunyaven de la realitat on s'havien d'implantar. Així, el sistema es va plantejar com un sistema que pogués ser utilitzat pels usuaris finals de forma eminentment interactiva, orientat a l'ús directe en l'entorn dels potencials usuaris. A continuació esmentem els elements arquitectònics més rellevants de MILORD.

*La simplicitat com a factor determinant en l'ús per part dels experts.* L'ús de les regles de producció com a llenguatge de base per definir les unitats de coneixement proporciona un marc simple on els experts es troben força còmodes a l'hora d'expressar-se. Al mateix temps permet utilitzar sistemes de validació extrets dels esquemes lògics que suporten els mecanismes de deducció útils per una definició incremental segura de les Bases de coneixements.

*El raonament aproximat com a element determinant de l'arquitectura.* La major part dels dominis que plantegen problemes complexos utilitzen esquemes de raonament que s'allunyen del marc booleà. La modelització d'aquests esquemes és determinant del tipus d'arquitectura que es proposa. El marc on s'inscriu MILORD és el de la lògica difusa, que permet la utilització de termes lingüístics per modelitzar la incertesa. Aquesta decisió ve determinada, novament, per l'intent d'apropar-se al tipus de representació de la incertesa que utilitzen els experts, fonamentalment mitjançant termes lingüístics. Així,

tant els fets com les regles i les estratègies, estan ponderats lingüísticament. Aquest element determina els algorismes d'avaluació proposats.

*La credibilitat de les aplicacions.* Un dels aspectes que els usuaris finals valoren molt és que la seqüència de preguntes realitzades pel sistema segueixin una línia de raonament semblant a la que ells mateixos farien. Aquesta característica no és en absolut menyspreable, ja que determina la utilització que es farà de les aplicacions desenvolupades. L'arquitectura de control ha estat fortament determinada per l'assoliment d'aquesta característica.

*La modularització com la forma de definir grans aplicacions.* El desenvolupament d'aplicacions de gran mida requereix d'eines que permetin el disseny per parts, la reutilització de fragments de les bases de coneixements i el disseny incremental. Aquest element ha determinat la definició de mòduls que englobin regles i metaregles com a unitats que es poden imbricar, i que modelitzen mecanismes d'herència. Aquesta modularització permet tanmateix estructurar correctament, al mateix temps, aquells subdominis que posseïxin una estructura ben definida.

*La distinció de nivells de coneixements com una ajuda a l'estructuració de problemes mal definits.* La separació dels diferents tipus de coneixements: associatius, estructurals, hipotètics i heurístics, utilitzats en la resolució d'un problema ajuda a millorar l'estructura de molts dominis, els quals degut a la seva complexitat, o bé degut a la seva dispersió, es troben mal definits o incomplets. Així es pot veure la metodologia proposada, no únicament com una metodologia de construcció de sistemes experts ben estructurats, sinó com una metodologia útil per als experts, per conèixer millor els processos de raonament que segueixen. Així no sembla forassenyat pretendre que s'introdueixi com a element en els ensenyaments d'àrees com la medicina, la biologia, o d'altres, el disseny de sistemes basats en els coneixements, com a reflexió metodològica dels futurs experts sobre la seva àrea de coneixements. Aquest punt pot ser un dels punts d'incidència dels sistemes experts en dominis on el seu ús, contràriament al que es podria pensar, és pobre o nul.

*Descripció formal de la semàntica del llenguatge.* A diferència de les presentacions de la majoria de llenguatges de desenvolupament de sistemes experts, que en el millor dels casos proporcionen una semàntica operacional, en la present memòria s'ha donat la semàntica denotacional de la major part dels constructes de MILORD. Aquest esforç és

necessari per desambiguar els diferents elements del llenguatge, així com per permetre la reutilització fàcil de les idees aquí presentades en futurs llenguatges de desenvolupament de sistemes experts. Al mateix temps aquest esforç ha permès fixar i amillarar certs punts de l'arquitectura que estaven dissenyats amb un grau de generalitat baix.

La implementació de MILORD s'ha dut a terme sobre VAXLISP, dialecte de CommonLisp, en màquines VAX de la companyia DEC amb operatiu VMS. Tanmateix, existeix una versió també en CommonLisp per màquines SUN amb operatiu UNIX. La codificació de MILORD ha representat la programació d'aproximadament 9000 línees de codi LISP.

### 1.3 Principals contribucions

*Un marc estructurat per a la resolució de problemes en funció dels tipus de coneixements. Capítol dos.*

Les arquitectures de desenvolupament de sistemes experts havien tingut en compte fins ara els tipus de coneixements que es manipulaven des d'un punt de vista descriptiu i no estructural [Szolovits, 1985 1988], [Clancey, 1985a].

L'arquitectura proposada en canvi utilitza els tipus de coneixements com a elements essencials d'estructuració. Parteix del principi que diferents tipus de coneixements determinen diferents tasques, així per exemple els coneixements associatius estan relacionats amb els processos deductius, mentre que els coneixements heurístics estan lligats als processos de restricció dels espais de cerca. La separació explícita dels elements de l'arquitectura, segons el tipus de coneixements que representen, facilita l'estructuració de les diferents tasques que es volen modelitzar en una aplicació, evitant la barreja que els sistemes experts de primera generació feien, on, en un mateix element de representació, coexistien sovint elements heurístics, associatius i relacionals, per exemple una premissa d'una regla podia representar el coneixement heurístic sobre l'oportunitat de validar la conclusió, mentre la resta de premisses podien determinar l'associació entre la conclusió i un conjunt de símptomes.

*Una representació i gestió dels coneixements incerts i incomplets adaptables al raonament habitual dels experts. Capítol tres.*

En els sistemes experts de primera generació ha estat habitual proposar marcs generals de raonament, als quals les aplicacions havien d'emmotllar-se [Duda et al, 1979] [Shortliffe, 1976]. La realitat indica que dins d'un marc general de solució de



problemes no hi ha un model de raonament, de combinació i propagació d'evidències, vàlid per totes les aplicacions. Així hem definit una aproximació a la gestió dels coneixements incerts en la qual, i dins el marc global de la lògica difusa, l'expert té una incidència directa en la determinació dels processos de combinació d'evidències. Si bé el tipus de gestió dels coneixements incerts és global dins una aplicació concreta, la seva extensió a fer la combinació d'evidències local a les subtasques d'una aplicació és fàcil gràcies als mecanismes de modularització introduïts. Al mateix temps hem optat per una representació de la incertesa de forma simbòlica, per tal d'evitar els greus problemes que la gestió numèrica ha provocat en molts sistemes [Zadeh, 1979 1983] [Godo i Sierra, 1988].

*Un formalisme de modularització per facilitar la programació de grans aplicacions. Capítol quatre.*

L'empaquetament d'elements de representació ha estat realitzat, en general, no com un sistema d'ajuda a l'estructuració de les aplicacions, sinó com un mecanisme de divisió de l'aplicació en trossos petits, "Rules sets" a KEE, que facilitessin la tasca dels mecanismes d'inferència. Els treballs desenvolupats entorn del llenguatge ML [Harper et al, 1986], han inspirat el mecanisme de modularització definit. Amb aquest procés de modularització els mòduls juguen un paper essencial en l'estructuració de les tasques d'una aplicació, permeten la reutilització de parts de les aplicacions, l'abstracció funcional de diferents parts de les aplicacions, així com la possibilitat de definir comprovacions de tipus entre especificacions de bases de coneixements i la seva programació concreta. S'han definit mòduls genèrics que permetin una reutilització del codi més alta, així com nivells d'abstracció elevats. Avui dia, totes aquestes característiques van en la línia de permetre la definició i manteniment de grans bases de coneixements, coll d'ampolla del disseny de sistemes basats en coneixements.

Aquests elements utilitzats en els llenguatges informàtics no havien estat utilitzats encara en els llenguatges de la Intel·ligència Artificial, tot i que sembla que hi ha un cert interès darrerament en intentar línies semblants a la proposada en aquesta tesi [Sannella i Wallen, 1987], [Akkermans et al, 1989], [Agustí i Sierra, 1989].

*Una arquitectura multi-nivell que permet separar clarament els elements del domini dels elements de control. Capítol cinquè.*

La separació entre els elements de domini i els elements de control ha estat una característica enunciada, encara que en moltes ocasions no aconseguida, per tots els sistemes experts des del seu origen. Possiblement el primer sistema amb una separació

clara entre els elements de control i els elements de domini sigui HEARSAY [Eрман et al, 1980], amb una descripció abstracta a diferents nivells de les solucions i coneixements de control per focalitzar-se sobre els elements de solució. Hi ha abundants exemples sobre l'ús d'elements implícits de control en sistemes basats en regles [Aikins 1983]. L'arquitectura proposada proporciona els mecanismes necessaris per realitzar aplicacions on els elements de control estiguin perfectament separats, i definits explícitament. Altres treballs han desenvolupat arquitectures on es diferencien els nivells de control [Beetz, 87], [Bylander i Chandrasekaran, 1986], [Clancey i Bock, 1986], [Genesereth, 1983]. Les aproximacions realitzades fins ara es poden classificar en dos tipus:

Aproximació al meta-control des de la modularització: És el cas de HERACLES [Clancey i Bock, 1986] i de les tasques genèriques [Bylander i Chandrasekaran, 1986], on el control s'associa a una jerarquia de tasques/subtasques. El control és en aquests sistemes un element que acompanya a la tasca de forma inseparable, no existeix, per tant una jerarquització dels elements de control de forma independent dels elements del domini..

Un únic nivell de Meta-control separat del llenguatge de base: És el cas de CATWEAZLE [Beetz, 87] i de Teiresias [Davis i Lenat, 1982].

En l'arquitectura de MILORD es separa la modularització dels elements de control que s'estructuren en quatre nivells, definint una arquitectura combinació dels dos enfocaments. Així, per una banda es defineix una estructura modular dels constructes del domini que engloba els elements de control referents a la utilització dels coneixements de domini (herència, visibilitat, abstracció/particularització) i per altra els elements de control referents a la selecció dels constructes a utilitzar (estratègies de resolució).

#### *Una aplicació en el camp mèdic pionera a Espanya. Capítol sisè.*

Utilitzant l'arquitectura MILORD s'ha desenvolupat una aplicació: PNEUMON-IA pionera a Espanya en dos aspectes:

- 1.- És la primera aplicació mèdica utilitzada regularment en el servei d'urgències d'un hospital (L'Aliança de Mataró). Des de l'octubre de 1988 l'aplicació PNEUMON-IA ha estat utilitzada en més de 30 casos de pneumònies extrahospitalàries amb gran acceptació per part dels metges que la utilitzen.

- 2.- És la primera aplicació sobre la qual s'ha desenvolupat una validació multiexpert .Ha demostrat un nivell de competència comparable a la dels experts humans que han participat en la validació: cinc experts de reconegut prestigi de l'àrea de Barcelona.

## 1.4 Esquema dels capítols de la tesi

Aquesta tesi ha estat estructurada en sis capítols i 6 apèndixs. S'han deixat els aspectes més tècnics als apèndixs per facilitar la lectura de la tesi.

El capítol dos està dedicat a la representació dels coneixements de domini a MILORD. En aquest capítol es fa una reflexió inicial sobre els tipus de coneixements a manipular, per passar a continuació a definir els elements de representació que s'utilitzen a MILORD per representar els coneixements de domini. Aquests elements són els fets, les regles, les metaregles, els mòduls i els grafs de relacions funcionals. Per cadascun d'aquests elements es defineix la sintaxi i la seva semàntica. Una definició formal de la semàntica d'aquests elements es pot trobar a l'apèndix 2. Es pretén al llarg d'aquest capítol mostrar la capacitat d'aquests elements per representar els tipus de coneixements descrits al començament. Es dona especial èmfasi a l'explicació dels grafs de relacions funcionals que permeten definir una relació de subsumpció sintàctica i semàntica entre els conjunts de fets i que defineix un dels criteris d'ordenació de regles utilitzats pels motors d'inferència. Finalment, es fa una descripció del procés de compilació de la sintaxi externa cap a uns objectes semàntics propers a la representació interna. La resta d'elements de representació, sobre tot els de control, es fa en ulteriors capítols.

El capítol tres conté la descripció detallada dels mecanismes de raonament aproximat definits a MILORD. Després d'una descripció inicial dels tres models més importants per representar el raonament aproximat: el probabilista, l'evidencial i el possibilístic, es passa a l'anàlisi de les raons que ens han fet triar el model possibilístic com a òptim per les necessitats que els problemes abordats plantegen. Dins el model triat es descriuen les dues semàntiques associades als termes lingüístics que l'expert defineix per a la seva aplicació. Aquestes interpretacions estan basades en els intervals difusos, i en la lògica multivaluada. S'arriba a la conclusió que ambdues interpretacions són satisfactòries, encara que la segona ofereix certs avantatges des del punt de vista de la comoditat de definició. La definició del mecanisme de subsumpció utilitzant les relacions definides al capítol dos sobre els grafs de relacions funcionals clou el capítol.

El capítol **quatre** està destinat a descriure la modularització del llenguatge MILORD utilitzant tècniques d'especificació formal. Aquestes tècniques, inspirades en el model del llenguatge ML [Harper et al, 1986], permeten definir mecanismes com l'amagament d'informació o l'herència que són imprescindibles per definir grans bases de coneixements reutilitzables i fàcilment validables. Es defineixen mecanismes de comprovació de tipus que faciliten la tasca d'estructuració dels dominis a modelitzar. Aquesta aproximació a la modularització s'ha desenvolupat únicament sobre els elements de representació del domini, fets, regles i metaregles referents a l'aplicació de les regles. L'extensió de la modularització a la resta de nivells de control és un camí obert que comença a despertar interès en altres grups de recerca [Akkermans et al, 1989].

L'aproximació feta al capítol quatre té un abast més ampli que el de l'arquitectura de MILORD, donat que defineix un mètode de modularització fàcilment utilitzable en altres llenguatges de desenvolupament de sistemes experts. La semàntica d'aquest nivell, descrita a l'apèndix quatre, pot servir per a ser reutilitzada per qualsevol altre llenguatge sobre el que es vulguin definir constructes de modularització.

El capítol **cinquè** fa una descripció inicial dels tipus de raonament més comuns en resolució de problemes, per passar a descriure en detall els diferents components estructurals de l'arquitectura multi-nivell de MILORD. Per cadascun dels nivells de l'arquitectura es descriuen els elements de representació de domini que hi intervenen, així com els elements de control encarregats de la seva utilització. Els diferents nivells de l'arquitectura s'han fet correspondre al diferents tipus de coneixements implicats en els tipus de problemes que s'intenta resoldre. Els nivells de l'arquitectura són els següents:

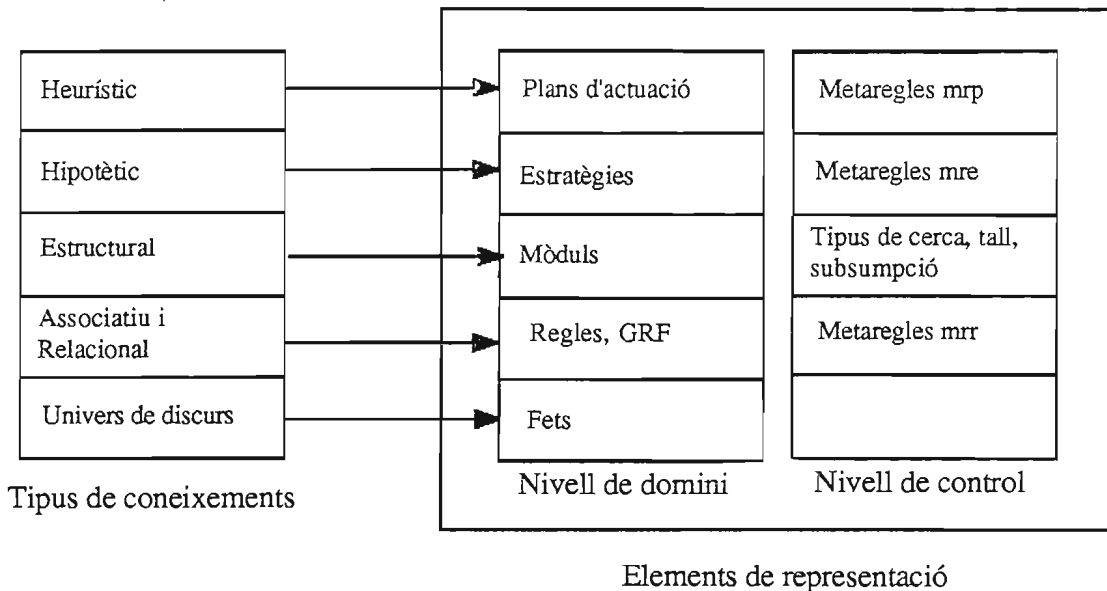


Fig 1.1 Arquitectura multi-nivell de MILORD.

*Nivell d'univers de discurs:* referent als elements conceptuals del domini, els fets.

*Nivell associatiu i relacional:* referent a les associacions i relacions existents entre els conceptes, representades com a regles i grafs de relacions funcionals. En aquest nivell es defineix el primer tipus de metaregles, que en aquest cas controlen l'aplicabilitat de les regles.

*Nivell estructural:* referent als agrupaments de les regles i fets en paquets anomenats mòduls, i als paràmetres que defineixen la forma en que aquests mòduls hauran de ser avaluats.

*Nivell hipotètic:* referent al procés d'establiment d'hipòtesis, que han estat definides en forma d'estratègies generades pel segon tipus de metaregles de l'arquitectura: les metaregles que generen les estratègies de resolució.

*Nivell heurístic:* referent a la combinació de les regles en el que s'anomenen plans d'actuació, elements que dirigeixen el procés d'execució de MILORD. En aquest nivell, l'últim, es defineix el tercer tipus de metaregles de l'arquitectura, aquelles encarregades de la combinació d'estratègies per formar plans d'actuació i de resoldre els possibles conflictes entre estratègies.

Al llarg d'aquest capítol cinquè també es presenten les estructures sintàctiques dels diferents components, deixant a part la semàntica que serà descrita a l'apèndix dos.

L'últim apartat del capítol cinquè està destinat a la descripció del cicle de control de MILORD, i permet acabar d'entendre el funcionament de l'arquitectura multi-nivell de control.

El capítol sisè conté la descripció de la implementació de MILORD i de les quatre aplicacions que han estat desenvolupades fins al moment utilitzant MILORD. Aquestes aplicacions han estat en moltes ocasions les responsables de les decisions preses, sempre amb un nivell d'abstracció, sobre l'aplicació que les motivava. Aquestes aplicacions han estat desenvolupades amb la col·laboració d'experts d'àrees geogràfiques molt diferents, i ha suposat un alt grau d'esforç per part seva. De les aplicacions desenvolupades dues ho són en l'àmbit de la medicina, una en l'àmbit de les arquitectures VLSI i l'altre en la del control adaptatiu:

**PNEUMON-IA:** Aplicació desenvolupada en col·laboració amb l'Albert Verdaguer, cap de medicina interna de l'Aliança de Mataró. Aplicació al diagnòstic de pneumònies extrahospitalàries. Aquesta aplicació és la primera aplicació mèdica que s'ha instal·lat a Espanya en un servei d'urgències, i la primera també sobre la qual s'està realitzant una validació multiexpert. Conté aproximadament 800 regles i 100 metaregles.

**RENOIR:** Aplicació desenvolupada en col·laboració amb en Miquel Belmonte, metge adjunt de reumatologia de l'Hospital General de Castelló. Aplicació al diagnòstic de malalties reumàtiques. Actualment es troba en fase de validació al mateix Hospital General de Castelló. Conté aproximadament 500 regles i 80 metaregles.

**VLSI:** Aplicació desenvolupada en col·laboració amb la Catedra d'informàtica de la Universitat de Dortmund a la RFA. Aplicació a la selecció d'arquitectures de xips. Actualment en fase d'utilització al mateix departament. Conté aproximadament 300 regles i 10 metaregles.

**SOLAR:** Aplicació desenvolupada en col·laboració amb en Rafael Sanz, professor titular de l'Escola d'Enginyers Industrials de la Universitat de Santiago de Compostela. Aplicació al control d'una planta d'energia solar. Actualment és un petit prototipus en fase de desenvolupament. Conté 40 regles i cap metaregla.

Finalment s'inclou un capítol de conclusions on es fa un repàs a les aportacions

d'aquesta tesi i sis apèndixs on es detallen certs aspectes més tècnics. El formalisme de descripció de la semàntica dels constructes ha estat la semàntica denotacional. Tanmateix, s'ha fet al llarg de la tesi una descripció informal dels constructes sintàctics que pot ser suficient per al lector que no estigui interessat en una descripció massa detallada. La sintaxi presentada al llarg de la tesi és una versió més llegible de la sintaxi implantada.





## 2 Representació de Coneixements de domini a MILORD

RATIO COGNOSCIBILIS, RATIO APPETIBILIS.

### 2.1 Introducció

Dins la bibliografia d'intel·ligència artificial s'utilitza el terme "representació de coneixements" allà on la informàtica utilitza "estructuració de les dades". És interessant remarcar les diferències semàntiques entre ambdues expressions. El verb *estructurar* significa donar estructura, arranjar, disposar les parts d'una cosa. Implica una actitud activa per part de qui realitza l'acció, les dades són en aquesta acció l'element passiu. *Representar* significa "fer present" alguna cosa per símbols, imatges, etc. L'actitud de qui representa és intentar donar una idea clara d'allò que vol representar sense modificar el seu significat. Si quan s'estructura, les estructures i els objectes a estructurar són entitats independents, quan es representa, les representacions i els objectes formen un tot inseparable.

En el cas de la informàtica el que convé és utilitzar estructures de representació sobre objectes dels quals no ens interessa la semàntica. Podrem crear fitxers indexats sobre el DNI d'una persona o sobre la matrícula d'un cotxe.

En les aplicacions de la IA és necessari realitzar una anàlisi més detallada entre:

- a) *elements de representació*, i
- b) *tipus de coneixements a representar*.

Els estudis sobre els elements de representació són abundants a la IA. En aquests estudis es fa un repàs a diferents esquemes de representació sense analitzar en profunditat quins tipus de coneixements s'hi representarien millor i perquè. En aquest sentit aquests estudis són essencialment semblants als de la informàtica, on es presenten estructures de representació de dades.

Estudis més recents es concentren sobre el problema de la taxonomia dels coneixements [Szolovits, 1985], [Clancey, 1985], [Chandrasekaran, 1983]. Aquests treballs estudien els diferents tipus de coneixements des del punt de vista dels processos cognitius que els éssers humans desenvolupen. Així, s'analitzen el coneixement associatiu, heurístic o hipotètic. Aquests estudis si bé resulten interessants per l'enfocament que proporcionen, són necessàriament aproximacions incompletes, generalitzacions de determinades aplicacions o dominis. Dins aquest capítol presentarem la visió que tenim dels tipus de coneixements, donat que això determinarà les decisions posteriors sobre els elements arquitectònics.

La descripció de la representació dels coneixements de les arquitectures de desenvolupament de SE s'ha fet tradicionalment donant una descripció dels elements de representació. Dir que el sistema és basat en *regles*, basat en "*frames*", *orientat objecte*, etc., acostuma a ser suficient per a que l'usuari tingui una idea del tipus d'aplicacions que hi podrà desenvolupar i amb quin grau d'esforç.

En la descripció de la representació dels coneixements realitzada en aquest capítol plantejarem una estratègia diferent. Primer introduïrem els tipus de coneixements que han inspirat l'arquitectura de MILORD, després parlarem dels elements de representació utilitzats i la relació entre uns i altres. D'aquesta manera seguirem una metodologia de presentació més propera a l'anàlisi del procés de resolució de problemes realitzat a MILORD.

Cal establir una distinció entre el que considerarem *coneixements de domini* i *coneixements de control* al llarg d'aquest treball. Direm *coneixements de domini* als coneixements referents al camí seguit entre les dades del problema i la solució trobada, així, els fets, les regles i els mòduls com a contenidors de regles, seràn considerats elements pertanyents als *coneixements del domini*. Direm *coneixements de control* als coneixements referents a la determinació del camí d'obtenció de la solució, així les metaregles i els mecanismes explícits de control seran *coneixements de control*. Això és així, independentment del tipus de conceptes que ambdós tipus de coneixements utilitzin per estructurar-se. Les regles, considerades coneixements de domini, i les metaregles, considerades coneixements de control, poden utilitzar els mateixos fets com a elements de les seves premisses.

Els elements de representació que seran objecte d'estudi en aquest capítol són únicament els referents a coneixements de domini: fets regles i mòduls. La resta de elements referents a coneixements de domini i els elements referents a coneixements de control seran tractats en ulteriors capítols.

## 2.2 Tipus de Coneixements

Farem aquí una anàlisi sobre els tipus de coneixements que han orientat el disseny de MILORD, i quines aplicacions poden abordar-se amb aquest conjunt de tipus de coneixements.

Els coneixements es poden entendre com a formalitzacions de relacions entre conceptes, obtingudes o no a partir de l'experiència.

Els coneixements es poden agrupar en dues categories: *coneixements teòrics* i *coneixements empírics*. Els coneixements teòrics representen els cossos doctrinals complets i sistemàtics sobre una determinada ciència. Aquests coneixements estan estructurats en forma de models, representacions simplificades (o abstractes) de la realitat. Els coneixements empírics concerneixen la pràctica, en tant que oposat a l'especulatiu i teòric. No cal dir que els dominis on els sistemes experts es situen són una barreja d'ambdós tipus de coneixements. En ciències biològiques, i més concretament en el camp de la medicina, s'utilitzen determinats models científics: models físics, químics, matemàtics, etc., encara que els coneixements pràctics són una peça clau en tasques complexes com el diagnòstic o el tractament. La preponderància d'un tipus de coneixements o d'uns altres determina fortament el tipus d'arquitectura que els sistemes experts tindran. Si ens trobem per exemple, en el camp de les matemàtiques on els coneixements teòrics són els fonamentals, els sistemes deductius no reben el nom de sistemes experts, sinó que són anomenats sistemes de deducció automàtica, encara que s'han desenvolupat sistemes que potencien els coneixements pràctics com és el cas de MACSYMA [MACSYMA, 1974].

### 2.2.1 Coneixements teòrics

Els models científics hauran de ser explotats, allà on es posseeixin. Malhauradament les aplicacions més comunes en el camp dels sistemes experts es situen en dominis on els models científics són marcadament incomplets: medicina, biologia, etc. Malgrat això, aquests dominis utilitzen com a submodels models físics o químics que és necessari representar. Per exemple: l'auscultació dels sorolls patològics pulmonars és explicada per les lleis de l'acústica. Si tenim present que la transmissió del só es més eficient en els sistemes sòlids que en els gasosos podem comprendre l'aparició d'un *buf tubàric*<sup>1</sup> quan hi ha una condensació de l'espai aeri<sup>2</sup> de mida

---

<sup>1</sup> Só respiratori que recorda l'efecte produït al bufar per un tub causat pel pas de l'aire per la tràquea.

<sup>2</sup> Alvéols ocupats per materials no gasosos.

suficient. La regla:

R01005 Si Buf\_tubàric

**Llavors**

**concloure consolidació\_espai\_aeri és segur**

" Principi físic: el soroll es transmet millor en medis sòlids que en medis líquids o gasosos. El soroll traquial que normalment no és audible a través del pulmó, mitjà gasós, es fa audible en l'auscultació si el mitjà gasós es transforma en sòlid, es a dir en el cas d'una condensació pulmonar "

té, per tant, una explicació no merament empírica sino basada en un model físic.

A part dels models comuns a la major part de les ciències aplicades: models físics, químics i matemàtics, en el cas concret de la medicina s'han desenvolupat alguns models parcials específics<sup>1</sup> que descriurem en els següents apartats, donat que ens interessen particularment per ésser la medicina el camp on s'inscriuen les més grans aplicacions de MILORD:

- (1) *models empírico-quantitatius* com podrien ésser els models farmacològics
- (2) *models fisiològics* i els seus derivats com ara els fisiopatològics.

### 2.2.1.1 Models empírico-quantitatius

Els coneixements empírico-quantitatius en les ciències mèdiques es basen en models quantitatius d'algun aspecte de la fisiologia humana. Aquests tipus de coneixements són útils en la clínica quan es tracten certs aspectes de la medicina com ara la farmacologia.

Per exemple la indicació i administració d'un determinat antibiòtic per tractar una infecció depèn de l'acció del fàrmac sobre el microorganisme, del comportament farmacocinètic de l'antibiòtic i de l'experiència clínica.

En una pneumònia pneumocócica l'elecció de penicil·lina es basa en primer lloc en el coneixement de la sensibilitat del *S. pneumoniae* a aquest antibiòtic.

Per poder dosificar el fàrmac és necessari tenir en compte el model farmacocinètic de la penicil·lina (la seva absorció per diferents vies d'administració, el seu volum de distribució, el seu lligam a les proteïnes, i la seva taxa de metabolització,

---

<sup>1</sup> Cal tenir en compte que encara que aquests es basin en últim terme en models més generals com poden ser els físics i químics, aquest reduccionisme a les fonts últimes del coneixement no és el més adient quan cal enfocar la resolució de problemes pràctics en els que, en l'estat del coneixement actual, no es disposen de totes les dades necessàries i encara aquestes són ambígues en moltes ocasions.

expressada com a vida mitjana ( $T_{1/2}$ ).

Hi ha dades no relacionables amb al model farmacològic que determinen la seva elecció com són: l'experiència prèvia amb el fàrmac, els possibles efectes indesitjables d'aquest, la comoditat en l'administració i el cost econòmic. També en la dosificació influeixen dades purament empíriques com és la necessitat o no d'aconseguir una determinada concentració de l'antibiòtic superior  $n$  vegades a la concentració mínima inhibidora, o aspectes més pràctics com és la durada del tractament.

La combinació de dades empíriques i de dades quantitatives basades en determinats models (com és el farmacocinètic) fa que aquests models siguin anomenats empírico-quantitatius.

### 2.2.1.2 Models fisiopatològics

Els models fisiològics o fisiopatològics expresen el funcionament o la disfunció de diferents aspectes de la fisiologia humana. La característica principal d'aquest tipus de coneixements és el seu caràcter explicatiu. La seva representació en els sistemes informàtics ha estat en general en forma de xarxes de relacions causa-efecte que reben el nom, moltes vegades, de *coneixements causals*.

Encara que els avantatges d'una representació més explícita i més profunda són evidents no sempre és convenient retrocedir a models cada vegada més específics per explicar les malalties i les seves manifestacions. Encara que el coneixement mèdic es pogués reduir sempre a les seves bases físico-químiques seria molt ineficient i de nulla utilitat des del punt de vista del clínic raonar d'aquesta manera. Els clínics, en la pràctica mèdica, tenen models de més alt nivell i només ocasionalment prenen en consideració nivells més profunds. Per altra banda el coneixement és moltes vegades tan incomplet que els programes basats en models molt profunds difícilment són efectius.

### 2.2.1.3 Coneixement Definicional

Des d'un punt de vista epistemològic es consideren les definicions com a proposicions per les que es fixen o delimiten els conceptes. Fonamentalment són eines d'ajut en l'estructuració del coneixements en jerarquies d'abstraccions a diferents nivells. És comú que el seu ús estigui determinat per l'acord dels experts del domini, o per raons històriques. En molts casos l'ús de definicions correspon a processos d'abstracció qualitativa [Clancey, 1985a], mitjançant els quals els experts abstrueixen de les dades observables elements útils des del punt de vista dels processos de raonament posteriors.

Clancey distingeix tres tipus de coneixements definicionals o factuais:

*Abstracció definicional*: Abstracció, canvi d'escala, a partir de característiques necessàries d'un objecte.

*Abstracció qualitativa*: Tipus d'abstracció definicional quan les dades que s'abstreuen són numèriques.

*Generalització*: Abstracció realitzada a través d'una jerarquia de tipus.

Per exemple la leucocitosi és una abstracció qualitativa representada per les següents regles:

R03002 Si leucòcits > 20000 llavors **concloure** leucocitosi és segur

R03005 Si leucòcits > 10000 i leucòcits < 15000 llavors

**concloure** leucocitosi és possible

### 2.2.2 Coneixements empírics

La resolució de problemes en els dominis esmentats ha de fonamentar-se també en coneixements empírics, no derivats de models formals. Això es justifica per diverses raons:

- 1.- Els models formals no són complets.
- 2.- L'ús de models fonamentals, físics o químics, a nivell molt profund comporta ineficiència i un allunyament de la realitat en la resolució de problemes pels experts.
- 3.- Les dades disponibles no són suficients per utilitzar models formals, i encara aquestes són ambíguas en moltes ocasions.

Degut a això s'ha de fer un gran esforç per veure quins són els diferents tipus de coneixements empírics que utilitzen els experts i quina serà la seva millor representació.

Els coneixements empírics, en tant que coneixements obtinguts a través de l'experiència poden tenir dos orígens diferenciats:

*Coneixements subjectius o heurístics*: són associacions entre conceptes provinents de la pràctica de l'expert i difícilment objectivables. Per exemple la relació entre la brusquetat de la aparició d'una pneumònia i el seu origen bacterià.

*Coneixements estadístics*: el seu origen prové d'observacions freqüentistes de fenòmens i d'associacions entre conceptes. Moltes vegades aquests coneixements són compartits per diferents experts del domini o bé per la comunitat sencera. Per exemple el càncer de pulmó està relacionat amb el tabaquisme d'una manera estadísticament

significativa.

### 2.2.2.1 Coneixements Heurístics

Són uns tipus de coneixements basats en l'experiència de l'expert, ja sigui la pròpia o la que ha esdevingut comú en un domini específic. Aquests tipus de coneixements, a diferència d'altres, tenen una relació estreta amb els processos de raonament que l'han d'utilitzar.

Aquests tipus de coneixements són especialment útils en medicina donat que els models fisiopatològics en que les ciències biològiques es basen són sovint incomplets, i certes relacions entre els elements del raonament no poden ser explicats per un raonament de tipus causal. Tanmateix encara que els models fisiopatològics poguessin ser complets, són de tal complexitat que els experts no els solen emprar. En aquest darrer cas els coneixements de tipus heurístic podrien ser el resultat d'una compilació de coneixements més profunds. Es a dir, els conceptes utilitzats són conceptes que n'agrupen d'altres més elementals, i les relacions entre ells són una abstracció de les relacions que tenen entre sí aquests conceptes més elementals.

Els coneixements de tipus heurístic sovint estan mal definits i mancats de precisió. No ajuden a la sistematització dels coneixements i per això no se'n fa esment en les presentacions acadèmiques. Són coneixements estretament lligats a la pràctica, i són, per tant, indispensables per a la simulació del raonament de l'expert ja que restringeixen l'àmbit de la cerca de les solucions dels problemes i permeten dirigir-la cap a objectius versemblants. És a dir, els coneixements de tipus heurístic són útils no tan sols per representar part dels coneixements de domini sinó que, en els sistemes experts, les heurístiques del sistema indiquen sovint l'estratègia de resolució que s'ha de seguir, (veure 5.9).

Els coneixements heurístics poden entendre's com a explicacions d'allò que és típic. Allò que en una situació seria cert la major part de vegades [Clancey, 1985a], [Plaza, 1987].

En el camp de la clínica mèdica, l'heurística en l'elaboració d'un diagnòstic es pot expressar com l'establiment inicial d'un nombre limitat d'hipòtesis. Aquesta limitació del camp diagnòstic és un tret essencial de l'expert. Diversos treballs, entre els quals destaca [Kassirer i Gorry, 1978], han demostrat que l'expert en un determinat domini clínic genera hipòtesis de treball més aviat i en menor nombre que el metge no expert.

L'estratègia d'agrupament correspon a una altre heurística relacionada amb

l'establiment d'hipòtesis: es reuniran diverses hipòtesis en una més general i no es prendran en consideració per separat fins que no sigui necessari. D'aquesta manera l'expert no limita excessivament el camp inicial de les hipòtesis. En agrupar-les en una altra més genèrica treballa amb elles sense que calgui especificar-les ni rebutjar-les. Observi's un exemple d'agrupament en la següent figura:

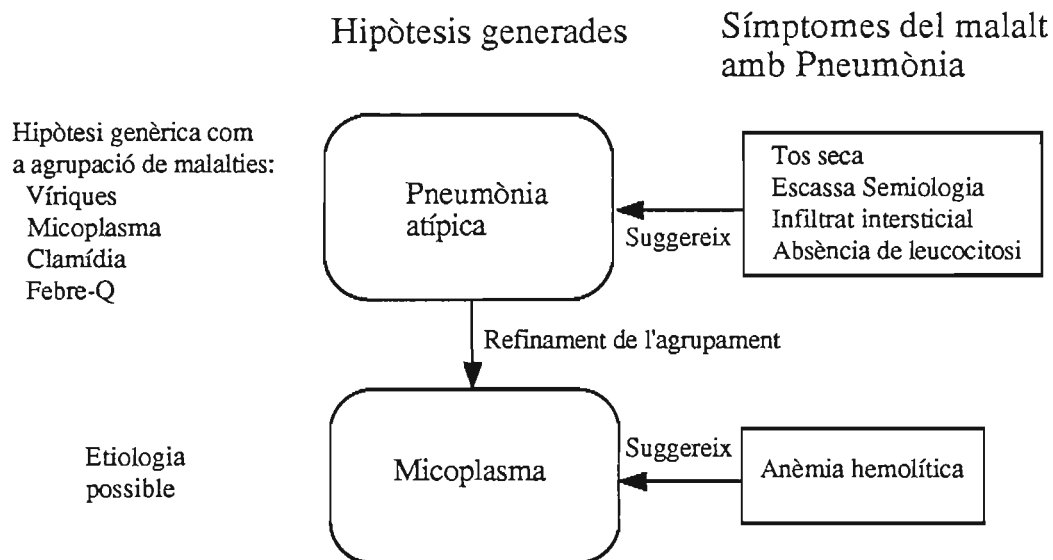


Fig. 2.1 Exemple d'heurística d'agrupament a l'aplicació PNEUMON-IA.

Aquests agrupaments heurístics són el que Clancey anomena classes de malalties dins el procés d'acarament heurístic.

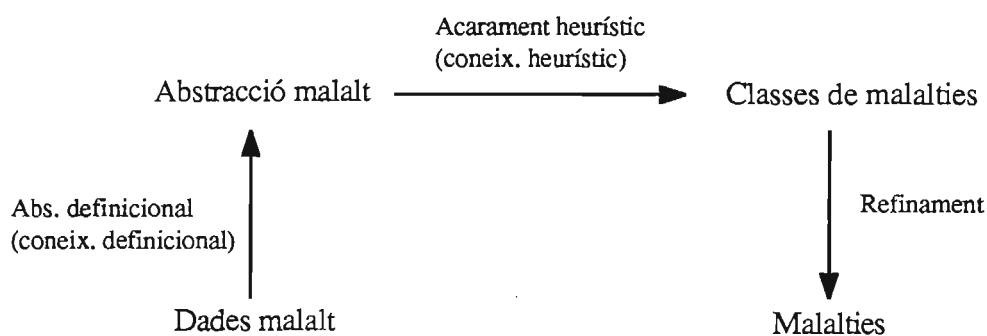


Fig. 2.2 Acarament heurístic en el model de Clancey.

Per ajudar als processos de raonament, es poden crear prototipus que agrupin fets que acostumen a associar-se en un determinat contexte. De la observació de signes i símptomes que es presenten agrupats en la clínica surgeix la síndrome. Les síndromes



es poden considerar com a prototipus de diverses situacions clíniques. Aquestes síndromes poden correspondre a una malaltia concreta o bé a diferents malalties. Quan s'identifica una única malaltia que dona lloc a la síndrome, en ocasions, i per raons d'hàbit, es conserva el nom de síndrome. Si són diferents les malalties identificades que causen una síndrome, és útil, des d'un punt de vista clínic, mantenir aquest concepte ja que permet d'una manera eficaç limitar el nombre d'hipòtesis generades.

Una síndrome també pot definir un conjunt de malalties d'origen comú. Així, es defineix la Síndrome de la Immunodeficiència Adquirida (S. I. D. A.), que inicialment agrupava una sèrie de malalties infeccioses o neoplàsiques<sup>1</sup> que afectaven a determinats grups de població. Quan s'identifica el virus de la immunodeficiència humana, roman el concepte de síndrome no ja des d'un punt de vista heurístic sino patogenètic o causal. En aquest punt la síndrome es correspon a un model fisiopatològic.

#### 2.2.2.2 Coneixements Estadístics

Els coneixements de tipus estadístic estan basats en la freqüència dels fets observats. L'aplicació del coneixement estadístic en un domini concret permet estudiar numèricament amb el màxim de precisió els fenòmens no completament coneguts d'aquell domini. L'aplicació exclusiva del coneixement estadístic implica un coneixement exhaustiu de les freqüències d'aparició dels fets observats en el domini en qüestió.

Per exemple un coneixement estadístic és el següent: el 80% de les pneumònies pneumocòciques produeixen esputs rovellats.

### 2.3 Elements de representació

La representació de coneixements de domini dins l'arquitectura de MILORD es situa en el marc dels sistemes basats en regles.

Les unitats bàsiques de representació són els fets, proposicions del càlcul proposicional o bé variables que poden posseir un conjunt finit de valors, conformant el que s'anomena lògica d'ordre 0+.

Les unitats inferencials són les regles, la lògica sobre la qual es basen és explicada al capítol 3. La seva característica més rellevant és l'ús de termes lingüístics per modelitzar la incertesa.

---

<sup>1</sup> Càncer.

Es defineixen conjunts de regles, que reben el nom de *mòduls* i que són les unitats estructurals de l'arquitectura.

Hi ha dos elements de representació del control: les estratègies i els plans d'actuació, que per la seva importància des del punt de vista del control han estat tractats apart en el capítol cinquè.

Es defineix un compilador que transformi una BC escrita en la sintaxi externa de representació a un conjunt d'objectes semàntics, llenguatge objecte, que són posteriorment interpretats. L'últim apartat d'aquest capítol descriu aquest procés de compilació en detall, destacant-ne els avantatges referents a l'eficiència en temps d'execució.

En els següents apartats farem una descripció dels elements a dalt esmentats. A l'apèndix II donarem la sintaxi i la semàntica denotacional de cada element de representació. Semàntica que servirà per entendre millor el procés de compilació explicat al final del present capítol.

### 2.3.1 Fets

Els fets modelitzen la conceptualització d'un domini d'expertesa concret. Són les unitats elementals a partir de les quals es contrueix la resta de l'arquitectura de MILORD. Representen les hipòtesis a validar, les dades observables, etc. Les premisses de les regles estan construïdes a partir de fets. Determinats fets possiran valor de certesa, que en el cas de la arquitectura de MILORD seran termes lingüístics, d'altres fets prendran valors numèrics, i d'altres prendran valors definits pels usuaris.

Els termes lingüístics que representen els valors de certesa són definits per l'expert en cada aplicació; d'aquí en endavant ens referirem al conjunt d'aquests termes amb el símbol *L*.

Els fets poden ser de dos tipus diferents:

*Predefinits pel sistema:*

**Booleans** : Són aquells fets el valor dels quals és *Cert*, *Fals* o *desconegut*<sup>1</sup>. Representen conceptes sobre els quals ens interessa la seva presència o absència en el

---

<sup>1</sup>Aquest valor és comú a tots els tipus de fets. Cal una petita aclaració en el seu ús per part dels motors d'inferència. Quan un fet no s'ha intentat validar el valor que posseeix, que és el valor que tenen tots els fets a l'iniciar una consulta, és *nil*. Ara bé, si un motor endavant necessita el valor d'un fet amb valor *nil* utilitzarà com a valor *desconegut*. Si un motor enrera el necessita, haurà d'engegar el procés d'obtenció del seu valor. Feta aquesta puntualització al llarg del text utilitzarem, quan no pugui causar confusió, el valor *desconegut* amb el benentès que aquest valor n'agrupa dos: *nil* i *desconegut*.

cas en curs. Un fet booleà seria per exemple tenir la *SIDA*.

**Difusos** : El valor és un d'entre el conjunt L de termes lingüístics definits per l'expert de l'aplicació o bé *desconegut*. Representen conceptes de difícil apreciació: bacterianicitat d'una malaltia, conceptes que presenten una graduació: *ser vell*, o bé conceptes de valoració subjectiva: *intensitat d'un dolor*. L'arquitectura de MILORD ha estat inspirada fonamentalment en la modelització d'aquest tipus de conceptes incerts. El capítol 3 es destina exclusivament al tractament de la incertesa.

**Numèrics** : El valor és un nombre real o bé *desconegut*. Correspon a dades quantitatives: nombre de leucòcits, nombre de processadors en un xip, etc.

**Classes** : El valor d'aquests fets és *buit*, *nobuit*, o *desconegut*. Altres fets mantenen relacions inclusives amb ells. El valor d'una classe serà *nobuit* si i només si algun dels fets relacionats inclusivament amb ella és present.

#### *Definits per l'expert*

**Enumerats** : L'expert té la possibilitat de definir fets per enumeració dels seus valors simbòlics. El valor d'un fet enumerat serà un subconjunt no buit del conjunt de valors definits, o bé *desconegut*.

### 2.3.1.1 Sintaxi de la creació de símbols

La definició d'un domini conceptual es realitza creant entrades a un *diccionari*. Cada entrada al diccionari consta d'un conjunt predeterminat de característiques del concepte. La sintaxi és la següent<sup>1</sup>:

<i>diccionari</i>	::= <b>Diccionari</b> <i>entrada rel-inclusives fidiccionari</i>
<i>entrada</i>	::= <i>fet entrada</i>   <i>fet</i>
<i>fet</i>	::= <b>fet</b> <i>fet-id atributs</i>
<i>atributs</i>	::= <i>nom pregunta tipus relació</i>   <i>nom pregunta tipus</i>

<sup>1</sup>Els símbols no terminals que no tenen regles de reescriptura es reescriuen en símbols des del punt de vista de LISP.

<i>nom</i>	::= <b>Nom</b> : cadena-caràcters
<i>pregunta</i>	::= <b>Pregunta</b> : cadena-caràcters
<i>tipus</i>	::= <b>Tipus</b> : def-tipus
<i>def-tipus</i>	::= <i>nomt</i>   <i>nomt</i> = <i>valors</i>   <i>valors</i>
<i>nomt</i>	::= <i>tipusid</i> <sup>1</sup>
<i>valors</i>	::= <i>valor valors</i>   <i>valor</i>
<i>relació</i>	::= <b>Relació</b> : <i>relació-id fet-id funció</i> <sup>2</sup>   <b>Relació</b> : <i>relació-id fet-id</i> <i>relació relació</i>
<i>rel-inclusives</i>	::= { Definit a 2.3.4 }

Aquestes característiques fan referència a diferents usos del concepte per part del sistema:

*fet-id* és la representació del concepte utilitzada en l'escriptura de la BC: regles, metaregles, etc. Acostuma a ser un nom mnemotècnic propi de l'expert i de vegades difícilment comprensible pels usuaris. Per exemple ADVP representa "Adictes a Drogues per Via Parenteral", a l'aplicació PNEUMON-IA.

*nom* és la representació utilitzada amb els interlocutors externs en processos com l'explicació del raonament, la presentació de resultats, la generació d'informes, etc.

*pregunta* representa la forma més correcta i menys ambigua, en opinió de l'expert, de preguntar el valor associat al concepte.

*tipus* determina el conjunt de possibles valors que poden ser associats a un fet determinat. Es permet la definició de tipus, i l'assignació d'un nom al tipus creat, de manera que es pugui referenciar el tipus més endavant. Es permet definir la llista de possibles valors (fets enumerats) sense donar nom en els casos en que no es vulgui crear un nom de tipus. Tanmateix es permet utilitzar els tipus predefinits.

*relació* defineix la relació funcional que existeix entre un concepte i la resta de conceptes de la BC. Les funcions són expressions lògiques sobre els conceptes del domini, i l'avaluació de les mateixes dona com a resultat un terme lingüístic que determina la intensitat de la relació. En cas que la relació existeixi independentment del context no es defineix cap expressió lògica associada. Un exemple de diccionari seria el que mostra la figura 2.3, la definició del qual és la següent:

<sup>1</sup>Els identificadors predefinits són booleà, difús, numèric i classe.

<sup>2</sup> Per una explicació detallada veure l'apartat de Grafs Dirigits Funcionals.

**Diccionari****Fet Dol-ple**

Nom: "Dolor pleurític"

Pregunta: "Pateix el malalt de dolor pleurític?"

Tipus: Booleà

Relació: pertany simp

**Fet Inici**

Nom: "Tipus d'inici"

Pregunta: "Com va ser l'inici: brusc o subagut?"

Tipus: Ini = brusc, subagut

Relació: pertany simp

**Fet D-esq**

Nom: "Desviació esquerra"

Pregunta: "Presenta el malalt desviació a l'esquerra?"

Tipus: Booleà

Relació: pertany lab

**Fet Leucocitosi**

Nom: "Leucocitosi"

Pregunta: "El nombre de leucòcits està per sobre de 10000?"

Tipus: Booleà

Relació: pertany lab

**Fet OH**

Nom: "Alcohòlic"

Pregunta: "El malalt, és alcohòlic?"

Tipus: Booleà

Relació: pertany ap-int

**Fet SIDA**

Nom: "Síndrome d'immunodeficiència adquirida"

Pregunta: "El malalt, té SIDA?"

Tipus: Booleà

Relació: pertany ap-int

**Fet Broncoaspiració**

Nom: "Broncoaspiració"

Pregunta: "Ha sofert el malalt una broncoaspiració?"

Tipus: Booleà

Relació: pertany ap-int

**Fet Pneum**

Nom: "Diagnòstic per Pneumococ"

Pregunta: ""

Tipus: Difús

Relació: pertany etio  
pertany diags**Fet Legio**

Nom: "Diagnòstic per Legionella"

Pregunta: ""

Tipus: Difús

Relació: pertany etio  
pertany diags**Fet Bact**

Nom: "Pneumònia Bacteriana"

Pregunta: ""

Tipus: Difús

Relació: pertany síndrome

**Fet Atip**

Nom: "Pneumònia Atípica"

Pregunta: ""

Tipus: Difús

Relació: pertany síndrome

**Fet Radiol**

Nom: "Dades Radiològiques"

Pregunta: "Es disposa de dades radiològiques?"

Tipus: Classe

Relació: inclòs exp-comp

**Fet Simp**

Nom: "Dades simptomatològiques"

Pregunta: "Posseïm dades simptomatològiques?"

Tipus: Classe

Relació: inclòs clín

**Fet Lab**

Nom: "Dades de laboratori"

Pregunta: "Es disposa de dades de laboratori?"

Tipus: Classe

Relació: inclòs exp-comp

**Fet Ap-int**

Nom: "Antecedents patològics d'interès"

Pregunta: "Té el pacient antecedents patològics d'interès?"

Tipus: Classe

Relació: inclòs factors

**Fet Clin**

Nom: "Dades clíniques"

Pregunta: ""

Tipus: Classe

Relació: inclòs malalt

**Fet Exp-comp**

Nom: "Dades d'exploracions complementàries"

Pregunta: ""

Tipus: Classe

Relació: inclòs malalt

**Fet Etiol**

Nom: "Etiologies"

Pregunta: ""

Tipus: Classe

Relació: inclòs malalt

**Fet Síndrome**

Nom: "Síndrome"

Pregunta: ""

Tipus: Classe

Relació: inclòs diags

**Fet Factors**

Nom: "Factors de risc"

Pregunta: "Es tracta d'un malalt amb algun factor de risc?"

Tipus: Classe

Relació: inclòs clín

**Fet Diags**

Nom: "Diagnòstics"

Pregunta: ""

Tipus: Classe

Relació: inclòs malalt

**Fet Malalt**

Nom: "Malalt"

Pregunta: ""

Tipus: Classe

fidiccionari

Hi ha un aspecte a remarcar sobre els grafs de conceptes definits en els diccionaris i la forma d'obtenir les dades. És prou coneguda la utilització en les premisses de fets que tenen com a única missió servir de guardes que controlin l'avaluació (obtenció de valor) d'altres premisses. Això té com a conseqüència:

- 1.- Aument de l'extensió de les regles.
- 2.- Disminució de l'explicabilitat de les regles.
- 3.- Barreja de coneixements de control i de domini amb la conseqüent dificultat de depuració.

Dins MILORD, i en la línia dels sistemes experts de segona generació, hem volgut donar eines per evitar l'ús d'aquestes guardes implícites fent que el mecanisme d'adquisició de les dades utilitzi el coneixement explícitament representat en els grafs de conceptes. Substituint la utilització implícita de coneixements de control per la declaració explícita de les interrelacions dels conceptes s'eviten els problemes que els sistemes de primera generació han mostrat.

L'algorisme d'adquisició de dades és l'expressat a la figura 2.4. Aquest algorisme fa referència a les condicions en les quals es pot realitzar la pregunta associada a un fet.

```

Funció demanar(Fet)
cas tipus(Fet)
  Classe: si pregunta(Fet) = "" o valor_fet(Fet) = nobuit o
          valor_fet(Fet) = buit
          llavors retorna(fracàs)
        fsi
        si  $\exists x \in \text{pares}(\text{Fet}) : \text{demanar}(x) = \text{èxit}$  o  $\text{pares}(\text{Fet}) = \emptyset$ 
          llavors
            preguntar(Fet)
            retorna(èxit)
          sinó retorna(fracàs)
        fsi
  Altrament
    si valor_fet(Fet)  $\neq$  nil
      llavors retorna(fracàs)
    fsi
    si  $\exists x \in \text{pares}(\text{Fet}) : \text{demanar}(x) = \text{èxit}$  llavors
      preguntar(Fet)
      retorna(èxit)
    sinó retorna(fracàs)
    fsi
  ficas
fifunció

```

Fig 2.4 Utilització de la xarxa conceptual per adquisició de dades. La funció retorna èxit si efectivament pregunta el fet, i fracàs en cas contrari.

Com a exemple d'utilització, vegi's la figura 2.5.

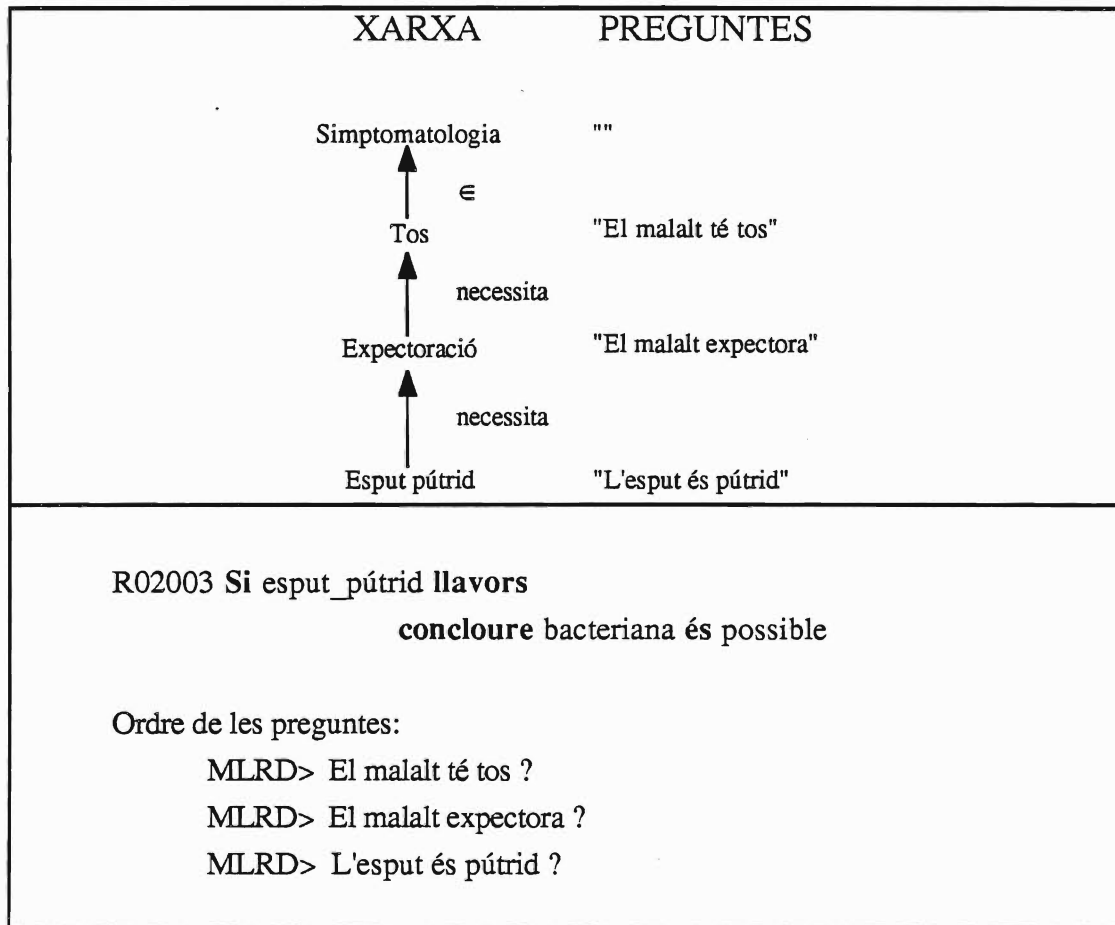


Fig 2.5 Utilització de la xarxa conceptual en l'avaluació de regles.

Com es constata a l'exemple anterior l'expert es troba molt lliure dels problemes de control a l'hora d'escriure les seves regles.

En el cas que el fet que aparegui en una premissa es trobi negat, com per exemple:

R03005 Si ¬esput\_pútrid llavors abaixar anaerobis 1

també s'ha de procedir de la mateixa manera que en el cas anterior. El fet ¬esput\_pútrid només té sentit en el contexte definit pels fets expectoració i tos. De fet la regla r03005, de manera similar a la regla r02003, és una forma equivalent de la regla:

R03005 Si tos i expectoració i ¬esput\_pútrid  
 llavors abaixar anaerobis 1

### 2.3.1.2 Predicació sobre els fets

Els fets són les unitats a partir de les quals es construeixen les regles d'inferència. Apareixen a les premisses de les regles i a les conclusions. A les premisses el que interessa és establir condicions sobre el valor dels fets i a les conclusions operadors que els modifiquin. En aquest apartat descriurem quins predicats existeixen a MILORD, en el següent apartat, referent a les regles d'inferència, descriurem la seva semàntica en el contexte de l'avaluació de les regles.

#### 2.3.1.2.1 Predicació a les condicions

Sintaxi de la definició de condicions sobre els fets<sup>1</sup>:

```

condició ::= no (condició) |
              no-dem (condició) |
              fet-id |
              fet-id és terme |
              fet-id = valors |
              càlcul operador-rel càlcul

operador-rel ::= < | > | ≥ | ≤ | = | /=
valors ::= valor o valors | valor
càlcul ::= valornumèric | fet-id | cardinal(fet-id) |
              ( càlcul operador-arit càlcul ) |
              ( fet-id := càlcul )

operador-arit ::= + | - | * | /
  
```

Els predicats utilitzats per construir condicions són de dos tipus, els que es poden aplicar depenent dels tipus dels fets i els que no.

<sup>1</sup> Allà on no hi hagi ambigüitat es podran suprimir els parèntesis.



Dividirem l'anàlisi dels tipus de condicions depenents dels fets segons els tipus dels fets als quals poden aplicar-se.

### 1.- Fets booleans

*fet-id* : L'aparició del fet aïlladament s'avalua en el valor de certesa booleà (cert, fals, desconegut) associat. Donat que el tipus de lògica en que ens movem és una lògica multivalent amb termes lingüístics, els valors *cert* i *fals* s'assimilen als extrems del conjunt ordenat de certeses. Un exemple seria:

**M01003 Si SIDA Llavors mòduls a visitar P-carinii**

SIDA és un fet booleà. Si fos cert el valor de la condició SIDA seria el més alt de l'escala de termes, per exemple a PNEUMON-IA seria:*segur*.

### 2.- Fets difusos

*fet-id*: El valor de certesa serà el valor associat al fet difús.

*fet-id és terme* : El valor serà *cert* si el terme lingüístic associat a *fet-id* es més gran o igual que *terme*. Més gran o igual s'ha d'entendre segons l'ordre definit sobre els termes dins el conjunt L. Per exemple:

**R04035 Si pneumococ és moderadament\_possible i esput = cgpp llavors  
apujar pneumococ 1**

### 3.- Fets classe

Els fets de tipus classe reben un tractament idèntic als fets de tipus booleà, interpretant els valors del fet de tipus classe quan són avaluats a la premissa d'una regla de la següent manera:

Valor	Avaluació
buit	fals
no buit	cert
desconegut	desconegut

Així la classe *lab* agrupa les dades de laboratori. Si la classe *lab* fos buida

activaria la següent metaregla, que inhibeix totes les regles que utilitzen a les seves premisses fets pertanyents a la classe *lab* :

**R25606 si no lab llavors inhibir regles pertany lab**

#### 4.- Fets de tipus definits per enumeració

*fet-id = valors*: El valor d'aquest predicat serà cert si el conjunt de valors de *fet-id* (un fet d'un tipus definit per l'expert pot posseir com a valor un subconjunt dels valors permesos) té una intersecció no buida amb la llista de *valors*. És una forma de disjunció permesa; si *valors* fos la llista  $v_1, v_2, \dots, v_n$  aleshores el predicat seria equivalent a:

$$fet-id = v_1 \text{ o } fet-id = v_2 \text{ o } \dots \text{ o } fet-id = v_n$$

Per exemple:

R05001 Si malalt = greu o molt\_greu i Antecedents\_interés = fumador o  
alcohòlic o malaltia\_resp\_crònica o cardiopatia i edat > 50 i  
sexe = home  
**llavors concloure legionella és moderadament\_possible**

Sobre els fets de tipus enumerat es defineix la funció **cardinal**, que dona el nombre d'elements que posseeix el conjunt valor del fet. Això és important quan s'han de prendre decisions a partir del nombre d'elements d'una certa característica que apareixen en un cas. Aquest fenomen es dona freqüentment a l'aplicació RENOIR pel diagnòstic de malalties reumàtiques, on certes decisions diagnòstiques es prenen en funció de la quantitat de símptomes presents. Per exemple a RENOIR és important saber quantes articulacions estan afectades. Això es pot modelitzar amb un fet enumerat sobre les articulacions, i la utilització de la funció cardinal sobre aquest fet.

Així doncs, la funció cardinal:  $fet-id \rightarrow N$  pot ser utilitzada en els càlculs numèrics que s'expliquen a continuació.

#### 5.- Fets numèrics

*càlcul operador-rel càlcul*: Tipus de predicació que permet comparar el valor dels fets, o bé de càlculs realitzats amb fets numèrics. Al mateix temps, i com a efecte lateral,

es permeten realitzar assignacions de càlculs o subcàlculs. Exemples són (la sintaxi s'ha adaptat per fer-ho més llegible):

febre  $\geq$  39.5

nombre\_de\_processadors = 1

R01089

Si (Polinuclears :=  $\left(\frac{(\text{Leucòcits} * \text{Polinucl}\%)}{100}\right) < 500$ )

**Llavors**

**Concloure** Neutropènia és segur

En aquesta regla, i com a efecte lateral, el valor de polinuclears és modificat i li és assignat el valor de multiplicar el tant per cent de polinuclears per la xifra de leucòcits del pacient.

Vegi's l'ús de la funció cardinal en la definició dels següents coneixements definicionals a l'aplicació RENOIR:

R02003

Si **cardinal**(artic\_afectades)  $\geq$  4

**Llavors**

**Concloure** poliartritis és segur

R02004

Si **cardinal**(artic\_afectades)  $>$  1 i

**cardinal**(artic\_afectades)  $<$  4

**Llavors**

**Concloure** oligoartritis és segur

Veiem ara els dos operadors aplicables a qualsevol tipus de condició: l'operador **no** i l'operador **no-dem**.

**no condició** : El valor de certesa d'aquest predicat és la negació del valor de certesa de *condició*. La negació es defineix com l'oposat en l'escala de termes definida.

<i>condició</i> <sup>1</sup>	<b>no condició</b>
$X_1$	$X_n$
$X_2$	$X_{n-1}$
...	
$X_n$	$X_1$
<i>desconegut</i>	<i>desconegut</i>

On  $L = \{X_1, X_2, \dots, X_n\}$ , llista ordenada de termes lingüístics.

Per exemple si el valor del fet enumerat *dades\_interés* fos { Transplantament, Alcohòlic } el valor de la següent condició:

**no dades\_interés = SIDA o Alcohòlic**

seria **fals**, es a dir  $X_1$

**no-dem condició** : El valor de certesa d'aquest predicat ve determinat per la següent taula, en funció del valor resultant d'avaluar *condició*.

<i>condició</i>	<b>no-dem condició</b>	
$X_1$	$X_n$	
$X_i$	$X_1$	$\forall i : n \geq i > 1$
<i>desconegut</i>	$X_n$	

On  $L = \{X_1, X_2, \dots, X_n\}$ , llista de termes lingüístics.

Aquest predicat és útil per esbrinar si una determinada condició és demostrable o no. Serà *cert* en el cas que la condició hagi rebut el mínim valor de certesa, o en el cas que el valor sigui desconegut. És un metapredicat, en el sentit que predica sobre el mateix funcionament del sistema. Està en la línia dels predicats d'introspecció [Maes, 1988].

---

<sup>1</sup>Aquesta llista conté els valors resultants d'avaluar *condició*. Tanmateix passa a la llista d'explicació del predicat **no-dem**.

La seva utilitat queda exemplificada en els següents exemples:

Exemple 1: Els malalts que no posseeixen antecedents significatius per sospitar una pneumònia pneumocòcica, com ara ser immunodeprimits, drogadictes, etc. tenen una aproximació diagnòstica diferent. Així si no podem demostrar que el malalt tingui aquests antecedents les assignacions d'evidències seran diferents, vegi's:

R04007

**Si no-dem** antec\_sig\_pneumococ i  
bacteriana

**Llavors**

**concloure** pneumococ és possible

Exemple 2: Tanmateix, Aquest predicat ens pot informar sobre una possibilitat diagnòstica que no ha tingut èxit, o bé que la estratègia no ha considerat. Així podem utilitzar aquest predicat per definir noves estratègies a partir de les prèvies. Per exemple si tenim una evidència alta de bacterianicitat i no hem demostrat res sobre atipicitat (o la estratègia anterior no l'ha considerat), llavors construirem una estratègia sobre la base dels agents bacterians:

M25884

**Si bacteriana és força\_possible i**  
**no-dem** atípica

**Llavors**

**mòduls a visitar**

pneumococ legionella enterobactèries anaerobis  
hemòfilus estafilococ virus estreptococ\_piògen  
meningococ branhamella tuberculosi pseudomones  
aspergillus nocàrdia

### 2.3.1.2.2 Predicació a les conclusions

A les regles de producció es permeten tres tipus d'accions sobre els fets a les conclusions: *concloure*, *apujar* i *abaixar*. Els tres corresponen a tipus diferents d'associacions entre fets com veurem a continuació:

*conclusió* ::= **concloure** *fet-id* és  $\beta$  |  
**apujar** *fet-id* *n* |  
**abaixar** *fet-id* *n*

**Concloure** *fet-id* és *terme*: Aquest predicat estableix una relació entre els fets de la premissa i de la conclusió tal i com està explicada al capítol 3. La més gran evidència de les premisses fa que el valor propagat a la conclusió sigui més gran. L'aplicació de diferents regles amb el predicat **concloure** fa que els diferents valors propagats es combinin com a disjuncions per donar el valor final al fet conclós.

**Apujar** *fet-id* *n* : Aquesta acció incrementa el valor de certesa de *fet-id* en *n* valors (seguint l'ordenació del conjunt de termes *L*). El comportament d'aquest predicat és útil quan es vol incrementar el valor d'un fet independentment del valor que en aquell moment poseís. En la combinació utilitzant els connectius **o**, la resposta a una aportació d'evidència d'un predicat **concloure**, no és lineal respecte al valor propagat. Per tant en casos en que el comportament de l'increment del valor hagi de ser lineal serà necessari utilitzar aquest predicat. Els motors d'inferències de MILORD apliquen aquestes regles només quan cap de les regles amb predicat **concloure** aplicables restin per aplicar-se.

**Abaixar** *fet-id* *n* : Aquesta acció té un comportament semblant a l'anterior. En aquest cas el que es fa és decrementar el valor de certesa en *n* valors. Donat que els connectius són monòtonament creixents, no hi ha cap mecanisme de fer decreixer el grau d'un determinat fet mitjançant predicats **concloure**. Moltes vegades les aproximacions a la solució tenen diferents estadis en l'obtenció de dades. Per exemple a les primeres hores determinades dades no poden ser utilitzades: hemocultius, serologies i dades evolutives (en el cas de PNEUMON-IA), i malgrat això es vol donar una aproximació diagnòstica, tenint en compte les dades utilitzables inicialment, i posteriorment, a la vista de la resta de les dades, afinar i rectificar l'anàlisi diagnòstica inicial. Aquesta aproximació, utilitzant dues fases en el procés classificatori, que no és incompatible amb la implementació de mecanismes de manteniment de la consistència, simplifica notablement el procés de construcció de regles al mateix temps que l'aproxima a la pràctica habitual en certs dominis. Evidentment aquells dominis on no hi ha una marcada seqüencialització en les dades disponibles no tenen necessitat d'utilitzar

els predicats d'**apujar** i **abaixar**.

Vegi's els següents exemples de regles utilitzant els predicats apujar i abaixar:

R04035 Si pneumococ és moderadament\_possible i esput = cgpp llavors  
**apujar** pneumococ 1

R04037 Si pneumococ i cavitació llavors **abaixar** pneumococ 2

### 2.3.2 Regles d'inferència

La utilització de regles com a mecanisme de representació dels coneixements és comú a molts entorns de desenvolupament de sistemes experts. L'estructura que posseeixen les regles en l'entorn MILORD és la següent:

<i>lregles</i>	::= <i>regla lregles   regla</i>
<i>regla</i>	::= <i>idregla Si premissa Llavors conclusió</i> <i>documentació</i>
<i>premissa</i>	::= <i>premissa i condició   condició</i>
<i>documentació</i>	::= <i>cadena-caràcters</i>

Les premisses de les regles de MILORD són conjuncions de condicions. L'avaluació de les diferents condicions es realitza d'esquerra a dreta en profunditat, de manera que aquest ordre determina la forma en que es realitzaran les preguntes. Aquesta avaluació s'atura quan la combinació d'evidències de les condicions no pot superar un llindar establert en el mòdul on està definida la regla. Dins el projecte SPES que es desenvoluparà al llarg dels propers tres anys al Centre d'Estudis Avançats de Blanes es modificarà aquest aspecte seqüencial de cara a obtenir paral·lelisme en l'avaluació de les premisses de les regles.

Les regles poden portar un camp de documentació que pot ser utilitzat per proporcionar l'explicació més profunda de la regla o bé per donar la referència bibliogràfica en cas que la regla no fos pròpia de l'expert. Per exemple:

R08007 Si no neutropènia i associades-dermatologia = ectima-gangrenosum

**Llavors concloure pseudomonas és molt possible**

"Molavi A, LeFrock JL. Enterobacteriaceae, Pseudomonas aeruginosa and acinetobacter. A Levison ME, ed. Boston, John Wright. PSG Inc.1984; 309-334"

Les regles posseeixen un únic identificador. Certes metaregles poden referir-se a aquest identificador per inhibir la regla associada. Una regla pot estar inhibida o activada. Inhibida implica que no podrà ser utilitzada en els processos deductius.

Les regles són aplicades una única vegada. i el seu règim és irrevocable, és a dir, un cop aplicada no es poden eliminar els seus efectes.

Un fet pot ser deduït per més d'una regla. En un moment determinat en que es vol avaluar un fet amb encadenament enrera es crea el conjunt conflicte, que és el conjunt de totes les regles utilitzables en un moment donat, és a dir les regles del mòdul sobre el que el sistema es focalitza (veure el següent apartat) i que no han estat ni aplicades anteriorment ni inhibides per cap metaregla. Un cop determinat el conjunt conflicte es determina l'ordre d'aplicació. L'ordre en el que s'aplicaran les regles per deduir el fet ve determinat per tres criteris que s'apliquen en l'ordre següent

- 1.- *Subsumció*: Les regles més específiques primer. (veure 3.3).
- 2.- *Certesa*: Les regles amb un grau de certesa més alt primer.
- 3.- *Ordre de l'expert*: L'ordre en que l'expert ha escrit les seves regles.

Aquests tres criteris defineixen un ordre total de les regles que és l'ordre que el sistema utilitza per avaluar-les. Quan una regla és aplicada l'ordre es modifica atès que totes les regles més generals que la regla aplicada han de sortir del conjunt conflicte.

Quan l'encadenament és endavant el mecanisme és exactament el mateix, excepte que el conjunt conflicte inicial coincideix amb totes les regles no aplicades o inhibides del mòdul de focalització.

### 2.3.3 Representació modular dels coneixements

Un mòdul és un agrupament d'elements de representació amb un objectiu concret. Generalment són agrupaments relacionats amb una jerarquia de classificació. Així cada mòdul correspondria a un nus en la jerarquia. La possibilitat de definir agrupaments es troba a la base dels mètodes de raonament utilitzats als sistemes experts. Els sistemes de primera generació generalment no permetien cap mena d'agrupament.



Consistien en un gran sac de regles sobre les quals s'aplicaven els algorismes d'encadenament. Quan la grandària de les aplicacions va anar augmentant es va veure imprescindible la possibilitat de compartimentar, i estructurar els coneixements i la seva representació: regles, metaregles, fets.

Els mòduls a MILORD han tingut dues fases de desenvolupament. Inicialment varem treballar sobre mòduls sense estructura entre ells; els mòduls eren agrupaments d'elements des d'un punt de vista de simplificació dels conjunts conflicte dels motors d'inferència i com a mitjà per a facilitar la manipulació i l'escriptura de les BC. Posteriorment varem redefinir aquests mòduls en un contexte d'especificació formal, on el mòdul passava a ser un objecte de representació més amb possibilitats de ser combinat i manipulat per altres mòduls.

Presentarem a continuació la sintaxi dels mòduls no estructurats de MILORD, i a l'apèndix II donarem la semàntica associada al procés de compilació. A més, donarem la semàntica associada a l'execució d'un mòdul.

<i>Nivell_estructural</i>	::= <b>Nivell Estructural</b> <i>unitat-estructural</i> <i>control-estructural</i>
	<b>finivell</b>
<i>unitat-estructural</i>	::= <b>Mòdul</b> <i>mòdul-id</i> <i>objectius regles metaregles</i>   <i>unitat-estructural unitat-estructural</i>
<i>objectius</i>	::= <b>Objectius:</b> <i>lobjectius</i>
<i>lobjectius</i>	::= <i>fet-id, lobjectius</i>   <i>fet-id</i>
<i>regles</i>	::= <b>Regles:</b> <i>lregles</i>
<i>lregles</i>	::= <i>regla lregles</i>   <i>regla</i>
<i>metaregles</i>	::= <b>Metaregles:</b> <i>lmrr</i>
<i>lmrr</i>	::= <i>mrr lmetaregles</i>   $\lambda$
<i>control-estructural</i>	::= { descrit al capítol 5 }

Un mòdul no estructurat, com es pot comprovar a la sintaxi, consta de tres elements fonamentals:

*Objectius:* Aquells fets que són deduïts en aquell mòdul i el valor dels quals interessa per les avaluacions de fets en altres mòduls.

*Regles:* Conjunt de regles agrupades al mòdul, que dedueixen els objectius marcats.

*Metaregles:* Conjunt de metaregles que controlen la inhibició i activació de les

regles dins el mòdul.

Quan l'avaluació d'un mòdul és amb encadenament enrera, es seleccionen els objectius d'esquerra a dreta, i per cada objectiu s'obté el conjunt conflicte de regles, i s'avaluen aquestes segons s'ha explicat a l'apartat anterior. Quan els objectius s'han exhaurit, l'avaluació del mòdul acaba.

L'avaluació d'un mòdul en encadenament endavant utilitza totes les regles com a conjunt conflicte, i s'atura quan no resta cap regla aplicable al conjunt conflicte o bé tots els objectius han estat avaluats amb el valor màxim de certesa.

A cada pas d'avaluació, aplicació d'una regla, o obtenció d'informació de l'exterior, les metaregles del mòdul són consultades utilitzant encadenament endavant, per tal d'actualitzar el conjunt conflicte.

En el capítol 4 presentarem la sintàxi de definició de mòduls estructurats. Aquests mòduls estructurats, són compilats en mòduls no estructurats com els que hem presentat aquí. La raó d'aquest disseny és únicament històrica, inicialment MILORD posseïa mòduls no estructurats i posteriorment es van introduir amb la pretensió de no modificar l'arquitectura existent.

Aquesta estructuració dels mòduls neix com a necessitat d'introduir constructes que ajudin en el disseny de grans bases de coneixements. Permeten la comprovació de tipus i la reutilització segura i fàcil de parts de les aplicacions. Aquest aspecte que ha estat potenciat en els llenguatges orientats objecte no ha estat utilitzat en els sistemes basats en regles.

### **2.3.4 Graf orientat de relacions funcionals**

Els fets de l'univers de discurs mantenen entre ells relacions diverses: inclusió, pertinença, prioritat, etc. Aquestes relacions representen una part molt important dels coneixements de domini: els coneixements estructurals.

Aquestes relacions poden ser representades, entre altres possibilitats, mitjançant un graf orientat on els arcs siguin les relacions i els nodes els fets. Si be en molts casos les relacions entre els fets poden ser considerades estàtiques, en d'altres apareix la necessitat de que siguin dinàmiques: relacions existents únicament en un determinat contexte. Per tal de poder representar aquestes situacions hem optat per la representació de les relacions en forma funcional.

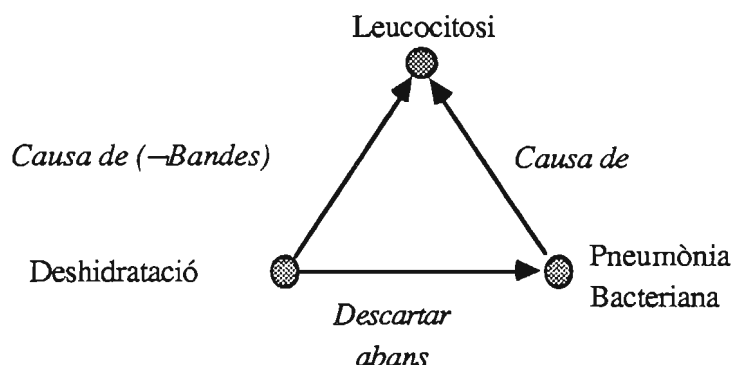


Fig 2.6 Xarxa de coneixements de control amb un GRF.

Aquestes relacions no són necessàriament de caracter booleà, sinó que moltes vegades poden posseir una certa graduació: relacions fortes, dèbils, etc. Per tal d'encabir aquesta característica utilitzarem funcions que ens donin valors sobre el conjunt de termes lingüístics utilitzats a l'aplicació. Aquestes funcions, des de tots els punts de vista són equivalents a les premisses de les regles d'inferència. S'entenen aquí les premisses com a funcions del conjunt de fets al conjunt de termes lingüístics.

La definició dels grafos de relacions es realitzada pels experts i es fa al mateix temps que la definició de l'univers de discurs. La sintaxi ja ha estat presentada a l'apartat 2.3.1.

**Def** Anomenarem **Graf de relacions funcionals (GRF)** a un graf orientat acíclic  $G = (V, E)$  on:

**V:** Conjunt de vèrtex (fets).

**E:**  $(v, w, r, f)$  on  $v, w \in V$ ,  $r$  relació i

$f: V^n \rightarrow L \cup \{\text{desconegut}\}$

essent  $L$  el conjunt de termes lingüístics.

**Exemple:** Considerem la relació "més important" i els fets fumador i alcohòlic, en el contexte d'una pneumònia bacteriana. És sabut que en el cas de pneumònies bacterianes el fet de l'alcoholisme és més important que el fet de ser fumador. Així doncs un element de  $E$  fora:

$(\text{alcohòlic}, \text{fumador}, \text{més\_important}, \text{bacteriana})$

o bé si ens interessés una resposta bivaluada podríem escriure:

$(\text{alcohòlic}, \text{fumador}, \text{més\_important}, \text{bacteriana} \text{ és } \beta), \beta \in L.$

Segons sigui el conjunt de fets coneguts en un moment donat el sistema considera com a graf de relacions un subgraf denominat *Graf actiu*, gestionat pel sistema i

actualitzat per cada modificació dels fets coneguts.

Def Anomenarem Graf actiu  $G$  associat a un GRF  $F$  en un instant de temps  $t$  a el parell  $(V,E)$  on:

$V$ : Conjunt de vèrtex (fets).

$E$ :  $(v,w,r,l)$  on  $v,w \in V$ ,  $r$  relació i

$\exists (v,w,r,f) \in F : f(x) = 1 \notin \{ \text{desconegut, fals} \}$  en l'instant de temps  $t$ .

Vegi's l'exemple següent:

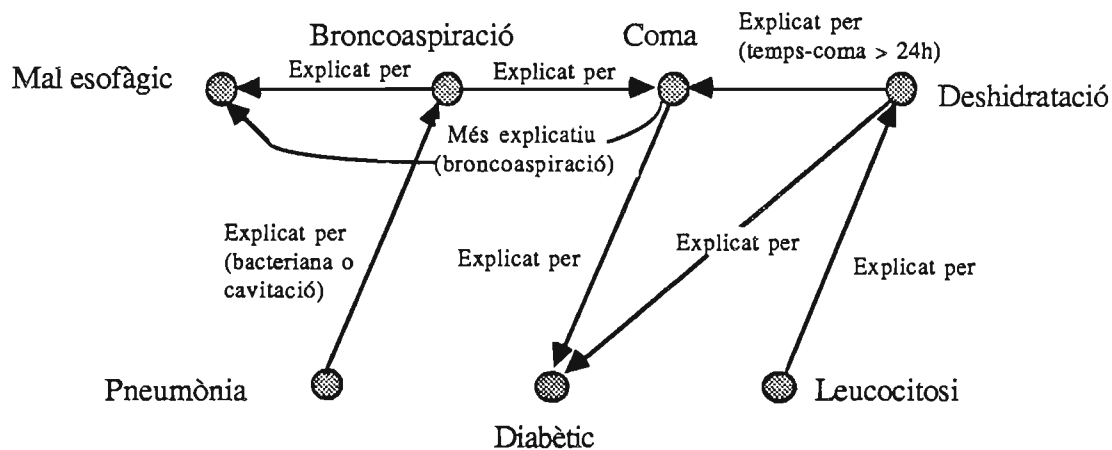


Fig 2.7 Xarxa Explicativa representada mitjançant un GRF.

Consideri's el GRF de la figura 2.7 i el cas d'un malalt amb una malaltia bacteriana amb un grau de certesa de força possible, sense estat de coma i amb una broncoaspiració. El graf actiu associat al representat a la figura 2.7 seria el representat a la figura 2.8.

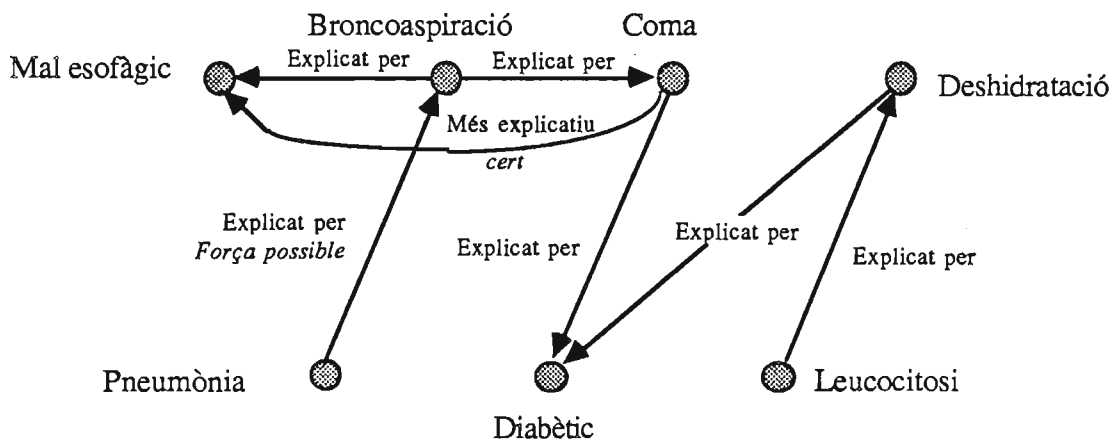


Fig. 2.8 Graf actiu associat al GRF de la figura 2.7

**Def** Donats dos grafs actius  $G = (V, E)$  i  $G' = (V', E')$  direm que  $G \subseteq G'$  si i només si

i.-  $V \subseteq V'$  i

ii.- Per a tot  $e=(v,w,r,l) \in E$  existeix  $e'=(v,w,r,l') \in E' : l \leq_L l'$   
on  $\leq_L$  és l'ordre definit sobre el conjunt de termes  $L$ .

D'entre totes les relacions que un expert pot utilitzar hi ha un subconjunt interessant per definir relacions de subsumpció. Aquestes relacions les anomenem inclusives i són les que donen una estructura jeràrquica a l'univers de discurs. Les relacions inclusives són definides per l'expert. No hem optat per donar un conjunt fixe de possibles relacions, per tal de fer més adaptat el lèxic a cada aplicació.

La definició d'aquest conjunt de relacions inclusives es realitza de la següent manera:

<i>rel-inclusives</i>	::= <b>Relacions inclusives</b> <i>llista-relacions</i>
	<b>firelacions</b>
<i>llista-relacions</i>	::= <i>relació-id</i> <i>llista-relacions</i>   $\lambda$

Per exemple a l'aplicació PNEUMON-IA les relacions inclusives són:

### **Relacions inclusives**

membre-de pertany-a inclós-a tipus-de  
**firelacions**

Hi ha una relació molt interessant entre els fets de cara a comprendre els algorismes d'avaluació de les regles i metaregles : l'*especificitat*. Amb ella podem establir relacions de preordre en els conjunts de fets i utilitzar aquests preordres en l'aplicació de les unitats inferencials. L'especificitat l'estudiarem inicialment des d'un punt de vista semàntic, després sintàctic, i finalment definirem la relació d'especificitat que subsumeix els punts de vista sintàctic i semàntic.

#### *Especificitat semàntica*

**Def** Donat  $G = (V, E)$  graf actiu, i  $F$  el conjunt de fets d'una aplicació ( $F$  finit), definim la relació  $E_G$  sobre  $F$  "*ser més específic semànticament*" com:

$$E_G: F \times F \rightarrow \{0, 1\}$$

$E_G(x, y) = 1 \Leftrightarrow$  existeix  $(x, y, r, l) \in E$  on  $r$  és una relació inclusiva.

**Propietat**

$$E_G(x, x) = 0, \forall x \in F$$

**Dem**

Immediata, ja que G és acíclic.

Ens interessa utilitzar el tancament transitiu de  $E_G$  sobre el graf definit per l'usuari, per tal d'arranjar correctament les relacions inclusives entre conceptes, de caràcter eminentment transitiu. Per això estenem la definició anterior:

**Def** Definim  $\bar{E}_G$  com el tancament transitiu de  $E_G$ . Es a dir:

$$\bar{E}_G(x, y) = 1 \Leftrightarrow \begin{array}{l} 1) E_G(x, y) = 1, \text{ o} \\ 2) \exists y': \bar{E}_G(x, y') = 1 \text{ i } \bar{E}_G(y', y) = 1. \end{array}$$

**Propietat**

$\bar{E}_G$  defineix un ordre estricte.

**Dem**

- 1.-  $\bar{E}_G(x, x) = 0$  per a tot  $x \in F$   
Per definició de  $E_G$  i G acíclic.
- 2.- Transitiva.

Per definició transitiva de  $\bar{E}_G$ .

Extenem ara la relació d'especificitat semàntica a les parts del conjunt F de fets finit:

**Def** Donat  $G = (V, E)$  graf actiu, i F el conjunt de fets d'una aplicació (F finit), definim la relació  $E_G^*$  sobre les parts de F,  $P(F)$ , " *ser més específic semànticament* " com:

$$E_G^*: P(F) \times P(F) \rightarrow \{0, 1\}$$

$$E_G^*(S_1, S_2) = 1 \Leftrightarrow S_1 \neq \emptyset \text{ i per a tot } y \in S_2, \text{ existeix } x \in S_1 \text{ tal que } \bar{E}_G(x, y) = 1$$

Nota: Observi's:  $E_G^*(\emptyset, S) = 0$  per a tot S i  $E_G^*(S, \emptyset) = 1$  per a tot  $S \neq \emptyset$

**Propietats**

a)  $E_G^*$  defineix un ordre estricte.

**Dem**

1.-  $E_G^*(S, S) = 0$

Si  $S = \emptyset$ , cert per definició.

Si  $S \neq \emptyset$ , suposem  $E_G^*(S, S) = 1$

sigui  $y_0 \in S$ , per la definició tenim que existeix  $y_1 \in S$  tal que  $y_1 \neq y_0$  i

tal que  $\bar{E}_G(y_1, y_0) = 1$ ,

de forma anàloga existeix  $\in S : y_2 \neq y_1, y_2 \neq y_0 : \bar{E}_G(y_2, y_1) = 1$

d'aquesta forma s'obtidria una successió infinita d'elements:  $(y_0, y_1,$

$y_2, \dots, y_n, \dots)$ . Això contradiu la hipòtesi de la finitud de  $F$ .

Conseqüentment, ha de ser  $E_G^*(S, S) = 0$

2.- Transitiva

Suposem  $E_G^*(S_1, S_2) = 1$  i  $E_G^*(S_2, S_3) = 1$ , llavors tenim

per a tot  $z \in S_3$  existeix  $y \in S_2$  tal que  $\bar{E}_G(y, z) = 1$ . Si  $y \in S_2$  llavors

existeix  $x \in S_1$  tal que  $\bar{E}_G(x, y) = 1$ , i per transitivitat de  $\bar{E}_G$  tenim

$\bar{E}_G(x, z) = 1$ , i per tant  $E_G^*(S_1, S_3) = 1$

b)  $E_G^*(S_1, S_2) = 1 \Rightarrow E_G^*(S_1, T) = 1$ , si  $T \subseteq S_2$

**Dem**

Si  $E_G^*(S_1, S_2) = 1$  tenim que per a tot  $x \in S_2$  existeix  $y \in S_1$

tal que  $\bar{E}_G(x, y) = 1$ ,

ara bé  $x \in T \Rightarrow x \in S_2$ , per tant per a tot  $x \in T$  existeix  $y \in S_1$  tal que

$\bar{E}_G(x, y) = 1$  llavors  $E_G^*(S_1, T) = 1$

c)  $E_G^*(S_1, S_2) = 1 \Rightarrow E_G^*(T, S_2) = 1$ , si  $S_1 \subseteq T$

**Dem**

Si  $E_G^*(S_1, S_2) = 1$  tenim per a tot  $x \in S_2$  existeix  $y \in S_1$  tal que  $\bar{E}_G(x, y) = 1$

ara bé  $x \in S_1 \Rightarrow x \in T$ , per tant per a tot  $x \in S_2$  existeix  $y \in T$

tal que  $\bar{E}_G(x, y) = 1$  llavors  $E_G^*(T, S_2) = 1$

*Especificitat sintàctica*

**Def** Donat  $G = (V, E)$  graf actiu, i  $F$  el conjunt de fets d'una aplicació ( $F$  finit), definim la relació  $I_G$  sobre les parts de  $F$ ,  $P(F)$ , "ser més específic sintàcticament" com:

$$I_G: P(F) \times P(F) \rightarrow \{0, 1\}$$

$$I_G(S_1, S_2) = \begin{cases} 1 & S_2 \subsetneq S_1 \\ 0 & \text{altrament} \end{cases}$$

**Propietats**

- a)  $SP_G(S, S) = 0$  Evident
- b) Transitiva. Evident

*Especificitat entre conjunts*

Acabem de veure que les relacions  $E_G^*$  i  $I_G$  defineixen dos ordres estrictes sobre les parts de  $F$ . A partir d'aquestes es defineix a continuació un altre ordre estricte que és una extensió dels anteriors.

**Def** Donat  $G = (V, E)$  graf actiu, i  $F$  el conjunt de fets d'una aplicació ( $F$  finit), definim la relació  $E_G^S$  sobre les parts de  $F$ ,  $P(F)$ , "ser més específic" com:

$$E_G^S: P(F) \times P(F) \rightarrow \{0, 1\}$$

$$E_G^S(S_1, S_2) = E_G^*(S_1, S_2 - S_1)$$

on  $S_2 - S_1$  representa la diferencia de conjunts.

**Propietats**

- a)  $E_G^S$  defineix un ordre estricte.

**Dem**

1.-  $E_G^S(S, S) = 0$

$$E_G^S(S, S) = E_G^*(S, S - S) = E_G^*(S, \emptyset) = 0$$

2.- Transitiva

Suposem  $E_G^*(S_1, S_2 - S_1) = 1$  i  $E_G^*(S_2, S_3 - S_1) = 1$ , llavors per a tot  $z \in S_3 - S_1$

Cas 1:  $z \in S_2$  llavors  $z \in S_2 - S_1$  i existeix  $x \in S_1$ :  $\bar{E}_G(x, z) = 1$

Cas 2:  $z \notin S_2$  llavors  $z \in S_3 - S_2$  i existeix  $y \in S_2$

tal que  $\bar{E}_G(y, z) = 1$



llavors

Cas 2.1:  $y \in S_1$  tal que  $\bar{E}_G(y, z)$

Cas 2.1:  $y \notin S_1$  llavors  $y \in S_2 - S_1$  i existeix  $x \in S_1$ :

$\bar{E}_G(x, y) = 1$  i per transitivitat de  $\bar{E}_G$  tenim  $\bar{E}_G(x, z) = 1$

b)  $E_G^S$  és una extensió de  $E_G^*$  i de  $SP_G$

Dem

- 1.-  $SP_G(S_1, S_2) = 1 \Rightarrow E_G^S(S_1, S_2) = 1$   
 $SP_G(S_1, S_2) = 1 \Rightarrow S_2 \subsetneq S_1$   
 $E_G^S(S_1, S_2) = E_G^*(S_1, S_2 - S_1) = E_G^*(S_1, \emptyset) = 1$
- 2.-  $E_G^*(S_1, S_2) = 1 \Rightarrow E_G^S(S_1, S_2) = 1$   
 $E_G^*(S_1, S_2) = 1 \Rightarrow S_2 - S_1 \subset S_2$   
 $\Rightarrow E_G^*(S_1, S_2 - S_1) = E_G^S(S_1, S_2) = 1$

La relació  $E_G^S$  ens defineix, doncs, un ordre estricte entre les parts dels conjunts de fets. Aquest ordre ens servirà per ordenar les regles tenint en compte el conjunt de fets utilitzats en la definició de les seves premisses. A l'apartat 3.3 definirem la relació de subsumpció entre regles que utilitzarà la definició de  $E_G^S$  feta aquí. Cal remarcar que sembla molt interessant poder barrejar la inclusió sintàctica i semàntica en una sola relació degut a la facilitat que això comporta a l'hora d'escriure regles.

### 2.3.5 Us dels GRF per definir jerarquies pures

L'estructuració de les relacions jeràrquiques entre conceptes porta en general a definir tantes jerarquies com subdominis tingui el sistema. Per exemple a l'aplicació PNEUMON-IA es defineixen jerarquies per: *dades microbiològiques, dades epidemiològiques, agents causants*, etc. Pot passar que el nombre de jerarquies i la seva estructuració variïn per un mateix domini segons l'expert o grup d'experts que desenvolupin una aplicació. En tot cas el que sempre és necessari és el maneig de diverses jerarquies pures<sup>1</sup> dins les aplicacions.

Un exemple de jerarquies mixtes podria ser la següent taxonomia basada en caràcters radiològics, temporals i de factors de risc:

<sup>1</sup>S'entén com a jerarquia pura aquella en la que els elements pertanyen a un sol subdomini: per exemple la jerarquia de bacteries causants d'una malaltia. Les jerarquies mixtes barregen elements de diferents subdominis.

*Pneumònies**Pneumònies cavitades*

*Agudes:*        *en malalts prèviament sans*  
                          *en immunocompromesos*

*Cròniques*     *en malalts prèviament sans*  
                          *en immunocompromesos*

*Pneumònies no cavitades*

*Agudes*         *en malalts prèviament sans*  
                          *en immunocompromesos*

*Cròniques*     *en malalts prèviament sans*  
                          *en immunocompromesos*

Una alternativa a aquesta jerarquia podria ser la següent:

*Pneumònies**En malalts prèviament sans*

*Agudes:*        *no cavitades*  
                          *cavitades*

*Cròniques:*    *no cavitades*  
                          *cavitades*

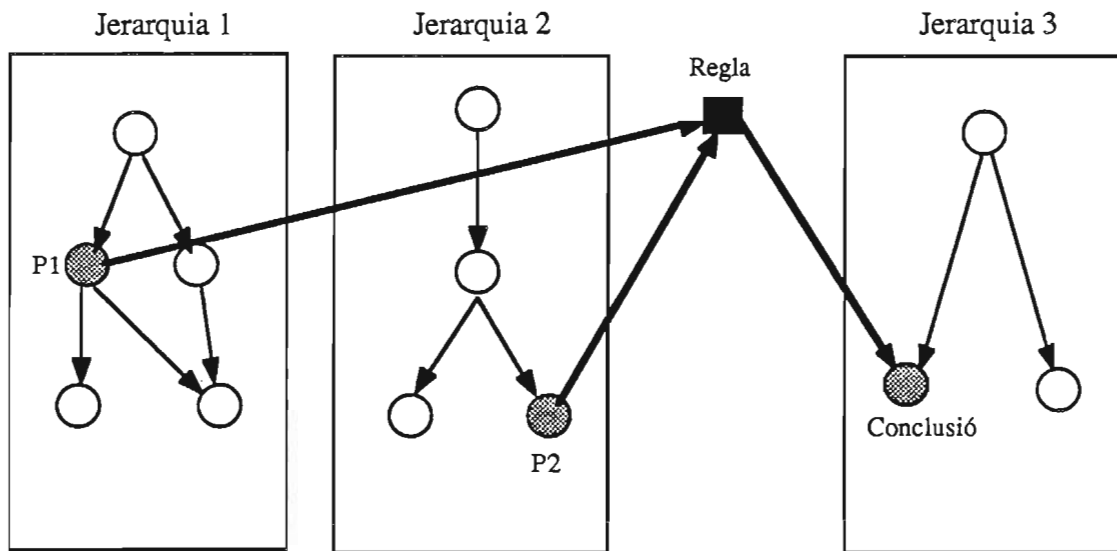
*En malalts immunocompromesos*

*Agudes:*        *no cavitades*  
                          *cavitades*

*Cròniques:*    *no cavitades*  
                          *cavitades*

Els processos de classificació es poden entendre com una forma de connectar aquestes jerarquies segons relacions heurístiques entre els nodes. D'aquesta manera es poden interpretar els coneixements heurístics com a lligams entre jerarquies diferents [Clancey, 1985a]. L'estructura d'aquests lligams és d'una gran importància a l'hora de definir arquitectures de SE. Hi ha una classificació d'aquests lligams que és especialment interessant, donat que generen arquitectures molt heterogènies:

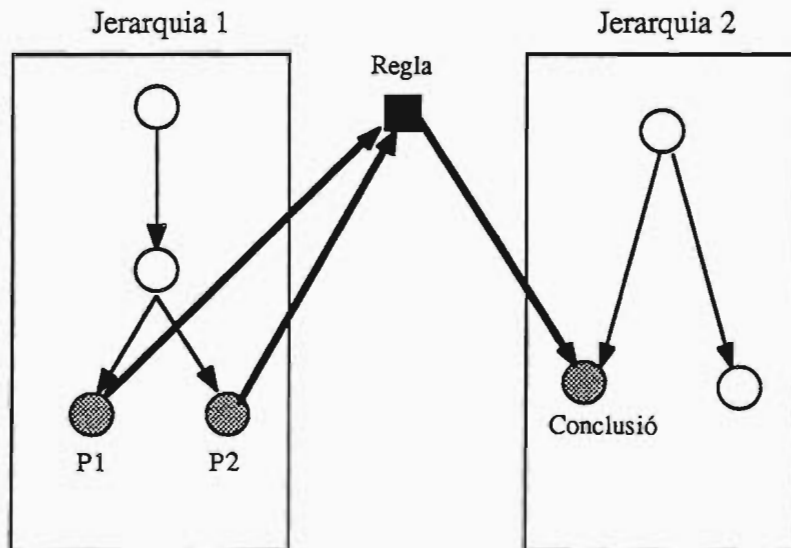
**Lligams mixtes:** es lliguen elements d'un conjunt de jerarquies amb elements d'una altra. Van ser els utilitzats majoritàriament en els sistemes experts de la primera època.



P1 i P2 són fets utilitzats a la premissa de Regla que dedueix conclusió

Fig. 2.9 Exemple d'ús de jerarquies mixtes per part d'una regla.

**Lligams simples:** són aquells que lliguen elements d'una única jerarquia amb elements d'una altra.



P1 i P2 són fets utilitzats a la premissa de Regla que dedueix conclusió

Fig. 2.10 Exemple d'ús d'una jerarquia pura per part d'una regla.

La classificació en jerarquies pures múltiples té en compte exclusivament conceptes que pertanyen a un determinat subdomini. Un exemple de jerarquia pura segons l'estat previ del malalt de pneumònies podria ser:

*Pneumònies*  
*En malalts sense factors de risc*  
*En malalts amb factors de risc*  
    *No Immunocompromesos*  
        *Alcohòlic*  
        *Broncoaspiració*  
        *Diabètic*  
        *EPOC*  
        *Fibrosi quística*  
        *Fumador*  
        ...  
    *Immunocompromés*  
        *Citostàtics*  
        *Corticosteroids*  
        *Càncer avançat*  
        *Leucòsi aguda*  
        *Limfoma*  
        *Mieloma*  
        ...

Aquests lligams, tant mixtes com simples, són representats dins MILORD com a regles d'inferència i com a metaregles (veure 5.8).

## 2.4 Compilació de les BC

Les bases de coneixements dels sistemes experts escrits en MILORD són inicialment compilats a un llenguatge de representació intermedi, i posteriorment interpretats. Aquesta execució en dues fases és comuna a molts llenguatges de programació, com ara LISP o PROLOG, i també a molts entorns de desenvolupament de SE (veure fig 2.11).

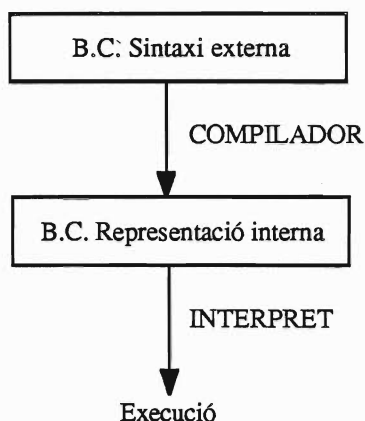


Fig 2.11 Esquema de les fases d'avaluació d'una BC a MILORD.

Les raons que suporten aquest esquema són diverses, entre les quals destaquen:

- 1 Permet extreure la màxima informació de la sintaxi del programa, com per exemple referències creuades, emmagatzemant-la per la fase d'interpretació.
- 2 Permet realitzar comprovacions sintàctiques: control de tipus, correcció de les premisses de les regles, etc. sense haver d'esperar al moment d'execució.
- 3 Permet optimitzar la representació fent-la el més adient possible a la forma de treballar dels mecanismes deductius i de control.

A l'apèndix II descriurem la semàntica dels constructes de MILORD donant part de les equacions semàntiques del compilador. Aquestes equacions treballen sobre uns objectes semàntics que representen l'esquelet de la representació interna. No es pretén representar en els objectes semàntics tota la representació interna, atès que seria molt llarg i, a més, no ajudaria a comprendre millor el sistema. Es pot veure el compilador com un procés en dos etapes, en la primera de les quals utilitzarem les equacions esmentades, generant objectes semàntics, i en la segona completarem aquests objectes transformant-los en objectes nous extesos. Aquest procés en dos etapes es pot veure a la figura 2.12.

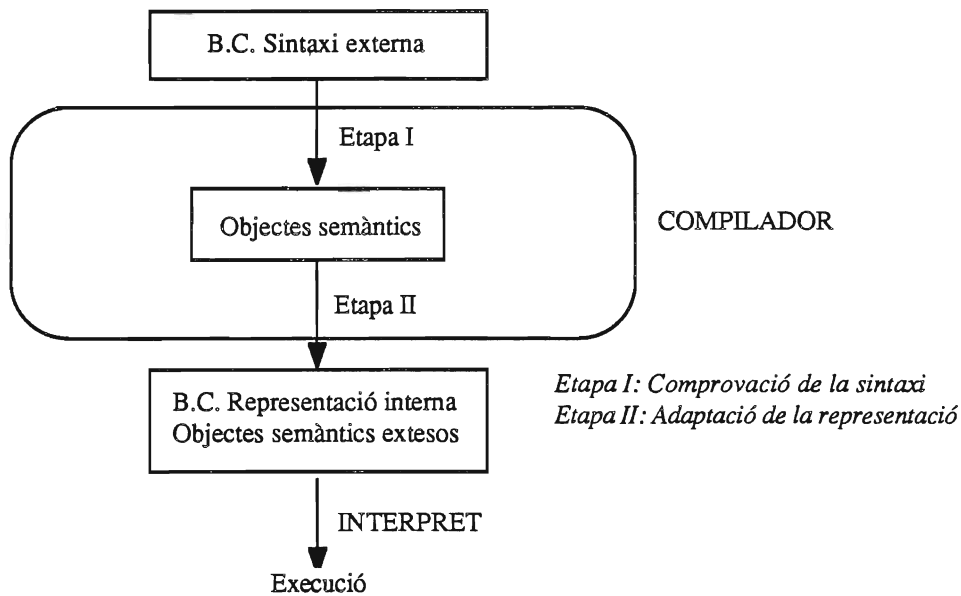


Figura 2.12 Etapes del compilador.

S'ha dividit el compilador en dues etapes per raons d'*eficiència* i *modularitat*. Eficiència, ja que dedicant una etapa a la generació de objectes semàntics que representin bàsicament el contingut sintàctic, la comprovació d'errors sintàctics, que malgrat l'experiència dels programadors sempre apareixen, d'aquesta manera es fa més ràpidament. Modularitat, ja que afegir noves característiques a la representació interna, implicarà la modificació de només una de les dues etapes, la segona.

Les funcions més importants que realitzen les dues etapes són:

Etapa I: Anàlisi lèxica i sintàctica; gestió de les taules de símbols.

Etapa II: Referències creuades.

El lector interessat pot adreçar-se a l'apèndix II on es descriu en més detall els diferents components del compilador.

### 3. Tractament del coneixement incert i incomplet

CERTUM EST QUIA IMPOSSIBILE

*Tertuliano*

#### 3.1. Introducció

En els darrers anys hi ha hagut un gran interès en el tractament de la incertesa als SE. Això és així degut a la necessitat d'enfrontar-se a dominis de coneixement on apareix informació inexacta, conflictiva i imprecisa. L'origen de la incertesa en el disseny de bases de coneixements té el seu origen en almenys quatre causes [Bonissone i Tong 1986]:

- 1.- *Manca de fiabilitat de la informació.* Limitacions dels aparells de mesura, apreciacions subjectives, etc.
- 2.- *Imprecisió inherent al llenguatge d'expressió.* Els coneixements són expressats per un o més experts en un llenguatge no formal, generalment en llenguatge natural.
- 3.- *Presència d'informació incompleta.* Malauradament la informació de que es disposa a l'hora de resoldre un cas no és tota la que hom voldria. Malgrat això els humans demostren gran habilitat en raonar en situacions on els coneixements són molt parcials.
- 4.- *Agregació d'informacions conflictives, contradictòries o redundants.*

El manegament de la incertesa interessa de cara a l'automatització de la regla d'inferència lògica *modus ponens*. Les diferències observables entre els diferents esquemes són el tipus d'incertesa a tractar i la forma de representar-la [Valverde et al, 1987].

Certs sistemes han restringit els coneixements a representar de forma que





permetrà deduir la certesa a posteriori de  $q$  a partir de la certesa de la regla  $p \rightarrow q$ , de la certesa a priori de  $q$  i de la certesa de  $p$ , actuant com a funció de *modus ponens*.

Tant les probabilitats a priori com les probabilitats condicionades han de ser proporcionades per l'expert. Donats els problemes que això pot comportar, Duda [Duda et al 1979] proposa una expressió més adient per la fórmula de Bayes tenint en compte els dos fets  $q$  i  $\neg q$ , que notarem  $\bar{q}$ . Aquesta formulació és la utilitzada al sistema expert PROSPECTOR [Duda et al, 1979].

De la fórmula de Bayes aplicada a  $q$  i  $\bar{q}$  s'obté:

$$\frac{P(q/p)}{P(\bar{q}/p)} = \frac{P(p/q) P(q)}{P(p/\bar{q}) P(\bar{q})}$$

definint

$$O(q/p) = \frac{P(q/p)}{P(\bar{q}/p)}, \quad O(q) = \frac{P(q)}{P(\bar{q})} \quad \text{i} \quad LS = \frac{P(p/q)}{P(p/\bar{q})}$$

s'obté

$$O(q/p) = LS * O(q)$$

on  $LS \in [0, +\infty]$  expressa el grau de *suficiència* de la regla  $p \rightarrow q$ , és a dir, un augment del valor de  $LS$  incrementa la probabilitat de  $q$  sota la hipòtesis de  $p$  certa.

De manera anàloga s'obté:

$$O(q/\bar{p}) = LN * O(q) \quad \text{on} \quad LN = \frac{P(\bar{p}/q)}{P(\bar{p}/\bar{q})}$$

i expressa el grau de *necessitat* de la regla  $p \rightarrow q$ , és a dir, un valor de  $LN$  menor que 1 disminueix la probabilitat de  $q$  sota la hipòtesi de  $p$  fals.

Aleshores, una regla tindrà associats dos valors  $LS$  i  $LN$ , el grau de suficiència i necessitat respectivament, que hauran de ser determinats per l'expert. De fet, atès que el grau de *suficiència* de la regla  $p \rightarrow q$  coincideix amb el grau de *necessitat* de  $\bar{p} \rightarrow q$ . En la pràctica l'expert avalua les regles:

$$p \rightarrow q$$

$$\bar{p} \rightarrow q$$

La relació entre els factors LS i LN d'una regla és:

$$LN = \frac{1 - LS P(p/\bar{q})}{1 - P(p/\bar{q})}$$

i s'acompleix:

$$LS = 1 \quad \text{sii} \quad LN = 1$$

$$LS > 1 \quad \text{sii} \quad LN < 1$$

Un dels problemes que s'han plantejat en els sistemes desenvolupats sota aquest marc és la violació d'aquestes relacions entre LS i LN per part dels experts. Aquestes relacions, com va ser observat per Duda et al [Duda et al, 1979], són de vegades massa restrictives, atès que es pot donar que sovint l'expert indica que l'observació de  $p$  ha d'incrementar la probabilitat de  $q$  ( $LS > 1$ ) però la falsedat de  $p$  vol que no tingui cap efecte sobre  $q$  ( $LN = 1$ ). Aquesta informació és inconsistent amb la relació establerta més amunt.

L'altre problema important dels sistemes probabilistes, com és el cas de PROSPECTOR, és la inconsistència de les assignacions de probabilitats a priori. Anem a veure-ho en més detall.

Els usuaris d'un sistema expert, en la majoria dels casos, no poden donar evidències absolutes a favor o en contra d'un determinat fet. Al contrari donen valors sobre l'evidència observada del fet:

$$P(p \mid \text{dades relevants}) = 0.6$$

Si anomenem  $p'$  aquestes dades relevants i volem obtenir un valor per  $P(q \mid p')$ , podem extreure de la teoria de probabilitats:

$$\begin{aligned} P(q \mid p') &= P(q, p \mid p') + P(q, \bar{p} \mid p') = \\ &= P(q \mid p, p') P(p \mid p') + P(q \mid \bar{p}, p') P(\bar{p} \mid p') \end{aligned} \quad (1)$$

Ara bé, si ja coneixem que  $p$  és cert o fals llavors es possible suposar que  $p'$  no porporioni informació addicional, és a dir:

$$P(q | p, p') = P(q | p)$$

$$P(q | \bar{p}, p') = P(q | \bar{p}) \text{ i s'acompleix}$$

$$P(q | p') = P(q | p) P(p | p') + P(q | \bar{p}) P(\bar{p} | p')$$

Si ara pensem en una regla "  $p \rightarrow q$  " dins una xarxa inferencial ( $p$  és també un fet deduïble dins la xarxa), l'expert ha de donar probabilitats a priori, no només a la conclusió  $q$  i a la regla, sinó també a  $p$ . Si les assignacions fossin consistents amb l'expressió

$$P(q | p') = P(q | p) P(p | p') + P(q | \bar{p}) P(\bar{p} | p')$$

tindriem el següent gràfic:

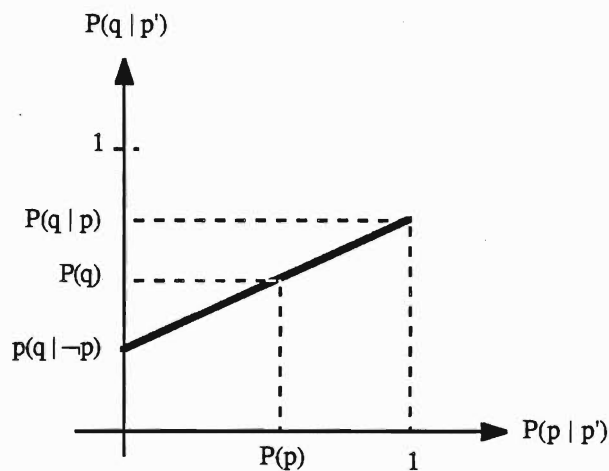


Fig. 3.1 Probabilitats a priori de  $p$  i  $q$  consistents.

malauradament la major part de les vegades la realitat és una altre i coincideix amb el següent dibuix on les probabilitats a priori no coincideixen amb el que l'equació prediu.

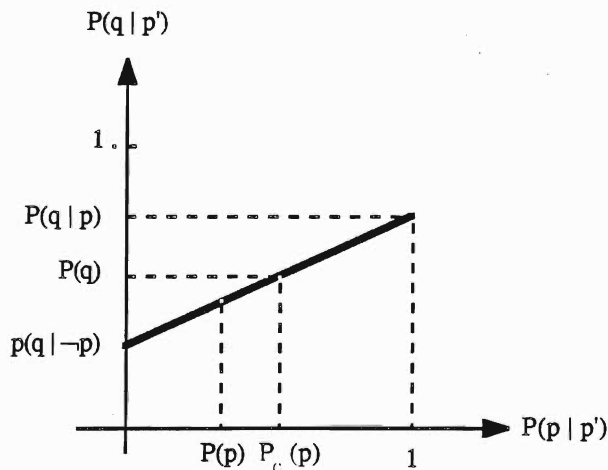


Fig. 3.2 Probabilitats a priori de  $p$  i  $q$  inconsistentes.

Exemple:

Donada la probabilitat a priori d'*etiologia pneumocòcica* de 0.35, s'estudiarà quina és la probabilitat condicionada en cas de que la pneumònia sigui *bacteriana* considerant la regla :

$$bacteriana \rightarrow pneumocòcica$$

amb uns valors donats per l'expert de:

$$LS = 2 \text{ i } LN = 0.5$$

Per altre part, preguntat l'expert sobre la probabilitat a priori de pneumònia bacteriana, aquest afirma que és 0.6. Calcularem la probabilitat consistent de *bacteriana* amb els valors LS i LN de la regla i la probabilitat a priori d'*etiologia pneumocòcica*.

a) Probabilitat condicionada

Signi  $q$  *pneumococ*;  $p'$  dades rellevants per establir el diagnòstic de *bacteriana* i  $p$  *bacteriana*.

sabem que:  $O(q/p) = \frac{P(q/p)}{1-P(q/p)} = LS * O(q)$ , i que  $O(p) = \frac{P(p)}{P(\bar{p})}$  de on

$$O(\text{pneumococ}) = \frac{P(\text{pneumococ})}{P(\bar{\text{pneumococ}})} = \frac{0.35}{0.65} = 0.538$$

i,

$$O(\text{pneumococ}/bacteriana) = LS * O(\text{pneumococ}) = 2 * 0.538 = 1.077$$

$$\text{de } O(q/p) = \frac{P(q/p)}{1-P(q/p)} \text{ tenim } P(q/p) = \frac{O(q/p)}{1+O(q/p)}$$

$$P(\text{pneumococ}/bacteriana) = \frac{1.077}{2.077} = 0.518$$

de la mateixa manera es calcula:  $P(\text{pneumococ}/\bar{bacteriana})$ , que és 0.211

b) Probabilitat consistent

Calcularem quina es la probabilitat de *bacteriana*,  $P_c(bacteriana) = P(p | p')$ , **consistent** amb la probabilitat a priori de *pneumococ*. Es a dir quan  $P(q | p') = P(q)$ .

Com ja es tenia previament:

$$P(q | p') = P(q | p) P(p | p') + P(q | \bar{p}) P(\bar{p} | p')$$

en aquest cas:  $P_c(p) = P(p | p')$  i  $P(q | p') = P(q)$

$$P(q) = P(q | p) P_c(p) + P(q | \bar{p}) P_c(\bar{p})$$

$P(\text{pneumococ}) =$

$$P(\text{pneumococ} | \text{bacteriana}) P_c(\text{bacteriana}) + \\ P(\text{pneumococ} | \neg \text{bacteriana}) (1 - P_c(\text{bacteriana}))$$

d'on:

$$P_c(\text{bacteriana}) = \frac{P(\text{pneumococ}) - P(\text{pneumococ} | \neg \text{bacteriana})}{P(\text{pneumococ} | \text{bacteriana}) - P(\text{pneumococ} | \neg \text{bacteriana})}$$

$$\text{i per tant, } P_c(\text{bacteriana}) = \frac{0.35 - 0.211}{0.518 - 0.211} = 0.45$$

La probabilitat a priori de bacteriana consistent amb les dades inicials fixades per l'expert seria de 0.45 mentre que el mateix expert havia indicat una probabilitat a priori de 0.6, els valors són, doncs, inconsistents.

La solució al problema exemplificat ha motivat diferents treballs [Duda et al, 1979]. Les solucions passen per la modificació de les assignacions de probabilitat a priori, per tal d'adaptar-les a les restriccions imposades pel compliment de les equacions de propagació. Aquestes modificacions poden portar, de vegades, a assignacions de probabilitats sense significat des del punt de vista de l'expert. Una altra solució és modificar l'equació de càlcul de les probabilitats condicionades de manera que es forci la consistència. D'aquesta manera la interpolació lineal de la equació (1) es pot modificar tal com apareix a la figura 3.3.

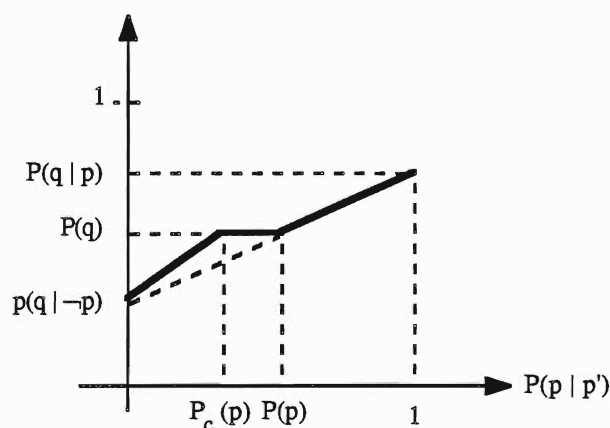


Fig 3.3 Modificació del càlcul de les probabilitats condicionades.

Quan més d'una regla dedueix la mateixa conclusió la combinació d'evidències en aquest mètode és la següent,

$$p(E_1 \dots E_n | H) = \prod_{i=1}^n P(E_i | H)$$

$$p(E_1 \dots E_n | \bar{H}) = \prod_{i=1}^n P(E_i | \bar{H})$$

que presuposa el que s'anomena independència condicional.

Una altra formulació basada en el model probabilista és l'emprada pel sistema MYCIN que va ser desenvolupat inicialment com una alternativa al raonament probabilistic. Defineix uns *factors de certesa* com a diferència de dues mesures obtingudes a partir de probabilitats subjectives. Heckerman [Heckerman, 1986] dona una interpretació probabilista dels factors de certesa a partir de les propietats que han de verificar. Aquesta interpretació engloba MYCIN dins el mateix marc probabilista que el mètode utilitzat a PROSPECTOR [Duda et al, 1979] i fa les mateixes hipòtesis de modularitat i d'independència condicional.

La combinació d'evidències utilitzada en el sistema MYCIN és la següent:

*Combinació paral·lela*

$$z = \begin{cases} x + y - xy & x, y \geq 0 \\ \frac{x + y}{1 - \min(|x|, |y|)} & x \text{ i } y \text{ de signe oposat} \\ x + y + xy & x, y < 0 \end{cases}$$

on  $x = CF(H, E_1)$   $y = CF(H, E_2)$   $z = CF(H, E_1 E_2)$  al graf:

$$E_1 \rightarrow H \text{ i } E_2 \rightarrow H$$

*Propagació*

$$z = \begin{cases} wx & w \geq 0 \\ -wy & w < 0 \end{cases}$$

on  $w = CF(E, E')$   $x = CF(H, E)$   $y = CF(H, \neg E)$   $z = CF(H, E')$  al graf

$$E' \rightarrow E \text{ i } E \rightarrow H$$

Les hipòtesis de modularitat i independència condicional que apareixen a la caracterització dels *factors de certesa* (FC) de MYCIN i en el cas de PROSPECTOR, són molt restrictives i poden portar a inconsistències quan es combinen les regles.

$FC(p, q, e)$ , el factor de certesa de la regla  $p \rightarrow q$ , indica un canvi de creència en la hipòtesi  $q$  donada l'evidència de  $p$  i el coneixement  $e$ . Els axiomes que donen lloc a problemes són els següents:

*modularitat:*  $FC(q,p,e) = FC(q,p,\emptyset) \equiv FC(q,p)$ . Vol dir que l'actualització de les evidències és independent de les evidències no considerades en l'escriptura de les regles.

*independència condicional:*

$$P(p \wedge p'/q) = P(p/q) * P(p'/q)$$

$$P(p \wedge p'/\bar{q}) = P(p/\bar{q}) * P(p'/\bar{q})$$

Així, s'ha demostrat [Johnson, 1986] que la independència condicional no és vàlida si existeixen més de dues hipòtesis mútuament excloents, amb la qual cosa la combinació de regles utilitzant els factors de certesa només és consistent si la xarxa inferencial forma un arbre. És a dir, si no existeixen regles del tipus:

$$p \rightarrow q_1$$

$$p \rightarrow q_2$$

Això és una restricció massa forta per els dominis on els sistemes experts són utilitzats. Veure [Heckerman, 1986] per més detalls.

### 3.1.2. La teoria de l'evidència

El model evidencial es basa en les anomenades *funcions de credibilitat*, definides sobre el conjunt de les possibles respostes a una pregunta donada, conjunt conegut com a "*Frame of discernment*". Les funcions de credibilitat donen el grau en que la resposta a la pregunta del conjunt està en cadascun dels seus subconjunts. Formalment:

**Def** direm que una funció  $Cr : 2^X \rightarrow [0, 1]$  és una credibilitat sobre el conjunt  $X$  si verifica:

- 1)  $Cr(\emptyset) = 0$

- 2)  $Cr(X) = 1$

- 3)  $Cr(A) \leq Cr(B)$  si  $A \subseteq B \subseteq X$

- 4)  $Cr(A_1 \cup \dots \cup A_n) \geq \sum_{I \subseteq \{1, \dots, n\}} (-1)^{|I|+1} Cr(\bigcap_{i \in I} A_i)$

**Def** Donada  $Cr$ , funció de credibilitat sobre  $X$ , es defineix una funció  $Pl: 2^X \rightarrow [0, 1]$  denominada de *plausibilitat* i definida per:

$$Pl(A) = 1 - Cr(\bar{A})$$

De les definicions es dedueix que  $Pl(A) \geq Cr(A)$  per tot  $A \subset X$ .

És interessant observar que les funcions de credibilitat són extensions de mesures de probabilitat:

**Prop**  $Cr$  és una credibilitat sobre  $X$  sii existeix  $S$  conjunt, existeix  $P$  probabilitat sobre  $2^S$  i existeix  $G: S \rightarrow 2^X : G(s) \neq \emptyset$  per tot  $s \in S$  i  $\bigcup_{s \in S} G(s) = X$ , de forma que

$$Cr(A) = P\{s \in S / G(s) \subset A\}$$

**Dem** Veure [Shafer, 1979]

El conjunt  $S$  es pot identificar amb el conjunt  $2^X - \{\emptyset\}$ , i si  $X$  és finit la probabilitat queda determinada per una funció  $m: 2^X - \{\emptyset\} \rightarrow [0, 1]$  que compleix

- 1)  $m(B) \geq 0$
- 2)  $\sum \{m(B) / \emptyset \neq B \subset X\} = 1$

aquesta funció rep el nom de funció d'"assignació de massa" i verifica:

$$Cr(A) = \sum \{m(B) / \emptyset \neq B \subset A\}$$

$$Pl(A) = \sum \{m(B) / B \cap A \neq \emptyset\}$$

d'aquesta manera la credibilitat reflexa el pes total de les evidències que impliquen  $A$  i la plausibilitat el pes total de les evidències que fan possible  $A$ , es a dir que no impliquen  $\bar{A}$ .

Donat un conjunt  $A \subset X$ , i suposant conegudes:

$$Cr(A) = \alpha \quad i \quad Pl(A) = \beta$$

existeix només una funció d'assignació de massa que ho compleixi i que compleixi el principi de *mínima especificitat*. Aquest principi aconsella mantenir-se sempre tan imprecís com es pugui, és a dir, maximitzar la diferència  $Pl(B) - Cr(B)$  per tot  $B \subset X$ .

En el cas de que només coneguem la credibilitat i plausibilitat d'un subconjunt  $A$ , amb els valors  $\alpha$  i  $\beta$  respectivament, l'assignació de massa que respecti el principi de mínima especificitat és la següent:

$$m(A) = \alpha, m(\bar{A}) = 1 - \beta, m(X) = \beta - \alpha$$

Per tal de fer inferències, és necessari posseir certs mecanismes de maneig de les funcions de credibilitat. Aquests mecanismes van ser definits per Dempster i Shafer i tenen el nom de: *regla de combinació*, *extensió cilíndrica* i *projecció*.

La *regla de combinació* de Dempster permet fusionar dues funcions de credibilitat sobre un mateix conjunt per obtenir-ne una de nova. Aquesta regla s'expressa en termes de les funcions d'assignació de masses

Siguin  $m_1$  i  $m_2$  funcions d'assignació de massa de dues funcions de credibilitat



$Cr_1$  i  $Cr_2$  sobre  $X$ .

Si definim

$$m(A) = \frac{m_1 \cap m_2(A)}{1 - m_1 \cap m_2(\emptyset)} \quad \forall A \subset X \quad A \neq \emptyset$$

$$\text{on } m_1 \cap m_2(A) = \sum \{m_1(A_1)m_2(A_2) / A_1 \cap A_2 = A\}$$

llavors  $m$  és l'assignació de massa de la credibilitat combinació de  $Cr_1$  i  $Cr_2$ ,

Siguin dos conjunts  $X$  i  $Y$ , i  $Cr_X$  una credibilitat sobre  $X$ , llavors:  
l'*extensió cilíndrica* de  $Cr_X$  sobre  $X \times Y$  és

$$Cr_{X \times Y}(B) = \begin{cases} Cr_X(A) & \text{si } B = A \times Y \\ 0 & \text{altrament} \end{cases}$$

amb una assignació de massa:

$$m_{X \times Y}(B) = \begin{cases} m_X(A) & \text{si } B = A \times Y \\ 0 & \text{altrament} \end{cases}$$

Sigui  $Cr_{X \times Y}$  una credibilitat sobre  $X \times Y$  amb una assignació de massa  $m_{X \times Y}$ , llavors  
la *projecció* de  $Cr_{X \times Y}$  sobre  $X$  és:

$$Cr_X(A) = Cr_{X \times Y}(A \times Y) \text{ per a tot } A \subset X, A \neq \emptyset$$

amb una assignació de massa:

$$m_X(A) = \sum_{\text{Proj}_X(B) = A} m_{X \times Y}(B)$$

Aquestes funcions, com veurem a continuació, ens permetran definir un mètode de raonament sobre els elements de representació.

La representació de la incertesa en aquesta teoria es fa sobre tres tipus d'elements:

- a.- *Fets elementals*. Representats de la forma "x és A" on A és un subconjunt de un conjunt X com els esmentats anteriorment, x és una variable que pren valors a X.
- b.- *Fets compostos*. Combinacions lògiques de fets elementals. Es transformen en

fets elementals referits a subconjunts del producte cartesià dels conjunts involucrats.

Conjunció: " $x_1$  és  $A_1$  i  $x_2$  és  $A_2$ " es transforma en " $(x_1, x_2)$  és  $A_1 \times A_2$ "  
 $A_1 \times A_2 \subset X_1 \times X_2$

Disjunció: " $x_1$  és  $A_1$  o  $x_2$  és  $A_2$ " es transforma en " $(x_1, x_2)$  és  $A_1 + A_2$ "

$$A_1 + A_2 = \overline{\overline{A_1} \times \overline{A_2}} \subset X_1 \times X_2$$

c.- *Regles elementals.* Reben una interpretació conjuntista ( $p \rightarrow q \equiv \neg p \vee q$ ) i per tant queden reduïdes al cas anterior.

La representació de la incertesa d'un fet  $F = "x \text{ és } A"$  vindrà donada per l'interval  $[Cr(A), Pl(A)]$  que conté la probabilitat de que passi el fet  $F$ . Els valors que defineixen els extrems de l'interval d'incertesa d'una regla admeten diferents interpretacions.

La incertesa d'una regla  $R = "x \text{ és } A \text{ llavors } y \text{ és } B"$  que es transforma en el fet " $(x, y)$  és  $\bar{A} + B$ " pot representar-se per dos valors  $\alpha, \beta \in [0, 1]$ . Una de les interpretacions possibles d'aquests valors  $\alpha$  i  $\beta$  és prendre

$$Cr(\bar{A} + B) = \alpha \quad Pl(\bar{A} + B) = \beta$$

on  $\alpha \leq \beta$  i aplicant el principi de mínima especificitat tindriem:

$$m_R(\bar{A} + B) = \alpha, m_R(A + \bar{B}) = \beta, m_R(X \times Y) = \beta - \alpha$$

Dels mecanismes d'inferència veurem el mètode general, aplicable sobre qualsevol conjunt de fets i regles, i el modus ponens aplicable sobre cadenes inferencials.

*Mètode general:*

Sigui  $F_1, \dots, F_m$  el conjunt dels fets,  $R_1, \dots, R_n$  el conjunt de regles i  $F$  el fet sobre el que volem establir la certesa:

- 1) Definim  $\Omega$  com el mínim conjunt que conté els conjunts  $F_i$   $i = 1, \dots, m$   $R_j$   $j = 1, \dots, n$  i  $F$ . Es a dir si a la base de coneixements hi ha  $p$  variables  $x_1, \dots, x_p$ , llavors  $\Omega = X_1 \times \dots \times X_p$
- 2) Es calcula l'extensió cilíndrica de les funcions d'assignació de massa  $m_{F_i}$ ,  $m_{R_j}$  i es combinen amb la regla de Dempster per obtenir una funció d'assignació de

massa  $m$  sobre  $\Omega$ .

3) Es projecta  $m$  sobre el conjunt  $Z$  al qual pertany  $F$ , i es calcula  $Cr(F)$  i  $Pl(F)$ .

Aquest mètode utilitza al mateix temps tota la base de coneixements. El cost computacional que això representa fa inviable el seu ús sobre Bases de Coneixements grans.

En els casos en que es pugui tenir una cadena inferencial es pot utilitzar el mecanisme de modus ponens:

*Modus ponens:*

L'esquema del modus ponens utilitzant aquesta aproximació seria: donats  $p$  i  $q$  i la regla  $p \rightarrow q$  amb

$$\begin{aligned} Cr(p) &= \alpha, & Pl(p) &= \beta \text{ on } \alpha \leq \beta \\ Cr(p \rightarrow q) &= \gamma & Pl(p \rightarrow q) &= \delta \text{ on } \gamma \leq \delta \end{aligned}$$

i utilitzant la interpretació esmentada i el principi de mínima especificitat tindriem:

$$\begin{aligned} m_F(p) &= \alpha, & m_F(\neg p) &= 1 - \beta, & m_F(\Pi) &= \beta - \alpha \\ m_R(\neg p \vee q) &= \gamma, & m_R(p \wedge \neg q) &= 1 - \delta, & m_R(p) &= \delta - \gamma \end{aligned}$$

on  $\Pi$  és una tautologia.

i amb la regla de combinació de Dempster obtenim

$$\begin{aligned} Cr(q) &= \alpha\gamma \\ Pl(q) &= 1 - \beta(1 - \delta) \end{aligned}$$

Finalment remarcar que existeix una estratègia per reduir la complexitat de càlcul del mètode general, que consisteix a dividir la xarxa inferencial en subxarxes de forma que es pugui establir una cadena inferencial entre elles i aplicar el modus ponens de cost computacional molt més baix.

Exemple: sigui el fet *bacteriana*, definit sobre el conjunt  $X$ , el fet *pneumococ* definit sobre el conjunt  $Y$  i la regla *bacteriana*  $\rightarrow$  *pneumococ* definida sobre  $X + Y$ .

$$\begin{aligned} Cr(bacteriana) &= 0.7, & Pl(bacteriana) &= 1 \\ Cr(bacteriana \rightarrow pneumococ) &= 0.6 & Pl(bacteriana \rightarrow pneumococ) &= 1 \end{aligned}$$

llavors

$$Cr(pneumococ) = 0.7 * 0.6 = 0.42$$

$$Cr(\neg pneumococ) = 1 * 0 = 0$$

$$Pl(pneumococ) = 1 - 1*(1 - 1) = 1$$

$$pl(\neg pneumococ) = 1 - 0.42 = 0.58$$

### 3.1.3. La teoria de la possibilitat

El model de raonament possibilístic està basat en la lògica difusa [Zadeh 1973], [Dubois i Prade, 1985], que representa la vaguetat inherent al llenguatge natural mitjançant subconjunts difusos.

Els *fets elementals* es representen de la forma [X és A] on X és una variable sobre l'univers U i A un terme lingüístic aplicable a la variable X; terme que restringeix els valors que pot prendre la variable X atès que [X és A]. Aquest terme està representat per un subconjunt difús  $\underline{A}$  de l'univers U. Per exemple [X és jove] està representat pel subconjunt difús *jove*:

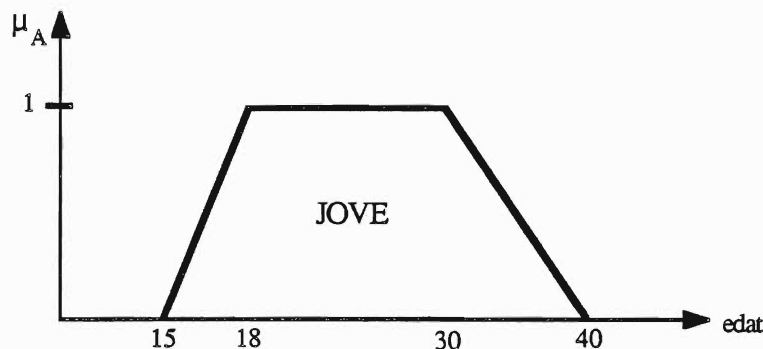


Fig. 3.4 Funció característica del subconjunt difús JOVE.

La representació de la incertesa d'un fet [X és A] ve determinada per una funció de distribució de possibilitat que s'identifica amb la funció característica generalitzada del subconjunt difús  $\underline{A}$ , és a dir, la funció de distribució de possibilitat és una funció:

$$\pi_A: U \rightarrow [0, 1] \text{ tal que } \pi_A(u) = \mu_A(u)$$

on  $\mu_A$  és la funció característica del subconjunt difús  $\underline{A}$ .

La funció de distribució de possibilitat  $\pi_A$  associa a cada  $u \in U$  la possibilitat que té X de prendre el valor  $u$  posat que [X és A].

Els *fets compostos* es representen mitjançant conjuncions, disjuncions i negacions de fets elementals. La distribució de possibilitat d'un fet compost es calcula a

partir de les distribucions de possibilitat dels fets elementals que el componen. Així, si els fets  $F = [X \text{ és } A]$  i  $G = [X \text{ és } B]$  tenen distribucions de possibilitat  $\pi_A$  i  $\pi_B$  definides sobre el mateix univers  $U$ , els fets compostos  $F \wedge G = [X \text{ és } A \text{ i } B]$ ,  $F \vee G = [X \text{ és } A \text{ o } B]$  i  $\neg F = [X \text{ és } \neg A]$  tenen per distribucions de possibilitat:

$$\pi_{A \wedge B}(u) = T(\pi_A(u), \pi_B(u))$$

$$\pi_{A \vee B}(u) = S(\pi_A(u), \pi_B(u))$$

$$\pi_{\neg A}(u) = N(\pi_A(u))$$

Aquestes funcions de combinació reben el nom de t-normes per les conjuncions, t-conormes per les disjuncions i negacions fortes per la negació [Alsina et al, 1983].

Una T-norma

$$T: [0, 1] \times [0, 1] \rightarrow [0, 1]$$

satisfà les següents propietats:

$$T(0,0) = 0$$

$$T(1,p) = T(p,1) = p$$

$$T(p,q) = T(q,p)$$

$$T(p,q) \leq T(r,s) \quad \text{si } p \leq r \text{ i } q \leq s$$

$$T(p, T(q,r)) = T(T(p,q), r)$$

Una T-conorma

$$S: [0, 1] \times [0, 1] \rightarrow [0, 1]$$

satisfà les següents propietats:

$$S(1,1) = 1$$

$$S(0,p) = S(p,0) = p$$

$$S(p,q) = S(q,p)$$

$$S(p,q) \leq S(r,s) \quad \text{si } p \leq r \text{ i } q \leq s$$

$$S(p, S(q,r)) = S(S(p,q), r)$$

Una negació forta

$$N: [0, 1] \rightarrow [0, 1]$$

satisfà les següents propietats

$$N(N(x)) = x$$

$$N(x) \leq N(y) \quad \text{si } x \geq y$$

Per operadors de negació forta  $N(x)$  apropiats, les T-normes i T-conormes són duals en el sentit de les lleis de DeMorgan [Trillas, 1979].

Alguns parells de T-normes i T-conormes duals són:

$T_0(x,y) = \begin{cases} \min(x,y) & \text{altrament} \\ 0 & \end{cases} = 1$	$S_0(x,y) = \begin{cases} \max(x,y) & \text{altrament} \\ 1 & \end{cases} = 0$
$T_1(x,y) = \max(0, x+y-1)$	$T_1(x,y) = \min(1, x+y)$
$T_{1.5}(x,y) = xy/[2-(x+y-xy)]$	$S_{1.5}(x,y) = (x+y)/(1+xy)$
$T_2(x,y) = xy$	$S_2(x,y) = x+y-xy$
$T_{2.5}(x,y) = xy/(x+y-xy)$	$S_{2.5}(x,y) = (x+y-2xy)/(1-xy)$
$T_3(x,y) = \min(x,y)$	$S_3 = \max(x,y)$

Aquests parells estan ordenats de la següent manera:

$$T_0 \leq T_1 \leq T_{1.5} \leq T_2 \leq T_{2.5} \leq T_3$$

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

Les regles elementals són representades de la forma  $R = \text{si } [X \text{ és } A] \text{ llavors } [Y \text{ és } B]$ , on  $[X \text{ és } A]$  i  $[Y \text{ és } B]$  són fets elementals sobre els universos  $U$  i  $V$ , respectivament. La incertesa s'aquestes regles ve representada per una relació difusa  $H$  en  $U \times V$ :

$$H(A,B)(u,v) = I(\pi_A(u), \pi_B(v))$$

on la funció  $I$ , funció d'implicació, representa el condicional en la lògica difusa.

Una funció d'implicació  $I$

$$I: [0, 1] \times [0, 1] \rightarrow [0, 1]$$

satisfà les següents propietats:

I decreixent respecte la primera variable i creixent respecte la segona

$$I(0,x) = 1$$

$$I(1,x) = x$$

$$I(x,I(y,z)) = I(y,I(x,z))$$

Les funcions d'implicació més comuns són:

- 1) **S-implicacions**, definides a partir d'una interpretació de la implicació  $a \rightarrow b$  com a  $\neg a \vee b$ . Així,  $I_S(p,q) = S(N(p),q)$ , essent  $S$  una  $T$ -conorma i  $N$  un operador de negació.

Exemples:

$$I_S^2(x,y) = 1-x+xy$$

$$I_S^1(x,y) = \min(1-x+y, 1)$$

- 2) **R-implicacions**, definides a partir d'una interpretació de la implicació com a forma condicional. Així,  $I_R(p,q) = T(p|q)$ , definint  $T(p|q) = \text{SUP} \{ c \in [0,1] / T(p,c) \leq q \}$  essent  $T$  una  $T$ -norma.

Exemples:

$$I_R^2(x,y) = \begin{cases} 1 & \text{si } y \geq x \\ \frac{y}{x} & \text{altrament} \end{cases}$$

$$I_R^1(x,y) = \min(1-x+y, 1)$$

El que resulta necessari en els sistemes basats en regles, és disposar de funcions que donin la certesa de la conclusió conegudes les certes de les premisses i de la regla. Les funcions que realitzen això són les *funcions generadores de modus ponens* (fgmp) [Valverde i Trillas, 1985]. Una vegada determinada quina funció d'implicació utilitzarem, és fàcil obtenir una fgmp associada a ella:

- 1) Per una  $S$ -implicació, generada a partir d'una  $T$ -conorma  $S$ , tenim  $m_I(p,r) = S(N(p)|r) = \text{INF} \{ c \in [0,1] / S(N(p),c) \geq r \}$
- 2) Per una  $R$ -implicació generada a partir d'una  $T$ -norma  $T$ , tenim  $m_I = T$ .

Per exemple vegin-se quines funcions d'implicació i quines fgmp tindriem per la parella dual  $(T_3, S_3)$ .

$$I_S^3(x,y) = \max(1-x,y) \qquad m_I(x,y) = \begin{cases} 0 & \text{si } y \geq 1-x \\ y & \text{altrament} \end{cases}$$

$$I_R^3(x,y) = \begin{cases} 1 & \text{si } y \geq x \\ y & \text{altrament} \end{cases} \qquad m_I = T_3$$

El modus ponens es generalitza en aquest contexte en la regla d'inferència:

$$\frac{([X \text{ és } A] \rightarrow [Y \text{ és } B]) \text{ és } \mu}{[X \text{ és } A] \text{ és } \tau} \qquad [Y \text{ és } B] \text{ és } \eta$$

considerant que  $\mu$  és una qualificació veritativa de la regla, s'obté:

$$\eta(y) = \sup_{I(x,y)=z} m_I(\tau(x), \mu(z))$$

Exemple: En el cas de la regla  $[X \text{ és } \textit{bacteriana}] \rightarrow [X \text{ és } \textit{pneumococ}]$  si el valor de certesa de la regla fos molt\_possible i la certesa de *bacteriana* fos força\_possible, utilitzant com a T-norma  $T_2$  i la seva funció de modus ponens associada a la R-implicació  $I_{R^2}$ , que coincideix amb  $T_2$   $m_{I_{R^2}} = T_2$  (veure 3.5), el valor de *pneumococ* podria ser:

$$\frac{([X \text{ és } \textit{bacteriana}] \rightarrow [X \text{ és } \textit{pneumococ}]) \text{ és } \textit{molt\_possible}}{[X \text{ és } \textit{bacteriana}] \text{ és } \textit{força\_possible}} \qquad [X \text{ és } \textit{pneumococ}] \text{ és } \eta$$

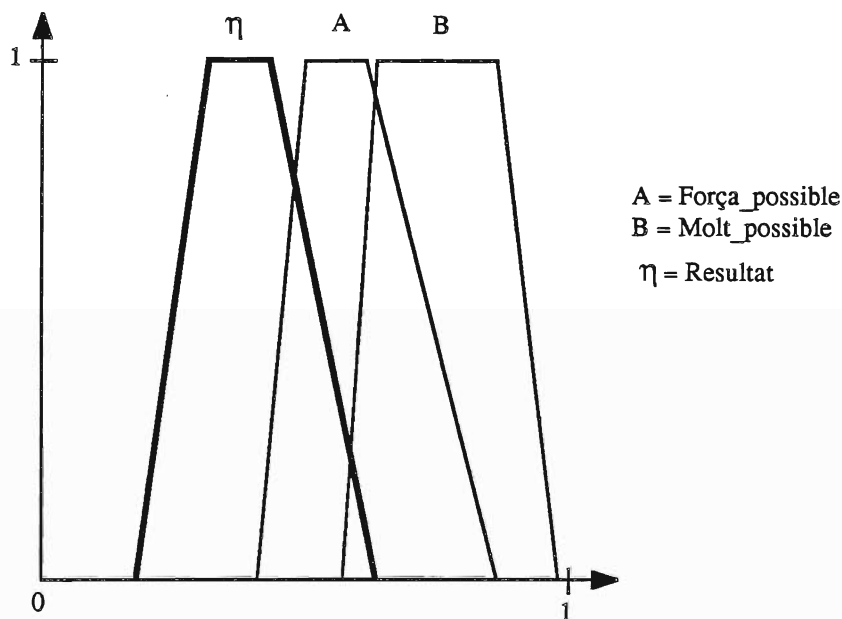


Fig. 3.5 Exemple de combinació de valors de certesa segons el modus ponens generalitzat.



## 3.2. Lògica difusa i coneixements incerts

Un sistema expert pot ser interpretat com un sistema d'informació que permet donar respostes a qüestions relatives a la informació continguda a la seva BC. Aquesta informació és en moltes ocasions imprecisa, incompleta i de vegades poc segura. Atès que el que s'espera del sistema són respostes a preguntes concretes, aquestes respostes hauran de contenir graus de certesa que ponderin la imprecisió de la informació obtinguda, així com la incertesa del coneixement enmagatzemat.

La major part de les aproximacions que s'han realitzat fins avui dia han estat a prop de la lògica booleana i de la teoria de les probabilitats.

Les aproximacions numèriques a la representació del coneixement incert presenten importants problemes entre els quals destaquen:

- 1.- Precisió forçada de l'expert i dels usuaris.
- 2.- Allunyament del llenguatge usual dels experts.
- 3.- Hipòtesi d'independència i exclusió mutua.
- 4.- Inconsistència dels experts i usuaris al donar valors numèrics

### 3.2.1. Termes lingüístics com a valors de certesa

Una aproximació basada en la teoria de possibilitats sembla la més adient per modelitzar la incertesa expressada lingüísticament. Els graus de certesa seran en aquesta aproximació termes lingüístics. Els termes lingüístics són depenents de l'aplicació i definits per l'expert. Aquests termes formen una escala totalment ordenada que serà la utilitzada per l'expert per desenvolupar el sistema i pels usuaris per a avaluar els fets de tipus difús. L'ús dels termes lingüístics ha estat destacat per diferents autors com a solució a les inconsistències sobre les ponderacions numèriques, inclús de situacions quotidianes, que els éssers humans cometem [Freksa i López de Mántaras, 1984].

La definició d'aquesta escala pot fer-se, bé per criteri de l'expert unicament, bé per algun tipus de consens entre diferents experts de l'àrea. En l'aplicació PNEUMONIA hem estat treballant amb l'escala proporcionada per l'expert, si bé s'ha realitzat un estudi entre els metges especialistes en medicina interna de l'àrea metropolitana de Barcelona, per tal d'obtenir un conjunt de termes comú (veure apèndix III), i comprovar si l'escala de l'expert era prou propera a la dels seus col·legues de l'entorn.

Les aplicacions desenvolupades fins ara amb MILORD utilitzen els següents

nombre de termes:

<u>APLICACIO</u>	<u>TERMES</u>
PNEUMON-IA	9
RENOIR	9
VLSI	5
SOLAR	5

Fig 3.6 Nombre de termes lingüístics utilitzats a les diferents aplicacions desenvolupades utilitzant MILORD.

El nombre de termes té variabilitat entre aplicacions però el nombre de termes cau dins els marges que els estudis psicològics de Beyth-Marom [Beyth-Marom, 1982] indiquen com a conjunts utilitzats pels experts a l'emetre els seus judicis : al voltant de 7 termes, més exactament  $7 \pm 2$ .

### 3.2.2. Semàntica dels termes lingüístics

S'han utilitzat diferents semàntiques associades als termes lingüístics dins l'arquitectura de MILORD: representar els termes com a intervals difusos, com a elements d'una lògica multivaluada i més recentment com a etiquetes lingüístiques. Farem aquí un repàs a les dues primeres [López de Mántaras et al, 1987], [Godo et al, 1987], [Godo et al, 1988], [Godo i Sierra, 1988], que són les que actualment estan completament implantades, i a les característiques de l'arquitectura per facilitar la definició i el manipulació eficient d'aquests conjunts de termes. La tercera semàntica pot trobar-se a la tesi en preparació de L. Godo al CEAB.

#### 3.2.2.1. Representació mitjançant intervals difusos

Una quantitat difusa és un conjunt difús sobre els reals. Es a dir una funció  $\mu_Q : \mathbb{R} \rightarrow [0, 1]$ .  $\mu_Q$  és vista com una distribució de possibilitat sobre els valors que pot prendre una variable. El *nucli* d'una quantitat difusa correspon als punts on  $\mu_Q = 1$ . Així si el nucli consisteix en un únic punt ens trobarem davant dels nombres difusos, i en el cas que sigui un interval davant dels intervals difusos. Aquests intervals representen bé

informacions imprecises. Mostren un avantatge respecte els intervals precisos, atès que amb aquests a l'hora d'estimar el valor d'una variable s'ha de ser o bé pessimista i definir intervals amplis o bé optimista i definir intervals restringits. El perill que ambdues opcions comporten a l'hora de fer-hi càlculs són: excessiva imprecisió i precisió il·lusòria respectivament. La representació amb intervals difusos permet una representació pessimista i optimista a l'hora, definint el suport<sup>1</sup> de forma pessimista i el nucli de forma optimista.

En el nostre cas cada terme lingüístic vindrà representat per un interval difús, aproximat, per raons d'eficiència, a un trapezi com es pot veure a la següent figura:

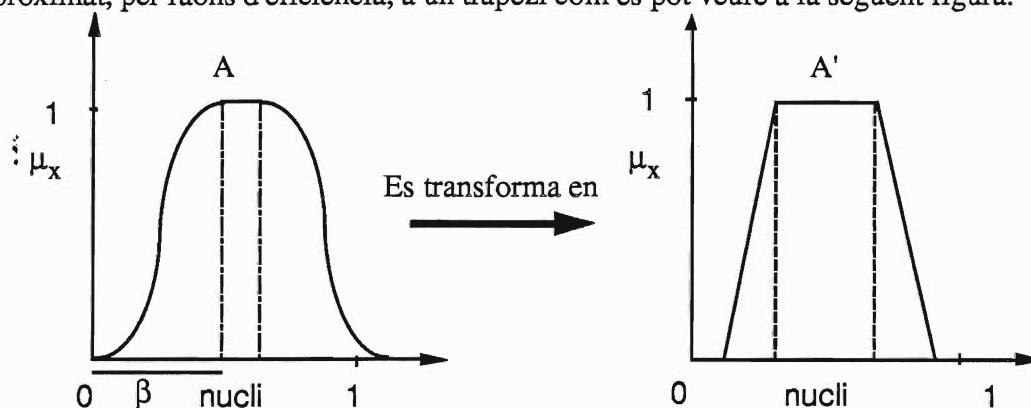


Fig. 3.7. Aproximació d'un interval difús general mitjançant un interval difús de forma trapezoïdal.

Si bé la representació del interval difús seria una funció complexa, la representació del trapezi és únicament la de quatre punts, els punts (a b c d) (veure fig 3.8). Per exemple la funció A (Fig 3.7) que representa un interval difús simètric es caracteritza pels límits de l'interval  $p_i$  i  $p_s$ , on la funció val 1, i la distància entre els extrems del nucli i els punts d'inflexió. La funció de pertinença és del següent tipus:

$$\mu_x(u) = \begin{cases} 0 & u \geq p_s + \beta \text{ o } u \leq p_i + \beta \\ \frac{2}{\beta^2}(u-p_i+\beta)^2 & p_i-\beta \leq u \leq p_i-\beta/2 \\ 1-2/\beta^2(u-p_i)^2 & p_i-\beta/2 \leq u \leq p_i \\ 1 & p_i \leq u \leq p_s \\ 1-2/\beta^2(u-p_s)^2 & p_s \leq u \leq p_s+\beta/2 \\ \frac{2}{\beta^2}(u-p_s-\beta)^2 & p_s+\beta/2 \leq u \leq p_s+\beta \end{cases}$$

<sup>1</sup> Suport(A) = { x :  $\mu_x > 0$  }

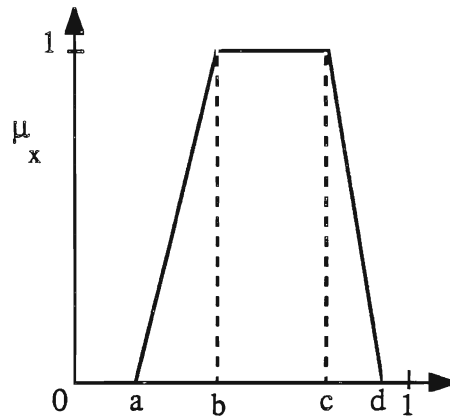


Fig. 3.8 Punts característics de la representació trapezoïdal d'un interval difús.

L'interval  $[0, 1]$  a les abscisses, sobre el qual es defineixen els trapezidis és un interval de valors de certesa. La interpretació semàntica d'aquesta representació es fa sobre la base de l'adequació entre el terme lingüístic representat i els valors de certesa de l'interval. Així la interpretació d'un trapezi  $(a, b, c, d)$  seria:

- 1.- El terme lingüístic és totalment adient per als valors compresos entre  $b$  i  $c$ . Totalment adient vol dir que el valor d'adequació 1.
- 2.- El terme no és gens adient per als valors a l'esquerra de  $a$  o a la dreta de  $d$ . Gens adient vol dir que el valor d'adequació 0.
- 3.- Entre  $a$  i  $b$  i entre  $c$  i  $d$  l'adequació del terme varia de forma lineal creixent i decreixent respectivament.

A sobre d'aquests trapezidis, en tant que intervals difusos, podem aplicar les operacions de l'aritmètica difusa. Aquestes operacions són l'extensió de les T-normes i T-conormes definides sobre l'interval  $[0, 1]$ . D'aquestes operacions podem veure a continuació uns quants exemples:

$$(a, b, c, d) + (a', b', c', d') = (a + a', b + b', c + c', d + d')$$

$$(a, b, c, d) * (a', b', c', d') = (a * a', b * b', c * c', d * d')$$

$$(a, b, c, d) / (a', b', c', d') = (d / a', c / b', b / c', a / d')$$

En termes d'aquestes operacions, el resultat d'aplicar la T-norma  $T_{2,5}$  sobre els trapezidis  $t_1 = (a_1, b_1, c_1, d_1)$  i  $t_2 = (a_2, b_2, c_2, d_2)$  seria

$$T_{2.5}(t_1, t_2) = ( d_1 * d_2 / (a_1 + a_2 + a_1 * a_2), \\ c_1 * c_2 / (b_1 + b_2 + b_1 * b_2), \\ b_1 * b_2 / (c_1 + c_2 + c_1 * c_2), \\ a_1 * a_2 / (c_1 + c_2 + c_1 * c_2))$$

Els termes utilitzats per l'expert són un conjunt finit, i els resultats de les operacions tal com han estat descrites no generen sempre trapezis del conjunt definit per l'expert. Davant d'aquesta situació hi ha dues opcions possibles:

- 1.- Arrossegar el resultat de les operacions aritmètiques, T-normes i T-conormes, al llarg de les cadenes inferencials i aproximar al final a un dels termes del conjunt.
- 2.- Aproximar en cada pas deductiu el resultat de les operacions aritmètiques a un dels termes definits per l'expert.

Ambdues opcions presenten avantatges i inconvenients. La primera té com a avantatge la no introducció d'errors d'aproximació i té com a inconvenient la impossibilitat de precalcular les operacions, essent necessari fer càlculs aritmètics en temps d'execució. La segona introdueix errors d'aproximació, però permet el precàlcul fora de línia. A MILORD hem optat per la segona opció, per les raons que la suporten i a més perquè el que a primera vista són avantatges de la primera opció i desavantatges de l'opció triada no ho són tant, ja que la precisió en els càlculs pot tenir com a conseqüència la relaxació dels trapezis, donant lloc a una imprecisió que els experts, com la nostra experiència constata, no presenten. Segon, si bé es produeixen errors al fer les aproximacions lingüístiques, aquests no són gaire importants si les fondàries dels arbres de deducció no són grans, com és el cas en moltes aplicacions que tenen el coneixement ben estructurat.

Degut a totes aquestes raons hem portat a terme la segona opció, fent una aproximació lingüística basada en els treballs desenvolupats per Piero Bonissone [Bonissone, 1979], [Bonissone i Decker, 1985].

Si  $L$  és el conjunt de termes lingüístics definits per l'expert, i  $t$  el resultat d'una operació entre trapezis, aproximar  $A$  consisteix a trobar el terme  $l \in L$  tal que tingui la similitud semàntica més gran amb  $A$ . Aquesta similitud resulta de la comparació entre les funcions de pertinença dels dos trapezis. Donat que no hi ha una definició precisa de la similitud, l'aproximació ha de trobar-se per procediments ad hoc. Una possible formalització del procés d'aproximació seria la següent:

## Def

$$\text{Aprox}[ A ] = l_{A'}$$

$$\text{on } d(\mu_A, \mu_{A'}) = \min \{ d(\mu_A, \mu_B), \forall l_B \in L \}$$

on A i A' són subconjunts difusos a l'interval [0 1], A és el trapezi sense etiquetar, l\_{A'} és el terme lingüístic associat a A', l\_B és el terme lingüístic associat a B i d és una distància.

Algunes distàncies utilitzades a la bibliografia són les següents:

$$d(\mu_A, \mu_B) = \text{Max}_i (\mu_A(u_i) - \mu_B(u_i))^2$$

les següents distàncies es calculen discretitzant l'interval [0, 1] en D punts. Les dues últimes són distàncies normalitzades.

$$d(\mu_A, \mu_B) = \sum_{i=1}^D (\mu_A(u_i) - \mu_B(u_i))^2$$

$$d(\mu_A, \mu_B) = \frac{1}{D} \sum_{i=1}^D | \mu_A(u_i) - \mu_B(u_i) |$$

$$d(\mu_A, \mu_B) = \sqrt{\frac{1}{D} \sum_{i=1}^D (\mu_A(u_i) - \mu_B(u_i))^2}$$

Per veure una discussió sobre les diferents distàncies veure [Bonissone, 1979]. Les distàncies presentades a dalt presenten diversos problemes que les fan poc aconsellables com a distàncies en un procés d'aproximació lingüístic, i entre els que destaca la necessitat d'un conjunt de termes gran per poder tenir una bona aproximació. Es defineix un espai de patrons en el que la mida dels conjunts de termes pugui ser més baixa, i la informació de cada terme sigui la mínima possible.

Per definir una distància sobre aquest espai de patrons, poden tenir-se en consideració diferents paràmetres dels conjunts difusos. Entre ells destaquen:

- 1.- Potència.
- 2.- Entropia.
- 3.- Primer moment.
- 4.- Biaix.

En el nostre cas, en que tenim conjunts ordenats d'etiquetes lingüístiques, n'hi ha prou amb utilitzar la potència i el primer moment per definir una distància. L'entropia és una mesura del grau de vaguetat del conjunt i és interessant quant els conjunts són solapats, de potència i primer moment semblants però d'entropia diferents. Els termes lingüístics de certesa no presenten aquesta característica. El mateix passa amb el biaix, que no és una característica rellevant en la definició de termes lingüístics.

Els paràmetres utilitzats en la definició de la distància són els següents:

$$\begin{array}{ll}
 \text{Potència} & P(A) = \int \mu_A(u) \, du \\
 \vdots & \\
 \text{Primer moment} & M(A) = \frac{\int x \mu_A(x) \, dx}{\int \mu_A(x) \, dx}
 \end{array}$$

El trapezis queden representats com a punts de l'espai  $R^2$  ( $P(A)$ ,  $M(A)$ ). La distància utilitzada a MILORD és, doncs, una distància normalitzada sobre  $R^2$ :

$$d_\beta(A, B) = d_\beta((P(A), M(A)), (P(B), M(B))) = \beta|P(A) - P(B)| + (1-\beta)|M(A) - M(B)|$$

Estudis empírics desenvolupats per nosaltres sobre el valor del paràmetre  $\beta$  demostren que el seu valor ha de caure dins l'interval  $[0.2, 0.5]$  per a que l'aproximació sigui prou bona en el cas de conjunts d'etiquetes distribuïts al llarg de l'interval  $[0, 1]$ .

En el cas d'una representació trapezoïdal ( $a$   $b$   $c$   $d$ ) la funció de pertinença  $\mu(x)$  és:

$$\mu(x) = \begin{cases} (x-a)/(b-a) & x \in [a \, b] \text{ i } a \neq b \\ 1 & x \in [b \, c] \\ (x-d)/(c-d) & x \in [c \, d] \text{ i } c \neq d \\ 0 & \text{altrament} \end{cases}$$

i els paràmetres tenen els següents valors:

$$P(A) = \frac{(c + d - b - a)}{2}$$

$$M(A) = \begin{cases} \frac{1}{3} \frac{(c^2 + d^2 + cd) - (a^2 + b^2 + ab)}{(c + d) - (a + b)} & c+d \neq a+b \\ a & \text{altrament}^1 \end{cases}$$

L'objectiu d'aquesta aproximació lingüística és el d'obtenir una representació tabular dels connectius utilitzats a l'aplicació. Aquesta taula es pot interpretar com una taula de veritat per una lògica multi-valuada. L'esquema del procés realitzat es pot veure a la següent figura:

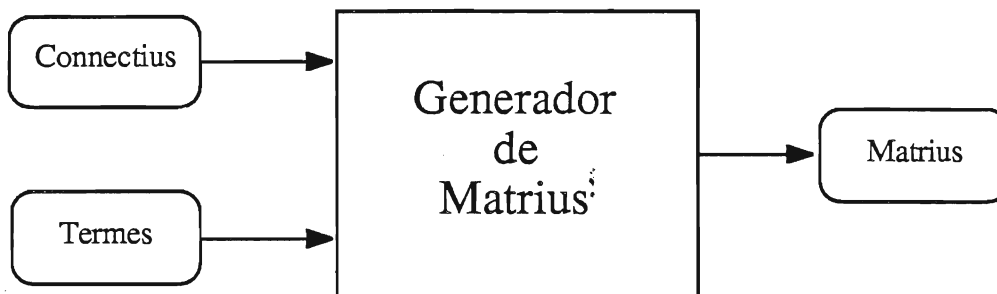


Fig. 3.9 Generació de les matrius que modelitzen els connectius.

El generador combina tots els termes utilitzant les funcions que defineixen els connectius, les aproxima fent servir el mètode explicat anteriorment, i produeix en forma de matrius els operadors que seran utilitzats posteriorment pels mecanismes d'inferència. Un exemple de matriu és :

<sup>1</sup> En el cas límit en que els quatre punts coincideixen ( $x x x x$ ) la funció trapezoïdal representa un valor de veritat puntual no difús. En aquest cas la potència és zero  $P(A) = 0$  i el primer moment es pot obtenir com a límit, per exemple, dels moments de funcions trapezoïdals del tipus: ( $a a x x$ ) quan  $a$  tendeix a  $x$ . essent el seu valor  $M(A) = x$ .



1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---

1	2	3	4	5	6	7	8	9
2	3	3	4	5	6	7	8	9
3	3	4	4	5	6	7	8	9
4	4	4	5	6	6	7	8	9
5	5	5	6	6	7	7	8	9
6	6	6	6	7	7	8	8	9
7	7	7	7	7	8	8	9	9
8	8	8	8	8	8	9	9	9
9	9	9	9	9	9	9	9	9

1	Gens possible
2	Molt poc possible
3	Lleugerament possible
4	Moderadament possible
5	Possible
6	Força possible
7	Molt possible
8	Pràcticament segur
9	Segur

Fig 3.10 Matriu generada per la T-conorma probabilista  $S_2$  amb  $\beta = 0.5$ 

### 3.2.2.2. Representació com elements d'una lògica multi-valuada

Si bé la representació dels termes lingüístics de certesa mitjançant conjunts difusos té una interpretació clara, es pot criticar que no és en absolut evident que les persones utilitzin una representació concreta pels termes lingüístics, i en tot cas resulta difícil provar que els trapezis siguin la millor representació. El que sembla més plausible és que les persones, els experts en el nostre cas, utilitzin els termes com a tals, i que els combinin segons uns esquemes determinats. De ser així l'ús de trapezis i de funcions de combinació s'hauria d'interpretar com un intent d'aproximar analíticament els processos cognitius desenvolupats pels experts.

Hi ha altres dificultats que es presenten a l'enginyer de coneixements quan tracta de representar el significat dels termes lingüístics:

- \* Sovint l'expert no es capaç de definir exactament el significat dels termes utilitzant una escala numèrica.
- \* El experts no demostren un alt grau d'acord sobre la representació d'un terme concret. [Kong et al, 1986] (veure apèndix III).

Si considerem els termes com a símbols d'una lògica multi-valuada, moltes de les dificultats desapareixen, conservant-se els elements de flexibilitat que proporciona l'aproximació utilitzant trapezis. Considerant els termes com a elements ordenats d'un

conjunt, i exigint certes propietats als operadors de combinació dels termes obtindrem resultats semblants als obtinguts en la representació trapezoïdal, però sense els problemes metodològics que aquella pugui presentar.

Anomenarem  $L$  al conjunt ordenat de termes lingüístics  $L = \{L_1, \dots, L_n\}$  acomplint-se  $L_i \leq L_j$  si  $i \leq j$ .

La cerca d'operadors de negació forta és simple seguint els resultats de [Esteva, 1981]. Les propietats a acomplir-se són com ja s'ha comentat:

$$\begin{array}{ll} \text{Decreixent monòtonament} & N(L_i) \geq N(L_j), \text{ si } i \leq j \text{ i, } j = 1, \dots, n \\ \text{Involució} & N(N(L_i)) = L_i, \text{ i } 1 \dots n \end{array}$$

i lúnica funció que les compleix (en el cas d'una lògica multi-valuada) és :

$$N(L_i) = L_k = L_{n+1-i}$$

Per exemple sigui  $L = \{fals, lleugerament\_possible, possible, molt\_possible, cert\}$  Llavors  $N(lleugerament\_possible) = molt\_possible$ ,  $N(cert) = fals$ ,  $N(possible) = possible$ .

És a la definició dels operadors binaris on trobem més possibilitats de joc. Exemplem la construcció dels operadors amb el cas dels connectius  $i$ , la construcció dels connectius  $o$  és dual.

Exigirem dels connectius  $i$  dos tipus de propietats:

- i.- Propietats que satisfan les T-normes
- ii.- Altres propietats desitjables

Les propietats de les T-normes traslladades al nostre cas són:

$$\begin{array}{lll} \text{Associativitat} & t(t(L_i, L_j), L_k) = t(L_i, t(L_j, L_k)) & i, j, k = 1..n \\ \text{Commutativitat} & t(L_i, L_j) = t(L_j, L_i) & i, j = 1..n \\ \text{Absorció} & t(L_1, L_i) = L_1 & i = 1..n \\ \text{Element Neutre} & t(L_n, L_i) = L_i & i = 1..n \\ \text{Monòtonament creixent} & t(L_i, L_j) \geq t(L_{i-1}, L_j) & i = 2..n \text{ j} = 2..n \end{array}$$

Altres propietats desitjables són:

$$\begin{array}{ll}
 \text{Operadors estrictes} & t(L_i, L_j) \neq L_1 \quad i, j \neq 1 \\
 \text{Suavitat}_\alpha & \text{si } t(L_i, L_j) = L_k, t(L_{i-1}, L_j) = L_p, i \\
 & t(L_i, L_{j-1}) = L_q \\
 & \text{s'ha d'acomplir donada } \alpha \in \mathbb{N}: \\
 & k - \alpha \leq p, q \leq k
 \end{array}$$

Aquestes propietats restringeixen el conjunt inicial de funcions  $f: L \times L \rightarrow L$ . El nombre total de funcions sense restriccions és de  $n^2$ , essent  $n$  el nombre de termes lingüístics. Amb aquest conjunt de restriccions el nombre de funcions restants queda notablement reduït [Godo i Sierra, 1988].

n	A	B
3	2	1
4	7	2
5	42	7
6	429	41

A: Propietats de les T-normes.

B: Propietats de les T-normes + Propietats desitjables.  
 $\alpha = 2$

Fig 3.11 Nombre d'operadors dependent de les propietats satisfetes.

A més de les propietats necessàries a complir i de les propietats desitjables, es poden aplicar tècniques de generació que redueixen encara més el nombre de funcions restants. Una tècnica força interessant és la de la *generació incremental*, consistent a utilitzar funcions d'ordre inferior com a plantilla per generar funcions d'ordre superior. Així, per exemple, utilitzar una matriu d'ordre 5, i intercalant-hi dos termes al conjunt existent, generar la matriu d'ordre 7 respectant els valors de la matriu d'ordre 5. Aquesta tècnica és interessant en les primeres fases de desenvolupament d'aplicacions, quan l'expert no té encara ben definit el conjunt de termes a utilitzar. Ampliar el conjunt de termes no representa doncs cap canvi en l'esquema previ de combinació d'evidències.

Un cop reduït el conjunt de matrius a un nombre raonable apliquem dos mètodes que redueixen encara més el nombre de matrius:

- 1) *Classificació de les matrius en classes significativament diferents*, i presentació dels prototipus de les classes als experts. La classificació l'hem realitzat a partir dels algorismes descrits a [Aguilar i Piera, 1987], [Desroches, 1987]. Hem comprovat que hi ha una gran correlació entre el nombre de classes significativament diferents generades per aquest mètode i les obtingudes a partir de la representació trapezoïdal utilitzant aproximacions lingüístiques [Bonissone, 1979, 1985, 1986].
- 2) *Assignació de valors a punts de la matriu*. Aquestes assignacions són restriccions addicionals sobre el conjunt de matrius, reduint-ne, per tant, el seu nombre. La metodologia utilitzada per fixar aquests punts consta de dues fases:
  - a.- Crear un arbre discriminant considerant els punt a fixar com a atributs de classificació [Quinlan, 1986]. D'aquesta manera podrem minimitzar el nombre de punts a fixar per part de l'expert.
  - b.- Presentar exemples del domini de l'expert per tal de que la fixació sigui més realista. Si bé actualment la selecció dels exemples s'ha de realitzar mitjançant converses entre l'enginyer de coneixements i l'expert, en un futur es realitzarà utilitzant tècniques d'adquisició de coneixements [Plaza, 1987] dins el marc del projecte DRUMS, acció bàsica de recerca de ESPRIT.

Aquesta segona metodologia ha estat utilitzada en el cas de PNEUMON-IA i s'han obtingut els següents punts fixats per l'expert:

*Punts fixats*

MOD-P	I	MOD-P	=	LLE-P
FOR-P	I	FOR-P	=	POSSI
MOD-P	I	LLE-P	=	MPO-P
LLE-P	I	LLE-P	=	MPO-P
MOL-P	I	PRA-S	=	MOL-P
MOL-P	I	FOR-P	=	FOR-P

on

GEN-P	=	Gens possible
MPO-P	=	Molt poc possible

LLE-P	=	Lleugerament possible
MOD-P	=	Moderadament possible
POSSI	=	Possible
FOR-P	=	Força possible
MOL-P	=	Molt possible
PRA-S	=	Pràcticament segur
SEGUR	=	Segur

La matriu generada segons aquesta aproximació va resultar ser pràcticament la mateixa que la generada amb els trapezis i l'aproximació lingüística. És un resultat que es podia esperar. No obstant l'expert va preferir aquest segon mètode atès que li era més còmode pensar en exemples de domini que no en funcions de combinació sobre les quals va haver de ser instruït i a més va haver d'experimentar fins trobar aquella que li era més propera.

Com es veurà més endavant, la sintaxi de definició de la lògica a utilitzar permet seleccionar entre els dos mètodes. En el futur s'adaptarà la sintaxi a un tercer esquema desenvolupat a la tesi de Lluís Godo.

### 3.2.2.3 Modus ponens a MILORD

Descriurem aquí com es realitza el modus ponens a MILORD. Sigui la regla general:

**Si  $C_1$  i  $C_2$  i ... i  $C_n$  Llavors concloure q és  $V_r$**

llavors el procés d'inferència que proporciona l'actualització del valor de certesa de la conclusió q a partir de les certes de les premisses de la regla es fa seguint el següent algorisme:

- 1.- Obtenir els valors de certesa de les condicions  $C_1, \dots, C_n$ . Siguin aquests  $V_1, \dots, V_n$ .
- 2.- Combinar els valors de certesa de les condicions per accessos successius a la matriu que modelitza el connectiu i. Anomenada TN.

$$V_{\text{combinat}} = \text{TN}(V_1, \text{TN}(V_2, \text{TN}(\dots(V_{n-1}, V_n) \dots)))$$

- 3.- Combinar el valor  $V_{\text{combinat}}$  amb el valor de certesa de la regla utilitzant la matriu que modelitza el modus ponens. Anomenada MP.

$$V_{\text{propagat}} = \text{MP}(V_{\text{combinat}}, V_r)$$

En els casos en què la funció de modus ponens es modelitza mitjançant la mateixa t-norma del connectiu i (R-implicacions), tindriem

$$V_{\text{propagat}} = \text{TN}(V_{\text{combinat}}, V_r)$$

- 4.- Combinar, si q posseïa un valor de certesa previ a l'aplicació d'aquesta regla, el valor propagat amb el valor anterior del fet q utilitzant la matriu que modelitza el connectiu o. Anomenada CN.

$$V_{q\text{-nou}} = \text{CN}(V_{\text{propagat}}, V_{q\text{-vell}})$$

### 3.2.2.3. Definició dels termes lingüístics

Com s'ha remarcat a l'apartat 3.2.1 on introduïem els termes lingüístics, el nombre de termes varia entre aplicacions, així com la semàntica associada. Es per tant un punt inicial de tota aplicació decidir quin conjunt de termes serà l'utilitzat i quina semàntica hi associarem. Si bé no està clara quina metodologia utilitzar a l'hora d'extreure de l'expert el conjunt de termes i la seva representació, diferents estudis han produït resultats satisfactoris.[Freksa i López de Mántaras, 1984] [Godo i Sierra, 1988]. MILORD no dona cap eina d'adquisició dels termes lingüístics, tot i que proporciona un conjunt de termes per defecte<sup>1</sup> que ha estat consensuat entre més de 25 especialistes mèdics de l'àrea metropolitana de Barcelona (veure apèndix III). Un estudi similar pot trobar-se a [Agustine K. et al, 1986].

La definició dels termes lingüístics té dos components, els noms dels termes i la semàntica associada. En el cas de triar una semàntica basada en els intervals difusos la seva representació seran trapezidis. En cas de la lògica multi-valuada la semàntica serà l'ordre entre ells.

La sintaxi de la definició dels termes és la següent.

<i>termes</i>	::= <b>Termes</b>
	<i>termesid</i>
	<i>semàntica</i>
<i>termesid</i>	::= <i>termeid</i> , <i>termesid</i>   <i>termeid</i>

<sup>1</sup> Aquest conjunt pot ser útil en cas d'aplicacions a la medicina.

*semàntica* ::= Representació trapezoïdal *valorstra* |  
**Representació multivaluada**  
*valorstra* ::= *valortra* | *valortra valorstra*  
*valortra* ::= *termeid* = *real real real real*

Per exemple a l'aplicació PNEUMON-IA la definició de termes és la següent:

### Termes

Impossible, Molt\_poc\_possible, lleugerament\_possible,  
 Moderadament\_possible, Possible, Força\_possible,  
 Molt\_possible, Pràcticament\_segur, segur

#### Representació trapezoïdal

Impossible	= 0 0 0 0
Molt_poc_possible	= 0 0 0.05 0.08
Lleugerament_possible	= 0.05 0.07 0.14 0.17
Moderadament_possible	= 0.10 0.15 0.35 0.45
Possible	= 0.25 0.35 0.55 0.65
Força_possible	= 0.45 0.55 0.75 0.85
Molt_possible	= 0.65 0.75 1 1
Pràcticament_segur	= 0.95 0.98 1 1
Segur	= 1 1 1 1

#### 3.2.2.4. Definició d'una lògica difusa concreta

En l'apartat anterior hem vist com els experts poden definir els termes lingüístics del seu domini. També és necessari definir els operadors que modelitzaran els connectius lògics  $\wedge$  i  $\vee$ . L'elecció dels operadors és de gran importància posat que determinen directament la relació entre els valors de certesa de les dades de l'aplicació i els valors de certesa de les solucions. Uns operadors "conservadors" faran que les certeses resultants puguin ser massa petites, i uns operadors "optimistes" que siguin excessives. Novament ens trobem amb una mancança d'ajudes en la determinació dels operadors òptims per una aplicació determinada, això és així degut fonamentalment a què en un domini d'aplicació es barregen diferents formes de raonament que utilitzen, en conseqüència, diferents connectius. Quan es tracta d'obtenir diagnòstics és més convenient ser optimista en el raonament de manera que no es menyspreï cap solució, mentre que quan

es tracta de recomanar una intervenció quirúrgica important és més prudent ser conservador.

Per tal de copsar aquesta propietat a MILORD hem adoptat una solució definida a partir de dos elements:

- 1) Uns operadors per defecte que s'utilitzaran mentre no es digui el contrari.
- 2) Permetre a les estructures de mòduls (veure cap 4) definir nous operadors vàlids localment en el seu raonament intern.

Aquesta decisió s'ha pres per facilitar que la gestió de la incertesa sigui un element de control del sistema en la línia del sistema MU [Cohen, 1987]. La combinació de l'evidència depèn del contexte on som. Per exemple una tasca de diagnòstic pot necessitar els connectius probabilistes i una tasca de tractament els connectius min-max.

La definició dels connectius per defecte es realitza de forma conjunta respecte a la definició dels termes lingüístics explicats a l'apartat anterior.

<i>lògica-defecte</i>	::= <b>Lògica per defecte</b> <i>termes</i> {ja descrit} <i>connectius</i> <b>fiLògica</b>
<i>connectius</i>	::= <b>Connectius</b> : <i>connec</i>
<i>connec</i>	::= <i>parelles-duals</i>   <i>desc-connec</i>   <i>externs</i>
<i>desc-connec</i>	::= <b>conjunció</b> = <i>funció</i> <sup>1</sup> <b>disjunció</b> = <i>funció</i>
<i>parelles-duals</i> <sup>2</sup>	::= <b>Lukasiewicz</b>   <b>Probabilistes</b>   <b>Zadeh</b>
<i>externs</i>	::= <b>Externs nom-fitxer</b>

Així tindriem per exemple la definició de la lògica per defecte de l'aplicació PNEUMON-IA

<sup>1</sup> Aquestes funcions són expressades en els exemples en un format flexible, en la realitat són expressions LISP.

<sup>2</sup> Aquests noms corresponen a les parelles presentades com a (T<sub>1</sub>, S<sub>1</sub>), (T<sub>2</sub>, S<sub>2</sub>) i (T<sub>3</sub>, S<sub>3</sub>) respectivament.



**Lògica per defecte**

Termes ...

**Connectius : Probabilistes****fiLògica**

Un exemple de definició de connectius podria ser:

**Lògica per defecte**

Termes ...

**Connectius :****conjunció** =  $xy/(x+y-xy)$ **disjunció** =  $(x+y-2xy)/(1-xy)$ **fiLògica**

La definició d'una lògica multi-valuada es faria de la següent manera:

**Lògica per defecte**

Termes ...

**Connectius : Externs "connectius.dat"****fiLògica**

"connectius.dat" és un fitxer que conté els operadors generats pel mètode descrit a l'apartat 3.2.2.2.

La definició de la lògica per defecte ha de ser prèvia a la definició sintàctica del nivell estructural i superiors, de manera que els termes lingüístics siguin coneguts pel compilador amb anterioritat a la compilació de les regles i metaregles que faran ús del conjunt de termes.

**3.2.3. Anàlisi de sensibilitat**

Un aspecte interessant d'estudi en els sistemes de raonament aproximat és la sensibilitat de les conclusions davant modificacions dels valors de certesa de les premisses. Això és important per dues raons fonamentals: permet establir els algorismes bàsics per fer un sistema de manteniment de la consistència, i dóna heurístiques de control sobre la informació més rellevant a recaptar per assolir una hipòtesi.

En els sistemes de manteniment de la consistència són necessaris algorismes eficients que actualitzin les certes dels fets de la BC a través de les unitats inferencials

que els relacionen. Anàlisis de sensibilitat que ens donin una idea sobre l'efecte de les modificacions en els fets a través de la xarxa, ens poden estalviar un gran nombre de càlculs innecessaris.

Un element important en el disseny d'algorismes de cerca és la decisió sobre quina serà la següent pregunta sobre la qual es focalitzarà el sistema, o bé quin fet és més interessant conèixer (recomanació de proves en sistemes mèdics). Hi ha diversos criteris que poden ajudar en la decisió:

- \* *Criteris de cost.*
- \* *Criteris de rellevància.*
- \* *Criteris de rendiment.*
- \* *Criteris de sensibilitat.*

Els primers tres tipus de criteris són de caracter semàntic: és l'expert qui els ha d'omplir de contingut, els de l'últim tipus en canvi són criteris de caracter sintàctic: s'obtenen estudiant únicament l'aspecte sintàctic del conjunt d'unitats inferencials. És per això que hem decidit dotar al sistema de certes eines d'anàlisi de sensibilitat que puguin ser utilitzades per l'usuari. Els criteris semàntics poden ser modelitzats utilitzant les eines de control que proporciona MILORD.

### 3.2.3.1. Criteris de cost

La informació demanada a l'usuari ha de ser la menys costosa possible segons criteris econòmics o socials. En les aplicacions mèdiques s'han d'evitar les anàlisis massa cares o molestes pel malalt. Els criteris de cost poden ser representats mitjançant metaregles que ens adrecin a mòduls on es recapta aquesta informació només en els casos en que sembli necessari fer-ho.

Exemple: En el cas d'un malalt amb pneumònia i SIDA, es poden plantejar dues opcions de realització de proves en cas de produir-se expectoració: Fer una anàlisi de l'esput, o fer un raspallat bronquioalveolar. La segona prova és més cara i molt més molesta pel malalt. És més interessant inicialment demanar la primera prova i realitzar la segona únicament en cas que la primera fracassés i no aportés informació concluent. L'esquema de recomanacions és:

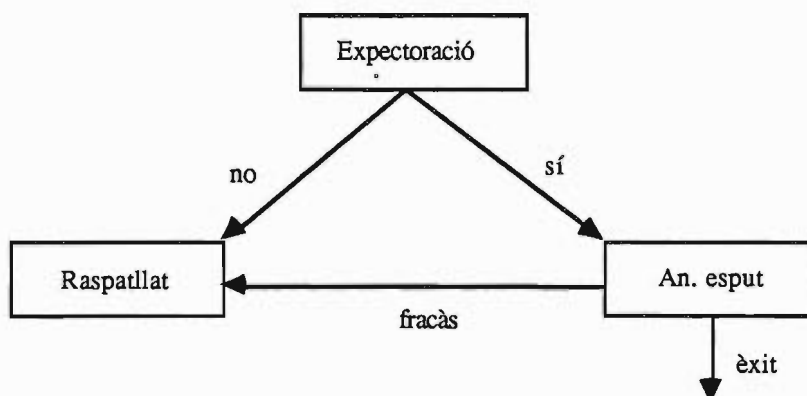


Fig 3.12 Recomanació de proves en el cas d'una SIDA

i les metaregles d'estratègia (veure 5.8.2) són

M03006 Si Expectoració i SIDA Llavors

**Moduls a visitar** Anàlisi\_esput

M03007 Si Expectoració i SIDA i Fracàs\_anàlisi\_esput

**Llavors Moduls a visitar** Raspatllat\_bronquioalveolar

M03004 Si no expectoració i SIDA

**Llavors Moduls a visitar** Raspatllat\_bronquioalveolar

### 3.2.3.2. Criteris de rellevància

Una informació molt rellevant pot fer-nos canviar d'estratègia de resolució, i per tant, millorar l'eficiència del sistema. A MILORD el mecanisme és que tota regla que tingui un grau de certesa més alt dins el conjunt conflicte serà tractada abans. Els criteris d'alta rellevància acostumen a ser inclosos en regles amb un alt grau de certesa. Tanmateix, la informació altament rellevant es pot utilitzar com a premisses de metaregles que ens modifiquin l'estratègia en curs.

### 3.2.3.3. Criteris de rendiment

La informació més general ha de ser demanada primer per tal de descartar subconjunts d'informació. Per exemple, és millor demanar si un malalt pateix d'algun tipus d'inmunodepressió abans de demanar si el malalt pateix d'una SIDA. Una resposta negativa a la primera pregunta eliminaria moltes de les possibles preguntes.

L'estructuració de forma jeràrquica dels conceptes als GRF proporciona el mecanisme desitjat. Tota precedència a l'hora de formular preguntes serà detallada a la xarxa per l'expert. (veure 2.3.1)

### 3.2.3.4. Criteris de sensibilitat

Aquella informació que suposadament provoqui una variació més gran de la certesa de l'objectiu és la que serà obtinguda primer. L'usuari del sistema podrà interaccionar amb el sistema per obtenir informació sobre les variacions de la solució que es puguin produir davant de variacions dels fets d'entrada. Aquesta anàlisi es realitza un cop el procés ha finalitzat<sup>1</sup>. Així un cop obtinguda una solució podem demanar quina prova o quin fet desconegut en aquell moment ens modificaria més notablement la solució obtinguda.

El tipus de preguntes que es permeten a l'usuari són les següents:

- 1.- Com variaria la solució si s'incrementés el valor del fet *fet-id* fins *terme* ?
- 2.- Com variaria la solució si es conegués un valor per al fet *fet-id*, i aquest valor fos *valor* ?
- 3.- Quina és la variació màxima admissible del valor del fet *fet-id* per mantenir-se inalterada la solució actual ?

Exemple: Considerem el següent conjunt de regles referents al diagnòstic de micoplasma:

R01001 **Si no vell i atípica Llavors concloure micoplasma és possible**

R01002 **Si conviu\_amb\_altres\_malalts\_pneumònia**

**Llavors concloure micoplasma és possible**

R01003 **Si atípica i otitis Llavors concloure micoplasma és força possible**

R01004 **Si crioaglutinines < (1 / 64)**

**Llavors concloure micoplasma és molt possible**

i considerem el cas d'un malalt jove amb una sospita de *força possible* que la malaltia sigui atípica i presència d'otitis. En aquesta situació les regles R01001 i R01003 donarien conjuntament el valor: *força possible*.

Veiem ara les respostes a les següents preguntes:

- 1.- Com variaria la solució si s'incrementés el valor de *atípica* fins *molt possible*?  
El valor passaria de *força possible* a *molt possible*.
- 2.- Com variaria la solució si es conegués un valor per al fet *crioaglutinines*, i aquest valor fos (1 / 256) ?  
El valor passaria de *força possible* a *molt possible*.

---

<sup>1</sup> Pot realitzar-se també al llarg del procés d'interacció. Ara bé, en general aquests criteris són útils de cara a comprovar l'estabilitat d'una solució. Es a dir, quines són les modificacions de les dades d'entrada que ens farien canviar de solució al problema.

3.- Com variaria la solució si es conegués un valor per al fet *conviu\_amb\_altres\_malalts\_pneumònia*, i aquest valor fos *cert* ?

El valor passaria de *força possible* a *molt possible*.

4.- Quina és la variació màxima admissible del valor del fet *atípica* per mantenir-se inalterada la solució actual ?

Qualsevol variació del fet *atípica* modificaria el valor final de *micoplasma*.

### 3.3. Coneixements incomplets. Subsumpció.

Un dels fenòmens més comuns en aplicacions com la medicina és la incompletesa de les dades: hi ha resultats de proves no disponibles encara, el malalt està en coma i no pot respondre preguntes, etc. Degut a això els experts tenen diferents nivells de precisió a l'emetre els diagnòstics o a l'establir estratègies depenent de la quantitat d'informació disponible. S'intenta aplicar sempre el nivell que utilitzi el màxim d'informació disponible. Passem a definir la subsumpció tenint presents les definicions dels Grafs dirigits de Relacions Funcionals.

**Def** Direm que una regla  $r_1$  és més específica que una altra  $r_2$  respecte a un graf actiu  $G$  si i només si

1.-  $\text{fet}(\text{conclusió}(r_1)) = \text{fet}(\text{conclusió}(r_2))$

2.-  $E_G(\text{fets}(\text{premissa}(r_1)), \text{fets}(\text{premissa}(r_2)))$

i en aquest cas direm que la regla  $r_2$  **subsumeix** la regla  $r_1$ .

Així doncs sobre el conjunt de regles que dedueixen un fet concret la relació subsumeix serà una relació d'ordre estricte com ja es va demostrar. A més aquesta relació no és estàtica, sinó que dinàmicament s'ha d'actualitzar segons avança el coneixement de les dades del cas i varia el graf actiu. L'explicació de la relació d'especificitat entre els fets es pot trobar a l'apartat 2.3.4. La seva més gran utilitat és la que s'explica aquí: permetre definir el criteri de subsumpció des d'un punt de vista semàntic, és a dir, tenint en compte les relacions inclusives entre els fets de les regles.

Aquesta característica, aplicar abans les regles més específiques, ens va obligar a establir els següents mecanismes de control:

1.- S'intentarà aplicar sempre la regla més específica.

2.- S'intentarà aplicar sempre la metaregla més específica.

- 3.- S'inhibiran les regles més generals que una aplicada.
- 4.- S'inhibiran les metaregles més generals que una aplicada.

Vegi's les següents regles que presenten una subsumpció únicament sintàctica (no hi ha inclusió dels fets a través del GRF).

**R02006 Si Alcohòlic i bacteriana Llavors**

**concloure pneumococ és força\_possible**

**R02007 Si Alcohòlic i bacteriana i cavitada Llavors**

**concloure pneumococ és moderadament\_possible**

**R06008 Si Alcohòlic i bacteriana Llavors**

**concloure BGN és moderadament\_possible**

**R06009 Si Alcohòlic i bacteriana i cavitada Llavors**

**concloure BGN és força\_possible**

Entre aquestes regles es compleix:

R06008 Subsumeix R06009, i

R02006 subsumeix R02007

*Nota bene:* els graus de certesa no són necessàriament creixents amb l'augment de l'especificitat d'una premissa, segons fan patent les dues primeres regles.

Amb aquest mecanisme l'expert s'assegura que mai dues regles que mantinguin una relació de subsumpció seran aplicades al llarg del mateix cas i que el coneixement més específic serà utilitzat sempre que es pugui.

# 4 Nivell estructural. Modularització dels coneixements del domini

FACILUS PER PARTES IN COGNITIONEM TOTIUS ADDUCIMUR

*Sèneca*

## 4.1. Introducció

En aquest capítol es presenta el nivell estructural de l'arquitectura de MILORD, fonamentada en una representació modular dels coneixements de domini. La tècnica de modularització que es descriurà és utilitzable per qualsevol llenguatge basat en regles, posat que l'estructura de mòduls resultant és pràcticament independent del llenguatge de base. Es descriuran, tanmateix, detalls del procés de construcció del compilador que transforma els mòduls del nivell estructural en mòduls no estructurats com els descrits al capítol 2.

La proposta d'aquest nivell de modularització intenta cobrir la mancança de primitives d'estructuració adients que la majoria de llenguatges basats en regles pateix. Els nivells de modularització d'aquests llenguatges acostumen a consistir en mòduls no estructurats semblants als descrits al capítol 2, com per exemple els *rule-sets* a KEE. Si tenim en compte que un dels problemes greus dels sistemes basats en coneixements és el manteniment i desenvolupament d'aplicacions de gran extensió, els ajuts a l'estructuració del coneixement semblen imprescindibles.

Les facilitats d'estructuració dels llenguatges de desenvolupament de sistemes basats en coneixements van des d'afegir elements únicament sintàctics al llenguatge de regles de base, fins a l'ús dels llenguatges orientats objecte [Shriver i Wegner, 1987].

La nostra proposta, [Agustí i Sierra, 1989], està basada en el sistema de mòduls del llenguatge funcional ML [Harper et al, 1986]. Altres treballs han utilitzat

aproximacions semblants en la modularització de llenguatges funcionals i lògics [O'Keefe, 1985], [Miller, 1986], [Sannella i Wallen, 1987], [Akkermans et al, 1989]. Aquesta aproximació ens permet conservar els elements de base del nostre llenguatge de regles: lògica difusa, motors d'ordre 0+, etc., i permeten al mateix temps experimentar en els conceptes de la programació orientada objecte.

La nostra aproximació no interpreta semanticament els mòduls, com es fa en [Sticklen et al, 1987] on els mòduls són anomenats "Knowledge groups" i són entesos com a "especialistes" en resoldre una tasca de solució de problemes, depenent estretament la seva estructuració de la gestió de la incertesa desitjada en cada grup. En la nostra aproximació un mòdul o submòdul s'ha d'entendre com una abstracció funcional. És a dir, com un tros de programa que ha estat abstret del seu contexte especificant-ne:

- 1) Els mòduls dels quals el mòdul abstret pot dependre. Interfase d'importació o requeriments.
- 2) Els resultats del mòdul utilitzables per la resta del programa. Interfase d'exportació.

El mecanisme de modularització que presentarem permetrà definir especificacions que representaran els components de visibilitat, interfícies d'importació i exportació, dels mòduls, i que permetran comprovacions de tipus. Tanmateix, es definiran mòduls paramètrics que doten al sistema d'una potència de representació molt superior a l'obtinguda amb els mòduls no estructurats.

L'interés més gran de la tècnica que proposarem és la seva reutilització sobre altres llenguatges basats en regles a un cost molt baix. Degut a això imposablem la restricció de separar el llenguatge de modularització del llenguatge de regles, així doncs, els mòduls no podran ser creats per les regles. Aquesta visió estàtica dels mòduls fa que els algorismes d'avaluació del llenguatge de base no hagin de ser modificats. Els motors, els sistemes de manteniment de la veritat, etc., d'un llenguatge que incorporés mòduls dinàmics, creats en temps d'execució, haurien de ser modificats substancialment.

Entre les motivacions per modularitzar les bases de coneixements podem destacar:

- 1) reduir la dificultat de disseny, verificació i manteniment de les BC.
- 2) disposar d'espais de noms diferents per cada mòdul, de forma que les interaccions per efectes laterals desapareguin.

Al llarg d'aquest capítol utilitzarem un exemple, que farà de fil conductor de l'explicació, extret del domini d'aplicació de PNEUMON-IA. L'aplicació PNEUMON-



IA, no ha utilitzat en el seu desenvolupament les primitives de modularització, atès que aquestes han estat incorporades a MILORD amb posterioritat al seu desenvolupament.

Al capítol 2 hem introduït la sintaxi dels mòduls no estructurats, que reproduïm a continuació:

```

Nivell estructural ::= Nivell Estructural
                    unitat-estructural
                    finivell
unitat-estructural ::= Mòdul idmodul
                    objectius regles metaregles |
                    unitat-estructural unitat-estructural
objectius          ::= Objectius: lobjectius
lobjectius        ::= fetid, lobjectius | fetid
regles            ::= Regles: lregles
lregles           ::= regla lregles | regla
metaregles        ::= Metaregles: lmetaregles
lmetaregles       ::= metaregla lmetaregles | λ
    
```

Cal remarcar que aquests mòduls no estructurats són les unitats d'estratègia, i les unitats d'execució. El mòduls estructurats hauran de transformar-se doncs en mòduls no estructurats per executar-se. El procés pot observar-se en la següent figura.

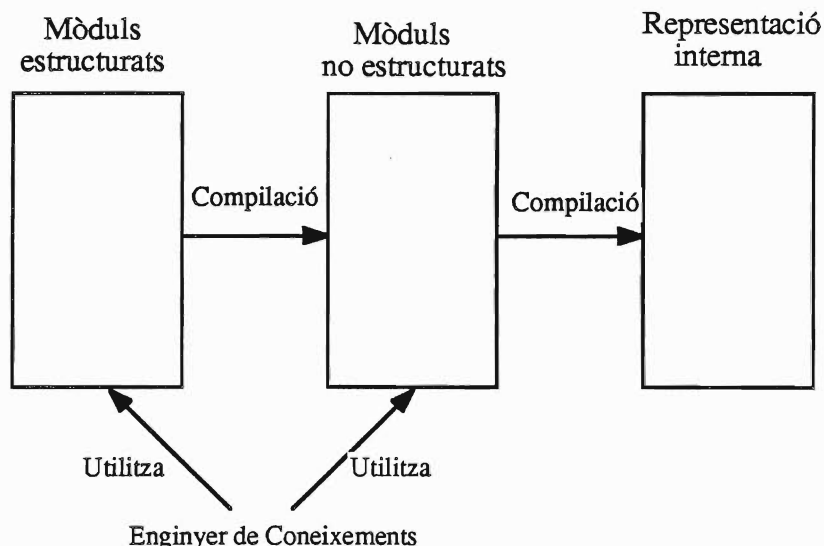


Fig 4.1 Compilació dels mòduls estructurats.

Aquesta dualitat de representació del nivell estructural, permet que les aplicacions desenvolupades fins ara sense modularització continuïn sent executables, i al mateix temps dóna a l'enginyer de coneixements dos esquemes on triar, que poden ésser utilitzats indistintament per diferents parts del programa.

## 4.2. Sintàxi del llenguatge de mòduls

Presentem aquí la sintaxi del nivell estructural utilitzant mòduls, i especificacions (quan es defineixi el nivell estructural cal decidir quin tipus de mòduls s'utilitzarà, si aquests o els mòduls no estructurats):

$$\begin{array}{l} \text{nivell estructural} \\ \text{declmod} \\ \text{vinclespec} \\ \text{vinclemod} \\ \text{listaparam} \\ \text{eqcamí} \\ \text{vincleconj} \\ \text{operador\_conjunt} \\ \text{exprespec} \\ \text{espec} \\ \text{expremod} \end{array} ::= \begin{array}{l} \text{Nivell estructural} \\ \text{declmod} \\ \text{finivell} \\ \text{Especificació vinclespec} \mid \\ \text{Mòdul vinclemod} \mid \\ \text{Conjunt vincleconj} \mid \\ \text{declmod declmod} \\ \{ \text{Heredar } id \text{ és equivalent a } \text{mòdul } id = id \} \\ idespec = exprespec \\ idmod[(l\text{listaparam})][: exprespec] = expremod \\ pid : exprespec [\text{compartir eqcamí}]; l\text{listaparam} \mid \lambda \\ id_1 = \dots = id_n \quad n > 1 \\ \{ idmod [(l\text{listaparam})]: exprespec = expremod \text{ és equivalent a } idmod \\ [(l\text{listaparam})] = expremod : exprespec \} \\ idconjunt = (fet_1 : tipus_1 ; \dots ; fet_n : tipus_n) \mid \\ idconjunt \text{ operador\_conjunt } idconjunt \\ \cup \mid \cap \mid / \\ idespec \mid \text{inici espec final} \\ \text{Mòdul } idmod : exprespec \mid \\ importats \mid \\ exportats \mid \\ espec espec \\ idcamí [( expremod_1 ; \dots ; expremod_n)] \mid \\ \text{inici decl final} \mid \end{array}$$

	<i>expremod</i> : <i>exprespec</i>
<i>idcamí</i>	::= <i>idmod</i>   <i>idmod</i> -> <i>idcamí</i>
<i>decl</i>	::= [ <i>capçal</i> ] <i>cos</i>
<i>capçal</i>	::= <i>declnucli</i>   <i>declmod</i>   <i>declmod declnucli</i>
<i>declnucli</i>	::= <b>Obrir</b> <i>idmod</i>   <i>importats</i>   <i>exportats</i>   <i>declnucli declnucli</i>
<i>importats</i>	::= <b>Importa</b> = ( <i>fet</i> <sub>1</sub> : <i>tipus</i> <sub>1</sub> ; ... ; <i>fet</i> <sub><i>n</i></sub> : <i>tipus</i> <sub><i>n</i></sub> )   <b>Importa</b> = <i>idconjunt</i>
<i>exportats</i>	::= <b>Exporta</b> = ( <i>fet</i> <sub>1</sub> : <i>tipus</i> <sub>1</sub> ; ... ; <i>fet</i> <sub><i>n</i></sub> : <i>tipus</i> <sub><i>n</i></sub> )   <b>Exporta</b> = <i>idconjunt</i>
<i>cos</i>	::= <i>precondició regles metareg</i>
<i>precondició</i>	::= <b>Precondició</b> <i>premissa</i>   $\lambda$
<i>regles</i>	::= <b>Regles</b> <i>lregles</i>   $\lambda$
<i>metareg</i>	::= <b>Metaregles</b> <i>lmrr</i>   $\lambda$
<i>premissa</i>	::= {definit a 2.3.2 }
<i>lregles</i>	::= {definit a 2.3.2 }
<i>lmrr</i>	::= {definit a 2.3.2 }

#### 4.2.1. Declaracions primitives

En els mòduls no estructurats no hi havia restriccions en quant a la visibilitat dels fets entre els mòduls. Aquest punt és un dels essencials en la modularització, com ja s'ha esmentat. Els fets que hagin de ser visibles hauran de ser declarats explícitament, així com els fets que la resta del programa podrà veure del mòdul en curs. Aquest mecanisme permet, doncs, l'amagament d'informació ("information hiding").

Els fets afirmats per l'usuari en temps d'execució són fets que no són exportats per cap mòdul del programes, i reben un tractament diferenciat dels altres. La declaració

**Importa** *fet*<sub>1</sub>, ..., *fet*<sub>*n*</sub>

indica que els fets *fet*<sub>1</sub>, ..., *fet*<sub>*n*</sub> seran visibles dintre del mòdul, i per tant podran formar part de premisses de regles i de metaregles, i que el seu valor serà obtingut en temps d'execució. Els fets declarats com a **Importa** no són exportats fora del mòdul. Si es volgués exportar algun dels fets importats s'hauria d'incloure també a la llista de fets **Exporta**. Obliguem a declarar els fets que s'utilitzaran com a

disciplina del programador. Els fets que realment s'utilitzin hauran de ser un subconjunt dels que es diu que s'utilitzaran.

Cada mòdul posseeix regles que realitzen una tasca concreta, i produeixen uns resultats; aquests resultats quan han de ser visibles fora del mòdul han de ser definits explícitament de la forma:

### **Exporta** *fet<sub>1</sub>, ..., fet<sub>m</sub>*

Tot fet exportat ha de ser deduïble dins el mòdul o bé importat, de no ser així el mòdul es considerarà erroni.

Distingim les declaracions dels fets que són importats, afirmats en temps d'execució, dels exportats degut a la seva natura diferent des del punt de vista de l'abstracció funcional. Al declarar un fet com a exportat indiquem que serà obtingut en el mòdul que el declara exportat, sense entrar en detalls de com serà obtingut. Al declarar un fet importat indiquem que no existeix cap mòdul que, mitjançant una abstracció funcional, l'exporti, i que el mòdul obtindrà aquest valor de la base de fets que existeixi en temps d'execució. Qui "exporta" aquest fet és, en definitiva, l'usuari en temps d'execució.

Les declaracions de regles i de metaregles es mantenen igual que en la declaració de mòduls no estructurats, excepte que la referència als fets té una petita modificació com es veurà en els següents apartats.

### **Precondició** *premissa*

Aquesta primitiva imposa les restriccions per l'execució de les regles i metaregles del mòdul. Si la premissa és falsa cap regla o metaregla del mòdul serà utilitzada, i per tant tots els fets exportats del mòdul tindran com a valor *desconegut*.

### **Regles** *lregles*

Aquesta primitiva és idèntica a la realitzada en els mòduls no estructurats, amb la diferència que la referència als fets ha de tenir en compte el camí d'accés a través dels submòduls.

### **Metaregles** *lmrr*

El comentari és exactament el mateix que en la primitiva precedent.

### 4.2.2. Declaració d'estructures

El llenguatge de modularització permet definir tres tipus diferents d'estructures: conjunts, especificacions i mòduls. Aquestes estructures són objecte d'estudi detallat en els següents paràgrafs.

## 4.3. Conjunts

Atès que és necessari declarar les interfases dels mòduls, i aquestes són conjunts de fets que s'importen i/o s'exporten, sembla útil definir conjunts de fets per tal de facilitar aquestes declaracions. Aquestes declaracions s'han d'entendre pura i simplement com a definicions de macros de lectura. Quan trobem, després de la declaració d'un conjunt, una referència al seu nom, l'efecte resultant serà la substitució del nom pels elements del conjunt. Vegin-se la definició dels següents conjunts de l'exemple sobre *pneumococ*:

**Conjunt** Analítiques\_generals = (exsudat\_pleural: Booleà)

**Conjunt** Tincions = ( gèrmen\_esput: [CGPP, CGPC, BGN, CBGN]; gram: Booleà; nombre\_PMM\_esput\_x100a: Numèric; nombre\_celepi\_esput\_x100a: Numèric; gèrmen\_pleura: [CGPP, CGPC, BGN, CBGN])

**Conjunt** Cultius = (gèrmen\_cultiu\_esput: [ Pneumococ, Klepsiella, Estafilococ, Hemòfilus ]; creixement\_cultiu\_esput: Booleà; gèrmen\_cultiu\_pleura: [ Pneumococ, Klepsiella, Estafilococ, Hemòfilus ]; gèrmen\_cultiu\_sang: [ Pneumococ, Klepsiella, Estafilococ, Hemòfilus ]; cultiu\_pleura+: Booleà; cultiu\_sang+: Booleà)

**Conjunt** Microbiologia = Tincions  $\cup$  Cultius

**Conjunt** Laboratori = Analítiques\_generals  $\cup$  Microbiologia

## 4.4. Especificacions

Una especificació és una descripció de les interfases dels mòduls en funció de que poden veure de la resta del program i de que aporten a la resta de programa. Les especificacions defineixen les relacions entre els mòduls. És obligatori definir les especificacions abans de definir mòduls que segueixin el patró per elles definit. A partir de la definició d'un mòdul es pot inferir quina és l'especificació associada, de fet la tasca que el compilador ha de realitzar és comprovar que l'especificació així inferida correspongui a l'exigida en la declaració del mòdul. Vegin-se els següents exemples:

```

Especificació Pneum =
Inici
  Exporta = (Pneumococ: Difús)
Final

Especificació C_C_R =
Inici
  Mòdul CL : Pneum
  Mòdul RX : Radiol
  Exporta : (Pneumococ : Difús)
final

Especificació Radiol =
Inici
  Mòdul Radiol_favor :
    Inici
      Importa = Radiologia
      Exporta = (Pneumococ_favor : Difús)
    final
  Mòdul Radiol_contra :
    Inici
      Importa = Radiologia
      Exporta = (Pneumococ_contra : Difús)
    final
  Exporta = (Pneumococ : Difús)
final

```

Les especificacions reflecteixen l'estructura de submòduls que tenen els mòduls, per exemple la segona especificació declara que els mòduls que la satisfacin hauran de tenir un submòdul que satisfaci l'especificació *Pneum* i un altre que satisfaci l'especificació *Radiol*.

La declaració d'un submòdul a l'especificació d'un mòdul implica que aquest submòdul serà visible des de fora del mòdul. És a dir, qualsevol mòdul que incorpori un mòdul que segueixi l'especificació *C\_C\_R* veurà un submòdul anomenat *CL* d'especificació *Pneum*.

Si un mòdul definit dins un altre vol ser amagat a l'exterior, la forma de fer-ho és no declarant-ho a l'especificació del mòdul que l'engloba (veure l'apartat ???). En el següent exemple es veu com a l'especificació *Exploració* no s'hi defineix cap submòdul, en canvi en el mòdul *Exp-físiques*, que compleix aquesta especificació, apareix el submòdul *BA = Bacterianicitat*. Això implica que qualsevol mòdul que utilitzi el mòdul *Exp-físiques* no tindria dret d'accés al submòdul *BA*.

```

Especificació Exploració =
Inici
  Importa = Simptomatologia
  Exporta = (Pneumococ: Difús)
final

```

Mòdul Exp-físiques : Exploració =

Inici

Mòdul BA = Bacterianicitat

Regles

R1 Si herpes\_labial i BA->Bacteriana

Llavors Pneumococ és molt\_possible

R2 Si monoartritis\_infecciosa i BA->Bacteriana > possible

Llavors Pneumococ és possible

R3 Si esput = rovellat i BA->Bacteriana > moderadament\_possible

Llavors Pneumococ és molt\_possible

R4 Si períodes\_esgarrifances = 1 i febre > 38.5 i BA->Bacteriana

Llavors Pneumococ és possible

R5 Si artritis i cardinal(articulacions\_afectades)

Llavors monoartritis\_infecciosa és lleugerament\_possible

final

Es pot veure que a partir de la definició d'un mòdul es pot extreure l'especificació del mòdul. El sistema de comprovació de la correcció de la definició d'un mòdul el que fa és un acarament entre l'especificació inferida i l'especificació exigida per aquell mòdul. L'especificació inferida haurà de *respectar* l'especificació exigida en el sentit de la següent definició:

Def una especificació  $S_1$  *respecta* una especificació  $S_2$  si

- 1) Els fets importats per  $S_1$  són un subconjunt dels fets importats per  $S_2$ .
- 2) Els fets exportats per  $S_1$  són un superconjunt dels fets exportats per  $S_2$ .
- 3) Els submòduls exportats per  $S_1$  són un superconjunt dels submòduls exportats per  $S_2$ .

Per tant podem dir que inferida(*Exp-físiques*) *respecta* *Exploració*.

Com a metodologia en el desenvolupament de les BC és molt interessant posseir els mecanismes de control que les especificacions proporcionen sobre el desenvolupament posterior dels mòduls. Inicialment es defineixen els grans blocs de l'aplicació, el compilador comprova que estan ben definits, i posteriorment, a l'hora de la creació de mòduls, el compilador pot assegurar-se que la definició ha seguit els esquemes marcats en les especificacions. Aquesta metodologia, utilitzada en llenguatges com ML, no ha estat utilitzada en els llenguatges de desenvolupament de BC. Creiem que el seu ús pot permetre avançar en el disseny de grans BC sense els enormes problemes de manteniment que actualment presenten.

## 4.5. Mòduls

La definició de mòduls, tal i com es pot veure a la sintaxi, té la forma

$$\text{Mòdul } idmod[(listaparam)][[: \text{exprespec}] = \text{exprmod}$$

on *exprmod* pot ser una d'entre les següents expressions:

- a) Un conjunt encapsulat de declaracions amb un abast limitat.
- b) Un identificador de mòdul previament definit.
- c) Una aplicació d'un mòdul genèric.

### 4.5.1. Declaracions encapsulades

El següent mòdul és un exemple de conjunt encapsulat de declaracions:

**Mòdul** Exp-físiques : Exploració =

**Inici**

**Mòdul** BA = Bacterianicitat

**Regles**

        R1 **Si** herpes\_labial i BA->Bacteriana

**Llavors** Pneumococ és molt\_possible

        R2 **Si** monoartritis\_infecciosa i BA->Bacteriana > possible

**Llavors** Pneumococ és possible

        R3 **Si** esput = rovellat i BA->Bacteriana > moderadament\_possible

**Llavors** Pneumococ és molt\_possible

        R4 **Si** periodes\_esgarrifances = 1 i febre > 38.5 i BA->Bacteriana

**Llavors** Pneumococ és possible

        R5 **Si** artritis i cardinal(articulacions\_affected)

**Llavors** monoartritis\_infecciosa és lleugerament\_possible

**final**

Dins les declaracions hi apareix un submòdul que és igual a un mòdul definit prèviament de nom *Bacterianicitat* i un conjunt de regles. No és necessari declarar els fets importats i exportats, posat que aquests ja ho van ser al moment de definir l'especificació *Exploració*. Si es volguessin escriure haurien de verificar la relació *respectar*, definida a l'apartat anterior, amb l'especificació *Exploració*.

### 4.5.2. Declaracions amb identificadors de mòdul

La declaració de submòdul en el mòdul Exp-físiques de l'apartat anterior és un exemple d'aquest tipus de declaracions. Mitjançant aquest mecanisme es poden construir



bases de coneixements jeràrquiques. És la forma de fer accessibles dins un mòdul tots els fets i submòduls definits dins un altre. Així, els fets definits al mòdul *Bacterianicitat* són accessibles dins el mòdul *Exp-físiques*. La forma d'accedir als fets es fa mitjançant un prefixe que indica el camí d'accés al fet. La declaració:

**Mòdul BA = Bacterianicitat**

a més de declarar que es vol tenir accés a tots els fets definits al mòdul Bact-atip es proporciona el nom que servirà per prefixar-los: BA. Si volguessim utilitzar el mateix nom que quan va ser declarat el mòdul escriuriem:

**Mòdul Bacterianicitat = Bacterianicitat**

o bé el seu equivalent:

**Heredar Bacterianicitat**

Aquesta prefixació serveix per distingir diferents instàncies del mateix fet en mòduls diferents. Així, per exemple la següent definició de mòdul:

**Mòdul Combinació\_clin\_rad : C\_C\_R =**

**Inici**

**Mòdul CL = Clínica**

**Mòdul RX = Radiologia**

**Precondició: CL->Pneumococ**

**Regles**

**R1 Si CL->Pneumococ Llavors Pneumococ és segur**

**R2 Si RX->Pneumococ Llavors Pneumococ és segur**

**final**

*CL* i *RX* fan referència a dos mòduls definits anteriorment i que exporten el fet *Pneumococ*. Com es pot veure la distinció entre el fet *Pneumococ* dels dos mòduls a les regles *R1* i *R2* és possible gràcies a la prefixació diferent d'ambdós fets. En aquest cas els dos fets *Pneumococ* fan referència a la evidència obtinguda a partir de dades de tipus *clínic* i l'evidència a partir de dades de caracter *radiològic*. Aquest mòdul combina les evidències obtingudes dels dos submòduls.

La prefixació es fa escrivint el nom del submòdul que conté el fet seguit de '-' i del nom del fet. Per exemple:

**Mòdul A =**

**Inici**

**Mòdul B =**

**Inici**

**Mòdul C =**

```

Inici
    Exporta fet
    ... {fet s'accedeix aquí com: fet }
    final
    ... {fet s'accedeix aquí com: C->fet }
    final
    ... {fet s'accedeix aquí com: B->C->fet }
final

```

Si es vol tenir accés a tots els fets definits dins un mòdul, exportats i no exportats, i sense haver de prefixar els seus noms es pot utilitzar la primitiva:

### Obrir idmòdul

aquesta primitiva es pot interpretar com un macro de lectura que realitza una còpia de la declaració del mòdul idmòdul dins el cos de definició del mòdul on es troba. Així si ens fixem en el mòdul *Combinació\_clin\_rad* i observem que les regles coincideixen amb les regles que ens proporciona el mòdul paramètric *Combinació\_o* (veure Apèndix V), podríem considerar aquest mòdul, *Combinació\_clin\_rad*, com a extensió de la *combinació\_o* afegint-hi una precondició. De la següent manera:

```
Mòdul Comb_o_clin_rad : Pneum = Combinació_o(Clinica, Radiologia)
```

```
Mòdul Combinació_clin_rad : C_C_R =
```

```
Inici
```

```
    Obrir Comb_o_clin_rad
```

```
    Precondició: CL->Pneumococ
```

```
final
```

Evidentment l'ús d'aquesta primitiva aplicada sobre dos mòduls que definissin el mateix fet faria que aquests dos fets passessin a ser el mateix fet dins el mòdul on s'obrissin. Es a dir, no hi hauria manera de distingir-los.

La utilització d'aquesta primitiva és útil quan allò que es pretén és l'extensió d'un mòdul afegint-li més definicions de fets, o bé afegint regles a la definició d'un fet ja definit dins el mòdul que es vol estendre.

### 4.5.3. Aplicació d'un mòdul genèric

Mitjançant aquesta definició es construeix un mòdul a partir de la aplicació d'un mòdul genèric (explicat al següent apartat) sobre un conjunt de paràmetres concrets. Per

exemple vegi's la declaració del mòdul *Antecedents\_Patològics* com a aplicació del mòdul genèric *Combinació\_i* sobre els paràmetres *Terreny\_malalt* i *Fàrmacs*;, a la vegada paramètrics sobre el mòdul *Bacterianicitat*:

```
Mòdul Antecedents_Patològics = Combinació_i(Terreny_malalt(Bacterianicitat),
                                             Fàrmacs(Bacterianicitat))
```

La utilització dels mòduls genèrics dóna a aquest esquema de mòduls no només la possibilitat d'establir jerarquies entre els mòduls d'una aplicació, sinó la possibilitat de realitzar abstraccions que generalitzin tasques realitzades moltes vegades en el disseny d'una BC.

## 4.6. Mòduls genèrics

Un mòdul genèric és una abstracció de mòduls que posseeixen una estructura semblant [Futatsugi et al, 1987]. Per exemple vegi's els dos mòduls següents

```
Mòdul Antecedents_Patològics =
Inici
  Mòdul TM = Terreny_malalt(Bacterianicitat)
  Mòdul FM = Fàrmacs(Bacterianicitat)
  Exporta: Pneumococ;
  Regles
    R1 Si no_dem TM->pneumococ i FM->pneumococ
      Llavors Pneumococ és segur
    R2 Si TM->pneumococ i no_dem FM->pneumococ
      Llavors Pneumococ és segur
    R3 Si TM->pneumococ i FM->pneumococ
      Llavors Pneumococ és segur
```

final

```
Mòdul Clínica =
Inici
  Mòdul AN = Antecedents
  Mòdul EX = Exp_físiques
  Exporta: Pneumococ;
  Regles
    R1 Si no_dem AN->pneumococ i EX->pneumococ
      Llavors Pneumococ és segur
    R2 Si AN->pneumococ i no_dem EX->pneumococ
      Llavors Pneumococ és segur
    R3 Si AN->pneumococ i EX->pneumococ
      Llavors Pneumococ és segur
```

final

L'estructura dels mòduls és idèntica excepte els submòduls que cada una posseeix. El que realitzen és una combinació de les evidències aportades tenint en compte

els casos en que en algún mòdul no s'ha pogut deduir res. En cas que cap dels mòduls produís cap evidència sobre el fet *Pneumococ*, no s'aplicaria cap regla i el valor de *Pneumococ* exportat seria *desconegut*. Sembla més correcte definir un mòdul genèric que es particularitzés en els submòduls concrets en cada cas. D'aquesta manera escriuriem el codi del mòdul una única vegada i evitariem inconsistències. Així el mòdul genèric abstracció d'aquests dos seria el següent:

```

Mòdul Combinació_i (X: Pneum; Y: Pneum) : Pneum
Inici
  Exporta: Pneumococ;
  Regles
    R1 Si no_dem X->pneumococ i Y->pneumococ
      Llavors Pneumococ és segur
    R2 Si X->pneumococ i no_dem Y->pneumococ
      Llavors Pneumococ és segur
    R3 Si X->pneumococ i Y->pneumococ
      Llavors Pneumococ és segur
final

```

*Pneum* és l'especificació que engloba als mòduls que exporten *Pneumococ*.

Els mòduls abstrerts passarien a construir-se de la següent manera:

```

Mòdul Antecedents_Patològics = Combinació_i(Terreny_malalt(Bacterianicitat),
  Fàrmacs(Bacterianicitat))

```

```

Mòdul Clinica = Combinació_i(Antecedents_patològics, Exp_físiques)

```

El mòdul genèric s'aplica sobre dos mòduls que segueixin l'especificació *Pneum* i retorna un mòdul que també segueix l'especificació *Pneum*. Aquesta descripció dels tipus dels arguments i del resultat es fa de cara a fer la comprovació dels paràmetres actuals i dels resultats.

Els mòduls genèrics tenen dret a accedir únicament als fets exportats pels seus paràmetres. Si es volguessin incorporar els submòduls de manera que fossin exportats s'haurien de declarar com a submòduls de la següent manera:

```

Mòdul Combinació_i (X: Pneum; Y: Pneum) : Pneum1
Inici
  Mòdul Primer = X
  Mòdul Segon = Y
  Exporta: Pneumococ;
  Regles
    { Conjunt de regles }
final

```

on *Pneum1* seria una especificació que extendria la de *Pneum* amb dos submòduls d'especificació *Pneum* i de nom *Primer* i *Segon*.

## 4.7. Semàntica de la modularització

### 4.7.1. Abstracció funcional

L'*abstracció funcional* és una tècnica utilitzada per limitar la interacció entre els mòduls i per obtenir especificacions simples que continguin una quantitat d'informació controlada pel dissenyador. El que es pretén és amagar detalls d'un mòdul als altres, de manera que alteracions futures no provoquin alteracions en altres mòduls. Volem que la interacció entre els mòduls es limiti a allò que l'especificació indiqui.

S'anomena *abstracció funcional* perquè el que abstraïem és la forma en la que calculem els fets, en oposició a abstracció de dades, on el que s'abstrau és la forma de representar les dades.

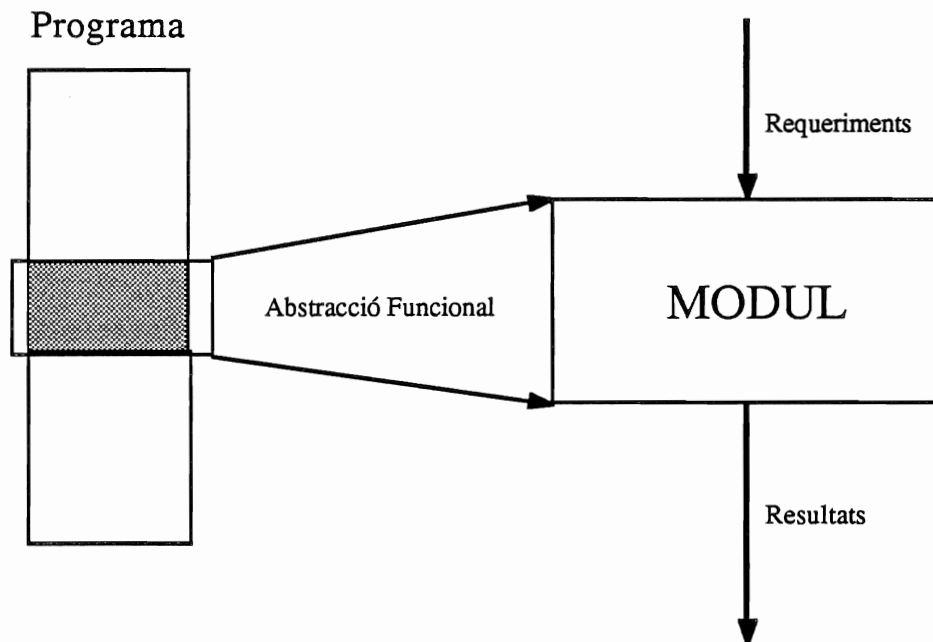


Fig 4.2 Procés d'abstracció funcional.

Per exemple el mòdul d'*exploracions físiques* té dues regles, la R4 i la R5, deduint la certesa de *monoartritis infecciosa*, que estan amagades a la resta de mòduls, atès que *monoartritis infecciosa* no és exportat per aquest mòdul. Ulteriors modificacions de la forma de deduir *monoartritis infecciosa* no afectaran a l'exterior d'aquest mòdul.

Mòdul Exp-físiques : Exploració =

**Inici**

Mòdul BA = Bacterianicitat

**Regles**

R1 Si herpes\_labial i BA->Bacteriana

Llavors Pneumococ és molt\_possible

R2 Si monoartritis\_infecciosa i BA->Bacteriana > possible

Llavors Pneumococ és possible

R3 Si esput = rovellat i BA->Bacteriana > moderadament\_possible

Llavors Pneumococ és molt\_possible

R4 Si periodes\_esgarrifances = 1 i febre > 38.5 i BA->Bacteriana

Llavors Pneumococ és possible

R5 Si artritis i cardinal(articulacions\_afectades)

Llavors monoartritis\_infecciosa és lleugerament\_possible

**final**

#### 4.7.2. Amagament d'informació

La modularització permet establir mecanismes de modularització al nivell estructural que abans no existien. L'amagament d'informació es fa a dos nivells: amagament de submòduls i amagament de fets.

#### 4.7.3. Declaracions compartides

Els mòduls poden tenir interaccions a través de submòduls comuns. Per exemple els mòduls *Terreny\_malalt* i *Fàrmacs* són tots dos paramètrics en un mòdul de tipus *Bact-atip*. Si volguéssim fer una combinació d'ambdós mòduls forçant que la particularització concreta de tipus *Bact-atip* fos la mateixa podríem fer-ho declarant que ambdós submòduls fossin el mateix utilitzant la primitiva **compartint**. Per exemple, si redefiníssim la funció *combinació\_i* com a la funció *combinació\_compartida* tindriem:

Mòdul Combinació\_compartida (X: Pneum\_ant; Y: Pneum\_ant  
compartint X->Bact = Y->Bact) : Pneum =

**Inici**

**Exporta:** Pneumococ;

**Regles**

R1 Si no\_dem X->pneumococ i Y->pneumococ

Llavors Pneumococ és segur

R2 Si X->pneumococ i no\_dem Y->pneumococ

Llavors Pneumococ és segur  
 R3 Si X->pneumococ i Y->pneumococ  
 Llavors Pneumococ és segur

**final**

Essent Terreny\_malalt i Fàrmacs els següents mòduls:

Mòdul Terreny\_malalt (Bact : Bact-atip) : Pneum\_ant =

**Inici**

Mòdul BA = Bact

**Regles**

R1 Si BA->Bacteriana i mes < 4

Llavors Pneumococ és molt\_possible

R2 Si BA->Bacteriana i mes > 4

Llavors Pneumococ és força\_possible

R3 Si BA->Bacteriana i grip Llavors Pneumococ és possible

R4 Si aspiració Llavors Pneumococ és possible

R5 Si BA->Bacteriana i no antecedents\_interés

Llavors Pneumococ és molt\_possible

R6 Si BA->Bacteriana i antecedents\_interés i no grip

Llavors Pneumococ és possible

**final**

Mòdul Fàrmacs (Bact : Bact-atip) : Pneum\_ant =

**Inici**

Mòdul BA = Bact

**Importa:** Antecedents

**Regles**

R1 Si BA->Bacteriana i (peni o amoxi o cefalosporines o eritro) i no evolució i dosis\_adients

Llavors Pneumococ és Lleugerament\_possible

**final**

i suposant dues instàncies de l'especificació Bact-atip: Bacterianicitat1 i Bacterianicitat2, tindriem que la següent aplicació de Combinació\_compartida seria correcta:

Combinació\_compartida(Terreny\_malalt(Bacterianicitat1), Fàrmacs(Bacterianicitat1))

i la següent, en canvi, no:

Combinació\_compartida(Terreny\_malalt(Bacterianicitat2), Fàrmacs(Bacterianicitat2))

La raó per voler compartir en aquest cas és que les instàncies de tipus Bact-atip es fan en funció del tipus de dades utilitzades per deduir Bacteriana i Atípica, en el cas de que les evidències sobre *pneumococ* provinguin de dues fonts diferents volem que les dades que han produït una certesa per Bacteriana i Atípica en ambdós mòduls sigui la mateixa.





# 5 Control Multinivell a MILORD

ENTIA NON SUNT MULTIPLICANDA PRAETER NECESSITATEM

*Guillem d'Ockham*

## 5.1 Introducció

Després de la descripció del llenguatge de representació dels coneixements relatius a associacions entre fets del domini i la seva modularització, explicat en els capítols anteriors, dedicarem aquest capítol a fer una visió global de l'arquitectura de control de MILORD. Tanmateix, presentarem aquells elements de representació de coneixements que no s'han definit fins ara degut a la seva íntima relació amb els mecanismes de control.

Quan es defineix una aplicació hi ha dos punts de vista sobre l'adequació d'una eina: "*Quins coneixements s'hi poden representar*" i "*Com poden ser utilitzats aquests coneixements*". El segon punt és l'aspecte on les eines acostumen a divergir més. Si bé s'han arribat a definir certs paradigmes majoritàriament acceptats en el tema de *com* representar el coneixement: "*frames*", *xarxes semàntiques*, *lògica*, *objectes*. No és aquest el cas pel que fa al control. Quan es parla d'una eina i es diu que utilitza "*frames*", regles de producció o xarxes semàntiques, tothom interpreta correctament aquest aspecte de l'arquitectura [Clancey, 1985b], [Cuenca et al, 1986], [Garijo, 1987]. En els elements relatius al control no s'utilitzen etiquetes que s'adeqüin a models coneguts i ben definits; en el millor dels casos es farà referència a aspectes tècnics del tipus: *cerca dirigida per les dades* o *pas de missatges*. Darrerament es troben a la bibliografia intents de definir models generals de control: *tasques genèriques* [Chandrasekaran 86], *tasques abstractes* [Clancey i Bock, 86], *arquitectura meta-nivell* [Genesereth 83]. Si bé tots aquests treballs no intenten proporcionar una formalització del control, sí que intenten obrir una discussió sobre els processos involucrats en el raonament i donen certes generalitzacions que poden ser útils en diferents camps

d'aplicació. Aquest capítol, dins de la mateixa línia, defineix un model arquitectònic de control, prou general com per a que pugui englobar diferents esquemes de control actualment vigents.

TEIRESIAS, construït en el contexte de MYCIN, va ser el primer sistema on es diferenciaven clarament dos nivells de coneixements, els coneixements de domini o "Object-level knowledge" i els coneixements de control o "Meta-level knowledge" [Davis, 1980], [Davis i Lenat, 1982], [Pitrat, 1987]. Moltes de les característiques essencials d'aquella arquitectura han estat incorporades a l'arquitectura de MILORD.

TEIRESIAS utilitza els coneixements de control exclusivament per un dels components del procés d'"invocació" de les unitats de coneixements o "knowledge sources" KS, el procés de refinament. Aquests processos són: recuperació, refinament i execució. A la fase de recuperació s'obté el subconjunt de KS a utilitzar per resoldre una tasca, a la fase de refinament s'eliminen part d'ells i s'ordenen per ,a la fase d'execució, executar una o més KS. Els coneixements de control són representats en forma de metaregles. Aquestes metaregles afecten a l'ordenació de les KS mitjançant dos criteris:

*Utilitar:* ponderació de la utilitat de la KS amb coeficients de certesa .

*Ordre parcial:* entre dos subconjunts de KS.

Una primera definició d'estratègia apareix a TEIRESIAS. S'entén per estratègia l'ordenació dels KS per la resolució d'una tasca. D'aquesta manera les metaregles de refinament de la bases de coneixements generen estratègies en el sentit de TEIRESIAS. Un esquema de l'arquitectura pot contemplar-se a la figura 5.1. Tanmateix TEIRESIAS planteja que el procés de refinament pot estendre's a les metaregles, mitjançant l'escriptura de metaregles de segon nivell, i així recurrentment definir arquitectures de qualsevol nombre de nivells.

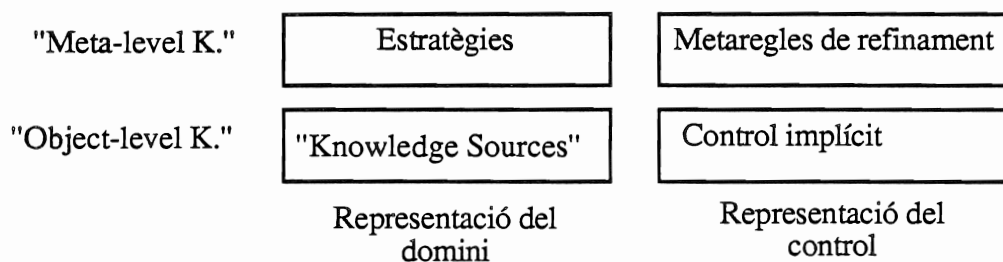


Fig. 5.1 Arquitectura de control a dos nivells de TEIRESIAS.

Certes característiques de TEIRESIAS han influït les ulteriors arquitectures multi-nivell:

- 1.- *Un mecanisme de raonament unificat.* El mateix motor d'inferències és utilitzat per les regles del nivell de domini i per les metaregles de control.
- 2.- *Una representació explícita del control.* Els coneixements referents al procés de refinament són representats explícitament en constructes accessibles pel programa mateix.
- 3.- *Concepte d'estratègia.* Entesa com a ordenació dels elements necessaris per resoldre un problema.

MILORD hereda aquestes característiques de TEIRESIAS, encara que no aplicades al problema de refinament. Els coneixements representats a TEIRESIAS en forma de metaregles que realitzaven l'ordenació de les regles del domini es realitza a MILORD gràcies als GRF. La utilització del mateix mecanisme de raonament ha estat realitzat al llarg de tots els nivells de MILORD, excepte de l'últim, que s'ha volgut separar dels coneixements dependents de l'aplicació. El concepte d'estratègia s'extén a estructures modulars de representació de coneixements, i no a la simple ordenació de les regles del domini, massa pobre per a la programació de grans bases de coneixements.

Una revisió de diferents arquitectures meta-nivell actuals pot trobar-se a [Maes i Nardi, 1988].

Inicialment farem una reflexió sobre els tipus de raonament més comuns als sistemes experts. Determinarem quins són i com afecten els esquemes de control dels llenguatges que els volen implantar. A continuació entrarem a descriure en detall l'arquitectura multi-nivell de MILORD.

## 5.2 Tipus de raonament en Resolució de Problemes

Abans d'entrar en detall sobre quines tècniques proporciona MILORD per representar diferents estratègies de resolució de problemes és interessant de fer un repàs de quines són les formes de raonament més comuns en les àrees d'aplicació dels sistemes experts. Ens hem inspirat fonamentalment en la forma de resoldre problemes en el domini mèdic [De lotto i Stefanelli, 1985], [Szolovits et al, 1988], [Patil, 1987], [Campbell, 1987]. S'ha fet així perquè les aplicacions d'alta complexitat més nombroses desenvolupades fins ara corresponen a la medicina: MYCIN [Shortliffe, 1976], CASNET [Weiss et al, 1978], SPHINX [Fieschi, 1983].

Els mètodes de raonament són en certa mesura independents de la representació utilitzada pels coneixements de domini. Classifiquem els tipus de raonament en dos

grups, que descriurem en els següents apartats:

- 1.- Mètodes de raonament classificatori.
- 2.- Mètodes de raonament generatiu.

En la medicina clínica, àrea on hem desenvolupat les principals aplicacions, s'utilitzen primordialment tres mètodes de raonament classificatori:

- 1 *Hipotètic-deductiu*,
- 2 *Classificació heurística* i
- 3 *Causal*.

Desde el punt de vista que aquí interessa, el tipus de raonament determina un tipus de control.

## 5.2.1 Raonament Classificatori

La classificació és un procés consistent en seleccionar un element d'entre un conjunt de possibilitats a partir d'observacions de característiques rellevants del cas a classificar. Es un dels problemes que ha rebut més atenció per part dels investigadors de la intel·ligència artificial [Mittal et al, 1984], [Sticklen et al, 1985], [Bylander i Mittal, 1986]. Els tipus de classificació poden anar des de la classificació directa mitjançant mètodes de reconeixement de formes, fins a processos de classificació estructurats desenvolupats en varies etapes.

El raonament de tipus classificatori és un tipus de raonament molt freqüentment utilitzat en medicina per resoldre problemes de tipus diagnòstic o terapèutic. Encara que ocasionalment el raonament de tipus classificatori es pot fer en un sol pas, com és el cas quan es troba una *dada patognomònica*, el més comú es que aquest mètode sigui iteratiu, es a dir, establint diferents nivells classificatoris que delimitin progressivament el problema i focalitzin el procés de raonament.

### 5.2.1.1 Establir/refinar

Un dels mètodes iteratius de classificació més simples és l'establir/refinar [Chandrasekaran i Mittal, 1983]. Mitjançant aquest mètode es segueix un camí en el que, tot anant dels fets més generals als més específics, les successives decisions preses restringeixen contínuament el rang de possibles solucions finals. La figura 5.2

mostra un exemple de recorregut d'una jerarquia de malalties anèmiques

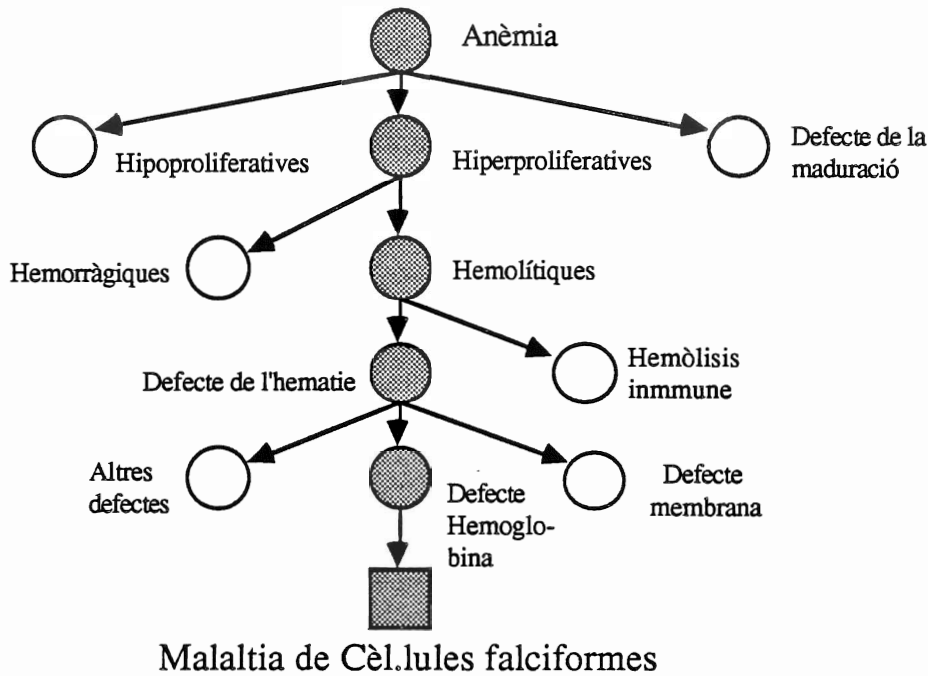


Fig 5.2 Camí seguit dins una jerarquia de malalties anèmiques utilitzant el mecanisme d'establir/refinar fins la detecció de la malaltia concreta. A cada pas es determina la subclasse per on continuar el procés.

### 5.2.1.2 Raonament hipotètic-deductiu

Aquest mètode de classificació és un mètode iteratiu, on a cada pas es generen hipòtesis de solució a partir de la informació disponible i s'hi obtenen dades que validin o rebutgin les hipòtesis generades.

En la pràctica clínica de la medicina, i d'una manera genèrica, es pot afirmar que el raonament diagnòstic és un raonament hipotètic-deductiu.

A partir de les dades que es coneixen d'un problema concret el clínic estableix una o varies hipòtesis diagnòstiques que a continuació passen a ser validades. A partir d'aquest moment el metge cercarà noves dades que li permetin restringir el nombre d'hipòtesis formulades, tot eliminant-ne les que siguin contradictòries amb l'estat del problema, i confirmar les hipòtesis restants.

### 5.2.1.3 Raonament causal

El raonament causal consisteix en trobar les relacions causals entre les manifestacions observables d'un fenòmen i les causes que el poden produir. El raonament causal es pot entendre com a recorreguts de grafs de relacions causa-efecte. Quan s'està en un nus d'alt nivell d'abstracció on hi ha una relació amb un grau d'evidència baix, es fa un recorregut d'un subgraf, resultat de l'expansió d'aquesta relació, situat en un nivell més profund, és a dir, on els coneixements són més detallats. Per tant, depenent del moment, s'utilitzen diverses parts del model de domini.

Aquest tipus de raonament està estretament vinculat amb la forma de representació de les associacions entre conceptes. De fet, i degut a la relació tant forta entre el tipus de raonament i la representació dels coneixements, s'acostuma a utilitzar el nom de models *causals* o *profunds* a la conjunció de la representació i el raonament.

Aquest tipus de raonament pot anar combinat amb altres tipus. Així, ens podríem trobar amb un mètode de classificació iterativa utilitzant raonament causal per cada pas de la iteració. També seria compatible amb mètodes de raonament generatiu.

La següent figura exemplifica el tipus de raonament causal:

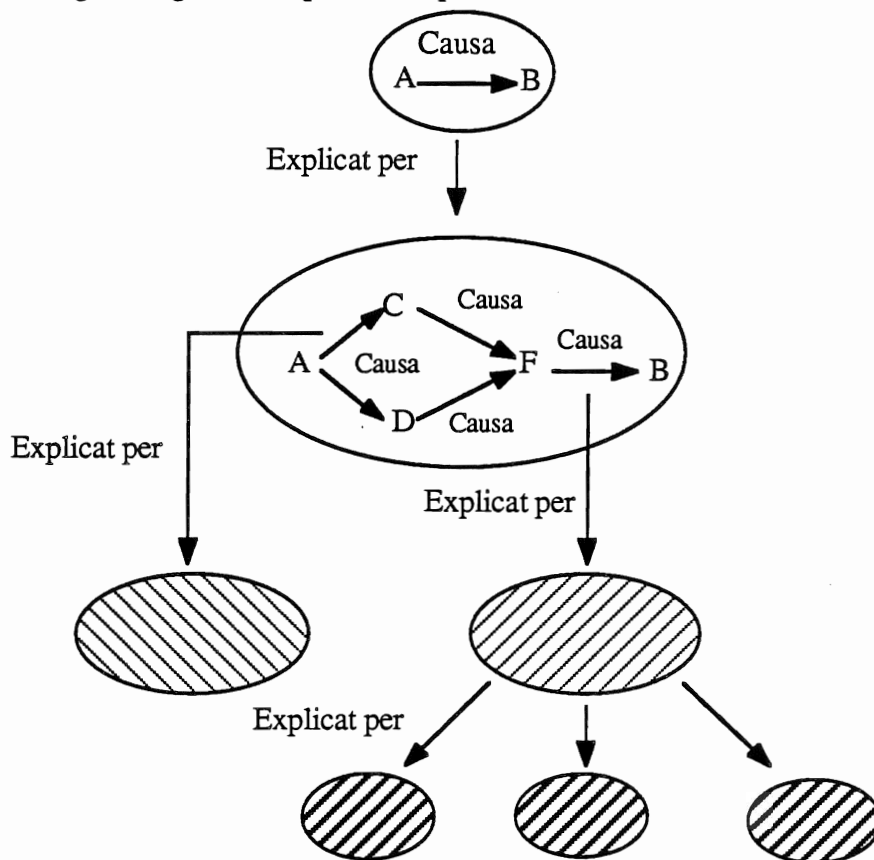


Figura 5.3. Possible representació dels coneixements de tipus causal

### 5.2.1.4 Classificació Heurística

Els mètodes de classificació poden dividir-se segons Clancey [Clancey, 1985a] en dos tipus: *classificació simple* i *classificació heurística*. Els mètodes de *classificació simple* són aquells en que les dades d'un cas o les seves abstraccions, han d'acabar-se directament amb les característiques de les solucions. Els mètodes de *classificació heurística*, en canvi, permeten realitzar acaraments heurístics entre les dades del cas i les característiques de les solucions. Aquests acaraments heurístics s'han d'entendre com associacions directes entre elements de diferents jerarquies. Per exemple, acaraments entre abstraccions de les dades d'un pacient i una jerarquia de possibles malalties. L'esquema d'aquest tipus de classificacions pot observar-se en la següent figura:

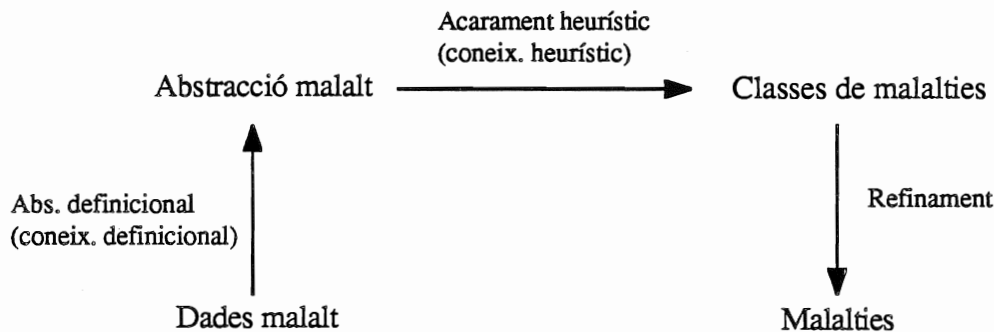


Fig. 5.4 Mètode de classificació heurística.

### 5.2.2 Raonament Generatiu

En determinats dominis d'aplicació els sistemes de classificació no són suficientment potents. Això es degut a la impossibilitat d'establir el conjunt de solucions a priori, bé per ser desconegudes, bé per ser aquest conjunt d'una mida inmanegable.

Aquests dominis, encara que no hi hagi gaire experiència en el disseny de sistemes experts, poden ser els de disseny, CAD o jocs. En medicina s'han desenvolupat alguns sistemes amb aquesta orientació ABEL [Patil, 1981], CADUCEUS [Pople, 1982].

ABEL utilitza com a objectes i solucions els PSM (Patient Specific Model), aquests PSM mantenen les interpretacions alternatives de l'estat del malalt, diferents etiologies possibles, diferents justificacions, etc. La cerca de la solució a un problema consisteix en la generació del PSM que millor s'adapti al cas en curs. Inicialment mitjançant un sistema de classificació, ABEL genera el primer PSM. A continuació

utilitza operadors que transformen els PSM en nous PSM. Els operadors són de tres tipus diferents:

- a) *Projecció*: Amplia el conjunt d'hipòtesis a manejar dins el PSM.
- b) *Agregació*: Combina causalment diferents nodes en un de més alt nivell d'abstracció.
- c) *Disgregació*: Separa hipòtesis en elements a més baix nivell per a que puguin ser suportats pel PSM en curs.

CADUCEUS utilitza dos mecanismes de classificació ortogonals : 1) una taxonomia de l'afectació dels òrgans i 2) un graf de relacions causals, i un conjunt d'operadors que generen un espai de cerca enorme de les possibles interpretacions de les dades, cada interpretació corresponent a un PSM.

### 5.3 Arquitectura de control a MILORD

L'arquitectura de MILORD (veure fig 5.5) és una arquitectura on s'han separat llurs elements al llarg de dues dimensions:

- a) *Dimensió domini/control*
- b) *Dimensió dels tipus de coneixements*

A la dimensió *domini/control* hem separat els elements que tinguin a veure amb *quines* solucions i *quins* mètodes de resolució tenim i els elements de *com* trobar aquestes solucions i *com* determinar els mètodes de resolució més adients. D'alguna manera és la distinció entre els elements *sintàctics* de l'arquitectura i els elements *semàntics* (que determinen la semàntica de la sintaxi).

A la dimensió dels *tipus de coneixements* separem els elements segons la seva relació amb els coneixements que representen.

Un dels punts interessants d'aquesta arquitectura és la visió dels elements d'un sistema com a punts en un espai bidimensional; les dimensions utilitzades són el fruit d'un procés de generalització a partir de l'experiència en el desenvolupament d'aplicacions.

Des del punt de vista de la dimensió del tipus de coneixements, l'arquitectura de MILORD està estructurada en cinc nivells de representació del domini i del control. Cadascun dels nivells és suportat pels nivells inferiors i modelitza uns tipus de coneixements concrets.

No es pretén amb aquesta arquitectura donar cabuda a tots els sistemes de solució de problemes existents. Sí creiem en tot cas que una arquitectura multinivell on



cada nivell tingui un comportament dissenyat de forma independent dels altres nivells és una bona metodologia de construcció de SE. El capítol 4 mostra com es pot substituir el disseny complet d'un dels nivells, l'estructural en aquest cas, sense que sigui necessari alterar la semàntica de la resta de nivells. Aquesta modularitat és absolutament desitjable de cara a obtenir arquitectures més generals que la present on es pugui especificar el nombre i el tipus dels nivells que una arquitectura concreta haurà de posseir.

Aquesta línia ha estat desenvolupada en informàtica, en el cas de les representacions, amb l'aparició dels tipus abstractes de dades. En el futur serà necessari aplicar-la en l'abstracció del control i parlar dels tipus de lògica, dels tipus de deducció o del fluxe dels programes com a objectes abstractes. Aquesta línia serà tema d'estudi dins el projecte SPES, finançat per la CICYT i que es desenvoluparà al llarg dels propers tres anys. També (encara que amb l'objectiu de dissenyar un metallenguatge per la validació de BC) estem adoptant aquest enfocament en el projecte VALID del programa ESPRIT. Tots dos projectes s'estan desenvolupant al Centre d'Estudis Avançats de Blanes.

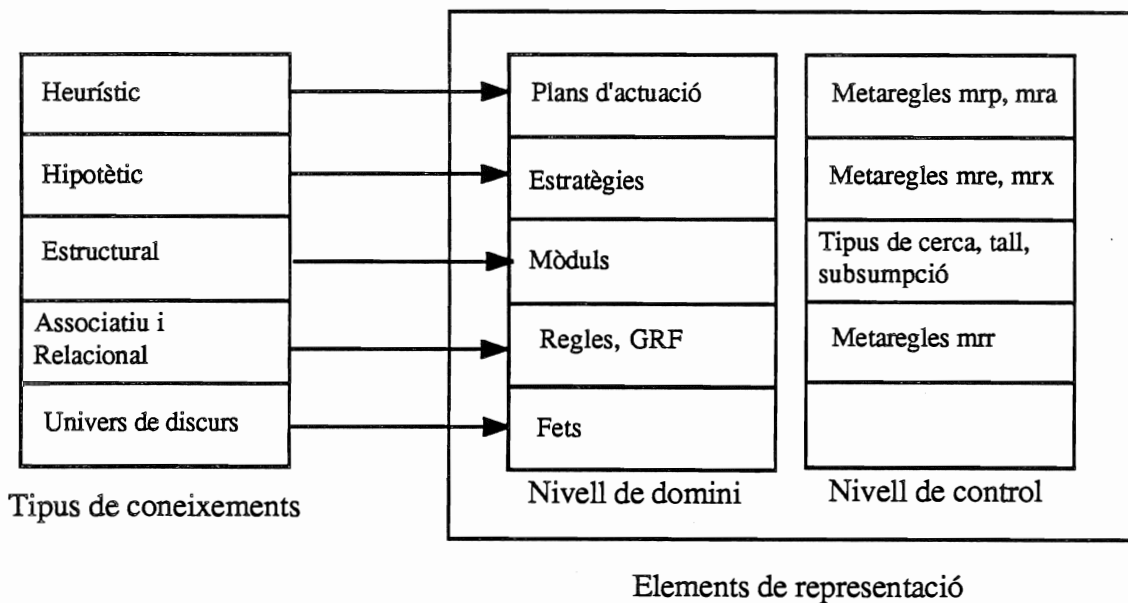


Fig 5.5 Arquitectura multi-nivell de MILORD.

Els diferents nivells de l'arquitectura són els següents:

### *Nivell d'univers de discurs*

Aquest nivell és el relatiu als conceptes que es manipularan a l'aplicació, descriu el seu tipus i la forma de ser obtinguts. Aquest nivell és l'interlocutor vàlid en la fase de conceptualització d'un sistema expert. Es a dir, en la decisió de quina

subàrea de coneixements serà la utilitzada en el sistema.

Aquest nivell és estàtic. No hi ha cap element de control que hi incideixi. En sistemes amb aprenentatge inductiu, on es generen nous conceptes per generalització o es modifiquen, conceptes previs sí que tindriem un element de control a aquest nivell.

### *Nivell associatiu i relacional*

A aquest nivell s'estableixen les relacions entre els fets mitjançant la definició d'un GRF i es defineixen les associacions entre conceptes fonamentades tant en models teòrics com empírics. El formalisme de representació d'aquestes associacions són les regles de producció i els GRF.

Els mecanismes de control fan referència a l'aplicabilitat de les regles i es basen en conceptes de l'univers de discurs i les seves relacions al GRF. Les unitats de control estan escrites en forma de metaregles.

### *Nivell estructural*

Aquest nivell agrupa els elements del nivell associatiu en mòduls per la realització d'una tasca concreta. Aquests mòduls poden ser de dos tipus diferents com ja s'ha explicat als capítols 2 i 4.

Els mecanismes de control afecten al procés d'avaluació del mòdul: tipus de cerca, tipus de lògica o llistats d'aplicació de regles. Aquests mecanismes són expressats de forma declarativa i vinculats a l'identificador de cada mòdul.

### *Nivell hipotètic*

Aquest nivell és el relatiu a les estratègies de resolució. Els elements de representació del domini són llistes d'identificadors de mòdul. Aquestes llistes no són definides per l'expert com elements estàtics, sinó que són generades per metaregles. És el nivell on s'implanten els mètodes de raonament, com ara l'hipotètic-deductiu o el classificatori per iteracions establir/refinar.

### *Nivell heurístic*

Aquest nivell, l'últim de l'arquitectura, és el responsable dels plans concrets d'execució del sistema.

Les metaregles d'aquest nivell són les responsables de la creació dels plans i utilitzen únicament informació referent al nivell hipotètic. En aquest sentit són

independents del domini i poden ser intercanviades entre aplicacions diverses. En l'actualitat el mateix conjunt de metaregles ha estat efectivament utilitzat per les aplicacions PNEUMON-IA i RENOIR.

Aquesta separació entre els diferents nivells de control i de representació dels coneixements permet diferents tècniques de disseny de BC:

- 1.- Permet un disseny descendent de la BC, atès que els coneixements de domini i el control estan separats i per tant poden tenir desenvolupaments independents. Per exemple, la definició d'estratègies i la resolució de conflictes entre elles: es pot definir primer les estratègies i integrar-les amb la resta de coneixements, decidint més tard com es solucionaran els conflictes entre elles. Tantmateix, la definició d'estratègies és independent de la definició dels elements de nivells inferiors o superiors.
- 2.- Permet definir diferents mètodes de resolució de problemes, posat que aquests depenen dels coneixements expressats al nivell hipotètic. Es pot diferenciar, per exemple, entre classificació simple o classificació heurística [Clancey, 1985a].

Un dels aspectes que ha influït més en el disseny d'una arquitectura multinivell de control ha estat l'intent de fer que el comportament del sistema fos el més creïble possible, en el sentit que seguís la mateixa estratègia que seguiria l'expert humà. En molts sistemes, disposar d'una informació altament rellevant per la solució del problema en curs no té una utilitat immediata en la modificació de la forma de resoldre el problema. Generalment, quan s'intenta verificar una hipòtesi els sistemes no es plantegen noves hipòtesis a demostrar fins que aquesta hipòtesi no ha estat verificada o rebutjada. Això és degut a la no separació entre diferents nivells de control respecte dels tipus de coneixements. Aquests sistemes no distingeixen clarament els moments en els que cada element de control ha d'entrar en actuació optant per algorismes on el control general es replanteja en moments predeterminats, d'una manera síncrona, fent servir l'analogia amb els sistemes operatius. Quan s'acaba tota una avaluació amb un dels motors d'inferència es comprova si s'ha de modificar l'estratègia en curs. En el cas de MILORD, el replantejament de l'estratègia de resolució es realitza cada vegada que la informació que es té del problema varia, bé perquè s'ha obtingut nova informació, (mitjançant l'aplicació d'alguna regla), bé perquè l'usuari ens ha proporcionat noves dades sobre el cas.

La credibilitat del sistema està relacionada amb aquest comportament, degut a que d'aquesta manera el sistema sempre segueix la hipòtesi més plausible, coincidint

teòricament amb les hipòtesis que els usuaris puguin plantejar-se en el moment de l'execució.

Els experts en medicina activen hipòtesis a partir de la introducció de símptomes individuals [Kassirer i Gorry 78]. Això obliga a replantejar el procediment de resolució tant bon punt és possible de manera que el sistema intenti seguir la seqüència d'hipòtesis que l'expert es faria en cada cas. MILORD es planteja com un sistema eminentment interactiu. En sistemes en els quals la interacció amb l'usuari és baixa o nul·la aquest criteri no és tan determinant.

L'arquitectura no està preparada per definir sistemes de raonament generatius, MILORD suposa totes les solucions del problema conegudes a priori. Es mou, per tant, dins el marc dels sistemes de classificació. Aquesta limitació ve donada per la representació dels elements de baix nivell en lògica d'ordre 0+. Modificant aquest esquema a lògica d'ordre 1 el mateix esquema arquitectònic podria servir per definir sistemes amb raonament generatiu.

Aquest model arquitectònic té característiques molt interessants des del punt de vista de l'enginyer de coneixements:

**Extensibilitat:** afegir nous coneixements de domini o nous coneixements de control és fàcil i segur. Degut a que ambdós tipus de coneixements estan expressats de forma declarativa.

**Depurabilitat:** Corregir la BC i validar-la és més fàcil degut a la modularitat de tota l'arquitectura.

**Explicabilitat:** Una de les coses que fan les BC poc explicatives és l'ús d'elements de control implícits en la seva escriptura, premisses extres a les regles, cerca cega, manca d'introspecció, etc. El que fa aquest esquema d'arquitectura és donar eines que permetin escriure BC amb tots els elements per fer una explicació del coneixement clara i no ambigua.

### 5.3.1 Metaregles

Les metaregles, inicialment introduïdes per Davis [Davis, 1980], són unitats de coneixements que fan referència a: el control en l'ús dels coneixements de domini expressats mitjançant regles, la generació d'estratègies de resolució, o bé a situacions excepcionals. Hi ha quatre tipus de metaregles segons els elements arquitectònics a que fan referència:

- i.- *Metaregles referents al nivell associatiu*: Són aquelles que tenen a veure amb l'aplicabilitat de les regles de la BC.
- ii.- *Metaregles referents a estratègies*: Són aquelles encarregades de construir les estratègies elementals.
- iii.- *Metaregles referents a excepcions*: Són les que responen a situacions anòmales o excepcionals.
- iv.- *Metaregles d'acabament*: Són les que defineixen les condicions d'acabament de l'execució del sistema.

Al llarg d'aquest capítol explicarem en detall cadascun d'aquests tipus.

Passarem a descriure en detall en els següents apartats els diferents nivells de l'arquitectura, amb la seva sintàxi i part de la semàntica.

## 5.4 Nivell de l'univers de discurs

Aquest nivell, el més elemental de tots, no té elements de control associats. El seu contingut és merament descriptiu. Consisteix en la definició del *diccionari* tal i com ja s'ha explicat a 2.3.1. No entrarem en més detalls atès que aquests van ser explicats convenientment en l'esmentat paràgraf.

## 5.5 Nivell associatiu i relacional

Aquest nivell és el que representa els coneixements associatius i relacionals. Aquest tipus de coneixements es representen utilitzant dos esquemes diferents. Les regles d'inferència i els GRF. Aquest esquemes ja han estat descrits al capítol 2. Aquí descriurem els elements de control que afecten l'ús de les regles. Aquests elements han estat escrits en format de metaregles.

### 5.5.1 Metaregles referents al nivell associatiu i relacional

Tots els elements del nivell associatiu i relacional (regles i GRF) s'escriuen de forma declarativa sense tenir en compte com i quan entraran en actuació. El control sobre ells és de dos tipus diferents. Un és el control implícit en l'entorn MILORD, avaluació esquerra dreta de les regles, algorismes d'avaluació segons lògica fuzzy, etc.

L'altre és explícit sota la forma de metaregles. Aquestes s'encarreguen de que el conjunt de regles actives i inhibides en cada moment sigui el correcte. Per tant, l'execució d'una metaregla d'aquest tipus tindrà com a resultat la modificació de l'estat del conjunt de regles de l'aplicació.

La sintaxi d'una metaregla referent a les regles és la següent:

```

mrr ::= id-meta Si premissa Llavors filtradors
filtradors ::= filtrador filtradors | filtrador
filtrador ::= inhibir regles relació-id fet-id |
               inhibir regles fet-id |
               inhibir regles llista-id-regla |
               desinhibir regles relació-id fet-id |
               desinhibir regles fet-id |
               desinhibir regles llista-id-regla |
               podar fet-id |
               empeltar fet-id
llista-id-regla ::= id-regla llista-id-regla | id-regla

```

**inhibir regles** *relació fet-id* : Inhibeix totes les regles que utilitzin com a premissa fets que mantinguin la relació *relació* amb el fet *fet-id*.

**desinhibir regles** *relació fet-id* : Operador invers de l'anterior. Activa les regles que mantinguin la relació *relació* amb el fet *fet-id*.

**inhibir regles** *llista-id-regla* : Inhibeix totes les regles pertanyents a *llista-id-regla* .

**inhibir regles** *fet-id* : Inhibeix totes les regles que utilitzin *fet-id* a alguna de les seves premisses.

**desinhibir regles** *fet-id* : Operador invers de l'anterior. Activa totes les regles que utilitzin *fet-id* a alguna de les seves premisses

**desinhibir regles** *llista-id-regla* : Activa totes les regles pertanyents a *llista-id-regla* .

**podar** *fet-id* : Inhibeix totes les regles que tinguin a les seves premisses algun fet que mantingui una relació inclusiva amb el fet *fet-id* (veure 2.3.4).

**empeltar** *fet-id* : Operador invers de l'anterior. Activa totes les regles que tinguin a les seves premisses fets que tinguin relacions inclusives amb *feti-d*.

## 5.6 Nivell estructural

Al nivell estructural es defineixen les unitats de solució de problemes. Aquestes unitats reben el nom de mòduls, que ja han estat presentats al capítol 2. Aquests mòduls representen l'estructura del procés de classificació. Són les solucions particulars i les classes de solucions. La seva definició és de dos tipus:

*Mòduls no estructurats*: Són els presentats al capítol 2, i no presenten mecanismes d'herència, o d'amagament d'informació.

*Mòduls estructurats*: Definites al capítol 4. Posseeixen eines més sofisticades de disseny: herència, amagament d'informació, abstracció funcional, etc.

Els elements de control presentats en aquest apartat són vàlids per tots dos

### 5.6.1 Estructures de control

Les estructures de control que explicarem aquí són els mecanismes implícits de control provistos per l'entorn en l'avaluació dels mòduls i dels seus components. Aquests mecanismes han de ser coneguts a l'hora de definir aplicacions, posat que són paramètrics, i l'usuari pot adaptar-los a les seves necessitats. Les estructures definides són les següents:

- 1.- *Nivells de tall*
- 2.- *Tipus de Cerca*
- 3.- *Subsumpció*

#### 5.6.1.1 Nivells de tall

Un dels mecanismes que ja ha estat comentat a l'hora de parlar de l'avaluació de les regles és l'existència de nivells de tall. Aquests nivells estableixen el valor de certesa mínim que ha de posseir una premissa per poder ser disparada la regla. A Mycin, que posseïa una escala numèrica d'avaluació de les certes el llindar estava a 0.2. Dins l'esquema de tractament de la incertesa de MILORD aquest llindar haurà de ser un terme lingüístic. Per defecte, el nivell de tall serà el segon terme, començant pel terme de menor grau de certesa de l'escala definida per l'aplicació.

### 5.6.1.2 Tipus de cerca

El procés elemental d'inferència és el *modus ponens* ja descrit. Ara bé, hi ha un problema de control associat que és: sobre quina regla intentar aplicar en cada moment el *modus ponens*. Aquest procés, que implica una cerca dins el conjunt de regles aplicables en cada moment, pot ser de dos tipus:

*Cerca dirigida per les dades:* En aquest cas la propera regla a aplicar vindrà determinada pel que coneixem del problema. Aquelles regles que tinguin el valor de certesa de la seva premissa per sobre del llindar seran aplicades.

*Cerca dirigida per les hipòtesis:* En aquesta cerca les regles són cercades en funció de quina és la conclusió que tenen. Aquelles regles que tinguin una conclusió referent a la hipòtesi en curs seran les seleccionades abans.

Aquestes cerques, de vegades, reben el nom d'encadenament endavant i encadenament enrera respectivament. Cada unitat d'estratègia o mòdul, té associat un tipus de cerca definit per l'expert. La decisió és normalment depenent de la tasca encarregada. Per defecte, si l'expert no diu res respecte a la cerca a utilitzar, se suposa cerca dirigida per les hipòtesis.

### 5.6.1.3 Subsumpció

El mecanisme de subsumpció, com ja s'ha explicat al capítol tres, fa referència a la manipulació de coneixements incomplets. Aquest mecanisme és a la base dels algorismes dels motors d'inferència i no pot ser modificat per l'actuació de les aplicacions.

La utilització de la subsumpció es veu en els següents algorismes referents als dos tipus d'encadenaments (algorismes simplificats on es focalitza l'atenció en l'ús de la relació de subsumpció).

#### *Encadenament endavant*

- 1.- Obtenir el conjunt conflicte  $S$  de regles aplicables del mòdul en curs que no estiguin inhibides o aplicades.
- 2.- Si  $S = \emptyset$  llavors acabar,



3.- Ordenar el conjunt conflicte segons les següents relacions per ordre de precedència:

Subsumpció: regles més específiques primer.

Certesa de les regles: regles més certes primer.

Ordre de l'expert: Les regles escrites abans primer.

4.- Aplicar la primera regla  $r_0$

5.- Inhibir totes les regles  $r_i$  tals que  $r_i$  subsumeix  $r_0$ .

6.- Anar a 1.

#### *Encadenament enrera*

1.- Obtenir el conjunt conflicte  $S$  de regles que dedueixen l'objectiu en curs.

2.- Si  $S = \emptyset$  llavors acabar,

3.- Ordenar el conjunt conflicte segons els criteris de l'algorisme anterior.

4.- Cercar enrera a partir de la primera regla  $r_0$ , definint el nou objectiu.

5.- Si  $r_0$  ha estat aplicada, llavors eliminar totes les regles  $r_i$  tals que  $r_i$  subsumeix  $r_0$ .

6.- Anar a 1.

### 5.6.2 Sintaxi de les estructures de control

La sintaxi de definició dels elements de control és:

<i>control-estructural</i>	::= Elements de control <i>controls</i>
<i>controls</i>	::= <i>control controls</i>   <i>control</i>
<i>control</i>	::= Mòdul <i>id-mòdul</i> <i>params-control</i>
<i>params-control</i>	::= Cerca = endavant   Cerca = enrera   Tall = <i>terme-id</i>   <i>params-control params-control</i>

**Cerca = endavant** indica que el tipus de cerca serà dirigida per les dades. Això implica:

- 1.- El sistema no farà preguntes sobre els fets desconeguts. És a dir, aquest encadenament serà interessant quan suposem conegudes les dades rellevants per l'aplicació d'un mòdul.
- 2.- L'aplicació de les regles es farà seguint l'estratègia de prioritat en profunditat ("*depth first*").
- 3.- El sistema aturarà els cicles d'encadenament quan, o bé no hi hagi cap més regla per aplicar, o bé quan els objectius del mòdul hagin estat assolits amb el màxim valor de certesa dins l'escala de termes definida.
- 4.- Quan un nou fet hagi estat deduït, es consultarà el nivell superior per veure si l'estratègia en curs pot variar. Si es així s'abandonarà l'avaluació del mòdul. Si més endavant es tornés al mòdul es continuaria l'avaluació a partir del punt on es va abandonar.

**Cerca = enrrera** indica que el tipus de cerca serà dirigida per les hipòtesis. El sistema intentarà validar els objectius del mòdul en l'ordre establert en la seva definició. Quan tots els objectius hagin estat exhaurits el sistema finalitzarà l'avaluació del mòdul. Aquesta cerca implica:

- 1.- Tot fet desconegut i exigit per la validació d'una regla serà demanat a l'exterior en cas que no es trobi cap regla que el dedueixi. En cas que fracassessin totes les regles que el deduïssin, el valor del fet seria considerat *desconegut*.
- 2.- L'aplicació de les regles es farà seguint l'estratègia de prioritat en profunditat.
- 3.- Quan un fet ha arribat al valor màxim de certesa, la resta de regles que el dedueixen i no han estat utilitzades, no s'aplicaran.
- 4.- Quan un nou fet hagi estat deduït o obtingut de l'exterior, es consultarà el nivell superior per veure si l'estratègia en curs pot variar. Si es així s'abandonarà l'avaluació del mòdul. Si més endavant es tornés al mòdul es continuaria l'avaluació a partir del punt on es va abandonar.

**Tall = terme** indica el llinard que haurà de superar la validació de la premissa d'una regla pertanyent al mòdul per a que aquesta pugui ser disparada. L'avaluació de la regla es farà en prospecció, en el moment en que sigui impossible la superació d'aquest llinard s'abandonarà l'aplicació de la regla, encara que restessin premisses per validar. Aquesta prospecció és dinàmica i s'estén al llarg dels arbres de deducció.

Exemple: Consideri's la regla:

R12024 Si Patró-intersticial i Atípica i Leucopènia  
Llavors virus és possible

on els tres fets de les premisses són de tipus difús. Si el nivell de tall fos *lleugerament possible*, i el valor de *Patró-intersticial* fos *possible*, i el de *Atípica* fos *lleugerament possible*, la seva combinació segons els connectius utilitzats a PNEUMON-IA seria *molt poc possible*, valor inferior al nivell de tall. En aquest cas no s'intentaria la demostració de la última premissa: *Leucopènia*.

## 5.7 Nivell hipotètic

Aquest nivell modelitza els coneixements referents a la definició d'hipòtesis de solució. Està intimament relacionat amb el raonament hipotètic-deductiu. L'element de representació de domini són les estratègies, representant les hipòtesis a validar. Els element de control són les metaregles que generen les estratègies.

### 5.7.1 Estratègies

Dins el procés de solució de problemes l'establiment d'estratègies de solució és un problema cabdal i representa en gran mesura el grau d'expertesa del sistema. Dins els sistemes de classificació, com ara el que ens ocupa, les estratègies representen les classes o elements sobre els que hi ha una evidència més alta de ser la solució al problema. Dins el contexte de MILORD l'unitat estratègica és el mòdul. És l'element de focalització del sistema. En cada moment el sistema intenta validar un mòdul segons s'ha explicat als capítols 2 i 4. L'estratègia ens indica sobre quins mòduls convé focalitzar i sobre quins no.

**Def** Una *estratègia*  $S$  és una partició d'un conjunt de mòduls  $M$  en tres classes diferents: Mòduls a visitar, Mòduls a eliminar i Mòduls encara per definir, i un grau de certesa associat a la partició.

$$S = ((MV, ME, MD), l) \quad l \in L.$$

Al conjunt de totes les estratègies possibles sobre un conjunt de moduls  $M$  l'anomenarem  $\Sigma_M$ .

Def Anomenarem **estratègia nulla** a l'estratègia  $S_0 = ((\emptyset, \emptyset, M), \text{cert})$ .

Def Direm que dues estratègies  $S_1$  i  $S_2$  són compatibles si i només si

$$1.- MV_1 \cap ME_2 = \emptyset, i$$

$$2.- ME_1 \cap MV_2 = \emptyset$$

El sistema obliga a definir una estratègia per defecte que serà l'estratègia inicial del sistema. Aquesta estratègia pot consistir en un únic mòdul d'obtenció d'informació, o bé en tota una estratègia complexa. Per exemple a l'aplicació SOLAR existeix una única estratègia i va ser definida com a estratègia per defecte. En cas que l'aplicació només tingui una estratègia, l'estratègia per defecte, no tindrà cap metaregla de tipus mre, i per descomptat tampoc de tipus mrx, metaregles del nivell estratègic encarregades de combinar estratègies.

La sintaxi d'aquest nivell és la següent:

<i>Nivell-hipotètic</i>	::= <b>Nivell Hipotètic</b> <i>lmre-mrx</i> <i>estrate-defecte</i> <b>finivell</b>
<i>lmre-mrx</i>	::= <i>mre</i>   <i>mrx</i>   $\lambda$   <i>lmre-mrx</i> <i>lmre-mrx</i>
<i>estrate-defecte</i>	::= <b>Estrategia inicial</b> <b>Visitar = l-id-mòduls</b>
<i>l-id-mòduls</i>	::= <i>id-mòdul</i> <i>l-id-mòduls</i>   <i>id-mòdul</i>

Els dos tipus de metaregles d'aquest nivell fan referència a les noves accions a prendre. Poden ser de dos tipus, referents a estratègies: *mre* i referents a excepcions: *mrx*. Els següents apartats les descriuen en detall.

### 5.7.2 Metaregles referents a estratègies

Aquestes metaregles són operadors que construeixen estratègies. Són elements que associen fets del domini de l'aplicació i mòduls de la BC. Aquests operadors

indueixen una partició sobre el conjunt de mòduls tal i com s'indica a continuació.

**Def** Una **metaregla referent a estratègies (mre)** és un operador del següent tipus:

$$\delta: U \rightarrow S : S \in \Sigma_M$$

Essent U l'univers de discurs.

La sintaxi que té una metaregla d'aquest tipus és la següent:

<i>mre</i>	::= <i>id-meta</i> <b>Si premissa</b> <b>Llavors constructors</b> <b>amb certesa</b> $\beta$
<i>constructors</i>	::= <i>constructor constructors</i>   <i>constructor</i>
<i>constructor</i>	::= <b>mòduls a visitar</b> <i>l-id-mòduls</i>   <b>mòduls a eliminar</b> <i>l-id-mòduls</i>   <b>mòduls a tractar</b> <i>l-id-mòduls</i>
<i>l-id-mòduls</i>	::= <i>idmodul l-id-mòduls</i>   <i>idmodul</i>

En aquest cas els *constructors* són suboperadors que ens defineixen la partició sobre el conjunt de mòduls. L'estratègia resultant de l'aplicació dels constructors ha de respectar la definició d'estratègia, es a dir, ha de generar una partició correcte de M. De no ser així, la metaregla es consideraria invàlida. Els constructors poden considerar-se operadors que s'apliquen sobre estratègies per produir-ne de noves.

Totes les metaregles són aplicades tan bon punt sigui possible, és a dir, en el moment immediatament posterior en el qual la seva premissa passi a tenir un grau de certesa diferent de fals. En qualsevol moment, per tant, pot ser generada més d'una estratègia diferent. Serà el nivell de coneixements heurístics l'encarregat de solucionar els possibles conflictes.

Sobre les metaregles també funciona el mecanisme de subsumpció (veure 3.3). D'aquesta manera podem utilitzar coneixements incomplets en la generació d'estratègies sabent que sempre serà utilitzada l'estratègia més específica possible amb la informació disponible.

Les següents metaregles tenen el seu únic fet de la premissa relacionat al GRF per una relació inclusiva *pertany(SIDA, Inmunodepressió)*, en el cas que el malalt tingués una SIDA només la metaregla més específica seria aplicada, és a dir la metaregla:

**M03024 Si SIDA Llavors Mòduls a visitar**

Bacteriana-atípica Pneumocistis\_carinii TBC  
 Citomegalovirus Criptococ Nocardia Aspergillus  
 Pneumococ Enterobacteries  
**Amb certesa possible**

en lloc de la metaregla:

**M03025 Si Inmunodeprimit Llavors Mòduls a visitar**

Bacteriana-atípica Enterobacteries Pseudomonas  
 Pneumococ Anaerobis Hemofilus estafilococ  
 legionella Pneumocistis\_carinii TBC Citomegalovirus  
 Criptococ Nocardia Aspergillus ;  
**Amb certesa moderadament\_possible**

Les metaregles són sempre avaluades amb encadenament endavant, el seu algorisme d'avaluació és molt similar al ja presentat pel cas de les regles.

La semàntica denotacional de la fase d'interpretació d'aquestes metaregles és explicat a l'apèndix II.

**5.7.2.1 Jerarquies pures versus jerarquies mixtes**

L'ús de lligams mixtes per construir metaregles de construcció d'estratègies presenta els següents inconvenients:

- a) Comporta una explosió combinatòria de metaregles generadores d'estratègia, essent moltes d'elles d'utilitat dubtosa.
- b) Es tendeix a fer metaregles tant específiques que les condicions de les seves premisses només les podien satisfer grups molt restringits de casos.
- c) Es perd la capacitat de controlar quines metaregles s'apliquen atesa l'elevada combinació de fets que poden intervenir en les condicions de les metaregles. L'expert està més atent a controlar quines metaregles i en quin ordre s'apliquen que en el contingut d'aquestes.

L'ús de jerarquies pures múltiples comporta els avantatges propis d'una més acurada estructuració del coneixement. Les estratègies generades a partir de metaregles amb condicions que continguin fets d'una sola jerarquia permeten

- a) Mostrar més clarament quina és l'estratègia generada a partir d'un fet relevant.
- b) Controlar realment quina serà l'aplicació de les metaregles atès que es podrà establir clarament quin és el grau de certesa d'aquestes. D'aquesta manera l'expert podrà donar més rellevància a metaregles que tinguin a les seves premisses condicions amb fets pertanyents a una determinada jerarquia pura per sobre de metaregles relacionades amb una altra.
- c) Impedir l'explosió combinatòria de metaregles.

Amb una millor estructuració de les metaregles que generen estratègies l'expert pot concentrar-se més en el contingut de la metaregla tot deixant al sistema el control de l'aplicació d'aquestes. Amb això s'aconsegueix un control més eficient i sistemàtic.

### 5.7.3 Metaregles referents a excepcions

La sintaxi que té una metaregla referent a excepcions és la següent:

```

mrx.           ::= id-meta Si premissa Llavors excepció
excepció      ::= solució definitiva fet-id |
                   avortar

```

Aquestes metaregles s'apliquen en certes situacions especials que provoquen l'acabament del procés. Aquestes situacions són:

**Solució definitiva:** Les metaregles que tenen una **solució definitiva** a la part de conclusió són utilitzades per representar patrons de l'univers de discurs que ens donen una prova absoluta sobre la solució d'un problema. En medicina aquests patrons són el que s'anomena *patrons patognomònics*. L'aplicació d'aquestes metaregles provoca l'aturada immediata del sistema. L'única solució presentada a l'usuari serà la que indiqui la metaregla.

Exemple: L'aparició d'una lesió cutània de tipus *ectima gangrenosum* indica una evidència absoluta de la presència a la sang de Pseudomonas. La metaregla que indica això és:

M02005 **Si** *ectima gangrenosum* **Llavors**  
**Solució definitiva** pseudomonas

**Avortar:** Les metaregles que conclouen avortar serveixen per modelitzar aquelles situacions que fan impossible la continuació del sistema:

\* *Casos fora del domini de l'aplicació.*

\* *Inconsistències greus entre les dades d'entrada.*

Exemple 1: L'aplicació PNEUMON-IA està dissenyada pel diagnòstic de les malalties extrahospitalàries en adults. Per tant una edat inferior als 14 anys en un pacient indica la no aplicabilitat del sistema:

**M02012 Si edat < 14 Llavors avortar**

Exemple 2: En presència d'un xoc sèptic (situació d'irrigació deficient dels òrgans vitals) les dades de l'hemograma (recompte i diferenciació de les cèl·lules sanguínies) han de presentar alteracions notables. De no ser així ens trobarem en presència d'una inconsistència greu de les dades. La metaregla següent expressa aquest fet:

**M03026 Si xoc sèptic i leucòcits < 10000 i leucòcits > 6000  
i no desviació\_esquerra Llavors avortar**

## 5.8 Nivell heurístic.

Aquest nivell modelitza coneixements heurístics referents al procés de resolució de problemes. Els elements de representació de domini són els plans d'actuació, i els elements de control són les metaregles de construcció dels plans.

### 5.8.1 Plans d'actuació

Les estratègies que han estat presentades a l'apartat anterior fan referència únicament a l'ordre d'anàlisi de les diferents solucions del sistema. Atès que, en un instant de l'execució, més d'una metaregla pot haver estat activada, és necessari una altra capa de meta-control, l'última, que sigui responsable de coordinar totes les estratègies i d'expressar els elements de control necessaris per a l'execució de les estratègies. L'element de representació de domini d'aquest últim nivell l'anomenem *pla*



*d'actuació*. Hem utilitzat la paraula *pla* per analogia amb els sistemes de planificació, un *pla d'actuació* agrupa totes les decisions i accions elementals a prendre per resoldre el problema en curs.

D'alguna manera aquest nivell i l'anterior tenen en comú la seva funció relacionada amb la selecció del tipus de mètode de resolució de problemes, i es podria entendre que els dos tipus de coneixements utilitzats són de caire heurístic. La diferència entre els dos és la següent: el nivell hipotètic té la missió de seleccionar subespais de l'espai de solucions i el nivell heurístic és l'encarregat de fer la cerca dins d'aquests subespais i establir les combinacions necessàries entre ells.

En tot moment el sistema segueix un únic pla d'actuació. Aquest pla conté informació únicament de control i és qui regeix tot el comportament d'alt nivell: Tipus de cerca a utilitzar, Mòduls a visitar i Moments d'aturada. Aquests elements de control són els següents:

- a) (*id-mòdul cerca*)
- b) **test final**

El control **test final** implica realitzar una avaluació de les metaregles referents a acabament que, en cas que alguna d'elles s'avalués amb èxit, comportaria l'acabament del procés de solució (veure 5.9.4). Aquests elements de control organitzats en forma de llistes representen el nucli dels plans. L'execució d'un pla representa l'avaluació dels elements de control seqüencialment fins el final, o bé fins que l'avaluació d'algun d'ells determini l'acabament del procés.

La representació d'un pla és la següent:

Def Anomenem *Pla d'actuació* a tot "frame" del següent tipus:

**Frame Pla**

- slot certesa:**  $l \in L$
- slot unitats:**  $(u_1, u_2, \dots, u_p)$  essent  $u_i$  elements de control.
- slot mòduls eliminats:**  $\{ m / m \in M \}$
- slot mòduls visitats:**  $\{ m / m \in M \}$
- slot conjunt suport:**  $\{ \beta / \beta \text{ és mre } \}$
- slot condició suport:** premissa( $\beta_1$ ) i premissa( $\beta_2$ ) i ... i premissa( $\beta_n$ )  
on  $\beta_1, \beta_2, \dots, \beta_n \in$  conjunt suport.
- slot combinadores:**  $\{ \beta / \beta \text{ és mrp } \}$

La *certesa* representa el grau de credibilitat del pla d'actuació. Es determina com a conjunció dels valors de certesa propagats per les metaregles pertanyents al *conjunt suport*. Posat que les funcions que modelitzen els connectius "i" en el contexte de la lògica difusa són funcions no creixents, afegir noves estratègies a un pla d'actuació implicarà una possible degradació en el valor de certesa del pla. D'alguna manera, el que s'intenta és afavorir aquells plans que estiguin suportats per poques estratègies, considerant que són plans més focalitzats, i per tant explicativament més convincents.

Les *unitats* ja han estat explicades. Els *mòduls eliminats* corresponen a aquelles tasques no executables pel sistema.

Els *mòduls visitats* indiquen la història de l'execució fins al moment present. Quins mòduls han estat avaluats.

El *conjunt suport* és el conjunt de metaregles de tipus mre que han estat definidores de les estratègies combinades en el pla. Aquest conjunt no és el mateix que el de totes les metaregles mre que han estat aplicades, atès que les metaregles encarregades de la combinació d'estratègies poden haver-ne descartat alguna.

La *condició de suport* representa la justificació conjunta del pla des del punt de vista dels coneixements de domini. Quantes menys metaregles mre s'hagin tingut en compte, més explicativa serà aquesta justificació.

Els *combinadors* són les metaregles del nivell heurístic que han combinat les diferents estratègies.

Si no obliguem a que el camp de combinadores sigui no buit tindriem que una *estratègia* és un *pla*.

## Def

Sigui

*id-meta* si premissa llavors constructors amb  
certesa  $\beta$

la metaregla que generi l'estratègia  $s = ((MV, ME, MD), \gamma)$ .

llavors el pla següent és defineix com a pla equivalent a l'estratègia  $s$ :

traduir\_pla(s) =

**frame** s

slot certesa:  $\gamma$

slot unitats: LU

slot mòduls eliminats: ME

slot mòduls visitats:  $\emptyset$   
 slot conjunt suport:  $\{ m \}$   
 slot condició suport: premissa(m)  
 slot combinadores:  $\emptyset$

essent  $LU = ((M_1 B_1) \dots (M_n B_n))$  on  
 $APLICONS^1 \llbracket constructors \rrbracket$  primer( $S_0$ ) =  $(M_1 \dots M_n)$  i  $B_i$  les cerques de  
 cada

$M_i$ ,

Nota: Utilitzarem la paraula estratègia en lloc de la paraula pla en la resta del capítol quan el pla correspongui directament a l'aplicació de la funció traduir\_pla sobre una estratègia concreta.

La sintaxi d'aquest nivell és la següent:

<i>Nivell-heurístic</i>	<b>::= Nivell Heurístic</b>
	<b>Generació de plans</b>
	<i>lmp</i>
	<b>Acabament</b>
	<i>lmra</i>
	<b>finivell</b>
<i>lmp</i>	<b>::= mmp lmp   <math>\lambda</math></b>
<i>lmra</i>	<b>::= mra lmra   <math>\lambda</math></b>

Les metaregles *mmp* i *mra* estan explicades en els següents apartats.

### 5.8.2 Generació de Plans d'actuació. Metaregles referents a plans

En un instant determinat de l'actuació del sistema és possible que més d'una metaregla mre pugui ésser disparada. Això ens produirà més d'una possible estratègia a seguir. Aquest nivell heurístic s'encarrega de la combinació de totes les estratègies actives en un moment qualsevol en un pla d'actuació.

Aquest nivell utilitza com a domini l'univers d'estratègies  $\Sigma_M$  i com a rang

---

<sup>1</sup> Veure apèndix II.

l'univers de plans d'actuació explicat a l'apartat anterior. Posat que, com s'ha vist, les estratègies són incloses dins els plans, definirem aquest nivell amb domini també sobre l'univers de plans, ja que això ens permetrà demanar propietats més interessants a aquests operadors. Aquest nivell de metaregles pot ser definit de forma semblant a les metaregles de definició d'estratègies.

**Def** Una metaregla referent a pla (*mrp*) és un operador del següent tipus:

$$\delta: P_M \times P_M \rightarrow P_M$$

La sintaxi que té una metaregla d'aquest tipus és la següent. A l'hora d'expressar els constructors farem referència als plans que són combinats amb els noms de *P*, que al mateix temps poden ser el resultat de la combinació d'altres plans.:

<i>mrp</i>	::= <i>id-meta</i> Si <i>premissa_pla</i> <b>Llavors</b> <i>combinador</i>
<i>combinador</i>	::= <b>Ockham</b> ( <i>def-pla</i> , <i>def-pla</i> )   <b>Prioritari</b> ( <i>def-pla</i> , <i>def-pla</i> )   <b>Primer</b> ( <i>def-pla</i> , <i>def-pla</i> )
<i>premissa_pla</i>	::= <i>premissa_certesa</i>   <i>premissa_objectius</i>   <b>compatibles</b> ( <i>def-pla</i> , <i>def-pla</i> )
<i>premissa_certesa</i>	::= <b>certesa</b> ( <i>def-pla</i> ) <i>operador_rel</i> <b>certesa</b> ( <i>def-pla</i> )
<i>premissa_objectius</i>	::= <b>generador</b> <i>operador_conjunt</i> <b>generador</b>
<i>generador</i>	::= <b>nil</b>   <b>objectius</b> ( <i>def-pla</i> )   <b>eliminats</b> ( <i>def-pla</i> ) <i>generador</i> <i>operació_conjunt</i> <i>generador</i>
<i>def-pla</i>	::= <i>combinador</i>   <i>variable</i>
<i>operador_conjunt</i>	::= =   ≠   ⊂   ⊆   ⊄
<i>operació_conjunt</i>	::= ∪   ∩   /

Les premisses d'aquestes metaregles són diferents de les premisses de les regles i de la resta de metaregles del sistema. Estan constituïdes per dos subtipus, les comparacions de les certes i les comparacions conjuntistes dels mòduls objectiu i dels mòduls eliminats dels plans d'actuació que s'han de combinar. Tanmateix, es defineix un predicat *compatibles*, que serà cert quan els mòduls de les unitats dels plans i els mòduls eliminats respectin la definició de compatibilitat establerta per les estratègies.

Els combinadors defineixen de quina manera podrem combinar dos plans d'actuació en un únic pla. Els tres combinadors definits fins ara són Ockham, Prioritari i

Primer:

### Ockham(P, P)

Aquest combinador aplica la heurística de la navalla d'Ockham. Segons aquesta heurística, quan des de dues fonts diferents obtenim diferents conjunts de mòduls que poden contenir la solució, el més probable és que la solució es trobi a la intersecció dels dos conjunts.

L'efecte d'aquest combinador és estructurar els plans com a seqüències de mòduls separats per **test final**. Imaginem que A i B són els objectius de dues estratègies X i Y, Llavors els objectius del pla combinació Ockham(X, Y) són:

$$A \cap B \text{ test final } A \cup B / A \cap B$$

La intersecció entre els dos conjunts es fa respectant l'ordre de l'estratègia amb un grau de certesa més alt. En cas que les certeses siguin les mateixes l'ordre dependrà de l'ordenació de les metaregles que han generat les estratègies.

Per exemple la següent metaregla

**M04025** Si  $\text{certesa}(x) > \text{certesa}(y)$  i  $\text{objectius}(x) \cap \text{objectius}(y) \neq \emptyset$  Llavors  
**Ockham(x, y)**

aplicada sobre els plans

**frame x**

**slot certesa:** Lleugerament\_possible

**slot unitats**<sup>1</sup>: micoplasma viruses viruses-bact clamídia febre-Q  
 tbc nocàrdia criptococ citomegalovirus p-carinii.

**slot mòduls eliminats:**  $\emptyset$

**slot mòduls visitats:** Bact-atip

**slot conjunt suport:** { R25883 }

**slot condició suport:** no (Bacteriana és possible) i

---

<sup>1</sup> En els exemples no representarem el tipus de cerca dins les unitats per fer més comprensibles els exemples.

Atípica és possible

slot combinadores:  $\emptyset$

**frame y**

slot certesa: força\_possible

slot unitats: bact-atip pneumococ legionella enterobactèries  
criptococ nocàrdia p-carinii tbc pseudomones  
aspergillus citomegalovirus

slot mòduls eliminats:  $\emptyset$

slot mòduls visitats:  $\emptyset$

slot conjunt suport: { R25903 }

slot condició suport: immunodeprimit

slot combinadores:  $\emptyset$

generaria el pla

**frame P**

slot certesa: força\_possible { =  $\vee$  (força\_possible,  
lleugerament\_possible) }

slot unitats: criptococ nocàrdia p-carinii tbc citomegalovirus  
test final pneumococ legionella enterobactèries  
pseudomones aspergillus micoplasma viruses  
viruses\_bact clamídia febre-Q

slot mòduls eliminats:  $\emptyset$

slot mòduls visitats: Bact-atip

slot conjunt suport: { R25903 R25883 }

slot condició suport: no (Bacteriana és possible) i

Atípica és possible i immunodeprimit

slot combinadores: { M04025 }

CADUCEUS utilitza un operador que és una aplicació d'aquesta heurística: Si s'investiga una malaltia d'un òrgan concret i es cerquen les causes d'una troballa clínica, és possible combinar ambdós problemes i concentrar-se en les malalties que puguin afectar l'òrgan i causar la troballa. MILORD, a diferència de CADUCEUS, permet que aquestes heurístiques siguin declaratives, i sigui l'expert qui determini en cada moment quan s'ha d'aplicar i quan no.

**Prioritari(P, P)**

Aquest operador determina que  $P_1$  és el primer pla que s'ha de utilitzar, i en cas que a la seva finalització no estiguem satisfets amb la solució trobada intentarem  $P_2$ . Per exemple la metaregla

M04032 Si  $\text{certesa}(x) > \text{certesa}(y)$  i  $\text{objectius}(x) \cap \text{objectius}(y) = \emptyset$   
**Llavors Prioritaria**(x, y)

aplicada sobre els plans

```

frame x
:
:   slot certesa: Lleugerament_possible
:   slot unitats: micoplasma viruses viruses-bact clamídia febre-Q
:                   tbc nocàrdia criptococ citomegalovirus p-carinii.
:   slot mòduls eliminats:  $\emptyset$ 
:   slot mòduls visitats: bact-atip
:   slot conjunt suport: { R25883 }
:   slot condició suport: no (Bacteriana és possible) i
:                           Atípica és possible
:   slot combinadores:  $\emptyset$ 

```

```

frame y
:
:   slot certesa: possible
:   slot unitats: bact-atip anaerobis pneumococ enterobactèries
:                   str-pio
:   slot mòduls eliminats:
:   slot mòduls visitats:  $\emptyset$ 
:   slot conjunt suport: { R25890 }
:   slot condició suport: broncoaspiració
:   slot combinadores:  $\emptyset$ 

```

generaria el pla

```

frame P
:
:   slot certesa: possible { = certesa( $P_2$ ) }
:   slot unitats: bact-atip anaerobis pneumococ enterobactèries

```

str-pio test final micoplasma viruses  
 viruses-bact clamídia febre-Q tbc  
 nocàrdia criptococ citomegalovirus p-carinii

slot mòduls eliminats:  $\emptyset$

slot mòduls visitats:  $\emptyset$

slot conjunt suport: { R25890 R25883 }

slot condició suport: broncoaspiraci i no (Bacteriana és possible) i Atípica és possible

slot combinadores: { M04032 }

### Primer(P, P)

Aquesta estratègia correspon simplement a la projecció sobre el primer element.

$$\text{Primer}(P_1, P_2) = P_1$$

Aquest combinador és necessari quan volem rebutjar un dels dos plans en conflicte. Per exemple vegi's les metaregles:

M04026 Si certesa(x) > certesa(y) i no compatibles(x, y)

**Llavors Primer(x, y)**

M04028 Si certesa(x) > certesa(y) i objectius(x)  $\subseteq$  objectius(y)

**Llavors Primer(x, y)**

la primera tria el pla més cert quan hi ha una incompatibilitat entre plans, i la segona tria els plans més específics quan la seva certesa és superior.

Els operadors definits a l'arquitectura de MILORD són els que l'experiència en el desenvolupament d'aplicacions ens ha proporcionat. La definició declarativa per part dels usuaris dels combinadors i la seva semàntica és un pas endavant a realitzar en el futur.

Volem que compleixin les propietats exigides per raons evidents. L'associativa representa el fet que l'estratègia en un moment donat sigui independent de l'ordre en que hem aplicat diferents mre contenint el mateix *combinador* a la conclusió. L'element neutre és necessari per definir l'estratègia inicial, que combinada amb qualsevol altre no la modifiqui.



Resta definir la semàntica dels operadors i comprovar les propietats exigides.

### 5.8.3 Semàntica dels combinadors

Descriurem únicament la semàntica d'interpretació dels combinadors d'aquest tipus de metaregles. L'avaluació de les premisses de les metaregles és similar a la de les regles.

#### 5.8.3.1 Objectes semàntics

##### I Plans

Domini  $p \in P = L \times \text{Llista de } \{M + \text{test final}\} \times \text{Llista de } M \times$   
 Llista de *id-meta*

{ Simplifiquem la definició de pla d'actuació. per contenir  
 únicament: certesa, unitats (sense les cerques), mòduls eliminats  
 i combinadores }

##### Operacions

$\text{pla\_buit} : P$

$\text{pla\_buit} = \text{traduir\_pla}(S_0)$   
 ( essent  $S_0$  l'estratègia buida )

$\text{compatibles} : P \times P \rightarrow B$

$\text{compatibles}(a, b) = \text{segon}(a) \cap \text{tercer}(b) = \emptyset \wedge$   
 $\text{segon}(b) \cap \text{tercer}(a) = \emptyset$

##### II Taula de plans

Domini  $\text{tp} \in TP = \text{idpla} \rightarrow P$

##### Operacions

$\text{taula\_pla\_buida} : TP$

##### III Llista d'interseccions

Domini  $\lambda : ( (u_1 v_1), (u_2 v_2), \dots, (u_n v_n) ),$   
 $v_i > v_j \forall i, j \ i < j$

##### Operacions

$\text{add\_lu} : \lambda \times \lambda \rightarrow \lambda$

$\text{add\_lu}(\lambda_1, \lambda_2) = ( (u_1, n_1+m_1), (u_2, n_2+m_2), \dots, (u_p, n_p+m_p) )$

$n_i = \begin{cases} x & \text{si } (u_i x) \in \lambda_1 \\ 0 & \text{Altrament} \end{cases}$

$$m_i = \begin{cases} x & \text{si } (u_i x) \in \lambda_2 \\ 0 & \text{Altrament} \end{cases}$$

Acomplint-se  $n_i + m_i \geq n_j + m_j$ , si  $i < j$

generar: Llista de  $\{M + \text{test final}\} \rightarrow \lambda$

generar(l) = X a *reverse*(l, m), *gen*(m,Y,l), *reverse*(Y, X)

*gen*([], [], \_).

*gen*([*test \_final* | R], S, N) :- M is N+1, *gen*(R,S,M).

*gen*([U | R], [[U, N] | S], N) :- *gen*(R, S, N).

regenerar:  $\lambda \rightarrow$  Llista de  $\{M + \text{test final}\}$

regenerar(l) = X a *regen*(l,X)

*regen*([], []).

*regen*([[U, N]], [U]).

*regen*([[U, N], [V, N] | R], [U, V | S]) :- *regen*(R, S).

*regen*([[U, N], [V, M] | R], [U, *test\_final*, V | S]) :-  
*regen*(R, S).

### 5.8.3.2 Equacions semàntiques

COMBINADOR: *combinador*  $\times$  TP  $\times$  id-meta  $\rightarrow$  (idpla  $\times$  TP)

COMBINADOR[[ *Prioritari*(P<sub>1</sub>, P<sub>2</sub>) ]] tp id-meta = (idpla tp')

pla<sub>1</sub> = PLA[[ P<sub>1</sub> ]] tp id-meta

pla<sub>2</sub> = PLA[[ P<sub>2</sub> ]] tp id-meta

p = PRIOR[[ ]] pla<sub>1</sub> pla<sub>2</sub> id-meta

*idpla* = gensym("Pla-")

tp' = [*idpla*  $\rightarrow$  p]tp

COMBINADOR[[ *Primer*(P<sub>1</sub>, P<sub>2</sub>) ]] tp id-meta = (*idpla* tp')

pla<sub>1</sub> = PLA[[ P<sub>1</sub> ]] tp id-meta

pla<sub>2</sub> = PLA[[ P<sub>2</sub> ]] tp id-meta

p = PRIMER[[ ]] pla<sub>1</sub> pla<sub>2</sub> id-meta

*idpla* = gensym("Pla-")

tp' = [*idpla*  $\rightarrow$  p]tp

COMBINADOR[[ *Ockham*(P<sub>1</sub>, P<sub>2</sub>) ]] tp id-meta = (*idpla* tp')

```

pla1 = PLA[[ P1 ]] tp id-meta
pla2 = PLA[[ P2 ]] tp id-meta
p = OCKHAM[[ ]] pla1 pla2 id-meta
idpla = gensym("Pla-")
tp' = [idpla → p]tp
⇔ ERROR SI no compatibles(tp(P1), tp(P2))
    MISSATGE "Plans no compatibles." id-meta
    RETORN COMBINADOR[[ Primer(P1, P2) ]] tp id-meta
    { si els plans són incompatibles canviem l'operador
      Ockham per l'operador Primer }

```

PLA:  $def-pla \times TP \times id-meta \rightarrow P$

```

PLA[[ variable ]] tp id-meta = p
p = tp(variable)

```

```

PLA[[ combinador ]] tp id-meta = p
(idpla tp') = COMBINADOR[[ combinador ]] tp id-meta
p = tp'(idpla)

```

PRIOR:  $P \times P \rightarrow P$

```

PRIOR[[ ]] tp1 tp2 id-meta = tp3
cert = ^ (primer(tp1), primer(tp2))
objec = segon(tp1) • (test final) • segon(tp2)
elimin = tercer(tp1) ∪ tercer(tp2) / primer(tp1)
meta = quart(tp1) • quart(tp2) • (id-meta)
tp3 = ( cert, objec, elimin, meta )

```

PRIMER:  $P \times P \times id-meta \rightarrow P$

```

PRIMER[[ ]] tp1 tp2 id-meta = tp3
tp3 = (primer(tp1), segon(tp1), tercer(tp1), quart(tp1) • (id-meta))

```

OCKHAM:  $P \times P \times \text{id-meta} \rightarrow P$

```

OCKHAM[ ] tp1 tp2 id-meta = tp3
  cert =  $\vee(\text{primer}(tp_1), \text{primer}(tp_2))$ 
  objec = regenerar(Add_lu(generar(segona(tp1)), generar(segona(tp2))))
  elimin = tercer(tp1)  $\cup$  tercer(tp2)
  meta = quart(tp1)  $\bullet$  quart(tp2)  $\bullet$  (id-meta)
  tp3 = ( cert, objec, elimin, meta )

```

#### 5.8.4 Condicions d'acabament. Metaregles d'acabament.

El procés de classificació de MILORD s'atura en un dels tres casos següents:

- 1.- El pla d'actuació s'ha exhaurit.
- 2.- S'ha executat una metaregla de terminació.
- 3.- S'ha produït una excepció.

El primer cas no requereix una especial atenció i el tercer ja ha estat explicat a l'apartat del nivell hipotètic. Les condicions de terminació formen part del coneixement de control que l'expert pot definir. Aquest coneixement està estructurat en les metaregles d'acabament de la BC.

Les metaregles de terminació, corresponents al segon cas, són les responsables de l'acabament del procés de classificació quan es considera que la solució és acceptable i no es considera necessari exhaurir l'estratègia en curs.

***mra* ::= id-meta si premissa llavors acabar**

Quan, en el pla d'actuació en curs, s'indica que s'analitzi si s'ha arribat o no al final del procés mitjançant una expressió del tipus:

**test final**

les metaregles *mra* són avaluades. Si alguna de les metaregles és disparada el procés s'atura.

Dues metaregles d'acabament són les següents. La primera mostra una satisfacció amb els resultats diagnòstics, i la segona mostra clarament que cap altre possibilitat diagnòstica pugui ser correcta:

M03005 Si Pneumocistis\_Carinii i no (Tuberculosi és possible) i  
no (Criptococcus és possible) **Llavors acabar**

M03022 Si neutropènia i Pseudomonas és possible **Llavors acabar**

#### 5.8.4.1 Resultats del sistema

Un cop el sistema ha acabat l'execució, el resultat consisteix en la llista de les solucions trobades ordenada de forma decreixent segons els valors de certesa lingüístics amb que s'ha deduït cadascuna d'elles.

Vegi's un exemple extret de la tesi doctoral en medicina de n'Albert Verdaguer (en preparació) sobre un cas real utilitzat en la validació del sistema expert PNEUMON-IA:

Resultats de l'aplicació del sistema:	
Objectiu	Valor
Pneumococ	Força possible
Legionella	Possible
BGN	Possible
Anaerobis	Lleugerament possible

#### 5.8.5 Solucions úniques versus solucions múltiples

El processos de *classificació determinista* són aquells en que tot element a classificar pertany a una única classe. Són exemples els sistemes on hi ha una taxonomia completament definida: classificació de vegetals, d'animals, etc. En aquests sistemes el resultat de la classificació és únic i, per tant, quan s'ha trobat la classe a la qual pertany l'objecte podem aturar el procés. Aquests processos es compliquen quan la incertesa en la pertinença d'un element a una classe fa que la solució al problema no pugui ser determinada exactament. El resultat serà, en aquests casos, un conjunt ordenat de possibles solucions.

En processos de *classificació no determinista* el problema és encara més complexe, perquè a més de la possibilitat de tenir incertesa en la pertinença d'un fet a una classe podem tenir solucions múltiples. Un exemple molt evident és el cas de les

aplicacions mèdiques, on un malalt pot tenir més d'una disfunció al mateix temps. Tenint en compte que aquestes disfuncions poden tenir interaccions, el disseny d'aquests sistemes suposa una complexitat netament superior a la dels sistemes de classificació discreta.

L'arquitectura de MILORD proporciona dos mecanismes que faciliten la tasca de programar aquestes aplicacions amb solucions múltiples : la definició per l'expert de les *condicions d'acabament*, i la *definició d'estratègies*.

Les *condicions d'acabament* permeten establir exactament en quins moments una solució és satisfactòria o no. És a dir, si unes assignacions de certes a determinades solucions es consideren prou bones podrem aturar el sistema en aquell moment sense haver de cercar més elements.

La *definició d'estratègies* permet plantejar-se més d'una solució al mateix temps. Això combinat amb la capacitat de decidir el moment d'acabament dóna una gran flexibilitat de programació.

## 5.9 Cicle de control a MILORD

En aquest apartat donarem el cicle de control global que segueix MILORD, tenint en compte tots els nivells i les interaccions entre ells.

La figura 5.6 representa els diferents nivells de l'arquitectura dividits en elements del nivell de domini i elements del nivell de control.

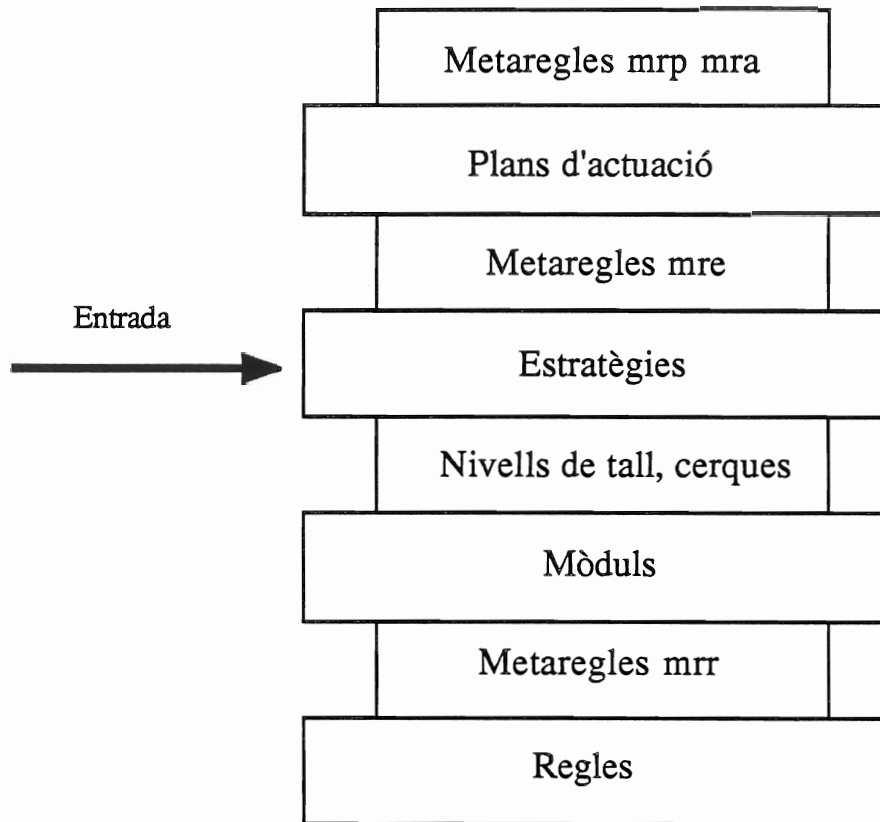


Fig 5.6 Cicle de control a MILORD.

Així, el punt d'entrada a MILORD és al nivell hipotètic, on inicialment existeix una estratègia per defecte. Es selecciona el primer mòdul de l'estratègia i es baixa al nivell estructural, on, utilitzant els paràmetres de control associats al mòdul, tall i cerca, s'inicia la seva avaluació. Es selecciona el conjunt de regles conflicte i es baixa al nivell associatiu on les metaregles mrr filtren el conjunt de regles per decidir les que són actives, i es passa a ordenar les regles segons els criteris ja esmentats: subsumpció, certesa i ordre. Es selecciona la primera regla i s'intenta aplicar (la forma dependrà del tipus d'encadenament). Aquest nivell acaba quan s'obté un fet de l'exterior o s'aplica una regla, moment en el qual tornem al nivell hipotètic on intentem aplicar alguna metaregla mre, per tal de veure si l'estratègia es modifica. Si cap regla és aplicable tornem a baixar, repetint el cicle ja descrit. Si alguna metaregla és aplicada, el nivell heurístic és el responsable de combinar la nova (o noves) estratègia generada amb l'estratègia en curs per tal d'obtenir el nou pla. Un cop obtingut, el nivell hipotètic selecciona el primer mòdul i repeteix el cicle esmentat. Si la primera acció del nou pla és un test final llavors s'avaluen les metaregles mra per veure si s'ha produït alguna

condició d'acabament; de ser així aturariem el cicle; en cas contrari seleccionariem la següent acció del pla i continuariem com s'ha explicat. La figura 5.7 mostra la interacció entre els diferents nivells de l'arquitectura de MILORD.

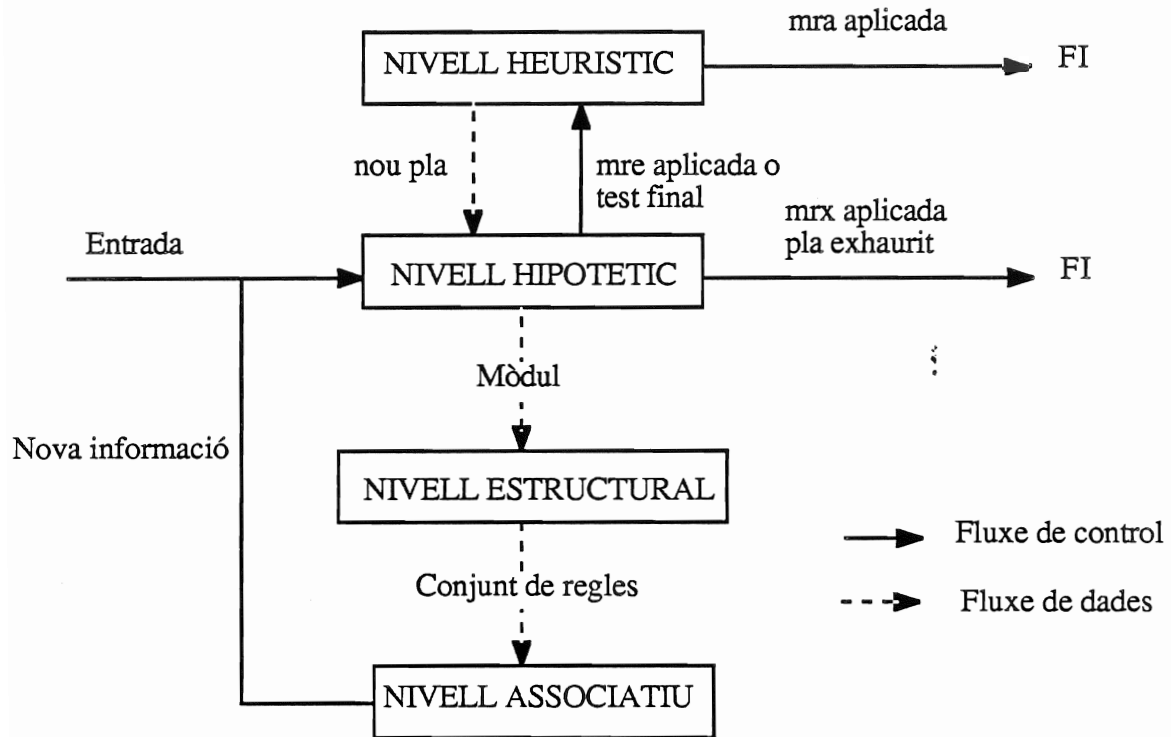


Fig. 5.7 Nivells de domini i de control de MILORD.



# 6 Aplicacions

OMNIS TURBA MALA, MEDICORUM PESSIMA

*Aforisme medieval*

Fins a l'escriptura d'aquesta memòria s'han desenvolupat quatre aplicacions utilitzant l'entorn MILORD. A continuació farem un repàs a totes elles veient-ne el seu estat i destacant els punts de l'arquitectura de MILORD que els han estat més importants.

## 6.1 PNEUMON-IA

La descripció de l'aplicació PNEUMON-IA [López de Mántaras et al, 1988], [Verdaguer et al, 1989], s'ha fet molt més en detall que les altres. Això s'ha fet per dues raons:

- 1.- Ha estat l'aplicació sobre la qual he estat treballant en col·laboració amb l'Albert Verdaguer durant més temps: dos anys i mig.
- 2.- Es una aplicació molt paradigmàtica del tipus d'aplicacions que considerem es poden desenvolupar millor utilitzant MILORD. És de gran complexitat i requereix un bon tractament de la incertesa i la modularitat. A més explota tots els recursos que l'arquitectura de MILORD proporciona.

La resta d'arquitectures s'ha descrit molt més breument, destacant-ne els seus trets més característics.

Haig d'agrair especialment a l'Albert Verdaguer la seva ajuda en la redacció d'aquest apartat.

### 6.1.1 Justificació del sistema expert PNEUMON-IA

El problema de la dificultat diagnòstica d'algunes malalties té una gran importància sanitària quan aquestes són freqüents i potencialment greus.

Les pneumònies són malalties freqüents. Es calcula que entre un 12.5% i un 15% de tots els pacients que demanen atenció mèdica per una malaltia aguda, pateixen una infecció de les vies respiratòries inferiors. A l'any 1983 a Espanya s'avaluà el nombre d'ingressos anuals per pneumònia en centres sanitaris en 76.9 per 100.000 habitants. Aquestes xifres representen només una part de la incidència real, ja que molts d'aquests processos no s'ingressen en els centres sanitaris a l'afectar a persones prèviament sanes que són tractades domiciliàriament.

El tractament d'un malalt afecte de pneumònia representa una urgència mèdica, ja que la mortalitat d'aquesta malaltia ve determinada d'una forma decisiva per l'inici precoç d'una teràpia dirigida contra els possibles agents etiològics causals. L'establiment del diagnòstic etiològic d'una pneumònia és difícil.

En la majoria d'ocasions es tracta d'una urgència mèdica en el sentit que cal iniciar un tractament mèdic sense esperar els resultats de les proves diagnòstiques. En aquesta situació és essencial utilitzar acuradament tota la informació que es disposa.

#### 6.1.1.1 Establiment d'hipòtesis

Així en el cas d'una pneumònia extrahospitalària es determinarà en primer lloc si és bacteriana, atípica o es dona en un contexte molt específic. Dintre d'una gran hipòtesi, el metge intenta aprofundir el coneixement de la malaltia que pateix el pacient plantejant-se subobjectius més restringits com són, en el cas de la pneumònia, el tipus de germen. En aquest procés coexisteixen diverses hipòtesis amb diferent grau de certesa.

La incertesa de les hipòtesis es reafirma quan revisem les dades bibliogràfiques que parlen de la freqüència dels diversos agents etiològics com a causa de pneumònia. Si es revisen les freqüències de les pneumònies comunitàries per legionella es trobaran incidències de la infecció per aquest agent que oscil·len entre el 0.8% i el 30%. Aquestes diferències són degudes a diferents factors, com ara el lloc geogràfic on es va fer l'estudi, el període de temps que comprén, la selecció prèvia dels malalts -només aquells que ingressaren a l'hospital, o només les pneumònies atípiques o d'etiologia desconeguda-, les tècniques emprades en els diferents estudis i els diferents criteris en la interpretació dels resultats.

Aquests valors d'incidència de legionella tan diferents fan que les aproximacions exclusivament freqüentistes (bayesianes) siguin de dubtosa utilitat per establir probabilitats d'un determinat tipus de pneumònia en un determinat malalt.

### 6.1.1.2 Tractament sota incertesa

Encara que en la gran majoria dels centres sanitaris que al nostre país atenen als malalts afectes de pneumònia es poden realitzar exploracions complementàries, aquestes no informen, generalment, de l'etiologia d'aquestes.

El metge que atén al malalt ha de fer una aproximació etiològica per iniciar un tractament. Aquestes aproximacions s'efectuen sota incertesa. La incertesa pot persistir durant tot el procés terapèutic, ja que en aquestes malalties el diagnòstic definitiu no és freqüent.

L'aproximació diagnòstica sota incertesa fa que el clínic estableixi algunes hipòtesis de treball molt generals i intenti refinar-les posteriorment.

Quan el metge hagi exhaurit tots els mitjants que té al seu abast per confirmar o refutar aquestes hipòtesis diagnòstiques disposarà d'un llistat de possibilitats. El grau de creença en aquestes possibilitats determinarà en bona part la seva conducta.

En el cas de les pneumònies, com en d'altres malalties infeccioses, no tan sols jugarà un paper important per al tractament la creença en aquestes hipòtesis diagnòstiques, sinó que es tindrà molt en compte l'estat del malalt. En efecte, en un malalt moderadament greu que pensem tèn moltes possibilitats de tenir una pneumònia pneumocòcica i poques que sigui per enterobactèries es farà un tractament d'espectre reduït pel pneumococ tot utilitzant la penicil·lina que és un antibiòtic molt eficaç en front d'aquest gèrm. Si, tenint les mateixes hipòtesis diagnòstiques, el malalt està molt greu, el més probable que farem serà donar un antibiòtic d'ampli espectre com una cefalosporina de tercera generació que cobreixi per igual els dos germens esmentats: *el metge no pot arriscar-se a observar l'evolució del malalt.*

### 6.1.1.3 Conclusió

En resum, el diagnòstic i tractament de les pneumònies és un exemple paradigmàtic, de com el metge, en certes situacions ha de decidir certes actuacions amb dades incompletes i, el que és més important, sota incertesa. I, en aquest camp, no compta només el coneixement que pot adquirir de fonts reconegudes, com les bibliogràfiques, sinó que és important el grau d'experiència que tingui en manejar aquest tipus de malalts, ja que d'aquests darrers coneixements es derivaran les heurístiques a

aplicar per tal que el diagnòstic i tractament del malalt amb pneumònia sigui el més segur i efectiu possible.

L'objectiu és, doncs, elaborar una primera aproximació diagnòstica que serveixi d'ajuda al metge en el tractament de les pneumònies extrahospitalàries en els malalts adults. Es tracta d'orientar les pneumònies en les primeres 24 hores, és a dir, quan el metge no disposa de la majoria de proves que li permetran afirmar o descartar un determinat diagnòstic.

## 6.1.2 Arquitectura

### 6.1.2.1 Nivell de l'univers de discurs: els fets

L'estructuració dels conceptes a PNEUMON-IA es fa segons diferents tipus de criteris.

- 1.- etiologies,
- 2.- dades generals,
- 3.- dades epidemiològiques,
- 4.- factors de risc o antecedents patològics d'interés,
- 5.- simptomatologia,
- 6.- semiologia,
- 7.- dades de laboratori i
- 8.- radiologia convencional.

L'estructuració conceptual té altres nivells. D'aquesta manera les dades de laboratori es classifiquen en dades analítiques generals i dades microbiològiques. Les dades microbiològiques estan agrupades segons el tipus de mostra: (1) hemocultius, (2) mostres de líquid pleural, (3) mostres d'esput, (4) mostres serològiques i (5) altres mostres. Els conceptes de cadascun d'aquests 5 conjunts estan agrupats segons les tècniques que s'hi apliquin. Cadascuna de les tècniques pot produir uns resultats en els que l'agrupació d'agents dependrà del coneixement del camp del domini i de l'operativitat de la classificació.

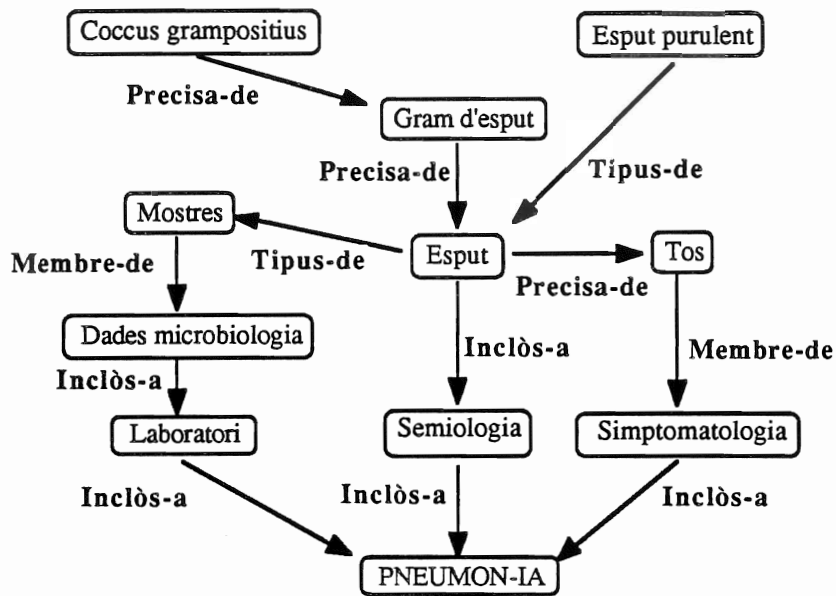


Fig. 6.1 Representació d'un subgraf de PNEUMON-IA on es mostren les relacions del fet: *esput*.

### 6.1.2.2 Nivell associatiu i relacional: les regles d'inferència i el seu control

A PNEUMON-IA els coneixements del domini expressats en forma de regles d'inferència s'estructuren en dos grans grups:

- 1.- aquelles regles que permeten una aproximació diagnòstica *inicial* i que habitualment es basen en dades epidemiològiques i clíniques i,
- 2.- les que *refinen* aquestes aproximacions a partir d'altres tipus de dades.

En la Base de Coneixements de PNEUMON-IA les conclusions que són *primeres aproximacions* a diagnòstics etiològics es representen mitjançant la grafia '-D'. D'aquesta manera s'escriurà Pneumococ-D o Legionella-D, etc. Aquesta gestió de noms era necessària abans d'introduir les primitives de modularització, on es posseeixen espais de noms locals.

Una vegada determinada la certesa d'aquesta primera aproximació, existeixen altres regles que refinen aquesta aproximació inicial per fer-la definitiva. Són les regles en que els predicats sobre els fets de les conclusions són *apujar* i *abaixar*. El predicat

*abaixar* és especialment important donada la monotonia creixent dels connectius o de MILORD, perquè no hi ha una altra manera de fer disminuir la certesa d'un fet determinat, quan, una vegada determinat, és tenen *de nou* arguments en contra.

Exemple:

R05042

**Si**

Legionella-D i

**no** (Antecedents-patològics-d'interés = alcohòlic) i

**no** (Antecedents-patològics-d'interés = hepatopatia\_crònica) i

transaminases > 40

**Llavors**

**apujar** Legionella<sup>1</sup> 2

i com exemple d'abaixar:

R05042

**Si**

Legionella-D i

estat = lleu

**Llavors**

**abaixar** Legionella 2

#### *Metaregles del nivell associatiu i relacional*

Les metaregles que poden inhibir o activar regles són aquelles que inhibeixen (1) una llista determinada de regles (2) les regles que utilitzin un determinat fet a alguna de les condicions de la seva premisa i (3) les regles que utilitzin com a premissa un fet que mantingui una determinada relació amb un altre fet.

El primer (1) tipus de metaregles van ser utilitzades profusament per l'expert a l'inici de la construcció de l'aplicació. Aquestes metaregles tenien greus inconvenients:

---

<sup>1</sup> El predicat *apujar* és fa sobre Legionella i no sobre Legionella-D. Existeix una regla que transforma la certesa final de Legionella-D en Legionella:

R05035

**Si**

Legionella-D

**Llavors**

**Segur** Legionella

És sobre aquest fet que s'apliquen els predicats *apujar* i *abaixar*. Existeixen regles d'aquests tipus a cadascun del mòdul diagnòstics de PNEUMON-IA

- a) Eren poc informatives. El seu examen no revelava el coneixement que implementaven ja que es referenciaven les regles mitjançant els símbols identificadors.
- b) La modificació i renumeració de regles d'inferència de la Base de Coneixements feia que s'hagués de revisar cadascuna d'aquestes metaregles.
- c) Afectaven només a *determinades* regles quan en realitat moltes d'elles eren metaregles més generals.

Aquest darrer punt expressa el fet que inicialment el coneixement de l'expert sobre el domini de PNEUMON-IA estava escassament estructurat i, per tant, particularitzava coneixements en cada situació concreta, quan, de fet, eren aplicables a situacions més generals. És per aquest fet que aquestes metaregles han anat desapareixent progressivament de la Base a mida que els coneixements han adquirit una millor estructuració.

### 6.1.2.3 Nivell estructural

Per tal d'estructurar les regles d'inferència de PNEUMON-IA, aquestes s'han agrupat en mòduls segons els objectius que persegueixen, és a dir, segons les seves conclusions. Els mòduls que configuren PNEUMON-IA són de diferents tipus:

- a.- Mòduls d'adquisició de dades
- b.- Mòdul sindròmic
- c.- Mòduls segons diagnòstics etiològics

La representació de a) i b) és pot esquematitzar en la figura 6.2

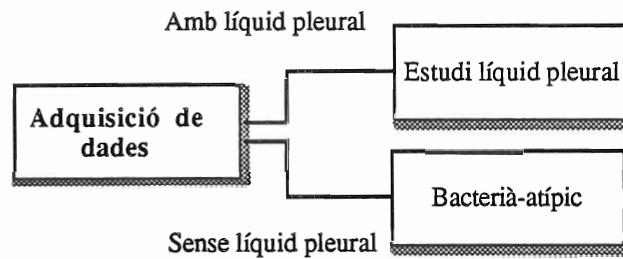


Fig. 6.2. Estructura dels mòduls d'adquisició de dades i sindròmic.

La representació de la figura 6.2. no pretén establir una jerarquia entre els diferents mòduls sinó que s'ha d'entendre com una estructura de condicions prèvies, es visita abans. En l'exemple, això significa que **abans** de tractar el mòdul sindròmic *bacterià-atípic* cal haver obtingut dades del mòdul adquisició de dades. D'aquest mòdul s'ha aïllat una subestructura que agrupa totes les dades que és poden demanar o inferir de l'examen d'un líquid pleural. Aquest mòdul, com el *bacterià-atípic*, també té com a condició prèvia el mòdul *adquisició\_de\_dades*

A partir del mòdul *bacterià-atípic* s'estructuren els mòduls dels diagnòstics etiològics tal com s'observa en la figura 6.3.



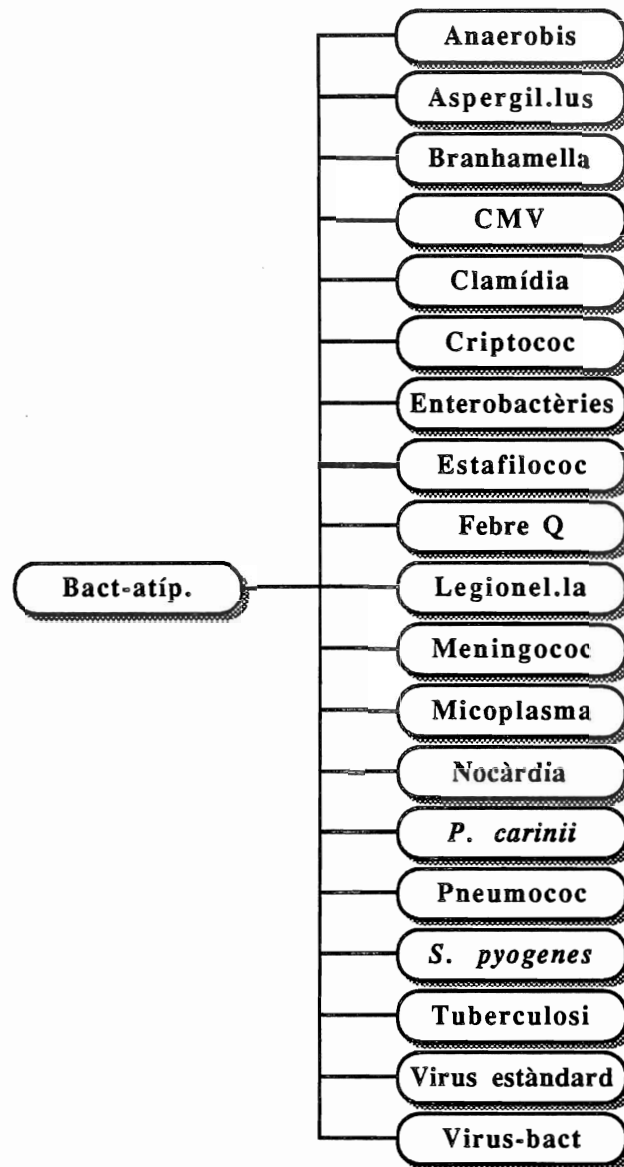


Fig. 6.3 Mòduls del nivell estructural de l'aplicació PNEUMON-IA

Cal fer algunes remarques sobre aquests mòduls:

No tots els mòduls tenen un sol diagnòstic etiològic. Això és degut a diferents motius: (1) algunes etiologies de les pneumònies extrahospitalàries són plurimicrobianes com és el cas de les pneumònies per gèrmens anaerobis (2) Les regles de producció aplicades a un grup d'etiologies són idèntiques com és el cas de les pneumònies víriques o de les pneumònies per enterobactèris (3) alguns diagnòstics són matitzacions d'un altre diagnòstic contingut en un determinat mòdul.

Els mòduls que agrupen les regles d'inferència per tal d'individualitzar els diagnòstics etiològics tenen una estructura interna que es pot diferenciar en dos seccions:

- a) D'una part s'agrupen aquelles regles que **orienten** sobre la possibilitat de que el clínic s'enfronti a una pneumònia ocasionada per un determinat germen. Aquestes regles donen una **primera aproximació** sobre aquesta possibilitat.

R04026

**Si**

bacteriana i  
herpes

**Llavors**

pneumococ és molt\_possible

- b) Per altra part s'agrupen una sèrie de regles que **refinen** les anteriors. Per poder avaluar aquestes regles és precís enmarcar-les en un contexte que ve donat pel primer conjunt de regles.

Exemple:

No es pot donar una certesa pel resultat d'un tenyiment d'un gram de l'esput si no s'emmarca el resultat d'aquesta prova en el contexte del malalt concret, és a dir, en la possibilitat prèvia o probabilitat subjectiva de la presència d'una determinada etiologia. Així tenim la regla:

R04035

**Si**

pneumococ és moderadament\_possible i  
gram-esput = CGPP o CGPC

**Llavors**

apujar pneumococ 1

La cerca que s'efectua, es fa sempre a partir de les hipòtesis.

#### 6.1.2.4 Nivell hipotètic: l'establiment d'estratègies

##### *Estratègies a partir dels factors de risc*

Les estratègies seguides per PNEUMON-IA simulen les hipòtesis que l'expert clínic genera en el procés diagnòstic i d'orientació etiològica d'una pneumònia. Les

hipòtesis generades més precoçment són aquelles que és relacionen amb els factors de risc del malalt. El coneixement d'aquests factors de risc (o la seva absència) determina prendre com a hipòtesi etiològica un determinat grup d'agents etiològics.

Exemple:

R25888

**Si**

no antecedents\_patològics\_d'interés

**Llavors**

**Mòduls a visitar**

BACT-ATIP PNEUM MYCO LEGION

VIRUSES VIRUSES-BACT CLAM

FQ TBC STR-PIO NOCARDIA

**Amb certesa moderadament\_possible**

R25894

**Si**

ADVP { Adicte a drogues per via  
parenteral }

**Llavors**

**Mòduls a visitar**

BACT-ATIP P-CAT TBC

CITOMEGALOVIRUS CRIPTOCOC

NOCARDIA ASPER PNEUM

ENTEROBACT

**Amb certesa moderadament\_possible**

R25891

**Si**

mucoviscidosi

**Llavors**

**Mòduls a visitar**

BACT-ATIP PSEUDO STAPH

**Amb certesa possible**

Aquestes tres metaregles que generen estratègies diagnòstiques diferents tenen com a premissa els factors de risc del malalt, és a dir, construeixen una hipòtesi prèviament a l'atenció de les dades d'exploració física i exploracions complementàries.

Ara bé, no tots els factors de risc tenen la mateixa importància a l'hora d'establir una hipòtesi de treball. PNEUMON-IA té en compte aquest factor mitjançant l'assignació de valors de certesa a aquestes metaregles. D'aquesta manera a partir de l'anterior exemple es pot observar que la regla que genera una estratègia a partir de *mucoviscidosi* té una etiqueta lingüística superior a la que genera hipòtesi a partir de la

manca d'*antecedents patològics d'interés* o d'*antecedents d'adicció a drogues per via parenteral*. Així queda explicitat el coneixement expert que la mucoviscidosi s'associa fortament a uns determinats agents patògens mentre que la associació entre manca d'*antecedents d'interés* i determinats microorganismes és més feble.

L'assignació de valors de certesa a les metaregles que generen estratègies permet graduar les estratègies en un malalt donat.

Exemple:

Si un malalt té com antecedents patològics la SIDA i l'alcoholisme s'apliquen les següents dos metaregles:

R25895

**Si**

SIDA

**Llavors**

**Mòduls a visitar**

BACT-ATIP P-CAR TBC

CITOMEGALOVIRUS CRIPTOCOC

NOCARDIA ASPER PNEUM ENTEROBACT

**Amb certesa** força\_possible

R25897

**Si**

alcohòlic

**Llavors**

**Mòduls a visitar:**

BACT-ATIP PNEUM ENTEROBACT LEGION

ANAEROB TBC STAPH STR-PIO

**Amb certesa** possible

La SIDA és més suggeridora que els antecedents d'alcoholisme per definir el grup de microorganismes que poden causar la pneumònia. Per tant l'estratègia que generi la metaregla que té com a premissa la SIDA té un grau de certesa superior a l'estratègia que parteixi d'alcoholisme.

#### *Estratègies a partir de les dades d'exploració clíniques*

En aquest grup es tenen en consideració les dades clíniques observables directament i per tant subministrades per l'usuari del sistema. Per tant no s'inclouen aquelles dades que són abstraccions qualitatives elaborades per PNEUMON-IA. La

manera com és generen estratègies a partir de metaregles que tenen com a premissa dades clíniques és similar a la forma de generar-se les estratègies del grup anterior.

*Estratègies a partir de les dades d'exploració complementàries*

Després de ser generades estratègies a partir de les dades clíniques, ja siguin de la Història Clínica o de l'exploració física, l'expert clínic pren en consideració les dades que li són subministrades per les exploracions complementàries: dades de laboratori i de radiologia

Les metaregles que generen estratègies a partir de les dades d'exploracions complementàries tenen premisses que comporten graus de certesa elevat. Això és explicable perquè els examens complementaris que generen poca certesa es valoren dins un contexte determinat, és a dir, dins un mòdul. Dit d'una altra manera: només aquells exàmens complementaris que tenen una especificitat prou evident són tinguts en compte a l'hora de generar una estratègia.

En les següents dues metaregles les condicions de les premisses corresponents a fets de laboratori i de radiologia es consideren prou suggeridores per construir una estratègia a partir d'elles mateixes.

R25898

**Si**

neutropènia

**Llavors**

**Mòduls a visitar**

BACT-ATIP PSEUDO ENTEROBACT

ASPER

**Amb certesa molt\_possible**

R25898A

**Si**

cavitacions

**Llavors**

**Mòduls a visitar**

BACT-ATIP ANAEROB STAPH

PSEUDO ENTEROBACT ASPER

**Amb certesa molt\_possible**

*Estratègies a partir de les dades deduïdes pel sistema*

Fonamentalment aquestes estratègies són construïdes per metaregles que tenen com a condició a avaluar si la pneumònia és bacteriana o atípica..

Un exemple d'estratègia construïda a partir de metaregles que tinguin com a premisses dades deduïdes pel sistema, podria ser el següent:

R25897

**Si**

empiema

**Llavors**

**Mòduls-a-eliminar**

MYCO VIRUSES CLAM FQ CMV P-CAR

**Mòduls a visitar**

BACT-ATIP PNEUM ANAEROB  
ENTEROBACT STAPH PSEUDO

**Amb certesa** força\_possible

on empiema és una condició deduïda per regles d'inferència de PNEUMON-IA.

#### 6.1.2.5 Nivell heurístic: els plans d'actuació

Les metaregles referents a plans de PNEUMON-IA són les següents:

*metaregla M01*

M01 **Si** objectius (x)  $\subset$  objectius (y) **Llavors Ockham** (x,y)

Exemple: Sigui un malalt neutropènic amb una pneumònia cavitada:

Estratègia X

R25898

**Si**

neutropènia

**Llavors**

**Mòduls a visitar** BACT-ATIP PSEUDO

ENTEROBACT ASPER

**Amb certesa** molt\_possible

Estratègia Y

R25898A

**Si**

cavitada

**Llavors**

**Mòduls a visitar** BACT-ATIP  
ANAEROB STAPH PSEUDO  
ENTEROBACT ASPER  
**Amb certesa** molt\_possible

Aplicant la M01 el pla d'actuació seria:

**BACT-ATIP PSEUDO ENTEROBACT ASPER test final ANAEROB STAPH**

*metaregla M02*

M02 **Si** certesa (x) = Certesa (y) i objectius (x)  $\cap$  objectius (y)  $\neq \emptyset$   
**Llavors** Ockham (x,y)

Exemple: Sigui un malalt diabètic amb una malaltia respiratòria crònica:

Estratègia X

R25900A

**Si**

diabètic

**Llavors**

**Mòduls a visitar** BACT-ATIP PNEUM  
ENTEROBACT LEGION ANAEROB  
TBC STAPH STR-PIO  
**Amb certesa** possible

Estratègia Y

R25901

**Si**

malaltia\_respiratòria\_crònica

**Llavors**

**Mòduls a visitar** BACT-ATIP PNEUM  
HEMOPH LEGION ENTEROBACT  
BRANHA STR-PIO  
**Amb certesa** possible

Aplicant la M02 el pla d'actuació seria:

**BACT-ATIP PNEUM ENTEROBACT LEGION STR-PIO test final**  
**HEMOPH BRANHA ANAEROB TBC STAPH**

*metaregla M03*

M03 Si certesa (x) > Certesa (y) i objectius (x)  $\cap$  objectius (y)  $\neq \emptyset$

**Llavors Ockham (x, Ockham (x,y))**

que determina, en primer lloc, l'avaluació de la intersecció dels objectius de x amb els objectius de y, després la resta d'objectius de x i, finalment, la resta d'objectius de y, és a dir:

**X  $\cap$  Y test final X / Y test final Y / X**

on / és la diferència de conjunts, X = objectius(x) i Y = objectius(y)

Exemple: Sigui un malalt diabètic amb una malaltia respiratòria crònica:

Estratègia X

R25895

**Si**

SIDA

**Llavors**

**Mòduls a visitar BACT-ATIP P-CAR**

TBC CITOMEGALOVIRUS

CRIPTOCOC NOCARDIA ASPER

PNEUM ENTEROBACT

**Amb certesa força\_possible**

Estratègia Y

R25883

**Si**

atípica és possible i

**no** (bacteriana és possible)

**Llavors**

**Mòduls a visitar MYCO VIRUSES**

VIRUSES-BACT CLAM FQ TBC

NOCARDIA CRIPTOCOC CMV P-

CAR

**Amb certesa lleugerament\_possible**

Aplicant la M03 el pla d'actuació seria:

P-CAR TBC CRIPTOCOC NOCARDIA CMV **test final** ASPER PNEUM

ENTEROBACT **test final** MYCO VIRUSES VIRUSES-BACT CLAM

FQ



Les situacions que poden sorgir quan s'avalua un cas a PNEUMON-IA amb estratègies X i Y que no interseccionen poden ser les següents:

*metaregla M04*

M04 Si certesa (x)  $\geq$  Certesa (y) i objectius (x)  $\cap$  objectius (y) =  $\emptyset$

**Llavors Prioritari (x,y)**

Exemple: Sigui un malalt amb una pneumònia bacteriana sense antecedents patològics d'interés que tingui una pneumònia cavitada :

Estratègia X

R25898A

**Si**

cavitada

**Llavors**

**Mòduls a visitar** BACT-ATIP  
ANAEROB STAPH PSEUDO  
ENTEROBACT ASPER  
**Amb certesa** força\_possible

Estratègia Y

R25888

**Si**

no antecedents\_patològics\_d'interés

**Llavors**

**Mòduls a visitar** BACT-ATIP PNEUM  
MYCO LEGION VIRUSES VIRUSES-  
BACT CLAM FQ TBC STR-PIO  
NOCARDIA  
**Amb certesa** possible

Aplicant la M04 el pla d'actuació seria:

ANAEROB STAPH PSEUDO ENTEROBACT ASPER **test final** PNEUM  
MYCO LEGION VIRUSES VIRUSES-BACT CLAM FQ TBC STR-PIO  
NOCARDIA

### 6.1.3 Situació actual

PNEUMON-IA es troba actualment en fase de validació, que descriurem a continuació. Posseeix a prop de les 800 regles, 100 metaregles i 25 mòduls no estructurats.

#### 6.1.3.1 Validació

La darrera fase en el desenvolupament d'un SE és la seva validació estadística [Soula et al, 1987], [Michel et al, 1987], [Porter et al, 1988].

##### *Material*

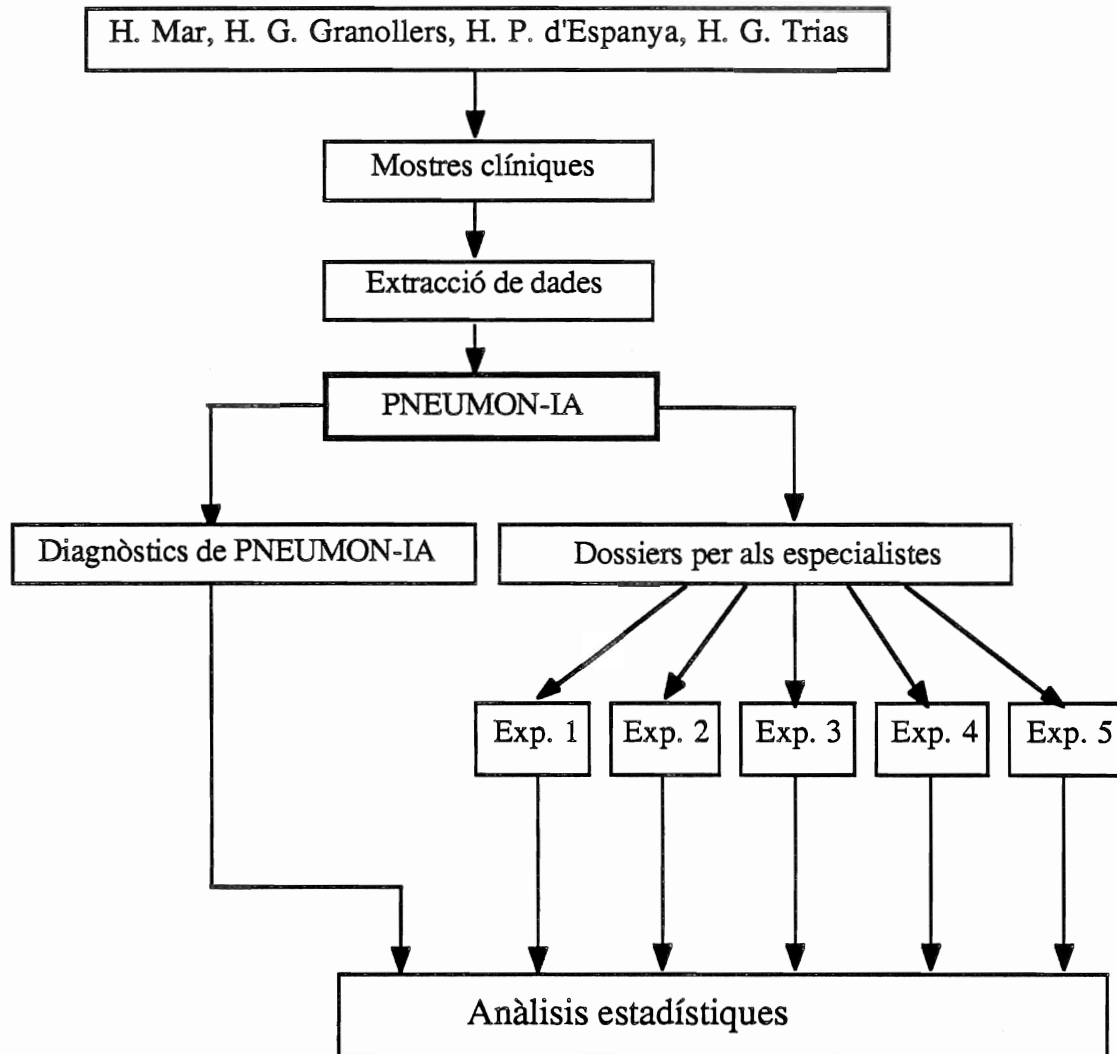
La validació realitzada sobre PNEUMON-IA [Verdaguer et al, 1989] es basa en 80 casos reals de pneumònies de quatre hospitals de l'àrea de Barcelona (Hospital del Mar, de Bellvitge, Germans Trias i Pujol, i General de Granollers). A cadascun d'ells el mètode d'extracció utilitzat ha estat aleatori, estratificat per diagnòstics.

S'han eliminat de les històries els diagnòstics conclusius, cas que hi fossin, així com les dades radiològiques. L'apreciació radiològica és molt important i els experts volen donar sempre la seva valoració. Degut a això s'han fotografiat les plaques de tòrax i s'han subministrat diapositives als experts que han intervingut en la validació.

Els dossiers amb les dades són generats per una utilitat que disposa MILORD, segons la qual, a partir de les dades d'un cas i utilitzant la informació del GRF, genera un informe on les dades es troben organitzades segons un ordre determinat: inicialment les dades de simptomatologia, després les semiològiques, etc. Veure VI.2.2

## Mètode

La figura 6.4 mostra quin és el procés de validació aplicat a PNEUMON-IA.



Els dossiers han estat estudiats per 5 experts de reconegut prestigi en l'àmbit de les pneumònies. Per cada cas l'expert ha proporcionat la llista dels diagnòstics plausibles en la seva opinió, amb una valoració de la certesa utilitzant els mateixos termes lingüístics que l'expert que va desenvolupar PNEUMON-IA. També s'han recopilat de cada expert l'opinió del tractament, de manera que la futura extensió de PNEUMON-IA en l'orientació terapèutica disposi del conjunt de dades per fer la validació. Vegi's el formulari que els experts omplen per cada cas:

## IDENTIFICACIO

Fitxa clínica: *nom del malalt.*

DIAGNOSTIC EXPERT HUMA: \_\_\_\_\_

Diagnòstic	Valor
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

## TRACTAMENT RECOMANAT

_____	_____
_____	_____

## RECORDATORI D'ETIQUETES

Gens Possible	0	0	0	0
Molt poc possible	0	0	0.05	0.08
Lleugerament possible	0.05	0.07	0.14	0.17
Moderadament possible	0.10	0.15	0.35	0.45
Possible	0.25	0.35	0.55	0.65
Força possible	0.45	0.55	0.75	0.85
Molt possible	0.65	0.75	1	1
Segur	1	1	1	1

A l'anàlisi estadística s'han utilitzat criteris de comparació de dos tipus diferents:

- 1.- Taules de freqüències segons el nombre de diagnòstics per cas, diagnòstics amb valor de certesa més alt, etc.
- 2.- Distància entre els vectors de termes de les 21 possibilitats diagnòstiques determinats pels experts i el sistema.

Els primers resultats d'aquestes anàlisis estadístiques han estat publicats a [Verdaguer et al, 1989], i demostren que el grau de competència del sistema és comparable al dels experts humans.

Presentem com a resultats part de les anàlisis realitzades sobre 32 dels 80 casos dels quals tenim resposta dels experts participants en la fase de validació.

La següent taula representa quantes vegades han estat considerats els diagnòstics pels cinc experts i pel sistema.

Taula 1. Diagnòstics considerats sobre 32 casos. S representa els resultats de PNEUMON-IA.

	E1	E2	E3	E4	E5	S	Mitjana
1.- Pneumococ	27	27	28	23	25	24	25.67
2.- Legionella	2	12	10	18	12	12	11
3.- Enterobactèries	8	15	14	11	10	16	12.33
4.- Hemòfilus	6	15	7	7	15	6	9.33
5.- Pseudomonas	0	0	0	0	1	0	0.17
6.- Anaerobis	7	10	9	12	4	9	8.50
7.- Estafilococ	5	2	3	5	1	2	3
8.- Meningococ	0	0	0	0	0	0	0
9.- Branhamela	0	1	0	0	0	0	0.17
10.- Estreptococ	0	1	2	3	0	0	1
11.- Tuberculosi	3	5	2	7	9	2	4.67
12.- Nocàrdia	1	0	1	1	0	0	0.5
13.- Criptococ	0	0	0	0	0	0	0
14.- Micoplasma	3	8	4	4	3	6	4.67
15.- Clamídia	0	7	1	4	1	7	3.33
16.- Febre Q	0	2	1	1	0	3	1.17
17.- Virus	0	1	1	0	3	5	1.67
18.- Complic. Bacter. de virus	0	0	0	0	0	1	0.17
19.- Citomegalovirus	0	0	0	0	0	0	0
20.- Aspergillus	0	0	1	1	2	1	0.83
21.- P. Caríni	0	0	0	0	0	0	0
Total diagnòstics	62	106	84	97	86	94	88.17
Mitjana Diagnòstics/cas	1.93	3.31	2.62	3.03	2.68	2.93	2.75

El càlcul de distàncies entre experts s'ha calculat utilitzant la distància "City-block" entre els diagnòstics de cada cas. En el cas de PNEUMON-IA hi ha 21 diagnòstics possibles. Cada expert dóna una ponderació amb terme lingüístic per alguns dels possibles diagnòstics que representem pel seu centre de gravetat. Si representem els diagnòstics no esmentats per un expert com a diagnòstics impossibles tindrem que un cas es pot representar per un vector de 21 elements compresos a l'interval [0 1].

$$E_{ik} = (D_1, D_2, \dots, D_{21})$$

Així la distància entre dos experts  $i, j$  per un cas  $k$  donat es calcula:

$$d(E_{ik}, E_{jk}) = \sum_{l=1}^{21} |E_{ikl} - E_{jkl}|$$

i la distància entre dos experts es calcula promitjant les distàncies entre tots els casos:

$$d(E_i, E_j) = \frac{\sum_{k=1}^n d(E_{ik}, E_{jk})}{n}$$

Vegi's a continuació una matriu que representa la distància entre els especialistes seguint la distància esmentada a dalt, amb  $n = 32$  i la distància normalitzada a [0 100] arrodonida:

	E1	E2	E3	E4	E5	S
E1		8	5	8	7	7
E2			6	8	7	7
E3				6	5	5
E4					7	6
E5						6
S						

Com es pot veure el sistema expert obté uns resultats molt bons respecte als experts. Fent una anàlisi dels "clusters" i obtenint un dendograma tenim la figura següent:

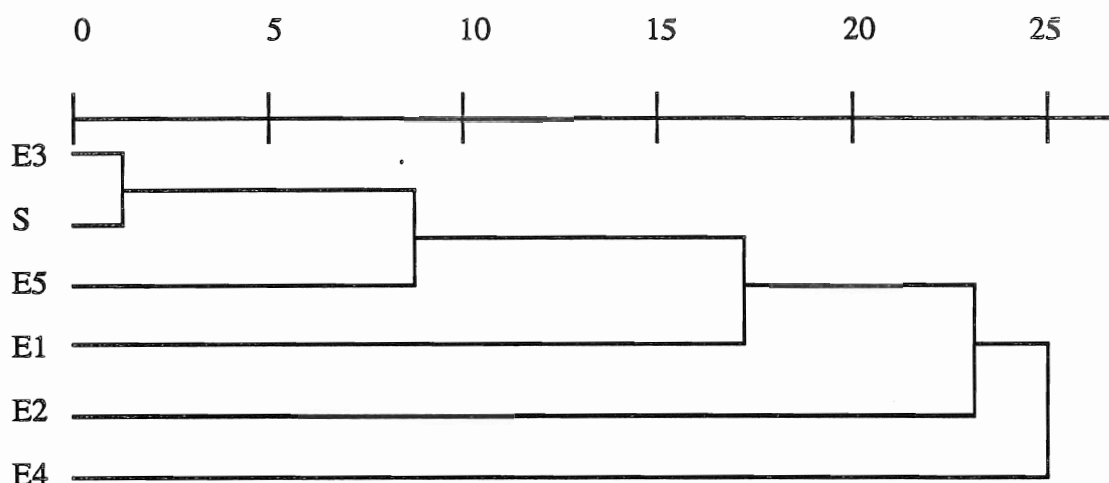


Fig. 6.5 Dendrograma dels "clusters".

On es pot veure que el primer agrupament es realitza entre l'expert E3 i el sistema PNEUMON-IA. Cal dir que preguntats tots els experts sobre el grau d'expertesa dels col·legues que participaven en la validació, l'expert E3 va ser el més afavorit en les puntuacions.

## 6.2 Renoir

L'aplicació Renoir, degut a ser una aplicació mèdica de característiques similars a l'aplicació PNEUMON-IA, serà tractada aquí de forma marcadament més breu.

### 6.2.1 Objectiu

La reumatologia és una de les especialitats mèdiques amb més grans dificultats diagnòstiques, degut tant a la incertesa que envolta les dades disponibles, com la classificació de malalties, que és en sí mateixa imprecisa i, sovint canvia a mesura que passa el temps.

Així, mentre en altres especialitats mèdiques existeixen proves diagnòstiques fiables i moltes vegades patognomòniques, en reumatologia freqüentment ni tan sols l'anatomia patològica és un criteri fiable per a classificar els malalts. Així la presència o absència de moltes de les dades que s'utilitzen no són practicament mai diagnòstiques per sí mateixes. Al mateix temps la classificació de les malalties es basa en multitud de símptomes, proves de laboratori i radiologia.

Totes aquestes característiques fan de la reumatologia un domini molt adient per al disseny de sistemes basats en coneixements, més encara perquè el nombre d'especialistes en aquest camp és relativament escàs. Són escassos els sistemes desenvolupats en aquests camps en altres països, i cap que coneguem en el nostre. A la Universitat de Missouri-Columbia s'està desenvolupant un sistema denominat AI/RHEUM, sobre el sistema EXPERT [Porter et al, 1988].

Tenint en compte que el nombre de les malalties de tipus reumàtic excedeix de dos-cents, hem de limitar-les a una xifra manipulable amb els nostres recursos. Degut a les característiques de treball d'en Miquel Belmonte, expert que ha realitzat el desenvolupament del sistema Renoir, s'han seleccionat dos grups de malalties: Les artropaties inflamatòries i les colagenosis. Per a més detalls consultar la tesi en preparació d'en Miquel Belmonte:

#### Artritis inflamatòries cròniques

- Artritis reumatoidea

- Artritis crònica juvenil

- Malaltia de Still

#### Malalties reumàtiques sistèmiques..Colagenosi

- Lupus eritematós sistèmic

- Dermatomiositis

- Esclerodèrmia

- Raynaud primari

- E.M.T.C. de Sharp

- Síndrome Sjögren primària

#### Vasculitis

- Arteritis temporal

- Polimiàlgia reumàtica

- Panarteritis nodosa

#### Esponsartrítides

- Espondilitis Anquilopoiètica

- Artritis psoriàsica

- Reiter

- Artritis enteropàtica

#### Artrosi

#### Artritis microcristal·lines

- Gota



Condrocalsinosi  
Artropaties bacterianes  
Artritis bacteriana no gonocòdica  
Artritis gonocòdica  
Artritis tuberculosa  
Febre reumàtica  
Reumatismes no articulars  
Fibromiàlgia  
Síndrome túnel carpià

En total 25 malalties.

### 6.2.2 Arquitectura

L'arquitectura, similar en essència a la del sistema PNEUMON-IA, defineix un mòdul per a cada malaltia que conté una serie de regles que bàsicament utilitzen els criteris diagnòstics de la malaltia. Els mòduls són 24 (alguns d'ells, especialment interessants, tenen un comentari explicatiu):

#### *Mòduls d'adquisició de dades i referents a síndromes*

##### MENU

CARACT-ARTIC : Característiques clíniques dels malalts amb afectació articular.

##### CONTADOR

MONOLIGOARTRITIS : Mòdul d'orientació de les malalties que afecten a una articulació.

RADIOLOGIA-ART : Dades referents a la radiologia.

SIMETRIA-ART : Mòdul per determinar la simetria de les articulacions afectades.

HEMOGRAMA : Dades referents a l'anàlisi de l'hemograma.

##### MAL-SISTEM

MULTIARTRITIS : Mòdul d'orientació de les malalties que afecten a varies articulacions.

CARACT-RAQUIS : Característiques clíniques dels malalts amb afectació de la columna.

RAQUIS-NOINFL : Característiques clíniques dels malalts amb afectació de la columna.

CAR-PELL

BIOQUIMICA : Dades de laboratori.

*Mòduls referents a malalties*

INFECCIO-ART

M-ARTR-REUM

M-FEBRE-REUM

M-LES

M-PALINDROMIC

M-STILL

M-ESPOND-ANQUIL

M-REITER

M-PSOR

M-ENTEROP

M-GOTA

Per a decidir les estratègies, el diagnòstic diferencial, s'han definit metaregles a partir de dades rellevants dels malalts, per exemple:

R00036

**Si**

poliartritis\_aguda és lleugerament\_possible

**Llavors**

**Mòduls a visitar** M-FEBRE-REUM M-STILL INFECCIO-ART M-ARTR-REUM M-LES M-ENTEROP M-REITER M-ESPOND-ANQUIL M-GOTA

**Amb certesa possible**

Les metaregles del nivell heurístic coincideixen amb les expressades a l'aplicació PNEUMON-IA.

### 6.2.3 Situació actual

Actualment l'aplicació Renoir es troba a la meitat de la fase de desenvolupament, i molt aviat entrarà en fase de validació.

La base de coneixements consta de :

431 regles organitzades en 25 mòduls.

450 fets.

18 mrr.

71 metaregles mre, mrrp i mra.

## 6.3 Disseny VLSI

Aquesta aplicació ha estat desenvolupada en col·laboració amb el departament d'informàtica de la Universitat de Dortmund<sup>1</sup>. Els experts alemanys en el camp del disseny VLSI que han desenvolupat l'aplicació són: Rudolf Felix, Peter Grabienski i Achim Höffmann [Felix et al, 1988].

Aquesta aplicació s'articula en el marc d'un projecte més ambiciós de l'esmentat departament anomenat ALICE-1 [Herrmann et al, 1988] consistent en un entorn de disseny que consta de cinc subsistemes:

**DACAPO:** Llenguatge de descripció algorísmic de hardware, cobrint tots els nivells d'abstracció des del nivell de sistema fins al nivell de transistor [Dacapo Manual, 1987].

**CHARM:** Planificador de l'arquitectura del chip. A partir de la forma compilada d'una descripció realitzada en el llenguatge DACAPO, CHARM genera un pla no topològic de l'arquitectura del xip.

**FLAIR:** És un "floorplanner" utilitzant una aproximació d'intel·ligència artificial [Bruck et al, 1988].

**IMAGE:** Aquest mòdul és l'encarregat de transformar blocs flexibles en substrat ("layout").

**LIMES:** Estimador de mòduls utilitzat per CHARM i FLAIR [Herrmann et al, 1988].

---

<sup>1</sup> Aquest treball ha estat subvencionat pel MEC espanyol i el DAAD Alemany mitjançant una Acció Integrada.

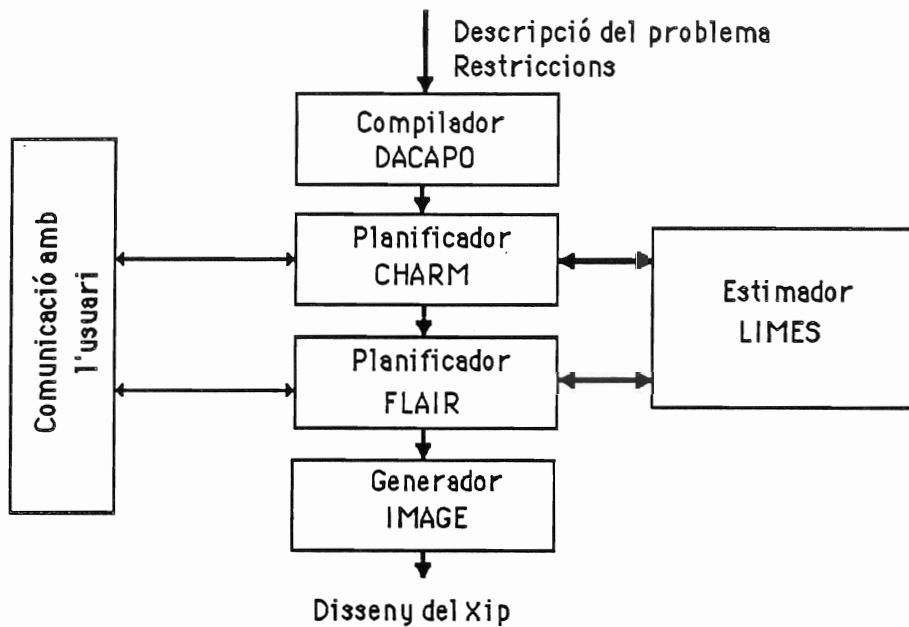


Figura 6.6 Sistema de disseny ALICE-1.

### 6.3.1 Objectiu

Hi ha una etapa en el procés de planificació de l'arquitectura d'un xip que és la selecció del tipus d'arquitectura que millor s'adapti a les necessitats funcionals del xip. Aquesta etapa és una subtasca del mòdul CHARM del projecte ALICE-1. Ara aquesta subtasca està realitzada mitjançant una arquitectura de pissarra. Amb el sistema desenvolupat s'han volgut potenciar els aspectes de raonament aproximat i raonament incomplet del procés de presa de decisions estudiat.

### 6.3.2 El problema de la selecció de l'arquitectura d'un chip

Un esquema d'arquitectura representa una classe de xips amb estructura i propietats similars. En són exemples l'arquitectura Von Neumann, vectorial SIMD, Sistòlica, etc. La classificació de les propietats dels xips utilitzada es basa en tres grups de propietats:

- 1.- Propietats dels elements de procés (veure figura 6.7)
- 2.- Propietats del sistema de connexió
- 3.- Propietats dels elements d'emmagatzemament

aquesta classificació està basada en els treballs de [Temme i Nitsche, 1988].

Propietats de les unitats de procés

#PE	$\frac{\#difPE}{\#PE}$	Complex. PE	Estrat. control	Sincro.	Arquitectura
1	1	alta	central	sincron	Von-Neumann
baix	1	baixa mitja	central	sincron	Vectorial Pipeline
mig-alt	$\ll 1$	arbitrar.	central	sincron	Vectorial SIMD
mig-alt	$\ll 1$	baixa	central	sincron	Vectorial Sistòlic
alt	$\ll 1$	baixa	distrib.	asincron	Vectorial Wavefront

Fig. 6.7 Exemple de classificació segons Temme. # vol dir *nombre de*, i PE són *processadors*.

El problema de selecció d'una arquitectura, sembla ser simple i pot ser entès com un procés en dues etapes:

- 1.- Extreure les característiques de classificació a partir de la descripció algorísmica.
- 2.- Aplicar la classificació de Temme.

Per la primera de les fases es proposa a ALICE-1 un mètode d'extracció i anàlisi descrit a [Temme, 1987]. Malauradament aquesta extracció no pot garantir que totes les propietats necessàries siguin obtingudes. A més la informació sobre propietats va des de la informació exacta, fins a les estimacions (ex. la complexitat dels processadors). Degut a això es fa necessària la manipulació de coneixements incomplets i incerts.

El resultat de la primera fase serà doncs un subconjunt de les propietats exigides per la classificació de Temme i malgrat això és volen realitzar classificacions. Aquest és l'objectiu de l'aplicació desenvolupada utilitzant MILORD

### 6.3.3 Arquitectura

#### 6.3.3.1 Raonament aproximat

El conjunt de termes lingüístics utilitzats a l'aplicació és de cinc:

{ **False, Almost false, May be, Almost true, True** }<sup>1</sup>

i els operadors que modelitzant els connectius són :

$$T(x,y) = \min(x,y)$$

$$S(x,y) = \max(x,y)$$

#### 6.3.3.2 Coneixements associatius

Al nivell estructural, els coneixements associatius estan estructurats en dues fases. La primera és la conclusió de les propietats desconegudes per l'usuari, però que poden ser deduïdes a partir de les propietats conegudes. La segona consta d'un conjunt de mòduls, un per cada tipus d'arquitectura, l'anàlisi dels quals ve determinada per l'actuació de les metaregles que determinen quins mòduls hauran de ser visitats i en quin ordre. Els mòduls d'aquesta aplicació són no estructurats<sup>2</sup>.

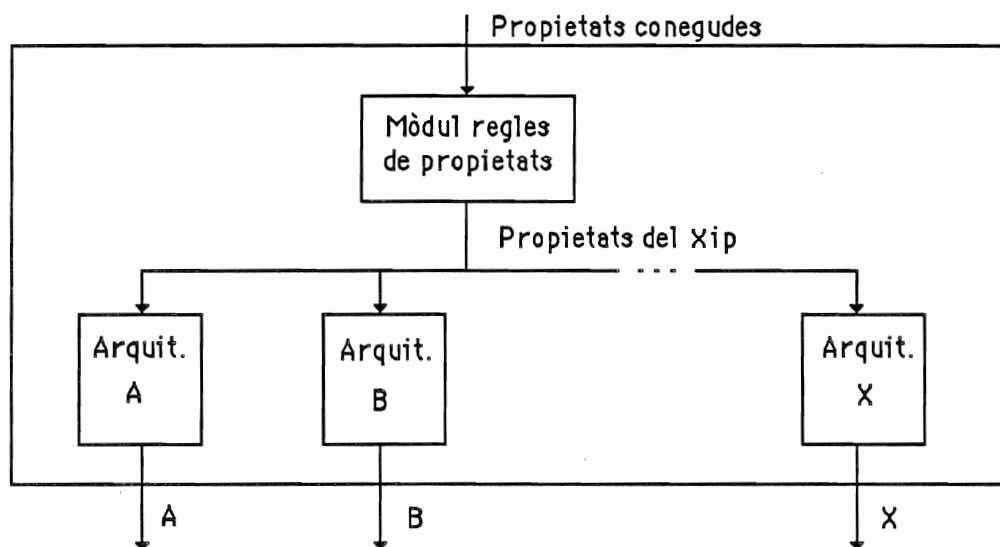


Fig 6.8 Fluxe de dades al nivell estructural de l'aplicació en VLSI.

<sup>1</sup> L'idioma utilitzat en aquesta aplicació és l'anglès.

<sup>2</sup> L'aplicació va ser desenvolupada uns mesos abans de la introducció dels mòduls estructurats a MILORD.

### *Mòdul de regles de propietats*

Aquest mòdul és el que conforma l'estratègia per defecte. Conté regles que intenten completar el conjunt de propietats a partir de les dades proporcionades a priori. Per exemple, si el nombre de processadors és elevat, es pot suposar que la seva complexitat serà baixa.

```

Si      number_of_pes = large
Llavors
        concloure high_complexity_of_pes és almost_false

```

Aquest mòdul s'executarà en encadenament endavant a partir de les dades proporcionades per l'usuari.

### *Mòduls de classificació de cada arquitectura*

Per cada arquitectura es defineix un mòdul, en el que s'estructuren regles de dos tipus: *Full rules* i *Subset rules* que passem a descriure a continuació:

*Full rules*: Aquestes regles corresponen als criteris de classificació pura proposada per en Temme. Totes les regles estàn ponderades amb el màxim valor de certesa *True*. Si les premisses d'una regla són totalment certes, l'arquitectura recomanada posseirà el màxim grau d'adequabilitat, i a més (suposant la classificació d'en Temme consistent) cap altre no posseirà el màxim grau d'adequabilitat. És un tipus de coneixement definicional el que aquestes regles expressen. Vegi's el següent exemple, on el fet *high\_complexity\_of\_pes* és un fet de tipus difús:

R02023

```

Si      number_of_pes = 1 i
        ratio = 1 i
        high_complexity_of_pes i
        control_strategy = central i
        synchronization = synchron i
        number_of_connection_systems = 1 i
        locality_of_connection_systems = global i

```

```

type_of_connection_systems = bus i
number_of_memories = 1 i
acces_to_memory = global

```

**Llavors**

```

concloure von_neumann és true

```

*Subset rules:* Les regles d'aquest tipus són regles que contenen premises definint conjunts incomplets de propietats. Degut a l'esquema de funcionament de la subsumció a MILORD, sempre s'intentarà aplicar la regla que posseeixi un subconjunt més ampli de premisses. d'aquesta manera estarem segurs que no hi haurà una sobrevaloració de cap de les propietats. Evidentment les regles d'aquest mòdul estan ponderades amb valors de certesa per sota del màxim valor: *true*. Per exemple:

R03020

```

Si    number_of_pes = 1 i
      ratio = 1 i
      high_complexity_of_pes i
      control_strategy = central i
      synchronization = synchron

```

**Llavors**

```

concloure von_neumann és almost_true

```

Degut al funcionament de la subsumció, sempre s'intentaran primer les regles *full*. Si aquestes no poden disparar-se, i per tant donar la màxima certesa a l'arquitectura, s'intentaran les regles *subset*, que podran, eventualment, donar un grau de certesa a l'arquitectura.

**6.3.3.3 Coneixements hipotètics**

El primer mòdul que el sistema ha de visitar sempre és el mòdul de les regles de propietats. Un cop visitat, i per tant completat el conjunt de propietats utilitzables s'establiran les estratègies pertinents per la resolució del problema. Aquest coneixement està organitzat en forma de metaregles, que segons el tipus de condicions que es manipulin podem classificar en tres tipus:



*Condicions necessàries:* Aquestes condicions expressen propietats que certes arquitectures han de posseir, i que, per tant, si estan absents en el cas en curs les inhabilita per ser candidates a solució. Per exemple, si el sistema de connexió és un bus, aleshores l'arquitectura no pot ser sistòlica.

M15004

**Si**      `type_of_connection_systems = bus`

**Llavors**

**mòduls a eliminar** `systolic_array_module`

**amb certesa** `true`

*Condicions suficients:* Hi ha una situació en la qual es considera que el sistema ha d'aturar la seva execució. Aquest és el cas en que una regla *full* ha estat aplicada, i per tant hem obtingut una màxima evidència per una arquitectura. Es tracta d'una situació excepcional, en la que tenim totes les propietats necessàries per classificar. Utilitzant metaregles d'excepcions aconseguim modelitzar aquesta característica. Cal notar que degut a la representació en ordre 0+, ens veiem obligats a escriure una metaregla per cada tipus d'arquitectura. Per exemple:

M15022

**Si**      `systolic_array és true`

**Llavors**

**Diagnòstic conclusiu** `systolic_array`

*Condicions de focalització:* Aquestes metaregles focalitzen sobre les arquitectures que tenen més possibilitats de ser solució. Utilitzen com a premisses condicions molt rellevants de certes arquitectures. Per exemple:

M15032

**Si**      `number_of_pes = high`

**Llavors**

**mòduls a visitar**      `systolic_array_module`

`SIMD_array_module`

`Wavefront_array_module`

**amb certesa** `almost_true`

Aquestes metaregles de focalització i les de condicions necessàries generen estratègies que són combinades utilitzant operadors d'Ockham.

Totes les metaregles de focalització no posseeixen una certesa *true*, degut a que el coneixement que manipulen és incomplet. Per tant, en cas de conflicte entre una metaregla de condició necessària, exemple: eliminar el mòdul A, i una metaregla de focalització, exemple: visitar el mòdul A, la metaregla de condició necessària serà prioritària sempre, com era desitjable.

### 6.3.4 Situació actual

Actualment el primer prototipus ha estat desenvolupat i ha estat ben rebut pels membres del departament dels experts de la Universitat de Dortmund que han intervingut en el seu desenvolupament. La seva complexitat no és gran: al voltant de les 250 regles i metaregles.

En un futur creiem que aquesta aproximació pot ser molt interessant per eines de CAD que a nivells alts d'abstracció hagin de manipular coneixements incerts i/o incomplets.

## 6.4 Control Industrial

La teoria avançada del control automàtic de processos ha tingut un impacte molt modest en l'automatització industrial, malgrat que ha donat bones solucions en els camps del control estocàstic i adaptatiu, en la coordinació de sistemes interconnectats i en l'optimització [Sanz et al, 1988].

El disseny de controladors autoadaptatius [Sanz, 1987] ha demostrat que a més de l'estimació de paràmetres i els algorismes de control, hi ha una gran quantitat d'heurístiques a considerar. A més, és necessari suportar canvis en les condicions de l'entorn i comportaments patològics dels estimadors deguts a problemes numèrics en els algorismes recursius d'estimació de paràmetres.

En aquesta aplicació s'ha desenvolupat un sistema de control adaptatiu que conté un nivell de supervisió utilitzant un supervisor basat en regles. No es pretén en aquesta aplicació substituir els procediments que actuen en temps real i que la teoria de control proporciona, amb gran èxit. Han estat proposats sistemes experts per supervisió automàtica a [Åström i Anton 1984], [Samaan, 1985], [Sanoff i Wellstead, 1985].

L'expert que ha col·laborat en el desenvolupament d'aquesta aplicació és el professor Rafael Sanz, membre del departament d'enginyeria elèctrica i de computadors i sistemes de la Universitat de Santiago de Compostela.

### 6.4.1 Objectiu

L'objectiu d'aquesta aplicació és implantar un procés supervisat posant en contacte una eina de desenvolupament de sistemes experts, MILORD, i algorismes de control adaptatiu per tal d'ajustar els paràmetres i triar les metodologies d'operació. L'esquema pot observar-se a la figura 6.9

Els dos elements del sistema són:

- 1) *Elements de control*. Proporcionen informacions numèriques i analògiques per ser processades per rutines procedimentals. Es proporcionen a aquest nivell respostes de control en temps real.
- 2) *Elements de supervisió*. Es manipula informació simbòlica expressada de forma declarativa. Les respostes es produeixen en un temps breu, similar al temps de raonament humà.

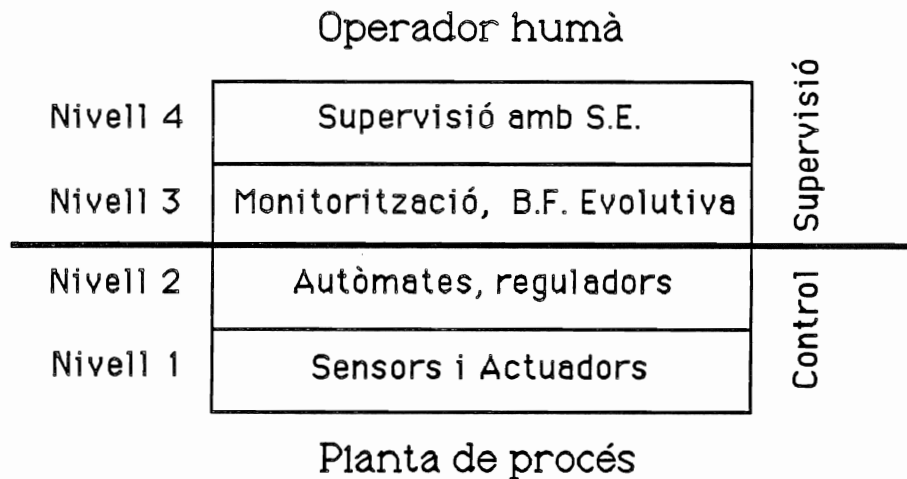


Fig. 6.9 Nivells entre l'operador humà i la planta de procés

S'ha utilitzat com a planta de procés per fer l'aplicació una planta d'energia solar situada a Andalusia. Pot trobar-se una descripció completa d'aquesta planta a [Rubio et al, 1987]. Les característiques més rellevants d'aquest model són la seva variabilitat respecte les irregularitats de la radiació solar, amb grans perturbacions quan apareixen núvols. L'amplitud d'aquestes perturbacions pot causar grans problemes en l'estimació dels paràmetres.

### 6.4.2 Arquitectura

El nivell 4 de l'esquema de la figura 6.9 és el nivell que ens ocupa en la present aplicació. Els diferents nivells han estat simulats utilitzant paquets PASCAL. La comunicació entre el simulador i el sistema expert s'ha realitzat utilitzant fitxers de comunicació i espera activa per part del simulador i del sistema expert.



Fig. 6.10 Comunicació entre els elements de l'aplicació.

Des del punt de vista de l'arquitectura de la Base de Coneixements desenvolupada, l'aplicació no presenta massa complexitat d'estructura. L'estratègia és invariant actualment, i consisteix en visitar un conjunt de mòduls en un ordre concret. Vegi's el següent esquema amb l'arquitectura.

Estratègia fixa: Visitar A B C

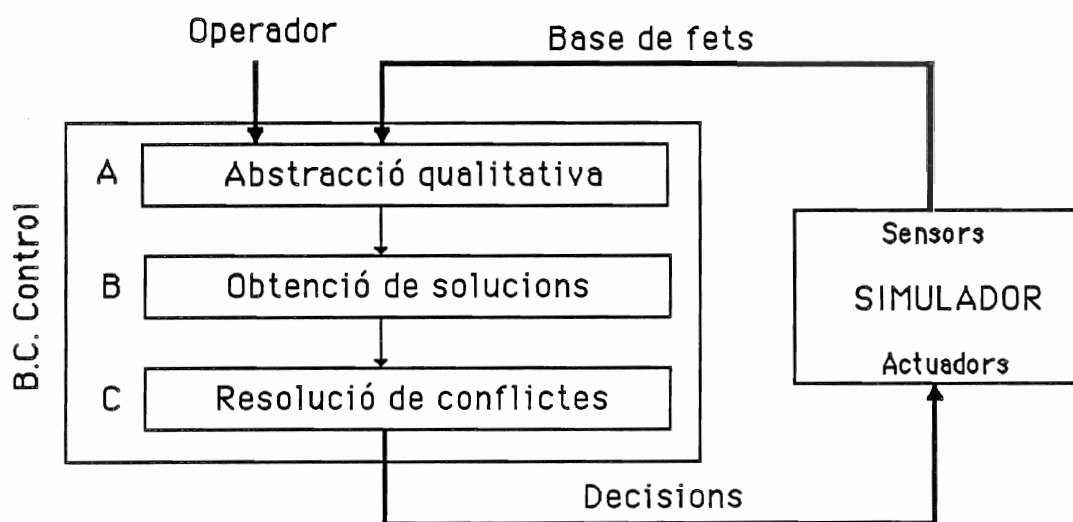


Fig. 6.10 Arquitectura de la B.C. Control.

Els tres mòduls tenen la següent semàntica:

*Abstracció qualitativa:* En aquest mòdul es transformen els valors numèrics en valors simbòlics amb assignació de certeses. Es pretén emular el procés cognitiu de l'operador davant la interpretació dels valors numèrics proporcionats pel nivell inferior de control.

*Obtenció de solucions:* Es modelitzen aquí les associacions entre les dades obtingudes pels sensors i les solucions, o modificacions dels paràmetres de la planta. Un exemple de regla seria:

**Si**      $\sigma$  = alt és segur i  
           $v$  = baix és possible i  
           $\varepsilon$  = alt és possible

**Llavors**

**concloure** reinicialitzar\_M és segur

on  $\sigma$  = traça de matriu de covariància generalitzada,  $v$  = variabilitat del valor estimat i  $\varepsilon$  = error.

*Resolució de conflictes:* Moltes vegades hi ha decisions conflictives que s'han de resoldre abans de comunicar al simulador. Per exemple s'ha pres la decisió d'apujar un paràmetre i d'abaixar-ne un altre amb el resultat d'un conflicte.

### 6.4.3 Situació actual

S'ha desenvolupat un prototipus de mida reduïda, aproximadament 50 regles, amb el qual s'ha comprovat la viabilitat de l'aproximació utilitzant un supervisor basat en la tecnologia de sistemes experts, i en concret l'ús del raonament aproximat per modelitzar les heurístiques dels operadors que supervisen els sistemes de control.

Aquesta orientació continuarà al llarg dels propers tres anys en el marc d'un projecte ESPRIT anomenat IPCES on es volen dissenyar aplicacions de nivells de supervisió en control de processos de fabricació.



## 7. Conclusions i línies futures

HAEC OLIM MEMINISSE IUBAVIT

*Virgili*

Hem presentat en aquesta memòria l'arquitectura de MILORD. Una arquitectura orientada a la resolució de problemes en classificació. Aquesta arquitectura ha estat implementada i utilitzada en el desenvolupament de diverses aplicacions. D'entre elles quatre han estat desenvolupades en col·laboració amb l'autor d'aquesta memòria, produint resultats significatius en quant a la competència d'una d'elles : PNEUMONIA. Els treballs desenvolupats, si bé no es consideren explotats fins a les seves últimes conseqüències, han mostrat diversos elements originals en el contexte de les arquitectures multi-nivell per al desenvolupament de sistemes experts.

*Un marc estructurat per a la resolució de problemes en funció dels tipus de coneixements.* S'han utilitzat com a elements vertebradors de l'arquitectura els tipus de coneixements implicats en el domini que es vol modelitzar. La separació explícita dels elements de l'arquitectura segons els tipus de coneixements que representen, facilita l'estructuració de les tasques que es volen modelitzar. S'han distingit cinc tipus de coneixements, a cadascun dels quals s'ha associat un nivell diferent d'arquitectura, obtenint com a resultat, aplicacions millor estructurades, més explicatives, i de gran facilitat de manteniment.

*Una representació i gestió dels coneixements incerts i incomplets, adaptables al raonament dels experts.* Partint originalment d'una lògica basada en una interpretació del termes lingüístics mitjançant intervals difusos, s'ha definit una lògica multivaluada de característiques semblants, adaptable al tipus de raonament simbòlic dels experts. Després d'una reflexió sobre la representació de la incertesa en diferents models hem arribat al convenciment que un model simbòlic de raonament millora els problemes que les aproximacions numèriques tradicionalment han mostrat. Tanmateix, la utilització d'un mecanisme de combinació d'evidències global per una aplicació sembla del tot

inadequat per a modelitzar correctament les diferents tasques de que es compona. La definició, per tant, de connectius diferents dependents dels mòduls a dissenyar, sembla una solució adient.

*Un formulisme de modularització que facilita la programació de grans aplicacions.* La superació dels sistemes experts de primera generació passa necessàriament per dotar les arquitectures de mecanismes potents de modularització. La metodologia de modularització proposada, reutilitzable fàcilment sobre arquitectures ja existents, dota al sistema de mecanismes d'herència, amagament d'informació i comprovació de tipus, que faciliten la programació eficaç i segura de grans aplicacions, coll d'ampolla tradicional de la Intel·ligència Artificial.

*Una arquitectura multinivell que permet separar els elements del domini dels elements de control.* La separació dels elements de domini dels elements de control té el seu origen a la mateixa aparició dels sistemes experts. En molts casos aquesta separació ha estat més anunciada que no pas aconseguida. A l'arquitectura de MILORD s'ha fet una distinció clara entre els elements de domini i de control per cadascun dels nivells de l'arquitectura. Basant-nos-en en el concepte de metaregla, s'han definit fins a cinc tipus diferents segons el nivell i la tasca que els corresponia. Aquests elements de control han estat representats de forma explícita utilitzant un mecanisme de raonament unificat amb els elements de domini. A més s'han seguit criteris d'uniformitat de la representació que permetrien eventualment i fàcilment l'ampliació dels tipus de metaregles emprats.

*S'han desenvolupat quatre aplicacions.* Una d'elles, PNEUMON-IA, ha estat una aplicació pionera a Espanya. Ha estat la primera en ser utilitzada habitualment pels metges d'un servei d'urgències en un hospital espanyol: L'Aliança Mataronina. Aquesta utilització, ha proporcionat moltes de les idees que es remarcaran a les línies futures d'investigació. Un altre aspecte pioner d'aquesta aplicació ha estat la seva validació multiexpert actualment conclosa i que ha proporcionat uns resultats molt encoratjadors. Aquesta validació multiexpert, per experts no participants en el desenvolupament de l'aplicació, ha estat la primera realitzada a Espanya i una de les poques realitzades a Europa

Moltes de les línies de recerca futures resulten evidents a partir de la lectura d'aquesta memòria. D'altres són fruit de projectes de recerca tot just començats que aprofundeixen en certs aspectes de l'arquitectura desenvolupada.



*Analitzar en profunditat la coexistència de diferents connectius associats als mòduls o àdhuc a les regles.* Els problemes de combinació d'evidències que això comporta i els tipus de manteniment de la consistència associats representen aspectes teòrics interessants.

*Aprofundir en la utilització de la teoria d'especificacions formals com a mecanisme d'estructuració de bases de coneixements.* Si bé a l'arquitectura actual de MILORD la modulabilització s'ha definit de forma estàtica, la seva extensió en la generació dinàmica de mòduls sembla un camí interessant en l'aproximació vers la programació orientada objecte.

*Estudiar la reformulació de MILORD en el contexte de sistemes concurrents.* L'adaptació a arquitectures paral·leles dels algorismes de resolució de problemes, així com els mecanismes de generació de regles i plans d'actuació, planteja reptes importants tant en allò referent a l'eficiència com en allò referent a la interacció amb l'usuari.

*En l'aspecte d'adquisició de coneixements,* resta encara molt de camí a fer en la definició de mètodes per a la construcció dels operadors matricials que defineixen els connectius a partir d'exemples del domini d'aplicació.

*Tècniques d'aprenentatge basat en casos.* A partir de la important bateria de casos de que disposem, obtinguts de la aplicació Pneumon-IA: 80 casos obtinguts de la validació multiexpert més 50 casos de la seva utilització al servei de urgències de l'Aliança Mataronina, plantegem definir tècniques d'aprenentatge basat en casos per deduir, no ja nous elements de domini, sinó estratègies de control que permetin realitzar una resolució dels problemes més eficient i més orientada.

Els resultats obtinguts a partir de l'estudi d'aquestes línies obertes seran aplicats sobre una nova aplicació d'abast molt més gran que les realitzades fins ara, en l'àmbit de la FOD (Febre d'origen Desconegut), continuant en la línia de multidisciplinarietat que tants bons resultats científics ens ha proporcionat fins ara.



# Apèndix I: BNF del llenguatge de MILORD

Tots els símbols no terminals d'aquesta gramàtica s'entenen com a àtoms LISP, o bé com a cadenes de caràcters si així s'indica.

```
aplicació ::= diccionari
           lògica_defecte
           nivell_estructural
           nivell_hipotètic
           nivell_heurístic
```

## I.1 Definició del nivell d'univers de discurs

```
diccionari ::= Diccionari entrada rel-inclusives fidiccionari
entrada    ::= fet entrada | fet
fet        ::= fet fet-id atributs
atributs   ::= nom pregunta tipus relació |
           nom pregunta tipus
nom        ::= Nom : string
pregunta   ::= Pregunta : string
tipus      ::= Tipus : def-tipus
def-tipus  ::= nomt | nomt = valors | valors
nomt       ::= tipusid1
valors     ::= valor valors | valor
relació    ::= Relació : relació-id fet-id funció2 |
           Relació : relació-id fet-id
```

---

<sup>1</sup>Els identificadors predefinitos són booleà, difús, numèric i classe.

<sup>2</sup>Per una explicació detallada veure l'apartat de Grafs Dirigits Funcionals.

*relació relació*

*rel-inclusives* ::= **Relacions inclusives** *llista-relaci*  
**firelacions**

*llista-relaci* ::= *relació-id llista-relaci* |  $\lambda$

## I.2 Definició dels nivells associatiu i estructural

### I.2.1 Definició de la lògica per defecte

*lògica-defecte* ::= **Lògica per defecte**  
 $\vdots$   
*termes*  
*connectius*  
**filògica**

*termes* ::= **Termes**  
*termesid*  
*semàntica*

*termesid* ::= *termeid ,termesid* | *termeid*

*semàntica* ::= **Representació trapezoidal** *valorstra* |  
**Representació multivaluada**

*valorstra* ::= *valortra* | *valortra valorstra*

*valortra* ::= *termeid = real, real, real, real*

*connectius* ::= **Connectius** : *connec*

*connec* ::= *parelles-duals* | *desc-connec* | *externs*

*desc-connec* ::= **conjunció = funció**<sup>1</sup>  
**disjunció = funció**

*parelles-duals* ::= **Lukasiewicz** | **Probabilistes** | **Zadeh**<sup>2</sup>

*externs* ::= **Externs** *descrip-fitxer*

### I.2.2 Definició dels mòduls no estructurats

*Nivell\_estructural* ::= **Nivell Estructural**

<sup>1</sup> Qualsevol funció en format infix.

<sup>2</sup> Aquests noms corresponen a les parelles presentades com a (T<sub>1</sub>, S<sub>1</sub>), (T<sub>2</sub>, S<sub>2</sub>) i (T<sub>3</sub>, S<sub>3</sub>) respectivament.

	<i>unitat-estructural</i>	
		<i>control-estructural</i>
		<b>finivell</b>
<i>unitat-estructural</i>	::= <b>Mòdul</b> <i>mòdul-id</i>	
		<i>objectius regles metaregles  </i>
		<i>unitat-estructural unitat-estructural</i>
<i>objectius</i>	::= <b>Objectius:</b> <i>lobjectius</i>	
<i>lobjectius</i>	::= <i>fet-id, lobjectius   fet-id</i>	
<i>regles</i>	::= <b>Regles:</b> <i>lregles</i>	
<i>lregles</i>	::= <i>regla lregles   regla</i>	
<i>regla</i>	::= <i>idregla</i> <b>Si</b> <i>premissa</i> <b>Llavors</b> <i>conclusió</i>	
		<i>documentació</i>
<i>premissa</i>	::= <i>premissa</i> <b>i</b> <i>condició</i>   <i>condició</i>	
<i>documentació</i>	::= <i>cadena-caràcters</i>	
<i>conclusió</i>	::= <b>concloure</b> <i>fetid</i> <b>és</b> $\beta$	
		<b>apujar</b> <i>fetid</i> <i>n</i>
		<b>abaixar</b> <i>fetid</i> <i>n</i>
<i>condició</i>	::= <b>no</b> ( <i>condició</i> )	
		<b>no-dem</b> ( <i>condició</i> )
		<i>fet-id</i>
		<i>fet-id</i> <b>és</b> <i>terme</i>
		<i>fet-id</i> = <i>valors</i>
		<i>càlcul</i> <b>operador-rel</b> <i>càlcul</i>
<i>operador-rel</i>	::= <   >   ≥   ≤   =   /=	
<i>valors</i>	::= <i>valor</i> <b>o</b> <i>valors</i>   <i>valor</i>	
<i>càlcul</i>	::= <i>valornumèric</i>   <i>fet-id</i>   <b>cardinal</b> ( <i>fet-id</i> )	
		( <i>càlcul</i> <b>operador-arit</b> <i>càlcul</i> )
		( <i>fet-id</i> := <i>càlcul</i> )
<i>operador-arit</i>	::= +   -   *   /	
<i>metaregles</i>	::= <b>Metaregles:</b> <i>lmrr</i>	
<i>lmrr</i>	::= <i>mrr</i> <i>lmetaregles</i>   $\lambda$	
<i>mrr</i>	::= <i>id-meta</i> <b>Si</b> <i>premissa</i> <b>Llavors</b> <i>filtradors</i>	
<i>filtradors</i>	::= <i>filtrador</i> <i>filtradors</i>   <i>filtrador</i>	
<i>filtrador</i>	::= <b>inhibir</b> <i>regles</i> <i>relació-id</i> <i>fet-id</i>	
		<b>inhibir</b> <i>regles</i> <i>fet-id</i>
		<b>inhibir</b> <i>regles</i> <i>llista-id-regla</i>

**desinhibir regles relació-id fet-id |**  
**desinhibir regles fet-id |**  
**desinhibir regles llista-id-regla |**  
**podar fet-id |**  
**empeltar fet-id**  
*control-estructural* ::= Elements de control *controls*  
*controls* ::= *control controls* | *control*  
*control* ::= Mòdul *id-mòdul*  
*params-control* ::= Cerca = endavant |  
Cerca = enrera |  
Tall = *terme-id* |  
*params-control params-control*

### I.2.3 Definició dels mòduls estructurats

{ Aquest nivell estructural és una alternativa al ja descrit }

*nivell\_estructural* ::= Nivell estructural  
*declmod*  
**finivell**  
*declmod* ::= Especificació *vinclespec* |  
Mòdul *vinclemod* |  
Conjunt *vincleconj* |  
*declmod declmod*  
{ Heredar *id* és equivalent a **mòdul id = id** }  
*vinclespec* ::= *idespec = exprespec*  
*vinclemod* ::= *idmod[(llista-param)][: exprespec] = expremod*  
*llista-param* ::= *pid : exprespec [compartir eqcamí]; llista-param* |  
λ  
*eqcamí* ::= *id1 = ... = idn* n > 1  
{ *idmod [(llista-param)]: exprespec = expremod* és equivalent a *idmod*  
*[(llista-param)] = expremod : exprespec* }  
*vincleconj* ::= *idconjunt = (fet<sub>1</sub> : tipus<sub>1</sub> ; ... ; fet<sub>n</sub> : tipus<sub>n</sub>) |*  
*idconjunt = idconjunt operador\_conjunt idconjunt*  
*operador\_conjunt* ::= *U | ∩ | /*

<i>exprespec</i>	::= <i>idespec</i>   <b>inici espec final</b>
<i>espec</i>	::= <b>Mòdul</b> <i>idmod</i> : <i>exprespec</i>   <i>importats</i>   <i>exportats</i>   <i>espec espec</i>
<i>expremod</i>	::= <i>idcamí</i> [( <i>expremod</i> <sub>1</sub> ; ... ; <i>expremod</i> <sub><i>n</i></sub> )]   <b>inici decl final</b>   <i>expremod</i> : <i>exprespec</i>
<i>idcamí</i>	::= <i>idmod</i>   <i>idmod</i> -> <i>idcamí</i>
<i>decl</i>	::= [capçal] <i>cos</i>
<i>capçal</i>	::= <i>declnucli</i>   <i>declmod</i>   <i>declmod declnucli</i>
<i>declnucli</i>	::= <b>Obrir</b> <i>idmod</i>   <i>importats</i>   <i>exportats</i>   <i>declnucli declnucli</i>
<i>importats</i>	::= <b>Importa</b> = ( <i>fet</i> <sub>1</sub> : <i>tipus</i> <sub>1</sub> ; ... ; <i>fet</i> <sub><i>n</i></sub> : <i>tipus</i> <sub><i>n</i></sub> )   <b>Importa</b> = <i>idconjunt</i>
<i>exportats</i>	::= <b>Exporta</b> = ( <i>fet</i> <sub>1</sub> : <i>tipus</i> <sub>1</sub> ; ... ; <i>fet</i> <sub><i>n</i></sub> : <i>tipus</i> <sub><i>n</i></sub> )   <b>Exporta</b> = <i>idconjunt</i>
<i>cos</i>	::= <i>precondició regles metareg</i>
<i>precondició</i>	::= <b>Precondició</b> <i>premissa</i>   $\lambda$
<i>regles</i>	::= <b>Regles</b> <i>lregles</i>   $\lambda$
<i>metareg</i>	::= <b>Metaregles</b> <i>lmrr</i>   $\lambda$
<i>premissa</i>	::= { definit a l'apartat anterior }
<i>lregles</i>	::= { definit a l'apartat anterior }
<i>lmrr</i>	::= { definit a l'apartat anterior }

### I.3 Definició del nivell hipotètic

<i>Nivell-hipotètic</i>	::= <b>Nivell Hipotètic</b> <i>lmre-mrx</i> <i>estrategia-defecte</i> <b>finivell</b>
<i>lmre-mrx</i>	::= <i>mre</i>   <i>mrx</i>   $\lambda$   <i>lmre-mrx lmre-mrx</i>
<i>estrategia-defecte</i>	::= <b>Estrategia inicial</b> <b>Visitar</b> = <i>l-id-mòduls</i>
<i>l-id-mòduls</i>	::= <i>id-mòdul l-id-mòduls</i>   <i>id-mòdul</i>
<i>mre</i>	::= <i>id-meta</i> <b>Si premissa</b> <b>Llavors constructors</b>

	<b>amb certesa <math>\beta</math></b>
<i>constructors</i>	::= <i>constructor constructors</i>   <i>constructor</i>
<i>constructor</i>	::= <b>mòduls a visitar</b> <i>lidmoduls</i>   <b>mòduls a eliminar</b> <i>lidmoduls</i>   <b>mòduls a tractar</b> <i>lidmoduls</i>
<i>lidmoduls</i>	::= <i>idmodul lidmoduls</i>   <i>idmodul</i>
<i>mr<sub>x</sub></i>	::= <i>id-meta</i> <b>Si premissa</b> <b>Llavors excepció</b>
<i>excepció</i>	::= <b>diagnòstic conclusiu fet-id</b>   <b>avortar</b>

## I.4 Definició del nivell heurístic

<i>Nivell-heurístic</i>	::= <b>Nivell Heurístic</b> <b>Generació de plans</b> <i>l<sub>mrp</sub></i> <b>Acabament</b> <i>l<sub>mra</sub></i> <b>finivell</b>
<i>l<sub>mrp</sub></i>	::= <i>mrp l<sub>mrp</sub></i>   $\lambda$
<i>l<sub>mra</sub></i>	::= <i>mra l<sub>mra</sub></i>   $\lambda$
<i>mr<sub>p</sub></i>	::= <i>id-meta</i> <b>Si premissa<sub>pla</sub></b> <b>Llavors combinador</b>
<i>combinador</i>	::= <b>Ockham</b> ( <i>def-pla</i> , <i>def-pla</i> )   <b>Prioritari</b> ( <i>def-pla</i> , <i>def-pla</i> )   <b>Primer</b> ( <i>def-pla</i> , <i>def-pla</i> )
<i>premissa<sub>pla</sub></i>	::= <i>premissa_certesa</i>   <i>premissa_objectius</i>   <b>compatibles</b> ( <i>def-pla</i> , <i>def-pla</i> )
<i>premissa_certesa</i>	::= <b>certesa</b> ( <i>def-pla</i> ) <b>operador_rel</b> <b>certesa</b> ( <i>def-pla</i> )
<i>premissa_objectius</i>	::= <b>generador</b> <b>operador_conjunt</b> <b>generador</b>
<i>generador</i>	::= <b>nil</b>   <b>objectius</b> ( <i>def-pla</i> )   <b>eliminats</b> ( <i>def-pla</i> ) <i>generador</i> <b>operació_conjunt</b> <i>generador</i>
<i>def-pla</i>	::= <i>combinador</i>   <i>variable</i>
<i>operador_conjunt</i>	::= =   $\neq$   $\subset$   $\subseteq$   $\not\subset$
<i>operació_conjunt</i>	::= $\cup$   $\cap$   /
<i>mra</i>	::= <i>id-meta</i> <b>si premissa</b> <b>llavors acabar</b>



# Apèndix II: Equacions semàntiques del compilador i interpret

## II.1 Compilació del nivell d'univers de discurs

### II.1.1 Objectes i operadors semàntics

Presentem aquí la descripció de part de la semàntica del llenguatge MILORD utilitzant semàntica denotacional. Primer descriurem quins són els objectes semàntics, i les operacions que hi podem realitzar, i a continuació les funcions de valuació o equacions semàntiques a partir de les quals es podrà construir el compilador. Cal recordar que el llenguatge MILORD és inicialment compilat a un llenguatge intermig i posteriorment aquest és interpretat. Quan hi hagi algun dubte sobre quin dels processos utilitzarà quines equacions ho explicitarem.

Per cada part de sintaxi descriurem primer els objectes semàntics que utilitzarem, amb les operacions sobre ells, i després les equacions que ens relacionen la sintaxi amb els objectes semàntics.

#### I Tipus

Domini  $\tau \in T = \text{id} \rightarrow \text{conjuntval}$

#### Operacions

tipredef: T

valors:  $\text{id} \rightarrow \text{Llistaval}$

#### *Predefinits:*

valors(booleà) = ( cert, fals )

valors(classe) = ( buit, nobuit )

valors(numèric) = R

valors(fuzzy) = L

Definits per l'usuari:

valors(idtipus) = ( v<sub>i</sub> )

## II Taula de fets

Domini  $\eta \in TF = id \rightarrow Fetcomp$

Fetcomp = (St × St × T × Llistarc × Llistarc)

{ Els elements corresponen a: nom, pregunta, tipus, relacions amb origen a id i relacions amb destí a id }

### Operacions

taulatfbuida: TF

afegir-inverses: TF → TF

Afegeix les relacions inverses. es a dir afegeix l'últim camp Llistarc a tots els fets que tenen relacions que apunten cap a ells. Per exemple si tenim la relació del fet A: *pertany B nil* llavors aquesta funció afegiria al fet B la relació : *pertany<sup>-1</sup> A nil* . Si el fet B no existís un error seria notificat i la relació *pertany B nil* esborrada del fet A. Aquesta relació és important per l'execució eficient de les metaregles mrr.

## II.1.2 Funcions de valuació

DICCIO: *diccionari* × T → T × TF

DICCIO[[ *Diccionari entrada fidiccionari* ]] tipred = (τ' η)

(τ η) = ENTRADA[[ *entrada* ]] predefinit taulatfbuida

τ' = afegir-inverses(τ)

ENTRADA: *entrada* × T × TF → T × TF

ENTRADA[[ *fet entrada* ]] τ η = (τ'' η'')

(τ' η') = FET[[ *fet* ]] τ η

(τ'' η'') = ENTRADA[[ *entrada* ]] τ' η'

ENTRADA[[ *fet* ]]  $\tau \eta = (\tau' \eta')$   
 $(\tau' \eta') = \text{FET}[[ \textit{fet} ]]$

FET:  $\textit{fet} \times T \times TF \rightarrow T \times TF$

FET[[ Fet *fetid atributs* ]]  $\tau \eta = (\tau' \eta')$   
 $(\text{Fetcomp } \tau') = \text{ATRIB}[[ \textit{atributs} ]]$   $\tau$   
 $\eta' = [\textit{fetid} \rightarrow \text{Fetcomp}] \eta$   
 $\Leftrightarrow$  ERROR SI  $\eta(\textit{fetid}) \neq \perp$   
 MISSATGE "Redefinició del fet:" *fetid*  
 RETORN  $(\tau \eta)$

ATRIB:  $\textit{atributs} \times T \rightarrow (\text{St} \times \text{St} \times T \times \text{Llistarc} \times \text{Llistarc}) \times T$

ATRIB[[ Nom : *string*<sub>1</sub> Pregunta : *string*<sub>2</sub>  
 Tipus : *def-tipus relació* ]]  $\tau = (\text{Fetcomp } \tau')$   
 $(\text{tipusid } \tau') = \text{DEFTIP}[[ \textit{def-tipus} ]]$   $\tau$   
 $\text{llista} = \text{RELACIO}[[ \textit{relació} ]]$   
 $\text{Fetcomp} = (\textit{string}_1 \textit{string}_2 \text{tipusid llista nil})$

ATRIB[[ Nom : *string*<sub>1</sub> Pregunta : *string*<sub>2</sub>  
 Tipus : *def-tipus* ]]  $\tau = (\text{Fetcomp } \tau')$   
 $(\text{tipusid } \tau') = \text{DEFTIP}[[ \textit{def-tipus} ]]$   $\tau$   
 $\text{Fetcomp} = (\textit{string}_1 \textit{string}_2 \text{tipusid nil nil})$

DEFTIP:  $\textit{def-tipus} \times T \rightarrow \textit{tipusid} \times T$

DEFTIP[[ *tipusid* ]]  $\tau = (\textit{tipusid } \tau)$   
 $\Leftrightarrow$  ERROR SI  $\tau(\textit{tipusid}) = \perp$   
 MISSATGE "Tipus indefinit:" *tipusid*  
 RETORN fail

DEFTIP[[ *tipusid = valors* ]]  $\tau = (\textit{tipusid } \tau')$   
 $\text{val} = \text{VALORS}[[ \textit{valors} ]]$   
 $\tau' = [\textit{tipusid} \rightarrow \text{val}] \tau$

```

⇔ ERROR SI  $\tau(\text{tipusid}) \neq \perp$ 
MISSATGE "Redefinint tipus:" tipusid
RETURN let x = gensym("Tipus-") in
  ( x , [x → val]η )

```

```

DEFTIP[[ valors ]]  $\tau = (\text{tipusid } \tau')$ 
  val = VALORS[[ valors ]]
  tipusid = gensym("Tipus-")
   $\tau' = [\text{tipusid} \rightarrow \text{val}] \tau$ 

```

VALORS: *valors* → llista de *valor*

```

VALORS[[ valor valors ]] = l'
  l = VALORS[[ valors ]]
  l' = valor•l

```

```

VALORS[[ valor ]] = l
  l = ( valor )

```

RELACIO: *relació* → llistarc

```

RELACIO[[ Relació relacióid fetid funció ]] = larc
  larc = (( relacióid fetid funció ))

```

```

RELACIO[[ Relació relacióid fetid ]] = larc
  larc = (( relacióid fetid nil ))

```

```

RELACIO[[relació1 relació2]] = larc
  larc1 = RELACIO[[ relació1 ]]
  larc2 = RELACIO[[ relació2 ]]
  larc = larc1•larc2

```

## II.2 Compilació de regles

En aquest apartat donem la semàntica formal des dels punt de vista de la fase de compilació i també d'interpretació. Per fer més llegibles les equacions hem utilitzat com a entrada de la fase d'interpretació les estructures sintàctiques, i no els objectes semàntics resultat de l'etapa II (veure figura 2.12).

### II.2.1 Objectes i operadors semàntics

El resultat de l'etapa I de la fase de compilació és la obtenció d'una taula que relacioni cada identificador de regla amb una descripció de la representació interna dels seus components. Reb com a entrada el resultat de la compilació del diccionari, es a dir la taula de tipus definits i la taula de fets :  $\tau$  ,  $\eta$ . Aquí donem només els objectes i equacions semàntiques corresponents a la comprovació sintàctica, l'etapa I de la figura 2.12. L'etapa II realitzaria modificacions sobre les taules de fets i de regles, com per exemple les referències creuades. Alla on més endavant es faci referència a les taules de regles i fets suposarem les taules exteses, però conservant el mateix nom que aquí.

#### III Regles

Domini  $\rho \in TR = \text{reglaid} \rightarrow \text{components}$   
 components = ( premissa, conclusió)  
 Operacions  
 taulargbuida: TR

### II.2.2 Funcions de valuació de l'etapa I

COMPREGS:  $lregles \times T \times TF \times TR \rightarrow TR$

COMPREGS[[ *regla lregles* ]]  $\tau \eta \rho = \rho''$   
 $\rho' = \text{COMPREG}[[ \text{regla} ]] \tau \eta \rho$   
 $\rho'' = \text{COMPREGS}[[ lregles ]] \tau \eta \rho'$

COMPREGS[[ *regla* ]]  $\tau \eta \rho = \rho'$   
 $\rho' = \text{COMPREG}[[ \text{regla} ]] \tau \eta \rho$

COMPREGS: *regla* × T × TF × TR → TR

COMPREG[[ *idregla Si premissa Llavors conclusió* ]] τ η ρ = ρ'  
 ρ' = [idregla → (premissa, conclusió) ]ρ  
 ⇔ ERROR SI CHKPRESSA[[ *premissa* ]] τ η = false ∨  
 CHKCONCLUSIO[[ *conclusió* ]] τ η = false  
 MISSATGE "Regla incorrecte. Prosseguim." *idregla*  
 RETURN ρ

CHKPRESSA: *premissa* × T × TF → B

CHKPRESSA[[ *premissa i condició* ]] τ η = b  
 b = CHKPRESSA[[ *premissa* ]] τ η ∧  
 CHKCONDICIO[[ *condició* ]] τ η

CHKPRESSA[[ *condició* ]] τ η = b  
 b = CHKCONDICIO[[ *condició* ]] τ η

CHKCONDICIO: *Condició* × T × TF → B

CHKCONDICIO[[ *no condició* ]] τ η = b  
 b = CHKCONDICIO[[ *condició* ]] τ η

CHKCONDICIO[[ *no\_dem condició* ]] τ η = b  
 b = CHKCONDICIO[[ *condició* ]] τ η

CHKCONDICIO[[ *fetid* ]] τ η = b  
 b = true  
 ⇔ ERROR SI tercer(η(*fetid*)) ≠ booleà ∧  
 tercer(η(*fetid*)) ≠ difús  
 MISSATGE "Tipus incorrecte com a condició." *fetid*  
 RETURN false

CHKCONDICIO[[ *fetid* és *terme* ]]  $\tau \eta = b$   
 $b = \text{true}$   
 $\Rightarrow$  ERROR SI  $\text{tercer}(\eta(\textit{fetid})) \neq \text{difús}$   
 MISSATGE "Tipus incorrecte com a premissa." *fetid*  
 RETORN false  
 $\Rightarrow$  ERROR SI *terme*  $\notin L$   
 { utilitzem L com a global, per ser estrictes caldria passar L  
 com a paràmetre de CHKCONDICIO }  
 MISSATGE "Terme lingüístic incorrecte." *terme*  
 RETORN false

CHKCONDICIO[[ *fetid* = *valors* ]]  $\tau \eta = b$   
 $b = \text{true}$   
 $\Rightarrow$  ERROR SI  $\text{membre}(\text{tercer}(\eta(\textit{fetid})),$   
 { booleà, numèric, difús, classe } )  
 MISSATGE "Tipus incorrecte com a premissa." *fetid*  
 RETORN false  
 $\Rightarrow$  ERROR SI no inclós( $\text{VALORS}[[ \textit{valors} ]], \tau(\text{tercer}(\eta(\textit{fetid})))$ )  
 MISSATGE "Valors incorrectes:"  $\text{VALORS}[[ \textit{valors} ]]$  /  
 $\tau(\text{tercer}(\eta(\textit{fetid})))$   
 RETORN false

CHKCONDICIO[[ *càcul1* *operadorrel* *càcul2* ]]  $\tau \eta = b$   
 $b = \text{CHKCALCUL}[[ \textit{càcul1} ]]$   $\tau \eta \wedge \text{CHKCALCUL}[[ \textit{càcul2} ]]$   $\tau \eta$   
 $\Rightarrow$  ERROR SI no  $\text{membre}(\textit{operadorrel}, \{>, \geq, <, [, =\})$   
 MISSATGE "Operador relacional incorrecte." *operadorrel*  
 RETORN false

CHKCALCUL:  $\textit{càcul} \times T \times \textit{TF} \rightarrow B$

CHKCALCUL[[ *valor numèric* ]]  $\tau \eta = b$   
 $b = \text{true}$

CHKCALCUL[[ *fetid* ]]  $\tau \eta = b$   
 $b = \text{true}$

⇒ ERROR SI tercer( $\eta(fetid)$ )  $\neq$  numèric

MISSATGE "Tipus incorrecte com a premissa." *fetid*

RETORN false

CHKCALCUL[[ **cardinal**(*fetid*) ]]  $\tau \eta = b$

$b = \text{true}$

⇒ ERROR SI membre(tercer( $\eta(fetid)$ ),

{booleà, numèric, difús, classe} )

MISSATGE "Tipus incorrecte com a premissa." *fetid*

RETORN false

CHKCALCUL[[ (*càcul1* **operatorarit** *càcul2* ) ]]  $\tau \eta = b$

$b = \text{CHKCALCUL}[[ \textit{càcul1} ]] \tau \eta \wedge \text{CHKCALCUL}[[ \textit{càcul2} ]] \tau \eta$

⇒ ERROR SI no membre(*operatorarit*, {+, -, \*, /})

MISSATGE "Operador aritmètic incorrecte." *operatorarit*

RETORN false

CHKCALCUL[[ (*fetid* := *càcul* ) ]]  $\tau \eta = b$

$b = \text{CHKCALCUL}[[ \textit{càcul} ]] \tau \eta$

⇒ ERROR SI tercer( $\eta(fetid)$ )  $\neq$  numèric

MISSATGE "Tipus incorrecte com a premissa." *fetid*

RETORN false

CHKCONCLUSIO: *conclusió*  $\times T \times TF \rightarrow B$

CHKCONCLUSIO[[ **concloure** *fetid* és  $\beta$  ]]  $\tau \eta = b$

$b = \text{true}$

⇒ ERROR SI tercer( $\eta(fetid)$ )  $\neq$  difús

MISSATGE "Tipus incorrecte com a conclusió." *fetid*

RETORN false

⇒ ERROR SI  $\beta \notin L$

{ utilitzem L com a global, per ser estrictes caldria passar  
L com a paràmetre }

MISSATGE "Terme lingüístic incorrecte."  $\beta$

RETORN false



CHKCONCLUSIO[[ **apujar fetid n** ]]  $\tau \eta = b$   
 $b = \text{true}$   
 $\Leftrightarrow$  ERROR SI tercer( $\eta(\text{fetid})$ )  $\neq$  difús  $\vee$  no sencer( $n$ )  
 MISSATGE "Conclusió incorrecte." *fetid n*  
 RETORN false

CHKCONCLUSIO[[ **abaixar fetid n** ]]  $\tau \eta = b$   
 $b = \text{true}$   
 $\Leftrightarrow$  ERROR SI tercer( $\eta(\text{fetid})$ )  $\neq$  difús  $\vee$  no sencer( $n$ )  
 MISSATGE "Conclusió incorrecte." *fetid n*  
 RETORN false

VALORS: *valors*  $\rightarrow$  conjunt de *valor*

VALORS[[ *valor o valors* ]] =  $c$   
 $c = \{ \text{valor} \} \cup \text{VALORS}[[ \text{valors} ]]$

VALORS[[ *valor* ]] =  $c$   
 $c = \{ \text{valor} \}$

### II.2.3 Objectes i operadors semàntics de la interpretació

#### IV. Valors de certesa

Domini  $v \in VC = \text{Conjunt de símbols} \cup$   
 $\{ \text{cert, fals, desconegut} \}$

#### Operacions

no:  $VC \rightarrow VC$   
 no\_dem:  $VC \rightarrow VC$   
 i:  $VC \times VC \rightarrow VC$   
 o:  $VC \times VC \rightarrow VC$   
 imp:  $VC \times VC \rightarrow VC$   
 decalar:  $VC \times Z \rightarrow VC$

A més es defineixen les operacions de comparació habituals, tenint en compte l'ordre establert entre les

etiquetes al conjunt L.

#### V. Fets (ampliació de II)

Domini  $\eta \in TF = id \rightarrow Fetcomp \times Valorfet$

$Valorfet = ( VC + R + \{buit, nobuit\} + conjunt )$

#### Operacions

tipus:  $id \times TF \rightarrow T$

$tipus(i, \eta) = tercer(primer(\eta(i)))$

valor:  $id \times TF \rightarrow Valorfet$

$valor(i, \eta) = second(\eta(i))$

actualitzar:  $TF \times id \times VC \rightarrow TF$

$actualitzar(\eta, i, v) =$

(case tipus(i,  $\eta$ ) de

difús  $\rightarrow [i \rightarrow o(valor(i, \eta), v)]\eta$

$\square$  altrament  $\rightarrow error("Tipus incorrecte.")$ )

valuat:  $id \times TF \rightarrow B$

$valuat(i, \eta) = (valor(i, \eta) \neq \perp)$

cardinal:  $id \times TF \rightarrow N$

$cardinal(i, \eta) =$

member(tipus(i,  $\eta$ ), {booleà, difús,

numèric, classe }

$\rightarrow error("Cardinal aplicat a$

tipus incorrecte")

$\square$  cardinalitat( $\tau$ (tipus(i,  $\eta$ )))

#### VI Parametres de control

Domini  $p \in P = Engine \times VC$

#### Operacions

(\* Les projeccions: primer i segon \*)

### II.2.4 Funcions de valuació de la interpretació

REGLA:  $Regla \times TF \times VC \times P \rightarrow TF$

REGLA  $\llbracket idregla \text{ Si premissa Llavors conclusió } \rrbracket \eta \vee p = \eta'$

$$\begin{aligned}
 t &= \text{PREMISSA}[\textit{premissa}] \eta \vee p \\
 \eta' &= (t \geq \text{second}(p) \rightarrow \text{CONCLUSIO}[\textit{conclusió}] \eta \vee t \\
 &\quad \square \eta)
 \end{aligned}$$

PREMISSA:  $\textit{Premissa} \times \text{TF} \times \text{VC} \times \text{P} \rightarrow \text{VC}$

$$\begin{aligned}
 \text{PREMISSA}[\textit{premissa i condició}] \eta \vee p &= v'' \\
 v' &= \text{PREMISSA}[\textit{premissa}] \eta \vee p \\
 v'' &= (v' < \text{second}(p) \rightarrow v' \\
 &\quad \square i(v', \text{CONDICIO}[\textit{condició}] \eta \vee p))
 \end{aligned}$$

$$\begin{aligned}
 \text{PREMISSA}[\textit{condició}] \eta \vee p &= v' \\
 v' &= \text{CONDICIO}[\textit{condició}] \eta \vee p
 \end{aligned}$$

CONDICIO:  $\textit{Condició} \times \text{TF} \times \text{VC} \times \text{P} \rightarrow \text{VC} \times \text{TF}$

$$\begin{aligned}
 \text{CONDICIO}[\textit{no condició}] \eta \vee p &= (v'', \eta') \\
 (v', \eta') &= \text{CONDICIO}[\textit{condició}] \eta \vee p \\
 v'' &= \text{no}(v')
 \end{aligned}$$

$$\begin{aligned}
 \text{CONDICIO}[\textit{no\_dem condició}] \eta \vee p &= (v', \eta') \\
 (v', \eta') &= \text{CONDICIO}[\textit{condició}] \eta \vee p \\
 v'' &= \text{no\_dem}(v')
 \end{aligned}$$

$$\begin{aligned}
 \text{CONDICIO}[\textit{fetid}] \eta \vee p &= (v', \eta') \\
 (v', \eta') &= (\text{cas tipus}(\textit{fetid}, \eta) \text{ de} \\
 &\quad \text{booleà} \rightarrow \text{VALUAR}[\textit{fetid}] \eta \text{ p} \\
 &\quad \square \text{difús} \rightarrow \text{VALUAR}[\textit{fetid}] \eta \text{ p} \\
 &\quad \square \text{altrament} \rightarrow \text{error}(\text{"Tipus incorrecte."}))
 \end{aligned}$$

$$\begin{aligned}
 \text{CONDICIO}[\textit{fetid és terme}] \eta \vee p &= (v'', \eta') \\
 (v', \eta') &= \text{CONDICIO}[\textit{fetid}] \eta \vee p \\
 v'' &= v' \geq_{\perp} \textit{terme} \rightarrow \text{cert fals}
 \end{aligned}$$

$$\text{CONDICIO}[\textit{fetid = valors}] \eta \vee p = (v', \eta')$$

$$\begin{aligned}\eta' &= \text{VALUAR}[\textit{fetid}] \eta \textit{p} \\ \textit{vset} &= \text{VALORS}[\textit{valors}] \\ \textit{v}' &= \text{valor}(\textit{fetid}, \eta') \cap \textit{vset} \neq \emptyset \rightarrow \text{cert fals}\end{aligned}$$

$$\begin{aligned}\text{CONDICIO}[\textit{càlcul1 operadorrel càlcul2}] \eta \textit{v p} &= (\textit{v}', \eta'') \\ (\text{val}_1, \eta') &= \text{CALCUL}[\textit{càlcul1}] \eta \textit{v p} \\ (\text{val}_2, \eta'') &= \text{CALCUL}[\textit{càlcul2}] \eta' \textit{v p} \\ \textit{v}' &= \text{apply}(\textit{operadorrel}, \text{val}_1, \text{val}_2) \rightarrow \text{cert fals}\end{aligned}$$

CALCUL:  $\textit{càlcul} \times \text{TF} \times \text{VC} \times \text{P} \rightarrow \text{R} \times \text{TF}$

$$\begin{aligned}\text{CALCUL}[\textit{valornumèric}] \eta \textit{v p} &= (\text{val}, \eta) \\ \text{val} &= \textit{valornumèric}\end{aligned}$$

$$\begin{aligned}\text{CALCUL}[\textit{fetid}] \eta \textit{v p} &= (\text{val}, \eta') \\ \eta' &= \text{VALUAR}[\textit{fetid}] \eta \textit{v p} \\ \text{val} &= \text{valor}(\textit{fetid}, \eta')\end{aligned}$$

$$\begin{aligned}\text{CALCUL}[\textit{cardinal}(\textit{fetid})] \eta \textit{v p} &= (\text{val}, \eta') \\ \eta' &= \text{VALUAR}[\textit{fetid}] \eta \textit{v p} \\ \text{val} &= \text{cardinal}(\text{valor}(\textit{fetid}, \eta'))\end{aligned}$$

$$\begin{aligned}\text{CALCUL}[\textit{(càlcul1 operadorarit càlcul2)}] \eta \textit{v p} &= (\text{val}, \eta'') \\ (\text{val}_1, \eta') &= \text{CALCUL}[\textit{càlcul1}] \eta \textit{v p} \\ (\text{val}_2, \eta'') &= \text{CALCUL}[\textit{càlcul2}] \eta' \textit{v p} \\ \text{val} &= \text{apply}(\textit{operadorarit}, \text{val}_1, \text{val}_2)\end{aligned}$$

$$\begin{aligned}\text{CALCUL}[\textit{(fetid := càlcul)}] \eta \textit{v p} &= (\text{val}, \eta'') \\ (\text{val}, \eta') &= \text{CALCUL}[\textit{càlcul}] \eta \textit{v p} \\ \eta'' &= [\textit{fetid} \rightarrow (\text{primer}(\eta'(\textit{fetid})), \text{val})]\eta'\end{aligned}$$

CONCLUSIO:  $\textit{conclusió} \times \text{TF} \times \text{VC} \times \text{P} \rightarrow \text{TF}$

$$\begin{aligned}\text{CONCLUSIO}[\textit{concloure fetid és } \beta] \eta \textit{v p} &= \eta' \\ \eta' &= [\textit{fetid} \rightarrow \text{o}(\text{valor}(\textit{fetid}, \eta), \text{imp}(\text{v}, \beta))]\eta\end{aligned}$$

CONCLUSIO[[ *apujar fetid n* ]]  $\eta \vee p = \eta'$   
 $f' = [fetid \rightarrow decalar(valor(fetid, \eta), n)]\eta$

CONCLUSIO[[ *abaixar fetid n* ]]  $\eta \vee p = \eta'$   
 $\eta' = [fetid \rightarrow decalar(valor(fetid, \eta), -n)]\eta$

VALUAR: *fetid*  $\times$  TF  $\times$  P  $\rightarrow$  TF

VALUAR[[ *fetid* ]]  $\eta \ p = f'$   
 $f' = first(p) = endavant \rightarrow \eta$   
 $\square (valuat(fetid, \eta) \rightarrow \eta$   
 $\square FET[[ fetid ]] \eta \vee p)$

VALORS: *valors*  $\rightarrow$  conjunt

VALORS[[ *valor o valors* ]] = c  
 $c = \{ valor \} \cup VALORS[[ valors ]]$

VALORS[[ *valor* ]] = c  
 $c = \{ valor \}$

Vegi's a continuació la semàntica operacional associada a la interpretació de les regles segons la sintaxi expressada més amunt. S'avaluen les condicions de la premissa, es combina aquest resultat amb el grau de certesa de la regla, i finalment s'avalua la conclusió tenint en compte el valor de certesa propagat. El següent algorisme n'indica la semàntica:

```

procedure EVAL_REGLA (regla,alfa);
  {Aquest procediment avalua una regla de MILORD}
  var
    valor_propagat: Terme = Cert;
    Prem: List;
  begin
    {Càlcul de la certesa associada a la premissa de la regla}
    Prem := premissa(regla);
    while valor_propagat /= desconegut and
      valor_propagat  $\geq$  alfa and Prem /= nil do begin
      valor_propagat := T(valor_propagat,
        eval_prem(CAR(Prem)));
      Prem := CDR(Prem)
    end

```

```

end;
{Combinació amb el valor de certesa de la regla}
valor_propagat := m(valor_propagat, certesa(regla));
eval_conclusió(conclusió(regla), valor_propagat)
end. {EVAL_REGLA}

function EVAL_PREM (condició): Term;
{Aquesta funció avalua una condició}
begin
  case condició
    fetid : return(certesa(fetid));
    no condició1 : return(negacio(EVAL_PREM(condició1)));
    no-dem condició1: if EVAL_PREM(condició1) = desconegut
                      then return(Cert) else return(Fals);
    fet és  $\beta$  : if posició(certesa(fetid))  $\geq \beta$ 
                then return(Cert) else return(Fals);
    fet = valor1 o valor2 o ... o valorn :
    ;      if pertany(valor(fet), {valor1, valor2, ..., valorn})
          then return(Cert) else return(Fals);
    càlcul1 operador_rel càlcul2 :
          if AVAL_CAL(càlcul1) operador numèric AVAL_CAL(càlcul2)
          then return(Cert) else return(Fals);
  end
end. {EVAL_PREM}

```

Nota: Essent T i m les funcions que modelitzen la conjunció i el modus ponens respectivament.

## II.3 Compilació del nivell estructural

### II.3.1 Objectes i operacions semàntiques

VII Taula unitats estructurals

Domini  $\nu \in TM : id \rightarrow \text{Mod-comp}$

Mod-comp = Llista de fetid  $\times$  TR  $\times$  TMR

{ TMR taula de metaregles, serà definida al capítol  
cinquè }

Operacions

taulaebuida: TM

### II.3.2 Funcions de valuació de l'etapa I

L'execució dels mòduls és tractada quan es parla de les estratègies al capítol

5.

El compilador realitza una comprovació de l'existència dels objectius a la

taula definida a partir del diccionari i crida a les equacions de compilació de regles definides a l'apartat anterior i a les equacions de compilació de metaregles referents a regles.

COMPNIVEST: *nivell estructural* × T × TF → TM

COMPNIVEST[[ **Nivell Estructural**

*unitat-estructural finivell* ]] τ η = υ

υ = UNITES[[ *unitat-estructural* ]] τ η taulauebuida

UNITES: *unitat-estructural* × T × TF × TM → TM

UNITES[[ **Mòdul idmodul objectius**

*regles metaregles* ]] τ η υ = υ'

taulareg = COMPREGS[[ *regles* ]] τ η

taulameta = COMPMETA[[ *metaregles* ]] τ η taulareg

obj = OBJECT[[ *objectius* ]] η

υ' = [ *idmodul* → ( obj taulareg taulameta ) ] υ

⇨ ERROR SI obj = nil

MISSATGE "Mòdul incorrecte." *idmòdul*

RETORN υ

OBJECT: *objectius* × TF → Llista de *fetid*

OBJECT[[ **Objectius: lobjectius** ]] η = 1

1 = LOBJ[[ *lobjectius* ]] η

LOBJ: *lobjectius* × TF → Llista de *fetid*

LOBJ[[ *fetid, lobjectius* ]] η = 1

l' = LOBJ[[ *lobjectius* ]] η

1 = *fetid*•l'

⇨ ERROR SI η(*fetid*) = ⊥

MISSATGE "Objectiu indefinit." *fetid*

RETORN l'

```

LOBJ[[ fetid ]]  $\eta = 1$ 
      1 = (fetid)
 $\Leftrightarrow$  ERROR SI  $\eta(\text{fetid}) = \perp$ 
      MISSATGE "Objectiu indefinit." fetid
      RETURN ()

```

### II.3.3 Semàntica denotacional de la interpretació de les mre

I Estratègies

Domini  $e \in S = (\text{Llista de } M)^3 \times L$

Operacions

$S_0 : S$

AVALMRE:  $mre \times S \rightarrow S$

AVALMRE[[ *id-meta Si Premissa Llavors*

*constructors amb certesa*  $\beta$ ]]<sup>1</sup> = e

p = AVALPREM[[ Premissa ]]

e = p  $\geq_L$  fals  $\rightarrow$  (APLICONs[[ constructors ]] primer( $S_0$ ),  
m(p,  $\beta$ ))

□  $S_0$

APLICONs: *constructors*  $\times$  (Llista de M)<sup>3</sup>  $\rightarrow$  (Llista de M)<sup>3</sup>

APLICONs[[ *constructor constructors* ]] e = e''

e' = CONSTRUCT[[ *constructor* ]] e

e'' = APLICONs[[ *constructors* ]] e'

CONSTRUCT: *constructor*  $\times$  (Llista de M)<sup>3</sup>  $\rightarrow$  (Llista de M)<sup>3</sup>

CONSTRUCT[[ *mòduls a visitar lidmoduls* ]] e = e'

e = (MV, ME, MD)

<sup>1</sup> De fet l'equació semàntica s'aplicaria sobre la representació generada pel compilador. Utilitzem la sintaxi externa per comprendre millor el procés realitzat.



$$Q = \text{LIDMOD}[\textit{lidmoduls}]$$

$$e' = (MV \bullet Q, ME, MD / Q)$$

CONSTRUCT[ mòduls a eliminar *lidmoduls* ]  $e = e'$

$$e = (MV, ME, MD)$$

$$Q = \text{LIDMOD}[\textit{lidmoduls}]$$

$$e' = (MV, ME \bullet Q, MD / Q)$$

CONSTRUCT[ mòduls a tractar *lidmoduls* ]  $e = e'$

$$e = (MV, ME, MD)$$

$$Q = \text{LIDMOD}[\textit{lidmoduls}]$$

$$e' = (Q \bullet MV, ME, MD / Q)$$

LIDMOD: *lidmoduls*  $\rightarrow$  Llista de *idmodul*

$$\text{LIDMOD}[\textit{idmodul lidmoduls}] = 1$$

$$l' = \text{LIDMOD}[\textit{lidmoduls}]$$

$$1 = \textit{idmodul} \bullet l'$$

$$\text{LIDMOD}[\textit{idmodul}] = 1$$

$$1 = (\textit{idmodul})$$



# Apèndix III: Conjunt consensuat de termes

## III.1 Introducció

La utilització de termes lingüístics pressuposa que els usuaris i l'expert o experts que dissenyin el sistema comparteixin un mateix significat respecte als valors de les etiquetes que s'utilitzin per qualificar la certesa de regles i fets de tipus difús.

Per tal d'utilitzar els termes lingüístics de certesa més adients per als usuaris del sistema, va semblar útil efectuar una enquesta a una població de metges de l'àrea metropolitana de Barcelona que tenien el català com a llengua materna, ja que la base de coneixements de PNEUMON-IA seria escrita en aquest idioma.

## III.2 Enquesta

Durant el mes de setembre de 1987 es va passar una enquesta a un grup de 50 metges, aproximadament, caracteritzats per efectuar la seva activitat professional en el camp de l'assistència i ser especialistes en diverses branques de la Medicina i de la Cirurgia.

A l'enquesta es demanava que al costat de cada terme lingüístic proposat s'indiqués un interval de valors compresos entre 0 i 100. Aquests valors significarien els límits inferior i superior de les certeses compatibles amb cadascun dels termes.

La ubicació dels termes dins l'enquesta va ser establerta aleatoriament. Es va demanar també, si volien proposar algun altre terme a més dels proposats. En cas afirmatiu es demanava quins intervals de certesa se li atribuiria.

Finalment es va demanar que a partir de la llista de termes proposats i els termes afegits és confeccionés una escala ordenada i suficientment matitzada amb els termes més adients per ser utilitzats a la pràctica clínica

### III.2.1 Resultats

Es va rebre resposta de 28 metges, amb una experiència professional que presentava una mitjana de 10 anys i amb una desviació estàndar de 6.

Els resultats, segons l'escala d'utilització s'expressen a la taula A1.1.

**Taula A1.1.** Resultats de l'enquesta

<u>Interval de certesa</u>			
Etiqueta lingüística	Mitjana	Desviació s.	Utilització
Segur (i) <sup>a</sup>	96.0	6.9	
Segur (s) <sup>b</sup>	99.9	0.3	
			<u>23 (85%)<sup>c</sup></u>
Impossible (i)	0.04	0.2	
Impossible (s)	1.9	3.1	
			<u>22 (82%)</u>
Possible (i)	38.3	16.7	
Possible (s)	59.3	17.8	
			<u>21 (78%)</u>
Probable (i)	55	18.1	
Probable (s)	74.3	16.3	
			<u>16 (59%)</u>
Improbable (i)	8.6	8.7	
Improbable (s)	25.2	18.4	
			<u>16 (59%)</u>
Pràcticament segur (i)	86.2	8.1	
Pràcticament segur (s)	96.8	4.7	
			<u>15 (56%)</u>
Molt possible (i)	71.8	12.1	
Molt possible (s)	88.6	11.3	
			<u>11 (41%)</u>
Força possible (i)	54	19	
Força possible (s)	74,8	13,8	
			<u>10 (37%)</u>
Raonable (i)	48.1	19.7	

Raonable (s)	73.3	19.5	9 (33%)
Moderadament possible (i)	2.1	4.9	
Moderadament possible (s)	11.6	20.2	7 (26%)
Molt poc possible (i)	8,9	8,6	
Molt poc possible (s)	23.6	20	7 (26%)
Lleugerament possible (i)	20.6	13.8	
Lleugerament possible (s)	36.6	19.3	7 (26%)
Gens possible (i)	2.1	4.9	
Gens possible (s)	11.6	20.2	6 (22%)

a. (i) significa el valor inferior de l'interval de certesa

b. (s) significa el valor superior de l'interval de certesa

c. La utilització significa la quantitat de metges que van seleccionar el terme a la llista des que ell utilitza.

### III.3 Elecció dels termes lingüístics

A la vista dels resultats obtinguts es va establir una ordenació de termes de manera que cubriessin d'una manera regular l'escala de valors compresa entre 0 i 100.

Amb els resultats obtinguts es va construir la següent escala:

Segur-----	96-100
Pràcticament segur -----	86-97
Molt possible-----	72-89
Probable -----	55-74
Possible-----	38-59
Improbable -----	9-25
Impossible-----	0-2

Donat que el sistema expert PNEUMON-IA es mou dins la teoria de les possibilitats i no utilitza mètodes de tipus probabilístic es va valorar que els termes més adients eren els que es relacionaven amb el mot 'possibilitat' i que s'evitaria utilitzar els derivats de 'probabilitat'. Per aquesta raó es va substituir el terme

'probable' per 'força possible' que presenta pràcticament el mateix interval de certesa (54-74). El mateix succeïa amb 'improbable' i 'molt poc possible' doncs els valors d'aquesta darrera etiqueta eren de 8.9 i 23.6

Per omplir el buit existent, va semblar convenient afegir dos etiquetes entre possible i molt poc possible i es van escollir 'moderadament possible' i 'lleugerament possible'. El valor final dels intervals es va ajustar a aquestes etiquetes utilitzades, tenint en compte la desviació estàndard dels extrems. Definint d'aquesta manera els extrems difusos dels intervals.

La llista definitiva de termes lingüístics va ser la següent:

**Impossible**  
**Molt poc possible**  
**Lleugerament possible**  
**Moderadament possible**  
**Possible**  
**Força possible**  
**Molt possible**  
**Pràcticament segur**  
**Segur**

La representació trapezoïdal dels termes lingüístics és pot observar a la figura A1.1:

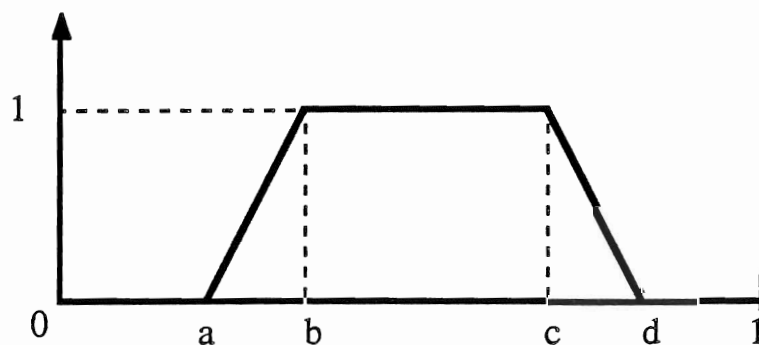


Figura A1.1 Representació dels termes lingüístics,

Valors de les abscisses	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>
Impossible -----	0	0	0	0
Molt poc possible -----	0	0	0.05	0.08
Lleugerament possible-----	0.05	0.07	0.14	0.17
Moderadament possible -----	0.10	0.15	0.35	0.45
Possible -----	0.25	0.35	0.55	0.65
Força possible -----	0.45	0.55	0.75	0.85
Molt possible -----	0.65	0.75	1	1
Practicament segur -----	0.95	0.98	1	1
Segur -----	1	1	1	1





# Apèndix IV: Equacions semàntiques de la modularització

## IV.1. Compilador

Presentem aquí les equacions semàntiques de la compilació de la sintaxi de la modularització del nivell estructural. Els objectes i operacions semàntiques estan definits en primer terme. El compilador està actualment en fase de desenvolupament.

Al final d'aquest apèndix presentem alguns exemples de compilació per facilitar la comprensió de les equacions.

Aquestes equacions no pretenen ser completes, però sí exemplificants de com es podria aplicar el mateix mecanisme de modularització sobre qualsevol altre sistema basat en regles. L'única cosa que d'una regla el sistema accedeix és a la seva conclusió i al seu identificador (elements que si s'exigeixen d'una regla). Altres aspectes com el tipus de lògica no afecten a l'anàlisi presentada.

### IV.1.1. Objectes i operacions semàntiques

Els símbols grecs que s'utilitzaran no corresponen amb els utilitzats a l'apèndix II. Els objectes, operacions i equacions presentats aquí s'han d'entendre independents d'aquells.

Per tots els objectes, es suposen existents les projeccions sobre els seus elements:

$$\text{primer } (A_1, A_2, \dots, A_n) = A_1$$

$$\text{segon } (A_1, A_2, \dots, A_n) = A_2$$

...

En aquestes equacions no s'ha tingut en compte el control dels tipus dels fets. Tenir-ho en compte implica la modificació de les equacions semàntiques de les regles, metaregles i precondicions.

## I Taula de submòduls

Domini  $\tau \in \text{taulasubmod} : \text{id-mòdul} \rightarrow \text{id-mòdul-intern}$

Les taules de submòduls donen la correspondència entre els noms dels submòduls i els seus noms interns. Aquests noms interns seran únics. La compartició de dos submòduls implica la compartició del seu nom intern.

## II Taula de fets

Domini  $\phi \in \text{taulafets} : \text{id-fet} \rightarrow (\text{id-fet-intern}, \text{tipus})$

Les taules de fets donen la correspondència entre els noms dels fets i els seus noms interns. Aquests noms interns seran únics. També conservem el tipus del fet. Aquestes taules són utilitzades en la definició dels conjunts.

## Operacions

int:  $\text{taulafets} \times \text{taulafets} \rightarrow \text{taulefets}$   
 { intersecció de taules de fets }

uni:  $\text{taulafets} \times \text{taulafets} \rightarrow \text{taulefets}$   
 { unió de taules de fets }

dif:  $\text{taulafets} \times \text{taulafets} \rightarrow \text{taulefets}$   
 { diferència conjuntista de taules de fets }

## III Taula de fets importats

Domini  $\delta \in \text{taulafetsimp} : \text{id-fet} \rightarrow (\text{id-fet}, \text{tipus})$

Les taules de fets importats donen la correspondència entre els noms dels fets importats per un mòdul i els seus noms interns. Aquests noms interns seran únics. Els fets importats es consideren globals i el seu nom intern serà el mateix que el nom extern.

## IV Taula de regles

Domini  $\rho \in \text{taularegles} : \text{id-regla} \rightarrow \text{id-regla-intern}$

Les taules de regles donen la correspondència entre els noms externs de les regles i els seus noms interns. Aquests noms interns seran únics.

## V Taula de metaregles

Domini  $\chi \in \text{taulametaregles} : \text{id-metaregla} \rightarrow \text{id-metaregla-intern}$

Les taules de metaregles donen la correspondència entre els noms externs de les metaregles i els seus noms interns. Aquests noms interns seran únics.

## VI Entorn d'especificacions

Domini  $\sigma \in \text{entorn\_espec} = \text{id-espec} \rightarrow \text{taulaespec}$   
 $\text{taulaespec} = (\text{taulasubmod}, \text{taulafets}, \text{taulafetsimp}, \text{taulafets},$

taulareg, taulametaregles )

Els entorns d'especificacions donen la relació entre un identificador d'especificació i els seus components. Representarem les instàncies  $\sigma$  de la següent manera  $\sigma = (\tau, \phi, \delta, \epsilon, \rho, \chi)$  on cada element representa:

- $\tau$ : Taula del submòdul.
- $\phi$ : Taula de fets. Utilitzada per enmagatzemar els elements pertanyents a un conjunt.
- $\delta$ : Taula de fets importats.
- $\epsilon$ : Taula de fets. Utilitzada per enmagatzemar els fets exportats.
- $\rho$ : Taula de regles.
- $\chi$ : Taula de metaregles.

Operacions

taulaespec<sub>0</sub> : taulaespec

{ taula d'especificacions buida }

$\sigma_0$ : entorn\_espec

{ entorn d'especificacions buit }

nousnomsint: taulaespec  $\rightarrow$  taulaespec

{ canvia tots els noms interns d'una taulaespec per nous noms interns }

taulasubmod: taulaespec  $\rightarrow$  taulasubmòdul

taulasubmod( $\sigma$ ) = primer( $\sigma$ )

$\cup$ : taulaespec  $\times$  taulaespec  $\rightarrow$  taulaespec

$(\tau_1, \phi_1, \delta_1, \epsilon_1, \rho_1, \chi_1) \cup (\tau_2, \phi_2, \delta_2, \epsilon_2, \rho_2, \chi_2) =$

$(\tau_1 \cup \tau_2, \phi_1 \cup \phi_2, \delta_1 \cup \delta_2, \epsilon_1 \cup \epsilon_2, \rho_1 \cup \rho_2, \chi_1 \cup \chi_2)$

{ entenent-se la unió de les taules com a unió de les entrades a les taules }

afegirsubmòdul: (idmod  $\times$  taulamod)<sup>n</sup>  $\times$  taulaespec  $\rightarrow$  taulaespec

Aquesta funció afegeix per cada *idmod* i *taulamod* un submòdul a *taulaespec*. A més per cada submòdul de *taulamod* concatena *idmod*, '-' i el nom extern del submòdul i els afegeix a *taulaespec*<sup>1</sup>.

VII Entorn de mòduls no paramètrics

Domini  $\mu \in$  entorn\_mod = id-mod  $\rightarrow$  taulamod

<sup>1</sup> Per exemple:

afegirsubmòdul( (A, (Modul-2, ([B $\rightarrow$ Mòdul-4],  $\phi, \delta, \epsilon, \rho, \chi$ ))), ( $\tau, \phi', \delta', \epsilon', \rho', \chi'$ ) =  
 ([B $\rightarrow$ Mòdul-4, A $\rightarrow$ B $\rightarrow$ Mòdul-4, A $\rightarrow$ Mòdul-2]  $\cup \tau, \phi \cup \phi', \delta \cup \delta', \epsilon \cup \epsilon',$   
 $\rho \cup \rho', \chi \cup \chi'$ ))

$\text{taulamod} = (\text{id-mod-int}, \text{taulaespec})$

Operacions

$\sigma_0$ :  $\text{entorn\_mod}$

{ entorn de mòduls buit }

acarar:  $\text{taulamod} \times \text{taulaespec} \rightarrow \text{taulamod}$

$\text{acarar}((\text{id-mod-int}, \text{taulaespec}_1), \text{taulaespec}_2)$   
 comprova si  $\text{taulaespec}_1$  respecta  $\text{taulaespec}_2$  en el sentit definit al capítol quatre. En cas que l'acarament sigui correcte es retorn una nova  $\text{taulamod}$  amb la substitució d' $\text{id-mod-int}$  per un identificador intern nou, i la restricció de  $\text{taulaespec}_1$  a  $\text{taulaespec}_2$ . Es a dir amb els fets importats de  $\text{taulaespec}_1$ , els fets exportats de  $\text{taulaespec}_2$  i els submòduls resultants de l'acarament dels submòduls de  $\text{taulaespec}_1$  a  $\text{taulaespec}_2$ .

Aquesta és la responsable de l'amagament d'informació.

submòdul:  $\text{idmod} \times \text{taulamod} \rightarrow \text{taulamod}$

$\text{submòdul}(\text{id-mod}, \text{taulamod})$  és el submòdul de nom  $\text{id}$  de la taula d'especificació, segon argument de  $\text{taulamod}$ . Es calcula obtenint de la taula d'especificació aquells mòduls i fets prefixats per  $\text{id-mod}$ .

### VIII Entorn de mòduls paramètrics

Domini  $\pi \in \text{entorn\_mod\_p} = \text{id-mod-p} \rightarrow (\text{llista-id-p}, \text{exprespec}, \text{taulaespec}, \text{expremod}, \sigma, \mu, \pi, \omega)$

Un mòdul genèric està representat per una 8-tupla on:

$\text{llista-id-p}$ : Llista dels identificadors dels paràmetres formals.

$\text{exprespec}$ : Expressió d'especificació declarada pel mòdul paramètric..

$\text{taulaespec}$ : Taula d'especificació calculada a partir dels arguments formals.

$\text{expremod}$ : Cos de la definició del mòdul.

$\sigma, \mu, \pi, \omega$ : Entorns d'avaluació.

Els mòduls genèrics són tractats com a macros que s'expandeixen en l'entorn existent al moment de la seva declaració.

Operacions

$\pi_0$ :  $\text{entorn\_mod\_p}$

{ entorn de mòduls paramètrics buit }

### IX Entorn de conjunts

Domini  $\omega \in \text{entorn\_con} = \text{id-conjunt} \rightarrow \text{taulacon}$

$\text{taulacon} = (\text{id-con-int}, \text{taulaespec})$

Operacions

$\omega_0$ : entorn\_con  
 { entorn de conjunts buit }

**X Codi**

Domini  $\varphi \in \text{codi} = (\text{regles}, \text{metaregles})$

**Operacions**

$\bullet$ :  $\text{codi} \times \text{codi} \rightarrow \text{codi}$

$(\text{reg}_1, \text{meta}_1) \bullet (\text{reg}_2, \text{meta}_2) = (\text{reg}_1 \bullet \text{reg}_2, \text{meta}_1 \bullet \text{meta}_2)$   
 { l'operador  $\bullet$  de la dreta és la concatenació de llistes. }

traduir:  $\text{codi} \times \text{taulaespec} \rightarrow \text{codi}$

Aquesta funció tradueix el codi que reb en codi del llenguatge pla de MILORD. Es a dir, es canvia les referències a fets que contenen les regles per referències als noms interns d'aquests fets. Aquesta relació ens la proporciona *taulaespec*. De la mateixa manera es tradueixen les metaregles. Un codi es tradueix només un cop, encara que s'apliqui un altre cop sobre ell la funció traduir. L'objectiu de tot el procés de compilació es el d'obtenir el resultat d'aquesta funció sobre el conjunt de la BC. De fet totes les equacions semàntiques tenen com a missió preparar *taulaespec*, de manera que la funció traduir pugui actuar.

**IV.1.2 Equacions semàntiques**

NIVESTR: *nivell estructural*  $\rightarrow$   $\text{taulamod} \times \text{codi}$

$\text{NIVESTR} \llbracket \text{Nivell estructural declmod finivell} \rrbracket = (\text{taulamod}, \text{codi})$   
 $(\text{taulamod}, \text{codi}) = \text{EXPREM} \llbracket \text{Inici declmod final} \rrbracket \text{taulaespec}_0 \sigma_0 \mu_0 \pi_0 \omega_0$

DECLMOD: *declmod*  $\times$  *taulaespec*  $\times$  *entorns*<sup>1</sup>  $\rightarrow$  *taulaespec*  $\times$  *codi*  $\times$  *entorns*

$\text{DECLMOD} \llbracket \text{Especificació vinclespec} \rrbracket \text{taulaespec } \sigma \mu \pi \omega =$   
 $= (\text{taulaespec}', \text{codi}, \sigma', \mu, \pi, \omega)$   
 $(\text{taulaespec}', \sigma') = \text{VINCLESPEC} \llbracket \text{vinclespec} \rrbracket \sigma \omega$   
 $\text{codi} = \emptyset$

<sup>1</sup> Per simplificar utilitzarem la següent abreviatura:

*entorns* : *entorn\_espec*  $\times$  *entorn-mod*  $\times$  *entorn\_mod\_p*  $\times$  *entorn\_con*

DECLMOD[[ *Mòdul vinclemod* ]] taulaespec  $\sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}', \text{codi}, \sigma, \mu', \pi', \omega )$   
 $( \text{taulaespec}', \text{codi}, \mu', \pi' ) = \text{VINCLEMOD}[[ \text{vinclemod} ]] \text{taulaespec } \sigma \mu \pi \omega$

DECLMOD[[ *Conjunt vincleconj* ]] taulaespec  $\sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}', \text{codi}, \sigma, \mu, \pi, \omega' )$   
 $( \text{taulaespec}', \omega' ) = \text{VINCLECON}[[ \text{vincleconj} ]] \text{taulaespec } \omega$   
 $\text{codi} = \emptyset$

DECLMOD[[ *declmod1 declmod2* ]] taulaespec  $\sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}'', \text{code1}.\text{code2}, \sigma'', \mu'', \pi'', \omega'' )$   
 $( \text{taulaespec}', \text{code1}, \sigma', \mu', \pi', \omega' ) =$   
 $= \text{DECLMOD}[[ \text{declmod1} ]] \text{taulaespec } \sigma \mu \pi \omega$   
 $( \text{taulaespec}'', \text{code2}, \sigma'', \mu'', \pi'', \omega'' ) =$   
 $= \text{DECLMOD}[[ \text{declmod2} ]] \text{taulaespec}' \sigma' \mu' \pi' \omega'$

VINCLESPEC:  $\text{vinclespec} \times \text{entorn\_espec} \times \text{entorn\_con} \rightarrow \text{id-espec} \times \text{taulaespec} \times$   
 $\text{entorn\_espec}$

VINCLESPEC[[ *idespec = exprespec* ]]  $\sigma \omega = ( \text{idespec}, \text{taulaespec}, \sigma' )$   
 $\text{taulaespec} = \text{EXPRESPE}[[ \text{exprespec} ]] \sigma \omega$   
 $\sigma' = [\text{idespec} \rightarrow \text{taulaespec}] \sigma$

VINCLEMOD:  $\text{vinclemod} \times \text{taulaespec} \times \text{entorns} \rightarrow \text{taulaespec} \times \text{codi} \times \text{entorn\_mod} \times$   
 $\text{entorn\_mod\_p}$

VINCLEMOD[[ *idmod = expremod* ]] taulaespec  $\sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}', \text{codi}, \mu', \pi )$   
 $( \text{taulamod}, \text{codi} ) = \text{EXPREMODO}[[ \text{expremod} ]] \text{taulaespec}_0 \sigma \mu \pi \omega$   
 $\text{taulaespec}' = \text{afegirsubmod}( ( \text{idmod}, \text{taulamod} ), \text{taulaespec} )$   
 $\mu' = [\text{idmod} \rightarrow \text{taulamod}] \mu$

VINCLEMOD[[ *idmod : exprespec = expremod* ]] taulaespec  $\sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}', \text{codi}, \mu', \pi )$   
 $( \text{taulamod}, \text{codi} ) = \text{EXPREMODO}[[ \text{expremod} ]] \text{taulaespec}_0 \sigma \mu \pi \omega$

$taulaespec'' = \text{EXPRESPE} \llbracket \text{exprespec} \rrbracket$   
 $taulamod' = \text{acarar}(\text{taulamod}, \text{taulaespec}'')$   
 $taulaespec' = \text{afegirsubmòdul}((\text{idmod}, \text{taulamod}'), \text{taulaespec})$   
 $\mu' = [\text{idmod} \rightarrow \text{taulamod}' ]\mu$

$\text{VINCLEMOD} \llbracket \text{idmod}(\text{llistaparam}): \text{exprespec} = \text{expremod} \rrbracket \text{taulaespec } \sigma \mu \pi \omega =$   
 $\text{VINCLEMOD} \llbracket \text{idmod}(\text{llistaparam}) = \text{expremod} : \text{exprespec} \rrbracket \text{taulaespec } \sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}, \text{codi}, \mu, \pi')$   
 $(\text{llista-id-p}, \text{taulaespec}') = \text{PLIST} \llbracket \text{llistaparam} \rrbracket \sigma \omega$   
 $\pi' = [\text{idmod} \rightarrow (\text{llista-id-p}, \text{exprespec}, \text{taulaespec}', \text{expremod}, \sigma, \mu, \pi, \omega)]\pi$   
 $\text{codi} = \emptyset$

$\text{VINCLEMOD} \llbracket \text{idmod}(\text{llistaparam}) = \text{expremod} \rrbracket \text{taulaespec } \sigma \mu \pi \omega =$   
 $= ( \text{taulaespec}, \text{codi}, \mu, \pi')$   
 $(\text{llista-id-p}, \text{taulaespec}') = \text{PLIST} \llbracket \text{llistaparam} \rrbracket \sigma \omega$   
 $\pi' = [ \text{modid} \rightarrow (\text{pid-list}, \emptyset, \text{taulaespec}', \text{expremod}, \sigma, \mu, \pi, \omega) ]\pi$   
 $\text{codi} = \emptyset$

**PLIST:**  $\text{llistaparam} \times \text{entorn\_espec} \times \text{entorn\_con} \rightarrow \text{llista-id-p} \times \text{taulaespec}$

$\text{PLIST} \llbracket \text{id-p}_1 : \text{exprespec}_1 ; \dots ; \text{id-p}_n : \text{exprespec}_n \rrbracket \sigma \omega =$   
 $= ( (\text{id-p}_1, \dots, \text{id-p}_n) \text{taulaespec} )$   
 $\text{taulaespec} = \text{ESPEC} \llbracket \text{Mòdul } \text{id-p}_1 : \text{exprespec}_1 \dots$   
 $\text{Mòdul } \text{id-p}_n : \text{exprespec}_n \rrbracket \text{taulaespec}_0 \sigma \omega$

**VINCLECON:**  $\text{vincleconj} \times \text{taulaespec} \times \text{entorn\_con} \rightarrow \text{taulaespec} \times \text{entorn\_con}$

$\text{VINCLECON} \llbracket \text{idconjunt} = (\text{fet}_1 : \text{tipus}_1 ; \dots ; \text{fet}_n : \text{tipus}_n) \rrbracket \text{taulaespec } \omega =$   
 $= ( \text{taulaespec}', \omega')$   
 $\text{taulaespec} = ( \tau, \phi, \delta, \epsilon, \rho, \chi )$   
 $\text{taulaespec}' = ( \tau, [\text{fet}_1 \rightarrow (\text{gensym}(\text{"fet-"}), \text{tipus}_1), \dots,$   
 $\text{fet}_n \rightarrow (\text{gensym}(\text{"fet-"}), \text{tipus}_n) ]\phi, \delta, \epsilon, \rho, \chi )$   
 $\omega' = [ \text{idconjunt} \rightarrow (\text{gensym}(\text{"Conjunt-"}), \text{taulaespec}') ]\omega$

$\text{VINCLECON} \llbracket \text{idconjunt} = \text{idconjunt}_1 \text{ operador\_conjunt } \text{idconjunt}_2 \rrbracket$   
 $\text{taulaespec } \omega' = ( \text{taulaespec}', \omega')$   
 $\text{taulaespec} = ( \tau, \phi, \delta, \epsilon, \rho, \chi )$

$\phi' = (\text{cas operador\_conjunt de}$   
 $\cup \rightarrow \text{uni}(\text{segon}(\text{segon}(\omega(\text{idconjunt}_1))), \text{segon}(\text{segon}(\omega(\text{idconjunt}_2))))$   
 $\sqcap \cap \rightarrow \text{int}(\text{segon}(\text{segon}(\omega(\text{idconjunt}_1))), \text{segon}(\text{segon}(\omega(\text{idconjunt}_2))))$   
 $\sqcap / \rightarrow \text{dif}(\text{segon}(\text{segon}(\omega(\text{idconjunt}_1))), \text{segon}(\text{segon}(\omega(\text{idconjunt}_2))))$   
 $\text{taulaespec}' = (\tau, \text{uni}(\phi \phi'), \delta, \epsilon, \rho, \chi)$   
 $\omega' = [ \text{idconjunt} \rightarrow (\text{gensym}(\text{"Conjunt-"} \text{taulaespec}') ] \omega$   
 $\Leftrightarrow \text{ERROR SI } \omega(\text{idconjunt}_1) = \perp$   
 $\text{MISSATGE "Conjunt inexistent." idconjunt}_1$   
 $\text{RETORN} (\text{taulaespec}, \omega)$   
 $\Leftrightarrow \text{ERROR SI } \omega(\text{idconjunt}_2) = \perp$   
 $\text{MISSATGE "Conjunt inexistent." idconjunt}_2$   
 $\text{RETORN} (\text{taulaespec}, \omega)$

EXPRESPE:  $\text{exprespec} \times \text{entorn\_espec} \times \text{entorn\_con} \rightarrow \text{taulaespec}$

EXPRESPE[[ *idespec* ]]  $\sigma \omega = \text{taulaespec}$   
 $\text{taulaespec} = \sigma(\text{idespec})$

EXPRESPE[[ *inici espec final* ]]  $\sigma \omega = \text{taulaespec}$   
 $\text{taulaespec} = \text{ESPEC} [[ \text{espec} ]] \text{taulaespec} \sigma \omega$

ESPEC:  $\text{espec} \times \text{entorn\_espec} \times \text{entorn\_con} \rightarrow \text{taulaespec}$

ESPEC[[ *Mòdul idmod : exprespec* ]]  $\text{taulaespec} \sigma \omega = \text{taulaespec''}$   
 $\text{taulaespec}' = \text{nousnomsint}(\text{EXPRESPE} [[ \text{exporespec} ]] \sigma \omega)$   
 $\text{taulaespec''} = \text{afegirsubmòdul} ( (\text{idmod}, (\text{gensym}(\text{"Module-"}), \text{taulaespec}')),$   
 $\text{taulaespec}$

ESPEC[[ *Importa = (fet<sub>1</sub> : tipus<sub>1</sub> ; ... ; fet<sub>n</sub> : tipus<sub>n</sub>)* ]]  $\text{taulaespec} \sigma \omega =$   
 $= \text{taulaespec}'$

$\text{taulaespec} = (\tau, \phi, \delta, \epsilon, \rho, \chi)$   
 $\text{taulaespec}' = (\tau, \phi, [ \text{fet}_i \rightarrow (\text{fet}_i \text{ tipus}_i), \forall i \in [1 \text{ n}]] \delta, \epsilon, \rho, \chi)$

ESPEC[[ *Importa = idconjunt* ]]  $\text{taulaespec} \sigma \omega = \text{taulaespec''}$   
 $\text{taulaespec}' = \omega(\text{idconjunt}) = (\tau, \phi, \delta, \epsilon, \rho, \chi)$   
 $\text{taulaespec''} = (\tau, \phi_0, \phi, \epsilon, \rho, \chi) \cup \text{taulaespec}$   
 $\Leftrightarrow \text{ERROR SI } \omega(\text{idconjunt}) = \perp$



MISSATGE "Conjunt inexistent." *idconjunt*  
 RETORN *taulaespec*  
 ESPEC[ *Exporta = (fet<sub>1</sub> : tipus<sub>1</sub> ; ... ; fet<sub>n</sub> : tipus<sub>n</sub>)* ] *taulaespec*  $\sigma \omega =$   
 $= \text{taulaespec}'$   
*taulaespec* = (  $\tau, \phi, \delta, \epsilon, \rho$  )  
*taulaespec'* = (  $\tau, \phi, \delta, [fet_i \rightarrow ((\text{gensym}(\text{"fet-"})) \text{tipus}_i), \forall i \in [1 \ n]]\epsilon, \rho$  )

ESPEC[ *Exporta = idconjunt* ] *taulaespec*  $\sigma \omega = \text{taulaespec}''$   
*taulaespec'* =  $\omega(\text{idconjunt}) = (\tau, \phi, \delta, \epsilon, \rho)$   
*taulaespec''* = (  $\tau, \phi_0, \delta, \phi, \rho$  )  $\cup$  *taulaespec*  
 $\Leftrightarrow$  ERROR SI  $\omega(\text{idconjunt}) = \perp$

MISSATGE "Conjunt inexistent." *idconjunt*  
 RETORN *taulaespec*

ESPEC[ *espec<sub>1</sub> espec<sub>2</sub>* ] *taulaespec*  $\sigma \omega = \text{taulaespec}''$   
*taulaespec'* = ESPEC[ *espec<sub>1</sub>* ] *taulaespec*  $\sigma \omega$   
*taulaespec''* = ESPEC[ *espec<sub>2</sub>* ] *taulaespec'*  $\sigma \omega$

EXPREMODO: *expremod*  $\times$  *taulaespec*  $\times$  *entorns*  $\rightarrow$  *taulamod*  $\times$  *codi*

EXPREMODO[ *idmod* ] *especparam*  $\sigma \mu \pi \omega = (\text{taulamod}, \text{codi})$   
*taulamod* =  $\mu(\text{idmod})$   
*codi* =  $\emptyset$   
 $\Leftrightarrow$  ERROR SI  $\mu(\text{idmod}) = \perp$

MISSATGE "Mòdul inexistent." *idmod*  
 RETORN *fracàs*

EXPREMODO[ *idmod->camí* ] *especparam*  $\sigma \mu \pi \omega = (\text{taulamod}, \text{codi})$   
*taulamod* = *submòdul*( *camí*,  $\mu(\text{idmod})$  )  
*codi* =  $\emptyset$   
 $\Leftrightarrow$  ERROR SI *submòdul*( *camí*,  $\mu(\text{idmod})$  ) =  $\perp$

MISSATGE "Mòdul inexistent." *idmod->camí*  
 RETORN *fracàs*

EXPREMODO[ *idcamí( expremod<sub>1</sub> ; ... ; expremod<sub>n</sub> )* ] *especparam*  $\sigma \mu \pi \omega =$   
 $= (\text{taulamod}, \text{codi})$   
 $\pi(\text{idcamí}) = (\text{id-p}_1, \dots, \text{id-p}_n, \text{exprespec}, \text{taulaespec}, \text{expremod}, \sigma', \mu', \pi', \omega')$

$(\text{taulamod}_i, \text{codi}_i) = \text{EXPREMOD}[\![ \text{expremod}_i ]\!] \text{ especparam } \sigma \mu \pi \omega, \forall i \in [1 n]$   
 $\text{taulamod}' = \text{acarar}(\text{gensym}(\text{"Module-"}),$   
 $\quad \text{afegirsubmòdul}(\{ (\text{id-p}_i, \text{taulamod}_i), \forall i \in [1 n] \},$   
 $\quad \quad \text{taulaespec}_0))$   
 $\quad \text{taulaespec}$   
 $\mu'' = [ \text{id-p}_i \rightarrow \text{submòdul}(\text{id-p}_i, \text{taulamod}'), \forall i \in [1 n] ] \mu'$   
 $\text{taulamod}' = (\text{nom-mòdul especparam})$   
 $(\text{taulamod}, \text{codi}') =$   
 $\quad = \text{EXPREMOD}[\![ \text{expremod} : \text{exprespec} ]\!] \text{ especparam } \sigma' \mu'' \pi' \omega'$   
 $\text{codi} = \text{codi}_1 \bullet \text{codi}_2 \bullet \dots \bullet \text{codi}_n \bullet \text{codi}'$   
 $\Leftrightarrow \text{ERROR SI acarar}(\text{gensym}(\text{"Module-"}),$   
 $\quad \text{afegirsubmòdul}(\{ (\text{id-p}_i, \text{taulamod}_i), \forall i \in [1 n] \},$   
 $\quad \quad \text{taulaespec}_0))$   
 $\quad \text{taulaespecdef}) = \perp$   
 MISSATGE "Mòdul inexistent." *idmod*  
 RETORN fracàs

$\text{EXPREMOD}[\![ \text{Inici decl final} ]\!] \text{ especparam } \sigma \mu \pi \omega = (\text{taulamod}, \text{codi})$   
 $(\text{taulaespec}, \text{codi}', \sigma, \mu', \pi', \omega) = \text{DECL}[\![ \text{decl} ]\!] \text{taulaespec}_0 \sigma \mu \pi \omega$   
 $\text{taulamod} = (\text{gensym}(\text{"Mòdul-"}), \text{taulaespec})$   
 $\text{codi} = \text{traduir}(\text{codi}', \text{taulaespec} \cup \text{especparam})$

$\text{EXPREMOD}[\![ \text{expremod} : \text{exprespec} ]\!] \text{ especparam } \sigma \mu \pi \omega = (\text{taulamod}, \text{codi})$   
 $(\text{taulamod}', \text{codi}) = \text{EXPREMOD}[\![ \text{expremod} ]\!] \text{ especparam } \sigma \mu \pi \omega$   
 $\text{taulamod} = \text{acarar}(\text{taulamod}', \text{EXPRESPE}[\![ \text{exprespec} ]\!] \sigma)$

DECL:  $\text{decl} \times \text{taulaespec} \times \text{entorns} \rightarrow \text{taulaespec} \times \text{codi} \times \text{entorns}$

$\text{DECL}[\![ \text{capçal cos} ]\!] \text{taulaespec } \sigma \mu \pi \omega =$   
 $\quad = (\text{taulaespec}'', \text{codi}_1 \bullet \text{codi}_2, \sigma', \mu', \pi', \omega')$   
 $(\text{taulaespec}', \text{codi}_1, \sigma', \mu', \pi', \omega') = \text{CAPÇAL}[\![ \text{capçal} ]\!] \text{taulaespec } \sigma \mu \pi \omega$   
 $(\text{taulaespec}'', \text{codi}_2) = \text{COS}[\![ \text{cos} ]\!] \text{taulaespec}'$

$\text{DECL}[\![ \text{cos} ]\!] \text{taulaespec } \sigma \mu \pi \omega = (\text{taulaespec}', \text{codi}, \sigma, \mu, \pi, \omega)$   
 $(\text{taulaespec}', \text{codi}) = \text{COS}[\![ \text{cos} ]\!] \text{taulaespec}$

CAPÇAL: *capçal* × *taulaespec* × *entorns* → *taulaespec* × *codi* × *entorns*

$$\begin{aligned} \text{CAPÇAL} \llbracket \text{declnucli} \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= ( \text{taulaespec}', \text{codi}, \sigma', \mu', \pi', \omega' ) \\ ( \text{taulaespec}', \text{codi}, \sigma', \mu', \pi', \omega' ) &= \text{NUCLI} \llbracket \text{declnucli} \rrbracket \text{taulaespec } \sigma \mu \pi \omega \end{aligned}$$

$$\begin{aligned} \text{CAPÇAL} \llbracket \text{declmod} \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= ( \text{taulaespec}', \text{codi}, \sigma', \mu', \pi', \omega' ) \\ ( \text{taulaespec}', \text{codi}, \sigma', \mu', \pi', \omega' ) &= \text{DECLMOD} \llbracket \text{declmod} \rrbracket \text{taulaespec } \sigma \mu \pi \omega \end{aligned}$$

$$\begin{aligned} \text{CAPÇAL} \llbracket \text{declmod declnucli} \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= \\ &= ( \text{taulaespec}'', \text{code}_1 \bullet \text{code}_2, \sigma'', \mu'', \pi'', \omega'' ) \\ ( \text{taulaespec}', \text{codi}_1, \sigma', \mu', \pi', \omega' ) &= \\ &= \text{DECLMOD} \llbracket \text{declmod} \rrbracket \text{taulaespec } \sigma \mu \pi \omega \\ ( \text{taulaespec}'', \text{codi}_2, \sigma'', \mu'', \pi'', \omega'' ) &= \\ &= \text{NUCLI} \llbracket \text{declnucli} \rrbracket \text{taulaespec}', \sigma', \mu', \pi', \omega' \end{aligned}$$

NUCLI: *declnucli* × *taulaespec* × *entorns* → *taulaespec* × *codi* × *entorns*

$$\begin{aligned} \text{NUCLI} \llbracket \text{Obrir idmod} \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= ( \text{taulaespec}'', \text{codi}, \sigma, \mu', \pi, \omega ) \\ ( (\text{id-mod-int}, \text{taulaespec}'), \text{codi} ) &= \text{EXPRMOD} \llbracket \text{idmod} \rrbracket \sigma \mu \pi \omega \\ \mu' &= [ \text{id}_i \rightarrow \text{submòdul}(\text{id}_i, \text{taulaespec}') , \\ &\quad \forall \text{id}_i \in \text{domini}(\text{taulasubmod}(\text{taulaespec}')) ] \mu \\ \text{taulaespec}'' &= \text{taulaespec} \cup \text{taulaespec}' \end{aligned}$$

$$\begin{aligned} \text{NUCLI} \llbracket \text{importats} \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= ( \text{taulaespec}', \text{codi}, \sigma, \mu, \pi, \omega ) \\ \text{taulaespec}' &= \text{ESPEC} \llbracket \text{importats} \rrbracket \text{taulaespec } \sigma \omega \\ \text{codi} &= \emptyset \end{aligned}$$

$$\begin{aligned} \text{NUCLI} \llbracket \text{exportats} \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= ( \text{taulaespec}', \text{codi}, \sigma, \mu, \pi, \omega ) \\ \text{taulaespec}' &= \text{ESPEC} \llbracket \text{exportats} \rrbracket \text{taulaespec } \sigma \omega \\ \text{codi} &= \emptyset \end{aligned}$$

$$\begin{aligned} \text{NUCLI} \llbracket \text{declnucli}_1 \text{ declnucli}_2 \rrbracket \text{taulaespec } \sigma \mu \pi \omega &= \\ &= ( \text{taulaespec}'', \text{code}_1 \bullet \text{code}_2, \sigma'', \mu'', \pi'', \omega'' ) \\ ( \text{taulaespec}', \text{codi}_1, \sigma', \mu', \pi', \omega' ) &= \\ &= \text{NUCLI} \llbracket \text{declnucli}_1 \rrbracket \text{taulaespec } \sigma \mu \pi \omega \\ ( \text{taulaespec}'', \text{codi}_2, \sigma'', \mu'', \pi'', \omega'' ) &= \\ &= \text{NUCLI} \llbracket \text{declnucli}_2 \rrbracket \text{taulaespec}', \sigma', \mu', \pi', \omega' \end{aligned}$$

COS:  $cos \times \text{taulaespec} \rightarrow \text{taulaespec} \times \text{codi}$

COS[[ *precondició regles metaregles* ]] *taulaespec* =  
 = ( *taulaespec'*,  $\text{codi}_1 \bullet \text{codi}_2 \bullet \text{codi}_3$  )  
 ( *taulaespec'*, *llista*,  $\text{codi}_1$  ) = REGLES[[ *regles* ]] *taulaespec*  
 ( *taulaespec''*,  $\text{codi}_2$  ) = META[[ *metaregles* ]] *taulaespec'*  
 ( *taulaespec'''*,  $\text{codi}_3$  ) = PRECON[[ *precondició* ]] *taulaespec''* *llista*

REGLES:  $\text{regles} \times \text{taulaespec} \rightarrow \text{taulaespec} \times \text{Llista idregla} \times \text{codi}$

REGLES[[ **Regles** *regla*<sub>1</sub> ... *regla*<sub>n</sub> ]] *taulaespec* = ( *taulaespec'*, *llista*, *codi* )  
*taulaespec* = (  $\tau$ ,  $\phi$ ,  $\delta$ ,  $\epsilon$ ,  $\rho$ ,  $\chi$  )  
*taulaespec'* = (  $\tau$ ,  $\phi$ , [ *fet*<sub>i</sub>  $\rightarrow$  ( *fet*<sub>i</sub> tipus-inferit ),  $\forall$  *fet*<sub>i</sub>  $\in$  *premissesregles*,  
 $\notin$  *fetsexportats*( $\tau$ )] $\delta$ ,  
 [ *fet*<sub>i</sub>  $\rightarrow$  ( *gensym*("fet-") difús),  $\forall$   $i \in [1 \ n]$   $i$   
*fet*<sub>i</sub> = *conclusió*(*id-regla*<sub>i</sub>)] $\epsilon$ ,  
 [ *id-regla*<sub>i</sub>  $\rightarrow$  *gensym*("Regla-"),  $\forall$   $i \in [1 \ n]$ ] $\rho$ ,  $\chi$  )  
 { *fet*<sub>i</sub> són identificadors de fets i *id-regla*<sub>i</sub> són identificadors de  
 regla. Tipus-inferit fa referència al tipus més general aplicable al fet  
 segons aparegui en la premissa de la regla. No desenvolupat aquí. }  
*llista* = ( *id-regla*<sub>1</sub> ... *id-regla*<sub>n</sub> )  
*codi* = ( ( *regla*<sub>1</sub> ... *regla*<sub>n</sub> ) () )

REGLES[[ $\lambda$ ]] *taulaespec* = ( *taulaespec'*, *llista*, *codi* )  
*llista* = ()  
*codi* = ()

META:  $\text{metaregles} \times \text{taulaespec} \rightarrow \text{taulaespec} \times \text{codi}$

META[[ **Metaregles** *meta*<sub>1</sub> ... *meta*<sub>n</sub> ]] *taulaespec* = ( *taulaespec'*, *codi* )  
*taulaespec* = (  $\tau$ ,  $\phi$ ,  $\delta$ ,  $\epsilon$ ,  $\rho$ ,  $\chi$  )  
*taulaespec'* = (  $\tau$ ,  $\phi$ , [ *fet*<sub>i</sub>  $\rightarrow$  ( *fet*<sub>i</sub> tipus-inferit ),  $\forall$  *fet*<sub>i</sub>  $\in$  *premissesmeta*,  
 $\notin$  *fetsexportats*( $\tau$ ),  $\notin$   $\epsilon$  ] $\delta$ ,  $\epsilon$ ,  $\rho$ ,  
 [ *meta*<sub>i</sub>  $\rightarrow$  *gensym*("Meta-"),  $\forall$   $i \in [1 \ n]$ ] $\chi$  )  
 { *fet*<sub>i</sub> són identificadors de fets, *meta*<sub>i</sub> són identificadors de  
 metaregla. Tipus-inferit fa referència al tipus més general aplicable  
 al fet segons aparegui en la premissa de la metaregla. No  
 desenvolupat aquí. }

$\text{codi} = ( () ( \text{meta}_1 \dots \text{meta}_n ) )$

$\text{META} \llbracket \lambda \rrbracket \text{taulaespec} = ( \text{taulaespec}, \text{codi} )$

$\text{codi} = \emptyset$

$\text{PRECON}: \text{precondició} \times \text{taulaespec} \times \text{Llista idregla} \rightarrow \text{taulaespec} \times \text{codi}$

$\text{PRECON} \llbracket \text{Precondició premissa} \rrbracket \text{taulaespec llista} = ( \text{taulaespec}', \text{codi} )$

$\text{taulaespec} = ( \tau, \phi, \delta, \epsilon, \rho, \chi )$

$\text{idmeta} = \text{gensym}(\text{"Precon"})$

$\text{taulaespec}' = ( \tau, \phi, [\text{fet}_i \rightarrow (\text{fet}_i \text{ tipus-inferit}), \forall i \in \text{premissa},$   
 $i \notin \text{fetsexportats de } \tau] \delta, \epsilon, \rho,$

$[\text{idmeta} \rightarrow \text{gensym}(\text{"Meta-"}), \forall i \in [1 \ n]] \chi )$   
 {Tipus-inferit fa referència al tipus més general aplicable al fet  
 segons aparegui en la premissa. No desenvolupat aquí. }

$\text{codi} = ( () ( \{ \text{idmeta Si no condició; Llavors inhibir regles llista} /$   
 $\text{premissa} = \wedge \text{condició}_i \} ) )$

$\text{PRECON} \llbracket \lambda \rrbracket \text{taulaespec llista} = ( \text{taulaespec}, \text{codi} )$

$\text{codi} = \emptyset$

## IV.2. Exemple de compilació

A continuació, i utilitzant les equacions semàntiques de la part IV.1, presentem quatre exemples de compilació utilitzant la tècnica descrita en aquest capítol.

Exemple1:

**Conjunt Radiologia** = (distribució:[Lobar, Broncopneumònica]; embassament:  
 Booleà; lòbuls: Numèric; cavitada: Difús)

$\text{DECLMOD} \llbracket \text{Conjunt vincleconj} \rrbracket \text{taulaespec}_0 \sigma_0 \mu_0 \pi_0 \omega_0$

$\text{VINCLECONJ} \llbracket \text{Radiologia} = (\text{distribució}:[\text{Lobar}, \text{Broncopneumònica}];$

$\text{embassament: Booleà; lòbuls: Numèric; cavitada: Difús}) \rrbracket \text{taulaespec}_0 \omega_0$

$\text{taulaespec}' = ( \tau_0, [\text{distribució} \rightarrow (\text{Fet-1} [\text{Lobar}, \text{Broncopneumònica}],$   
 $\text{embassament} \rightarrow (\text{Fet-2 Booleà}),$   
 $\text{lòbuls} \rightarrow (\text{Fet-3 Numèric}),$

cavitada→(Fet-4 Difús) ],  $\delta_0, \epsilon_0, \pi_0, \chi_0$ )

$\omega' = [\text{Radiologia} \rightarrow (\text{Conjunt-1 taulaespec'})]$

$\rightarrow (\text{taulaespec}' \omega')$

$\rightarrow (\text{taulaespec}' \emptyset \sigma_0 \mu_0 \pi_0 \omega') =$

(

( $\tau_0, [\text{distribució} \rightarrow (\text{Fet-1} [\text{Lobar, Broncopneumònica}],$

embassament→(Fet-2 Booleà),

lòbuls →(Fet-3 Numèric),

cavitada→(Fet-4 Difús) ],  $\delta_0, \epsilon_0, \pi_0, \chi_0$ )

$\emptyset \sigma_0 \mu_0 \pi_0$

[Radiologia→(Conjunt-1 taulaespec')]

)

Exemple 2:

**Especificació Pneum =**

**Inici**

**Exporta = (Pneumococ: Difús)**

**Final**

DECLMOD[[ **Especificació vinclespec** ]] taulaespec $_0$   $\sigma_0$   $\mu_0$   $\pi_0$   $\omega_0$

VINCLESPEC[[ **Pneum = exprespec** ]]  $\sigma_0$   $\omega_0$

EXPRESPE[[ **inici espec final** ]]  $\sigma_0$   $\omega_0$

ESPEC[[ **Exporta = (Pneumococ: Difús)** ]] taulaespec $_0$   $\sigma_0$   $\omega_0$

taulaespec' = ( $\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-5 difús})]$ ),  $\pi_0, \chi_0$ )

$\rightarrow$  taulaespec'

$\rightarrow$  taulaespec'

$\sigma' = [\text{Pneum} \rightarrow \text{taulaespec}']$

$\rightarrow (\text{Pneum taulaespec}' \sigma')$

$\rightarrow (\text{taulaespec}' \emptyset \sigma' \mu_0 \pi_0 \omega_0) =$

(

( $\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-5 difús})]$ ),  $\pi_0, \chi_0$ )

$\emptyset$

[Pneum → taulaespec']

$\mu_0 \pi_0 \omega_0$

)

Exemple 3:

**Mòdul** Combinació\_i (X: Pneum; Y: Pneum): Pneum =

**Inici**

**Regles**

R1 Si **no\_dem** X->pneumococ i Y->pneumococ

**Llavors** Pneumococ és segur

R2 Si X->pneumococ i **no\_dem** Y->pneumococ

**Llavors** Pneumococ és segur

R3 Si X->pneumococ i Y->pneumococ

**Llavors** Pneumococ és segur

**final**

$\sigma' = [\text{Pneum} \rightarrow (\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-5 difús})], \pi_0, \chi_0)]$

DECLMOD[[ **Mòdul** *vinclemod* ]] taulaespec<sub>0</sub>  $\sigma'$   $\mu_0$   $\pi_0$   $\omega_0$

VINCLEMOD[[ **Combinació\_i** (X: Pneum; Y: Pneum): Pneum = *expremod* ]]

taulaespec<sub>0</sub>  $\sigma'$   $\mu_0$   $\pi_0$   $\omega_0$

PLIST[[ X: Pneum; Y: Pneum ]]  $\sigma'$   $\omega_0$

ESPEC[[ **Mòdul** X: Pneum **Mòdul** Y: Pneum ]] taulaespec<sub>0</sub>  $\sigma'$   $\omega_0$

ESPEC[[ **Mòdul** X: Pneum ]] taulaespec<sub>0</sub>  $\sigma'$   $\omega_0$

EXPRESPE[[ Pneum ]]  $\sigma'$   $\omega_0$

taulaesp = s'(Pneum) =

= ( $\tau_0, \phi_0, \delta_0,$

[Pneumococ  $\rightarrow$  (fet-5 difús)],

$\pi_0, \chi_0$ )

$\rightarrow$  taulaesp

taulaesp' = nousnomsint(taulaesp) =

( $\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-6 difús})], \rho_0, \chi_0$ )

taulaesp'' = afegirsubmòdul((X (Module-1 taulaesp'))

taulaespec<sub>0</sub>) =

([X  $\rightarrow$  Module-1],  $\phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0$ ) QUE taulaesp..

$\rightarrow$  taulaesp''

ESPEC[[ **Mòdul** Y: Pneum ]] taulaesp''  $\sigma'$   $\omega_0$

EXPRESPE[[ Pneum ]]  $\sigma'$   $\omega_0$

taulaesp = s'(Pneum) =

$$\begin{aligned}
&= (\tau_0, \phi_0, \delta_0, \\
&\quad [\text{Pneumococ} \rightarrow (\text{fet-5 difús})], \\
&\quad \pi_0, \chi_0) \\
&\rightarrow \text{taulaesp} \\
\text{taulaesp}' &= \text{nousnomsint}(\text{taulaesp}) = \\
&\quad (\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-7 difús})], \rho_0, \chi_0) \\
\text{taulaesp}''' &= \text{afegirsubmòdul}((X (\text{Module-2 taulaesp}')) \\
&\quad \text{taulaesp}'') = \\
&\quad ([X \rightarrow \text{Module-1}, Y \rightarrow \text{Module-2}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0) \\
&\rightarrow \text{taulaesp}''' \\
&\rightarrow \text{taulaesp}''' \\
&\rightarrow ((X Y) \text{taulaesp}''') \\
(\text{llista-id-p}, \text{taulaesp}') &= ((X Y) \text{taulaesp}''') \\
\pi' &= [\text{Combinació}_i \rightarrow ((X Y), \text{Pneum}, \text{taulaesp}', \text{expremod}, \sigma', \mu_0, \pi_0, \omega_0)] \\
&\rightarrow (\text{taulaesp}_0, \emptyset, \mu_0, \pi') \\
\rightarrow (\text{taulaesp}_0, \emptyset, \sigma', \mu_0, \pi', \omega_0) &= \\
( \\
\text{taulaesp}_0, \emptyset, \sigma_0, \mu_0, \\
[\text{Combinació}_i \rightarrow ((X Y), \text{Pneum}, \\
&\quad ([X \rightarrow \text{Module-1}, Y \rightarrow \text{Module-2}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0), \\
&\quad \text{expremod}, \\
&\quad [\text{Pneum} \rightarrow (\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-5 difús})], \pi_0, \chi_0)], \\
&\quad \mu_0, \pi_0, \omega_0)], \\
\omega_0 \\
)
\end{aligned}$$

*expremod* = Inici

**Regles**

R1 Si no\_dem X->pneumococ i Y->pneumococ

**Llavors** Pneumococ és segur

R2 Si X->pneumococ i no\_dem Y->pneumococ

**Llavors** Pneumococ és segur

R3 Si X->pneumococ i Y->pneumococ

**Llavors** Pneumococ és segur

**final**



Exemple 4:

Mòdul Clínica : Pneum = Combinació\_i(Antecedents\_Patològics, Exp\_físiques) |

Suposem que els mòduls *Antecedents\_Patològics* i *Exp\_físiques* han estat compilats i que tenim una  $\mu$  que conté les referències a ambdós mòduls.

DECLMOD[[ Mòdul vinclemod ]] taulaespec<sub>0</sub>  $\sigma'$   $\mu$   $\pi'$   $\omega_0$   
 VINCLEMOD[[ Clínica : Pneum = *expremod* ]] taulaespec<sub>0</sub>  $\sigma'$   $\mu$   $\pi'$   $\omega_0$   
 EXPREMOD[[ Combinació\_i(Antecedents\_Patològics, Exp\_físiques) ]]  
 taulaespec<sub>0</sub>  $\sigma'$   $\mu$   $\pi'$   $\omega_0$

$\pi'$ (Combinació\_i) =  
 ((X Y), Pneum,  
 ([X  $\rightarrow$  Module-1, Y  $\rightarrow$  Module-2],  $\phi_0$ ,  $\delta_0$ ,  $\epsilon_0$ ,  $\rho_0$ ,  $\chi_0$ ),  
*expremod\_anterior*,  
 [Pneum  $\rightarrow$  ( $\tau_0$ ,  $\phi_0$ ,  $\delta_0$ , [Pneumococ  $\rightarrow$  (fet-5 difús)] ,  $\pi_0$ ,  $\chi_0$ )],  
 $\mu_0$ ,  $\pi_0$ ,  $\omega_0$ )

EXPREMOD[[ Antecedents\_Patològics ]] taulaespec<sub>0</sub>  $\sigma'$   $\mu_0$   $\pi'$   $\omega_0$   
 $\mu$ (Antecedents\_Patològics) = taulamod<sub>1</sub> =  
 (Mòdul-4 ( $\tau_0$ ,  $\phi_0$ ,  $\delta_0$ , [Pneumococ  $\rightarrow$  (fet-7 difús)],  $\rho_0$ ,  $\chi_0$ ))  
 $\rightarrow$  (taulamod<sub>1</sub>  $\emptyset$ )

EXPREMOD[[ Exp\_físiques ]] taulaespec<sub>0</sub>  $\sigma'$   $\mu_0$   $\pi'$   $\omega_0$   
 $\mu$ (Exp\_físiques) = taulamod<sub>2</sub> =  
 (Mòdul-5 ([BA  $\rightarrow$  Mòdul-Bacter],  $\phi_0$ , [herpes\_labial  $\rightarrow$   
 (fet-12, difús), esput  $\rightarrow$  (fet-13, [rovellat]),  
 períodes\_esgarrifances  $\rightarrow$  (fet-14, numèric), febre  $\rightarrow$  (fet-  
 15, numèric), artritis  $\rightarrow$  (fet-16, difús),  
 articulacions\_afectades  $\rightarrow$  (fet-17, [])], [Pneumococ  $\rightarrow$   
 (fet-8 difús)], [R1  $\rightarrow$  Regla-2, R2  $\rightarrow$  Regla-3, R3  $\rightarrow$   
 Regla-4, R4  $\rightarrow$  Regla-5, R5  $\rightarrow$  Regla-6],  $\chi_0$ ))  
 { Aquests tipus són els inferits a partir de les regles }  
 $\rightarrow$  (taulamod<sub>2</sub>  $\emptyset$ )

taulamod' = acarar((Module-4 ([X  $\rightarrow$  Mòdul-4,  
 Y  $\rightarrow$  Mòdul-5],  $\phi_0$ ,  $\delta_0$ ,  $\epsilon_0$ ,  $\rho_0$ ,  
 $\chi_0$ )),

$([X \rightarrow \text{Mòdul-1}, Y \rightarrow \text{Mòdul-2}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0) =$   
 { en aquest acarament Mòdul-4 i Mòdul-5 són restringits a  
 seguir l'especificació Pneum, es a dir a esportar només  
 Pneumococ. Els mòduls resultat de la restricció es diran  
 Mòdul-8 i Mòdul-9. Aquests mòduls tindran les següents  
 taulaespec:

$(\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-7 difús})], \rho_0, \chi_0)$

$(\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-8 difús})], \rho_0, \chi_0)$

}

(Mòdul-6  $([X \rightarrow \text{Mòdul-8}, Y \rightarrow \text{Mòdul-9}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0)$ )

especparam =  $([X \rightarrow \text{Mòdul-8}, Y \rightarrow \text{Mòdul-9}],$

$\phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0)$

$\mu'' = [X \rightarrow \text{Mòdul-8}, Y \rightarrow \text{Mòdul-9}]\mu_0$

EXPREMOD[[ *expremod* : Pneum ]] especparam  $\sigma' \mu'' \pi' \omega_0$

EXPREMOD[[ *inici decl final* ]] especparam  $\sigma' \mu'' \pi' \omega_0$

DECL[[ *cos* ]] taulaespec<sub>0</sub>  $\sigma' \mu'' \pi' \omega_0$

COS[[ *precondició regles metaregles* ]] taulaespec<sub>0</sub>  $\mu''$

REGLES[[ **Regles**

R1 **Si no dem** X->pneumococ i Y->pneumococ

**Llavors** Pneumococ és segur

R2 **Si** X->pneumococ i **no dem** Y->pneumococ

**Llavors** Pneumococ és segur

R3 **Si** X->pneumococ i Y->pneumococ

**Llavors** Pneumococ és segur ]] taulaespec<sub>0</sub>  $\mu''$

taulaespec' =  $(\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-15},$

difús)], [R1  $\rightarrow$  Regla-16, R2  $\rightarrow$  Regla-17,

R3  $\rightarrow$  Regla-17],  $\chi_0)$

$\rightarrow$  (taulaespec' (R1 R2 R3) ((regles) ()))

META[[  $\lambda$  ]] taulaespec'  $\mu''$

$\rightarrow$  (taulaespec'  $\emptyset$ )

PRECON[[  $\lambda$  ]] taulaespec'  $\mu''$

$\rightarrow$  (taulaespec'  $\emptyset$ )

$\rightarrow$  (taulaespec' ((regles) ()))

$\rightarrow$  (taulaespec' ((regles) ())  $\sigma' \mu'' \pi' \omega_0$ )

taulamod' = (Mòdul-18 taulaespec')

codi = traduir( ((regles) ()) taulaespec'  $\cup$  especparam) =

((R1 **Si no dem** fet-7 i Yfet-8

**Llavors** fet-15 és segur

R2 **Si** fet-7 i **no dem** fet-8

**Llavors** fet-15 és segur

R3 **Si** fet-7 i fet-8 **Llavors** fet-15 és segur) ())

$\rightarrow$  (taulamod' codi)

$$\begin{aligned}
 \text{taulamod} &= \text{acarar}(\text{taulamod}' \sigma'(\text{Pneum})) = \\
 &= (\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-15, difús}), \rho_0, \chi_0]) \\
 &\rightarrow (\text{taulamod codi}) \\
 &\rightarrow (\text{taulamod codi}) \\
 \text{taulaespec''} &= \sigma'(\text{Pneum}) \\
 \text{taulamodfinal} &= \text{acarar}(\text{taulamod taulaespec''}) = \text{taulamod} \\
 \text{taulaespec}' &= \text{afegirsubmòdul}((\text{Clínica taulamod}) \text{taulaespec}_0) = \\
 &([\text{Clínica} \rightarrow \text{Mòdul-23}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0) \\
 \mu' &= [\text{Clínica} \rightarrow \text{taulamod}] \mu \\
 &\rightarrow (\text{taulaespec}' \text{codi } \mu' \pi') \\
 \rightarrow (\text{taulaespec}', \text{codi}, \sigma', \mu', \pi', \omega_0) &=
 \end{aligned}$$

$$\begin{aligned}
 &( \\
 &([\text{Clínica} \rightarrow \text{Mòdul-23}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0), \\
 &((\text{R1 Si no\_dem fet-7 i Yfet-8 Llavors fet-15 és segur} \\
 &\quad \text{R2 Si fet-7 i no\_dem fet-8 Llavors fet-15 és segur} \\
 &\quad \text{R3 Si fet-7 i fet-8 Llavors fet-15 és segur}) ()), \\
 &[\text{Pneum} \rightarrow \text{taulaespecpneum}], \\
 &[\text{Clínica} \rightarrow \text{taulamod} \\
 &\quad \text{Antecedents\_Patològics} \rightarrow \text{taulaant} \\
 &\quad \text{Exp\_físiques} \rightarrow \text{taulaexp}], \\
 &[\text{Combinació\_i} \rightarrow ((X Y), \text{Pneum}, \\
 &\quad ([X \rightarrow \text{Module-1}, Y \rightarrow \text{Module-2}], \phi_0, \delta_0, \epsilon_0, \rho_0, \chi_0), \\
 &\quad \text{expremod}, \\
 &\quad [\text{Pneum} \rightarrow (\tau_0, \phi_0, \delta_0, [\text{Pneumococ} \rightarrow (\text{fet-5 difús})], \pi_0, \chi_0)], \\
 &\quad \mu_0, \pi_0, \omega_0)], \\
 &\omega_0 \\
 &)
 \end{aligned}$$



## Apèndix V: Exemple de modularització

Presentem aquí l'exemple complet utilitzat al capítol quatre per l'explicació del mecanisme de modularització. Aquest exemple correspon a la modularització del diagnòstic de pneumococ de l'aplicació PNEUMON-IA. A la figura V.1 es pot veure l'estructura de mòduls de l'exemple. Aquesta estructura es transportable, en essència a la resta de diagnòstics de l'aplicació. Això implica que en el desenvolupament ulterior de tota l'aplicació utilitzant aquest mecanisme produirà nivells d'abstracció més importants que els presents. En tot cas, de cara a l'exemplificació de la metodologia, creiem que l'exemple és prou ric com per demostrar la seva potència.

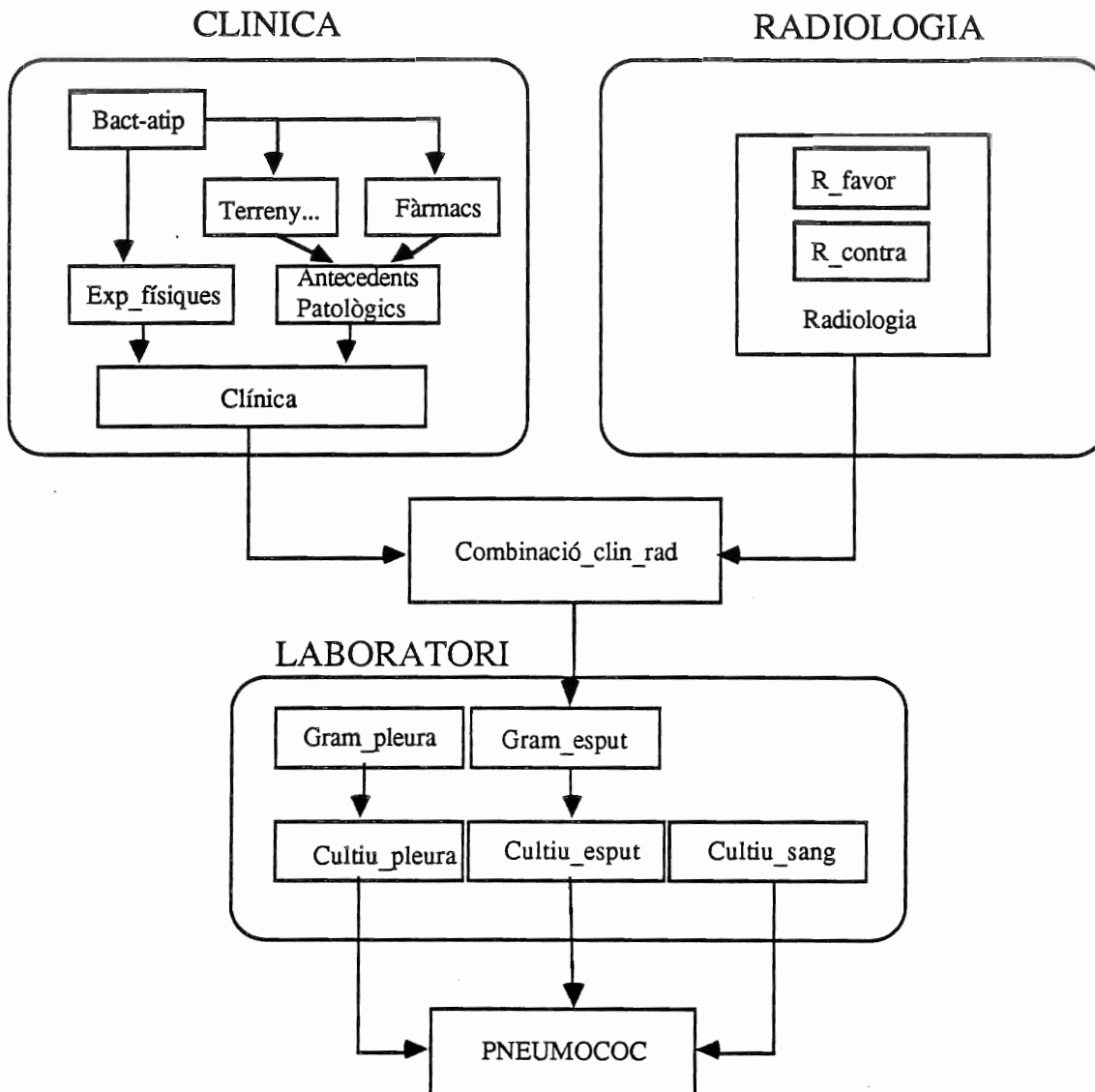


Fig V.1 Exemple de modularització del diagnòstic de Pneumococ a PNEUMON-IA.

## V.1. Declaració de CONJUNTS

**Conjunt Dades\_generals** = (mes: numèric)

**Conjunt Semiologia** = (esput: [rovellat, mucós, purulent, pútrid, hemoptoic]; febre: numèric; artritis: Booleà; articulacions\_afectades: numèric)

**Conjunt Simptomatologia** = (herpes\_labial: Booleà; períodes\_esgarrifances: numèric)

**Conjunt Antecedents** = (grip: Booleà; aspiració: Booleà; antecedents\_interés: Boolea; antibiòtics\_previs: Booleà; peni: Booleà; amoxi: Booleà; cefalosporines: Booleà; eritro: Booleà; evolució: Booleà; dosis\_adients: Booleà)

**Conjunt Dades clíniques** = Semiologia  $\cup$  Simptomatologia  $\cup$  Antecedents

**Conjunt Radiologia** = (distribució:[Lobar, Broncopneumònica]; embassament: Booleà; lòbuls: Numèric; cavitada: Difús)

**Conjunt** Analítiques\_generals = (exsudat\_pleural: Booleà)

**Conjunt** Tincions = ( gèrmen\_esput: [CGPP, CGPC, BGN, CBGN, flora\_mixta]; gram: Booleà; nombre\_PMM\_esput\_x100a: Numèric; nombre\_celepi\_esput\_x100a: Numèric; gèrmen\_pleura: [CGPP, CGPC, BGN, CBGN, flora\_mixta])

**Conjunt** Cultius = (gèrmen\_cultiu\_esput: [ Pneumococ, Klepsiella, Estafilococ, Hemòfilus ]; creixement\_cultiu\_esput: Booleà; gèrmen\_cultiu\_pleura: [ Pneumococ, Klepsiella, Estafilococ, Hemòfilus ]; gèrmen\_cultiu\_sang: [ Pneumococ, Klepsiella, Estafilococ, Hemòfilus ]; cultiu\_pleura+: Booleà; cultiu\_sang+: Booleà)

**Conjunt** Microbiologia = Tincions  $\cup$  Cultius

**Conjunt** Laboratori = Analítiques\_generals  $\cup$  Microbiologia

**Conjunt** Exploracions\_complementàries = Radiologia  $\cup$  Laboratori

**Conjunt** Tot = Dades\_generals  $\cup$  Dades clíniques  $\cup$  Exploracions\_complementàries

⋮

## V.2. Especificacions

{Les especificacions que siguin pròpies d'un únic mòdul es presentaran davant de la declaració del mòdul }

**Especificació** Bact-atip =

**Inici**

**Importa** = Tot

**Exporta** = (Bacteriana: Difús; Atípica: Difús)

**Final**

**Especificació** Pneum =

**Inici**

**Exporta** = (Pneumococ: Difús)

**Final**

**Especificació** Pneum\_ant =

**Inici**

**Importa** = Dades\_generals  $\cup$  Antecedents

**Exporta** = (Pneumococ: Difús)

**Final**

**Especificació** Pneum\_lab =

**Inici**

**Importa** = Analítiques\_generals  $\cup$  Laboratori

**Exporta** = (Pneumococ: Difús)

**Final**

## V.3. Mòduls auxiliars

**Mòdul** Combinació\_i (X: Pneum; Y: Pneum): Pneum =

**Inici**

**Regles**

R1 Si no\_dem X->pneumococ i Y->pneumococ  
Llavors Pneumococ és segur

R2 Si X->pneumococ i no\_dem Y->pneumococ  
 Llavors Pneumococ és segur  
 R3 Si X->pneumococ i Y->pneumococ  
 Llavors Pneumococ és segur

**final**

**Mòdul Combinació\_o (X: Pneum; Y: Pneum): Pneum =**

**Inici**

**Regles**

R1 Si X->pneumococ  
 Llavors Pneumococ és segur  
 R2 Si Y->pneumococ  
 Llavors Pneumococ és segur

**final**

**Mòdul Prevaldre(X:Pneum; Y:Pneum): Pneum**

**Inici**

**Regles**

R1 Si X->Pneumococ Llavors Pneumococ és segur  
 R2 Si no\_dem X->Pneumococ i Y->Pneumococ  
 Llavors Pneumococ és segur

**final**

## V.4. Mòduls referents a la CLINICA

**Mòdul Bacterianicitat: Bact-atip =**

**Inici**

**Regles**

{ Aquest Mòdul resta buit en pro de la millor llegibilitat de l'exemple. La decisió sobre la bacterianicitat o atipicitat d'una pneumònia ocuparia aproximadament el mateix tamny que el present exemple. }

**final**

**Especificació Exploració =**

**Inici**

**Importa =** Simptomatologia  
**Exporta =** (Pneumococ: Difús)

**final**

**Mòdul Exp-fisiques : Exploració =**

**Inici**

**Mòdul BA = Bacterianicitat**

**Regles**

R1 Si herpes\_labial i BA->Bacteriana  
 Llavors Pneumococ és molt\_possible  
 R2 Si monoartritis\_infecciosa i BA->Bacteriana > possible  
 Llavors Pneumococ és possible  
 R3 Si esput = rovellat i BA->Bacteriana > moderadament\_possible  
 Llavors Pneumococ és molt\_possible  
 R4 Si periodes\_esgarrifances = 1 i febre > 38.5 i BA->Bacteriana  
 Llavors Pneumococ és possible  
 R5 Si artritis i cardinal(articulacions\_affected)  
 Llavors monoartritis\_infecciosa és lleugerament\_possible

**final**

**Mòdul Terreny\_malalt (Bact : Bact-atip) : Pneum\_ant =**

**Inici**

**Mòdul BA = Bact**

**Regles**



R1 Si BA->Bacteriana i mes < 4  
     Llavors Pneumococ és molt\_possible  
 R2 Si BA->Bacteriana i mes > 4  
     Llavors Pneumococ és força\_possible  
 R3 Si BA->Bacteriana i grip Llavors Pneumococ és possible  
 R4 Si aspiració Llavors Pneumococ és possible  
 R5 Si BA->Bacteriana i no antecedents\_interés  
     Llavors Pneumococ és molt\_possible  
 R6 Si BA->Bacteriana i antecedents\_interés i no grip  
     Llavors Pneumococ és possible

**final**

Mòdul Fàrmacs (Bact : Bact-atip) : Pneum\_ant =

**Inici**

Mòdul BA = Bact

**Importa:** Antecedents

**Regles**

R1 Si BA->Bacteriana i (peni o amoxi o cefalosporines o eritro) i no evolució i dosis\_adients  
     Llavors Pneumococ és Lleugerament\_possible

**final**

Mòdul Antecedents\_Patològics : Pneum = Combinació\_i(Terreny\_malalt(Bacterianicitat),  
     Fàrmacs(Bacterianicitat))

Mòdul Clinica : Pneum = Combinació\_i(Antecedents\_Patològics, Exp\_fisiques)

## V.5. Mòdul referent a la RADIOLOGIA

**Especificació Radiol =**

**Inici**

Mòdul Radiol\_favor :

**Inici**

**Importa** = Radiologia

**Exporta** = (Pneumococ\_favor : Difús)

**final**

Mòdul Radiol\_contra :

**Inici**

**Importa** = Radiologia

**Exporta** = (Pneumococ\_contra : Difús)

**final**

**Exporta** = (Pneumococ : Difús)

**final**

Mòdul Radiologia: Radiol =

**Inici**

Mòdul Radiol\_favor =

**Inici**

**Regles**

R1 Si distribució = lobar

    Llavors Pneumococ\_favor és moderadament\_possible

R2 Si distribució = broncopneumònica

    Llavors Pneumococ\_favor és moderadament\_possible

R3 Si embassament

    Llavors Pneumococ\_favor és moderadament\_possible

R4 Si distribució = lobar i lòbuls = 1

```

                Llavors Pneumococ_favor és possible
final
Mòdul Radiol_contra =
Inici
    Regles
        R1 Si distribució = lobar i lòbuls > 1
            Llavors Pneumococ_contra és força_possible
        R2 Si cavitada
            Llavors Pneumococ_contra és força_possible
        R3 Si distribució = segmentària
            Llavors Pneumococ_contra és força_possible
final
Regles
    R1 Si no radiol_contra->Pneumococ_contra i
        radiol_favor->Pneumococ_favor
        Llavors Pneumococ és segur
final

Especificació C_C_R =
Inici
    Mòdul CL : Pneum
    Mòdul RX : Radiol
    Exporta : (Pneumococ : Difús)
final

Mòdul Combinació_clin_rad : C_C_R =
Inici
    Mòdul CL = Clinica
    Mòdul RX = Radiologia
    Precondició: CL->Pneumococ
    Regles
        R1 Si CL->Pneumococ Llavors Pneumococ és segur
        R2 Si RX->Pneumococ Llavors Pneumococ és segur
final

```

## V.6. Mòduls referents a LABORATORI

```

Especificació Gram =
Inici
    Mòdul AN : Pneum
    Importa = Antecedents ∪ Laboratori
    Exporta = (Pneumococ : Difús; Possibilitat_cultiu : Difús)
final

Mòdul Gram_esput =
Inici
    Mòdul AN = Antecedents_Patològics
    Precondició: esput ≠ mucós i no Antibiotics_previs
    Regles
        R1 Si AN->Pneumococ i gram i no_dem gèrmen_esput
            Llavors Pneumococ és força_possible
        R2 Si AN->Pneumococ Llavors Pneumococ és segur
        R3 Si AN->Pneumococ i gèrmen_esput = CGPP i cardinal(gèrmen_esput) = 1
            Llavors Pneumococ és força_possible
        R4 Si AN->Pneumococ i gèrmen_esput ≠ CGPP
            Llavors Pneumococ és possible
        R5 Si AN->Pneumococ i gèrmen_esput ≠ flora_mixta
            Llavors Pneumococ és segur
        R6 Si nombre_PMM_esput_x100a > 25 i

```

nombre\_celepi\_esput x100a < 25  
 Llavors Possibilitat\_cultiu és segur

**final**

**Especificació Cultiu =**

**Inici**

Mòdul GE : Gram  
 Importa = Laboratori  
 Exporta = (Pneumococ : Difús)

**final**

**Mòdul Cultiu\_esput : Cultiu =**

**Inici**

Mòdul GE = Gram\_esput  
 Precondició: GE->Possibilitat\_cultiu i no Antibiotics\_previs

**Regles**

- R1 Si GE->AN->Pneumococ i gèrmen\_cultiu\_esput = pneumococ  
 Llavors Pneumococ és molt\_possible
- R2 Si GE->AN->Pneumococ i gemen\_cultiu ≠ pneumococ i  
 cardinal(gèrmen\_cultiu\_esput) = 1  
 Llavors Pneumococ és lleugerament\_possible
- R3 Si GE->AN->Pneumococ i gèrmen\_cultiu\_esput i  
 cardinal(gèrmen\_cultiu\_esput) = 2  
 Llavors Pneumococ és força\_possible
- R4 Si GE->AN->Pneumococ i no creixement  
 Llavors Pneumococ és moderadament\_possible

**final**

**Mòdul Gram\_pleura : Pneum\_lab =**

**Inici**

Precondició: exsudat\_pleural

**Regles**

- R1 Si gèrmen\_pleura = CGPP o CGPC Llavors Pneumococ és segur
- R2 Si gèrmen\_pleura = CGP Llavors Pneumococ és molt\_possible
- R3 Si no gèrmen\_pleura = CGP  
 Llavors Pneumococ és gens\_possible

**final**

**Mòdul Cultiu\_pleura : Pneum\_lab =**

**Inici**

Precondició: exsudat\_pleural

**Regles**

- R1 Si gèrmen\_cultiu\_pleura = pneumococ  
 Llavors Pneumococ és segur
- R2 Si gèrmen\_cultiu\_pleura ≠ pneumococ  
 Llavors Pneumococ és gens\_possible

**final**

**Mòdul Cultiu\_sang : Pneum\_lab =**

**Inici**

**Regles**

- R1 Si gèrmen\_cultiu\_sang = pneumococ  
 Llavors Pneumococ és segur
- R2 Si gèrmen\_cultiu\_sang ≠ pneumococ  
 Llavors Pneumococ és gens\_possible

**final**

## V.7. Mòdul referent al diagnòstic de PNEUMOCOC

```
Mòdul Pneumococ : Pneum =  
  prevaldre( Combinació_o( Cultiu_sang,  
    prevaldre(Cultiu_pleura, Gram_pleura)),  
  prevaldre(Cultiu_esput, prevaldre(Gram_esput, Clínica)))
```

# Apèndix VI. Exemple d'interacció. Sortides.

En aquest apèndix és pretén exemplificar una interacció de l'usuari amb el sistema. Per mostrar-la s'ha utilitzat l'aplicació PNEUMON-IA amb un cas habitual en les consultes als Serveis d'Urgències o als metges de capçalera. El primer apartat mostra una reproducció de la interacció amb el sistema, i el segon certes sortides que poden ser d'utilitat a l'expert que desenvolupa una aplicació.

## VI.1. Interacció amb el sistema

En aquesta interacció es mostra una traça de les metaregles que el sistema utilitza. També es mostren diverses utilitats del sistema com són les d'explicar com s'ha produït un resultat, mostrar regles i mostrar en quin punt està el sistema en el moment en que es fa la pregunta.

Quan es desconeix la resposta a una pregunta es respon amb un <return> que a la transcripció apareix en blanc.

---

Quina opció desitges?

- 1      Compileu
- 2      Carregar
- 3      Executar
- 4      Suspendre
- 5      Explicar
- 6      Activar la sortida
- 7      Acabar la sortida
- 8      Menú d'utilitats
- 9      Acabar

MLRD> 3

.....

- Si en el decurs de la consulta no sap quines són les respostes adients, escriu 'valors' (o 'v') i obtindrà la llista de respostes possibles.

-Per accedir al menú d'utilitats i explicacions escriu una de les següents expressions: 'per què?' o '?'

-Si vol acabar abans d'arribar a la fi del cas escriu: 'acabar'

.....  
 Quin model vol fer servir (triï'n un):

Conversacional  
 No conversacional

MLRD > C

Quin identificador de fitxa té aquest malalt? Joan H

En quin hospital ha estat atès el malalt? L'Aliança

Quin metge ha proporcionat el cas? A. Verdaguer

Vol donar per avançat els resultats diagnòstics? No

.....  
*La pneumònia ha estat adquirida en mitjà extrahospitalari?*<sup>1</sup>

MLRD (booleà) > Sí

*Quin és l'estat del malalt atribuïble a la seva pneumònia?*

NOM	ABREVIACIONS
-----	--------------

Lleu	L
Moderadament greu	MOD
Greu	G
Molt greu	MOL

MLRD (enumerat) > L<sup>2</sup>

*Quants dies fa que s'inicià la clínica de pneumònia?*

MLRD (numèric) > 10<sup>3</sup>

*Dies que fa que s'ha diagnosticat la pneumònia?*

MLRD (numèric) > 1<sup>4</sup>

*Dia d'inici de la pneumònia*

MLRD (numèric) > 10

*Mes d'inici de la pneumònia:*

MLRD (numèric) > 10

*Any:*

MLRD (numèric) > 88

*Edat:*

MLRD (numèric) > 19

*Quin és el sexe del malalt?*

---

<sup>1</sup> Si es contesta negativament a aquesta qüestió el sistema acaba la interacció ja que està dirigit a les pneumònies extrahospitalàries

<sup>2</sup> Aquesta valoració és subjectiva

<sup>3</sup> Amb aquesta qüestió es valora el moment evolutiu de la malaltia

<sup>4</sup> PNEUMON-IA estableix quins exàmens complementaris poden estar disponibles en el moment de la interacció amb el sistema

<i>NOM</i>	<i>ABREVIACIO</i>
------------	-------------------

<i>Home</i>	<i>H</i>
-------------	----------

<i>Dona</i>	<i>D</i>
-------------	----------

MLRD (enumerat)> H

*Està el malalt en coma?*

MLRD (booleà)> No

*El pacient està desorientat?*

MLRD (booleà)> No

*Creu que el malalt té antecedents personals o patològics a destacar?*

MLRD (booleà)> No

.....  
Hem aplicat endavant les metaregles: R25888

La nova estratègia és:

ADQUISICIO-DADES

BACT-ATIP

PNEUM

MYCO

LEGION

VIRUSES

VIRUSES-BACT

CLAM

FQ

TBC

STR-PIO

NOCARDIA

.....  
*El malalt havia estat tractat amb antibiòtics?*

MLRD (booleà)> No

*Indiqui quin va ser l'inici de la pneumònia*

<i>NOM</i>	<i>ABREVIACIO</i>
------------	-------------------

<i>Brusca</i>	<i>BR</i>
---------------	-----------

<i>Subaguda</i>	<i>SU</i>
-----------------	-----------

MLRD (enumerat)> SU

*Estossega el malalt?*

MLRD (booleà)> Sí

*El malalt expectora?*

MLRD (booleà)> Sí

*De quin tipus és l'esput?*

<i>NOM</i>	<i>ABREVIACIONS</i>
------------	---------------------

<i>Mucós</i>	<i>MU</i>
--------------	-----------

<i>Purulent</i>	<i>PUR</i>
-----------------	------------

<i>Rovellós</i>	<i>HER</i>
-----------------	------------

<i>Pútrid</i>	<i>PU</i>
<i>Hemoptoic</i>	<i>HE</i>

MLRD (enumerat)> MU

*El malalt té dolor de tipus pleurític?*  
MLRD (booleà)> Sí

*Temperatura axilar:*  
MLRD (numèric)> 37.5

*Freqüència cardíaca:*  
MLRD (numèric)> 60

*Indiqui els signes de consolidació pulmonar que es trobin agrupats en una mateixa àrea anatòmica*

<i>SIGNES</i>	<i>ABREVIACIONS</i>
<i>Buf tubàric</i>	<i>BUFT</i>
<i>Estertors crepitants</i>	<i>CR</i>
<i>Augment de la transmissió de les vibracions</i>	<i>VV</i>
<i>Matidesa a la percussió</i>	<i>MAT</i>

MLRD (enumerat)>

*A l'exploració física hi ha semiologia de condensació?*  
MLRD (booleà)> No<sup>1</sup>

*Coneix altres dades clíniques -d'història o d'exploració-, a més de les pròpies de la pneumònia, que vulgui fer constar?*  
MLRD (booleà)> Sí

*Indiqui altres òrgans o sistemes afectats que vulgui fer constar:*

<i>NOM</i>	<i>ABREVIACIONS</i>
------------	---------------------

<i>Aparell cardiocirculatori</i>	<i>COR</i>
<i>Aparell digestiu</i>	<i>GI</i>
<i>Aparell genitourinari</i>	<i>GU</i>
<i>Aparell musculoesquelètic</i>	<i>REUM</i>
<i>Aparell respiratori / ORL</i>	<i>RES</i>
<i>Pell</i>	<i>DER</i>
<i>Sang / Limfàtics</i>	<i>HEMLIMF</i>
<i>Sistema nerviós</i>	<i>NEU</i>

MLRD (enumerat)> REUM GI RES

*Indiqui les dades clíniques gastrointestinals:*

<i>NOM</i>	<i>ABREVIACIONS</i>
<i>Diarrea</i>	<i>DIAR</i>
<i>Pancreatitis</i>	<i>PANCR</i>
<i>Citòlisi</i>	<i>GPT</i>
<i>Distensió abdominal</i>	<i>DIST</i>
<i>Peritonitis</i>	<i>ERIT</i>

<sup>1</sup> Es fa aquesta qüestió atès que a l'anterior pregunta s'ha respost amb *desconegut*



*Ulceres bucals* U-B  
*Hemorragia digestiva* HD  
 MLRD (enumerat)> DIAR DIST

Indiqui les dades clíniques musculoesquelètiques:

NOM	ABREVIACIONS
<i>Artràlgies</i>	ARTRALGIA
<i>Artritis</i>	ARTRITIS
<i>Miàlgies</i>	MIALG
<i>Osteïtis costal</i>	OC

MLRD (enumerat)> ARTRALGIA MIALG

.....  
 Hem aplicat endavant les metaregles: R25896D

La nova estratègia és:

BACT-ATIP  
 MYCO  
 VIRUSES  
 VIRUSES-BACT  
 CLAM  
**Test final**  
 ADQUISICIO-DADES  
 PNEUM  
 LEGION  
 FQ  
 TBC  
 STR-PIO  
 NOCARDIA

.....  
 Indiqui altres dades clíniques respiratòries o ORL

NOM	ABREVIACIONS
<i>Bronquièctasi</i>	BRONQ
<i>Amigdalitis</i>	AMIG
<i>Simptomatologia de vies altes</i>	CATARRO
<i>Otitis</i>	OTITIS
<i>Sinusitis/lesions nasals</i>	SIN

MLRD (booleà)> CATARRO

S'han practicat radiografies de tòrax?

MLRD (booleà)> Sí

Triï les dades radiològiques que observa a la placa de tòrax:

DADES	ABREVIACIONS
<i>Broncograma aeri</i>	BRAERI
<i>Imatges confluents</i>	CONFLUENT
<i>Condensació amb vores mal definides</i>	VMALDEF
<i>Patró intersticial</i>	INTER
<i>Embassament pleural</i>	EMBASS
<i>Cavitació(ns)</i>	CAV
<i>Lesions tuberculoses antigues</i>	LATBC
<i>Pneumotòrax</i>	NTX

Signes d'atelectasi  
 Atelectasis laminars  
 MLRD (enumerat)> INTER  
 ATEL  
 LAM

Quin grau de certesa té que l'infiltrat pulmonar sigui de tipus intersticial  
 MLRD (difús)> FORÇA POSSIBLE<sup>1</sup>

Indiqui, mitjançant l'abreviació corresponent, quin tipus de distribució predominant te(nen) el(s) infiltrat(s)

TIPUS DISTRIBUCIO	ABREVIACIONS
Lobar (un o més d'un)	LOBAR
Segmentari	SEG
Broncopneumònic o multisegmentari	BRNEUM
Difús	DF
Un infiltrat nodular-o arrodonit-	L-NOD
Diversos infiltrats nodulars	D-NOD

MLRD (enumerat)> SEG

Quin és el grau de certesa que té vostè que l'infiltrat sigui segmentari?  
 MLRD (difús)> FORÇA POSSIBLE

S'aprecia a la radiologia l'afectació de diferents segments?  
 MLRD (booleà)> No

A quins segments està localitzada la o les condensacions ?

NOM	ABREVIACIONS
Apical-lòbul superior dret	AD
Anterior-lòbul superior dret	AND
Posterior-lòbul superior dret	PD
Intern-Lòbul mig	IMI
Extern-Lòbul mig	EMI
Apical-Lòbul inferior dret	AID
Piràmide-basal-lòbul inferior dret	BD
Apico-posterior-lòbul superior es	ANE
Superior-lingular	SL
Inferior-lingular	IL
Apical-lòbul inferior esquerre	AIE
Piràmide-basal-lòbul inferior esquerre	BE

MLRD (enumerat)> IMI

S'han fet anàlisis de laboratori?  
 MLRD (booleà)> Sí

Nombre de leucòcits:  
 MLRD (numèric)> 9000

Quin és el percentatge de polinuclears?  
 MLRD (numèric)> 70

Percentatge de bandes o formes joves:  
 MLRD (numèric)> 0

<sup>1</sup>Amb aquesta qüestió es matisa la certesa que té l'usuari en l'interpretació de la radiografia de tòrax

.....

Els resultats del mòdul ADQUISICIO-DADES Són els següents:

<u>MODUL</u>	<u>DIAGNOSTIC</u>	<u>VALOR</u>
ADQUISICIO-DADES	DADES-MALALT	SEGUR

.....

Hem aplicat endavant les metaregles: R25896D

La nova estratègia és:

BACT-ATIP  
 MYCO  
 VIRUSES  
 VIRUSES-BACT  
 CLAM  
 PNEUM  
 LEGION  
 STR-PIO  
 TBC  
**Test final**  
 ADQUISICIO-DADES  
 FQ  
 NOCARDIA  
**Test final**  
 ENTEROBACT  
 ANAEROB  
 HEMOPH  
 STAPH  
 MENINGO  
 BRANHA

.....

*Al malalt li fa mal el cap?*

MLRD (booleà)> ?

Pot fer algunes de les següents preguntes:

?	Dóna el perquè del fet que es demana de la conclusió de la regla en curs
Enrera	Baixa en l'encadenament actual de les regles
Pila	Ensenya la pila d'encadenament de regles des del punt que estem fins el final
Actual	Mostra el punt en que ens trobem
Explicar	Ens dóna explicacions del procés de raonament
Trassa	Ens permet establir traces
<return>	Deixa l'ajut de PERQUE

MLRD > A

El fet Mal de cap és necessari per acabar de validar la regla R03023.

Les premisses validades fins ara són:

\* El valor de: Estat del malalt no és ni Greu ni Molt greu

Resten per validar les següents premisses:

\* Mal de cap  
 \* Bradicàrdia relativa

Això ens permetrà deduir:

\* Concloure Atípica

La creença en la regla és: Força Possible

Premi un <return> per continuar.

*Al malalt li fa mal el cap?*

MLRD (booleà)> Sí

*Indiqui quina és, al seu parer, l'evolució de la pneumònia:*

NOM

ABREVIACIONS

Aguda (<15 dies)

AG

Crònica (>30 dies)

PNCR

MLRD (enumerat )> AG

.....  
Hem aplicat endavant les metaregles: R25883

La nova estratègia és:

MYCO

VIRUSES

VIRUSES-BACT

CLAM

**Test final**

FQ

TBC

NOCARDIA

**Test final**

ADQUISICIO-DADES

PNEUM

LEGION

STR-PIO

CRIPTOCOC

CITOMEGALOVIRUS

P-CAR

.....  
Els resultats del mòdul BACT-ATIP són els següents:

MODUL

DIAGNOSTIC

VALOR

BACT-ATIP

Pneumònia atípica

Molt possible

BACT-ATIP

Pneumònia bacteriana

Moderadament possible

.....  
Els mòduls que resten són:

MYCO

VIRUSES

VIRUSES-BACT

CLAM

FQ

TBC

NOCARDIA

ADQUISICIO-DADES

PNEUM

LEGION

STR-PIO

CRIPTOCOC

CITOMEGALOVIRUS

P-CAR

.....  
*S'ha realitzat el test de Coombs ?*

MLRD (booleà)> No

*S'han determinat les crioaglutinines ?*

MLRD (booleà)> No

.....

Els resultats del mòdul MYCO són els següents:

<u>MODUL</u>	<u>DIAGNOSTIC</u>	<u>VALOR</u>
MYCO	Pneumònia per micoplasma	Força possible

Els mòduls que resten són:

VIRUSES	VIRUSES-BACT	CLAM	FQ	TBC
NOCARDIA	ADQUISICIO-DADES	PNEUM	LEGION	STR-PIO
CRIOPTOCOC	CITOMEGALOVIRUS	P-CAR		

*Hi ha infiltrats radiològics perihilars ?*

MLRD (difús)> Molt possible

Els resultats del mòdul VIRUSES són els següents:

<u>MODUL</u>	<u>DIAGNOSTIC</u>	<u>VALOR</u>
VIRUSES	Pneumònia per virus	Molt possible

Els mòduls que resten són:

VIRUSES-BACT	CLAM	FQ	TBC	NOCARDIA
ADQUISICIO DADES		PNEUM	LEGION	STR-PIO
CRIOPTOCOC	CITOMEGALOVIRUS	P-CAR		

El mòdul VIRUSES-BACT no ha produït cap resultat

Els mòduls que resten són:

CLAM	FQ	TBC	NOCARDIA	ADQUISICIO DADES
PNEUM	LEGION	STR-PIO	CRIOPTOCOC	
CITOMEGALOVIRUS	P-CAR			

*El pacient conviu o tracta quotidianament amb ocells ?*

MLRD (booleà)> No

Els diagnòstics finals als que hem arribat, per ordre de creença són els següents:

DIAGNOSTIC	VALOR
Pneumònia per virus	Molt possible
Pneumònia per mycoplasma	Força possible
Pneumònia per clamidia	Lleugerament possible

Quina opció desitges?

- 1 Compilar
- 2 Carregar
- 3 Executar
- 4 Suspendre
- 5 Explicar
- 6 Activar la sortida
- 7 Acabar la sortida
- 8 Menú d'utilitats

9 Acabar

MLRD&gt; 5

Preguntes que podem respondre

COM	ex: com febre
REGLA	ex: regla regla
DIAGNOSTICS	ex: diagnostics mòdul
RESULTATS	ex: resultats mòdul
FET	ex: fet NE
MODUL	ex: mòdul bact-atip
LLISTAMODULS	Enumera els mòduls de la Base
LLISTAREGLES	Enumera les regles aplicades
OBRIR	Comença el Dribble
TANCAR	Acaba el Dribble
<return>	Per acabar

EXPL&gt; COM ATIP-D

El fet ATIP-D ha estat deduït de les següents regles:

R03014 R03026.

EXPL&gt; REGLES R03014 R03026

Regla R03026

Les premisses de la regla són:

- \*No Existència d'antecedents patològics
- \*La certesa de la Creença en el fet: patró intersticial és superior a Possible

Això ens permetrà deduir:

- \*Concloure Atípica diagnòstic

La creença en la regla és: Força possible

Regla R03014

Les premisses de la regla són:

- \*Tos
- \*Expectoració
- \*El valor d'esput és MU
- \*El valor d'Evolució no és PNCR

Això ens permetrà deduir:

- \*Concloure Atípica diagnòstic

La creença en la regla és: Possible

## VI.2 Sortides del sistema

Un cop el sistema ha finalitzat l'execució d'un cas es poden demanar certes sortides d'utilitat per a l'expert, tal i com s'ha comentat al capítol 6. Aquí presentem dos d'aquests fitxers de sortida, l'*informe d'execució* i l'*informe de validació*.

### VI.2.1 Informe d'execució

En aquest informe es dóna la següent informació:

*Identificació:* Dades referents a la identificació del cas. En el cas de l'aplicació PNEUMON-IA l'identificador del malalt, de l'hospital i del metge que el tracta.

*Fets observats:* Totes aquelles dades obtingudes pel sistema expert mitjançant interacció amb l'exterior.

*Fets deduïts (no diagnòstics):* Fets deduïts pel sistema que no són considerats solucions sinó subobjectius de les solucions o d'altres subobjectius.

*Diagnòstics del sistema:* Llista ordenada dels diagnòstics obtinguts pel sistema.

*Regles utilitzades:* Llista de totes les regles utilitzades en el procés de solució.

*Metaregles utilitzades:* Llista de totes les metaregles aplicades, sense distingir el seu tipus.

*Plans d'actuació seguits pel sistema:* Llista dels moments de canvi de pla d'actuació degut a l'aplicació d'alguna metaregla. És una representació abreujada de la que s'ha presentat com a plans d'actuació al capítol 5.

Els informes d'aquesta mena es generen amb l'extensió ".inf".

---

Sistema MILORD Base de Coneixements PNEUMON-IA  
 DATA: Dimecres 1-mar-1989 18:55:12

---

#### IDENTIFICACIO

Fitxa clínica: Joan H.  
 Doctor: A. Verdaguer

Hospital: L'Aliança

#### FETS OBSERVATS

Fet	Nom intern	valor
1) Pneumònia extrahospitalària	NE	Sí
2) Estat del malalt	ESTAT-MALALT	Lleu
3) Coma	COMA	No
4) Desorientació	DESORIENTAT	No
5) Nombre de dies des de l'inici	DIES-INICI	10
6) Nombre de dies des del diagnòstic	DIES	1
7) Dia de l'inici de la pneumònia	DIA	1
8) Mes de l'inici de la pneumònia	MES	10
9) Any	ANY	88
10) Edat	EDAT	19
11) Sexe	SEXE	Home
12) Antecedents patològics	AP-INTERES	No
13) Antibiótics previs	ABS	No
14) Forma d'instauració	INICI	Subagut

15) Tos	TOS	Sí
16) Expectoració	EXPECTORACIO	Sí
17) Esput	ESPUT	Mucós
18) Dolor pleurític	DOLOR-PLEURITIC	Sí
19) Temperatura	TEMPERATURA	37.5
20) Freqüència cardíaca	FRE-CARD	60
21) Signes de consolidació	SIGNES-CON	Desconegut
22) Semiologia de condensació	SEMIOL-CON	No
23) Dades clíniques associades	TE ASSOC	Sí
24) Tipus dades associades	ASSOCIADAS-GRA	Reumatològiques Gastrointestinals Respiratòries
25) Dades gastrointestinals	ASSOCIADAS-GI	Diarrea Distensió abdominal
26) Dades reumatològiques	ASSOCIADAS-REU	Artràlgia Miàlgia
27) Dades Respiratòries	ASSOCIADAS-RES	Catarro
28) Existència de radiologia	RX-FET	Sí
29) Dades Radiològiques	DADES-RX	Intersticial
30) Certesa Patró intersticial	RX-INTER-S	Força possible
31) Distribució de l'infiltrat	RX-DISTR	Segmentària
32) Certesa segmentària	SEG-S	Força possible
33) Plurisegmentària	RX-SEG-VAr	No
34) Localització de l'infiltrat	LOCALITZACIO-S	Inter-lòbul-mig
35) Analítica	ANALAB	Sí
36) Nombre de leucòcits	LEUCOCITS	9000
37) Polinuclears totals	POLIS	6300
38) Percentatge de polinuclears	PP	70
39) Polimorfonuclears en banda	BANDA	0
40) Evolució	EVOLUCIO	Aguda
41) Mal de cap	MAL-CAP	Sí
42) Convivència amb pneumònies	ALTREPN	No
43) Realitzat test de Coombs	FET-COOMBS	No
44) Determinades les crioaglutinines	FET-CRIO	No
45) Convivència amb ocells	OCELL	No
46) Perihilar	PERIH	Molt possible

## FETS DEDUITS (No Diagnòstics)

Fet	Nom intern	valor
1) Neutrofília	NEUTROFILIA	Gens possible
2) Desviació a l'esquerra	DESV-ESQ	Gens possible
3) Bacteriana	BACT	Moderadament possible
4) Atípica	ATIP	Molt possible

## DIAGNOSTICS SISTEMA

Diagnòstic	Valor
1) Pneumònia vírica	Molt possible
2) Pneumònia per micoplasma	Força possible
3) Pneumònia per clamídia	Lleugerament possible



REGLES UTILITZADES

R20001	R20002	R20005	R20018	R20020	R18012	R17009
R17010	R17011	R17013	R17016	R03009	R03014	R03024
R03026	R03028	R01001	R01007	R01008	R01009	R01009A
R01009B	R01009C	R01028	R01029	R01030	R01057	R01077
R01078	R01094	R01101.				

METAREGLES UTILITZADES

R25882A R25883 R25888 R25896D R25602

PLANS D'ACTUACIO SEGUITS PEL SISTEMA

- 1) Combinació: estratègia defecte i estratègia de R25888  
 Objectius: ADQUISICIO-DADES BACT ATIP PNEUM MYCO LEGION VIRUSES  
 VIRUSES-BACT CLAM FQ TBC STR-PIO NOCARDIA  
 Objectius eliminats:  
 Objectius visitats: ;  
 Conjunt suport: R25888  
 Condició suport: **no** ap-interès  
 Certesa suport: Possible
  
- 2) Combinació: 1 i estratègia de R25896D  
 Objectius: BACT ATIP MYCO VIRUSES VIRUSES-BACT CLAM test final  
 ADQUISICIO-DADES PNEUM LEGION FQ TBC STR-PIO NOCARDIA  
 Objectius eliminats:  
 Objectius visitats:  
 Conjunt suport: R25888 R25896D  
 Condició suport: **no** ap-interès i associades-reum = artralgia  
 Certesa suport: Possible
  
- 3) Combinació: 2 i estratègia de R25882A  
 Objectius: BACT ATIP MYCO VIRUSES VIRUSES-BACT CLAM PNEUM  
 LEGION STR-PIO TBC test final ADQUISICIO-DADES FQ NOCARDIA  
 test final ENTEROBACT ANAEROB HEMOPH STAPH MENONGO  
 BRANHA  
 Objectius eliminats:  
 Objectius visitats:  
 Conjunt suport: R25888 R25896D R25882A  
 Condició suport: **no** ap-interès i associades-reum = artralgia i bact  
 Certesa suport: Possible
  
- 4) Combinació: 2 i estratègia de R25883  
 Objectius: MYCO VIRUSES VIRUSES-BACT CLAM test final FQ TBC  
 NOCARDIA test final ADQUISICIO-DADES PNEUM LEGION STR-PIO  
 CRIPTOCOC CITOMEGALOVIRUS P-CAR  
 Objectius eliminats:  
 Objectius visitats: BACT ATIP  
 Conjunt suport: R25888 R25896D R25883  
 Condició suport: **no** ap-interès i associades-reum = artralgia i **no** (bact és possible) i  
 atip és possible  
 Certesa suport: Possible

## 8) Final d'execució

Objectius: **test final** FQ TBC NOCARDIA **test final** ADQUISICIO-DADES  
PNEUM LEGION STR-PIO CRIPTOCOC CITOMEGALOVIRUS P-CAR

Objectius eliminats:

Objectius visitats: BACT ATIP MYCO VIRUSES VIRUSES-BACT CLAM

Conjunt suport: R25888 R25896D R25883

Condicció suport: **no** ap-interès i associades-reum = artralgia i **no** (bact és possible) i  
atip és possible

Certesa suport: Possible

## VI.2.2 Informe de validació

Aquest informe és una versió resumida de l'anterior, on només apareixen les dades externes del malalt. El seu objectiu és el de servir com a dades pels experts que han de validar el sistema. En el cas de PNEUMON-IA s'han suprimit les dades referents a interpretació radiològica donat que es suministren diapositives de les radiografies als experts

Aquest informes es generen amb l'extensió ".exp".

.....  
Sistema MILORD Base de Coneixements PNEUMON-IA  
DATA: Dimecres 1-mar-1989 18:55:12  
.....

### IDENTIFICACIO

Fitxa clínica: Joan H.

### FETS OBSERVATS

Fet valor

1) Pneumònia extrahospitalària	Sí
2) Estat del malalt	Lleu
3) Coma	No
4) Desorientació	No
5) Nombre de dies des de l'inici	10
6) Nombre de dies des del diagnòstic	1
7) Dia de l'inici de la pneumònia	1
8) Mes de l'inici de la pneumònia	10
9) Any	88
10) Edat	19
11) Sexe	Home
12) Antecedents patològics	No
13) Antibiótics previs	No
14) Forma d'instauració	Subagut
15) Tos	Sí
16) Expectoració	Sí
17) Esput	Mucós
18) Dolor pleurític	Sí
19) Temperatura	37.5
20) Freqüència cardíaca	60
21) Signes de consolidació	Desconegut
22) Semiologia de condensació	No

23) Dades clíniques associades	Sí
24) Tipus dades associades	Reumatològiques Gastrointestinals Respiratòries
25) Dades gastrointestinals	Diarrea Distensió abdominal
26) Dades reumatològiques	Artràlgia Miàlgia
27) Dades Respiratòries	Catarro
28) Existència de radiografies	Sí
29) Analítica	Sí
30) Nombre de leucòcits	9000
31) Polinuclears totals	6300
32) Percentatge de polinuclears	70
33) Polimorfonuclears en banda	0
34) Evolució	Aguda
35) Mal de cap	Sí
36) Convivència amb pneumònies	No
37) Realitzat test de Coombs	No
39) Determinades les crioaglutinines	No
40) Convivència amb ocells	No



# Bibliografia

- Aguilar J., Piera N. (1987) : "Connectives of Fuzzy Logic in Multivariate Data Classification", a AAVV: *Proceedings of International Workshop on Approximate Reasoning and Artificial Intelligence*, Granada.
- Agustine K., Barnett G., Mesteller F., Youtz C. (1986) : "How medical professionals evaluate expressions of probability", *The New England Journal of Medicine*, Sept., pp. 740-744.
- Agustí-Cullell J., Sierra C. (1988) : "Adding Generic Modules to Flat Regla-Based Languages", *Report CEAB-CSIC*, 88-13.
- Aikins J. S. (1983) : "Prototypical Knowledge for Expert Systems", *Artificial Intelligence*, 29, pp. 163-230.
- Akkermans H., Wielinga B., Schreiber G., Balder J. (1989) : "Towards a Formal Specification of Knowledge Models", (per atenció dels autors).
- Alsina C., Trillas E., Valverde L. (1983) : "On some logical Connectives for Fuzzy Set Theory", *J. on Mathematical Analysis and Applications*, Vol 93, 1, pp. 15-26.
- Astrom, K., Anton, J. (1984) : "Expert Control", a AAVV: *Proceedings IFAC 9th World Congress*, VI, pp. 220-245, Budapest.
- Beetz M. (1987) : *Specifying Meta-Level Architectures for Rule-Based Systems*, Diploma Thesis presentada a la Universitat Kaiserslautern.
- Beyth-Marom R. (1982) : "How Probable is Probable? A Numerical Taxonomy Translation of Verbal Probability Expressions", *Journal of Forecasting*, Vol 1, pp. 257-269.
- Bonissone P. (1979) : *The problem of linguistic Approximation in System Analysis*, Ph. D. dissertation of EECS Dept. in U. C. Berkeley, University Microfilms International Publications #80-14, 618, Ann Arbor, Michigan.

- Bonissone P., Decker K., (1985) : "Selecting Uncertainty and Granularity: An Experiment in trading-off Precision and Complexity", *KBS Working Paper*, General Electric Corporate Research and Development Center, Schenectady, New York.
- Bonissone P., Tong R.M. (1986) : "Editorial: Reasoning with uncertainty in expert systems", *Int. J. Man-machine Studies*, 22, pp. 241-250.
- Bruck R., Temme K., Wronn H. (1988) : "Flair - A Knowledge Based Approach to Intelligent Circuit Floorplanning", a AAVV: *Proceedings of International Workshop on Artificial Intelligence for Industrial Applications*, Hitachi City, Japan.
- Burke G., Carrette G. (1982) : "NIL Notes for Release 0", *Report from MIT Laboratory for Computer Science*, Decembre.
- Bylander T., Chandrasekaran B. (1986) : "Generic Tasks for Knowledge Based Reasoning : The "RIGHT" level of Abstraction for Knowledge Acquisition", *Technical Report*, November, Laboratory for Artificial Intelligence Research of The Ohio State University.
- Bylander T., Mittal S. (1986) : "CRSL: A Language for Classificatory Problem Solving and Uncertainty Handling", *The AI Magazine*, agost, pp. 66-77.
- Campbell E. J. M. (1987) : "El pensamiento diagnóstico", *The Lancet* (ed. Esp.), Vol 11, 2, pp. 66-69.
- Chandrasekaran B. (1986) : "Generic Tasks in Knowledge-Based Reasoning: High Level Building Blocks for Expert System Design", *Technical Report*, The Ohio State University, September.
- Chandrasekaran B., Mittal S.(1983) : "Conceptual Representation of Medical Knowledge for Diagnosis by Computer: MDX and Related Systems", a AAVV, *Advances in Computers*, (Yovits Ed.), Academic Press, pp. 217-293.
- Clancey W. J. (1985b) : "Software Tools for Developing Expert Systems", a AAVV: *Artificial Intelligence in Medicine*, De Lotto i Stefanelli (eds), Elsevier Science Publishers, pp. 155-178.

- Clancey W. J. (1985a) : "Heuristic Classification", *Artificial Intelligence*, 27, pp. 289-350.
- Clancey W. J., Bock C. (1988) : "Representing Control Knowledge as Abstract Tasks and Metarules", *KSL 85-16*, Stanford Knowledge Systems Laboratory.
- Cohen P. R. (1987) : "The Control of Reasoning Under Uncertainty: A Discussion of Some Programms", *COINS Technical Report*, 87-81, University of Massachusetts, Amherst.
- Cortés U., Sierra C., (1987) : *LISP*, Marcombo.
- Cortés U., Sierra C., Villar A. (1984) : "LISP manual de interprete", *Report de treball*, 84/01, Facultat d'Informàtica de Barcelona.
- Cuena J., Fernández G., López de Mántaras R., Verdejo F. (1986): *Inteligencia Artificial: Sistemas Expertos*, Alianza editorial.
- Dacapo Manual (1987) : *Dacapo Manual*, Dosis GMBH, Dortmund.
- Davis R. (1980) : "Reasoning about Control", *Artificial Intelligence*, 15, 3, pp. 179-222.
- Davis R., Buchanan B. G. (1984) : "Meta-level Knowledge", A AAVV: *Rule-based expert systems: The MYCIN experiments of the Stanford Heuristic Programming Project*, (Buchanan i Shortliffe eds), Addison-Wesley, Reading, Machasussets.
- Davis R., Lenat D. B. (1982) : *Knowledge-Based Systems in Artificial Intelligence*, (Advanced Computer Science Series), McGraw-Hill.
- De Lotto I., Stefanelli M. (eds) (1985) : *Artificial Intelligence in Medicine*, Amsterdam, Elsevier Science Publishers B. V., 1985.
- Desroches P. (1987) : "SYCLARE: Système de classification par apprentissage et reconnaissance des formes. Manuel d'utilisation", *Report de recerca*, CEAB/20, Blanes.
- Dubois D., Prade H. (1985) : *Théorie des possibilités. Applications à la représentation des connaissances en informatique*, Masson, Paris.

- Duda R., Gaschnig J., Hart P. (1979) : "Model design in the PROSPECTOR consultant system for mineral exploration", a AAVV *Expert Systems in the micro-electronic decade*, (Michie D. Ed), Edinburgh University Press, Edinburgh.
- Erman L., Hayes-Roth F., Lesser V., Reddy D. (1980) : "The Hearsay II speech understanding system: Integrating knowledge to resolve uncertainty", *ACM Computing Survey*, 12, pp. 213-253.
- Esteva F. (1981) : "On the form of negations in posets", a AAVV: *Proceedings of XIth I.S.M.L.*, pp. 228-231, Oklahoma.
- Felix R., Godo L., Höffmann A., Sierra C., Moraga C. (1988) : VLSI Chip Architecture Selection using reasoning based on Fuzzy Logic, a AAVV: *19th International Symposium on Multiple-Valued Logic*. (pendent de publicació).
- Fieschi M. (1983) : *SPHINX: un système expert d'aide à la décision en médecine*, Thèse d'Etat en Biologie Humaine presentada a la Université d'Aix-Marseille-II, Mars 1983.
- Freksa, C., López de Mantaras, R (1984) : "A learning system for linguistic categorization of 'soft' observations", a AAVV: *Colloque Assoc. Recherche Cognitive*, p 331-345, Orsay, França.
- Futatsugi K., Goguen J., Messeguer J., ada K. (1987) : "Parameterized Programming in OBJ2", a AAVV: *Proceedings of Ninth Int. Conference on Software Engineering*, IEEE Comp. Soc. Press, pp 51-60.
- Garijo J. (1985): "Entornos específicos para la construcción de sistemas expertos", a AAVV: *Inteligencia Artificial. Conceptos Técnicas y aplicaiones*, Marcombo, pp. 223-230.
- Genesereth M. R. (1983) : "An overview of Meta-Level Architecture", a AAVV: *AAAI-83*, pp. 119-124.
- Godo Ll., López de Mántaras R., Sierra C., Verdaguer A. (1987) : "Managing Linguistically Expressed Uncertainty in MILORD-application to Medical Diagnosis", a AAVV: *7th International Workshop Expert Systems and*



*their Applications*, Avignon, pp. 571-596. Reimprés (amb lleugeres modificacions) per invitació dels editors a *Artificial Intelligence Communications*, 1, 1.

Godo Ll., Sierra C. (1988) : "A New Approach to Connective Generation in the Framework of Expert Systems Using Fuzzy Logic", a AAVV: *Proceedings of 18th IEEE International Symposium on Multiple-valued Logic*, Palma de Mallorca, pp. 157-163.

Godo, L., López de Mántaras, R., Sierra C., Verdaguer A. (1988) : "MILORD: The Architecture and the Management of Linguistically Expressed Uncertainty", *Int. Journal of Intelligent Systems*, (en premsa).

Harper R., McQueen D., Milner R. (1986) : "Standard ML", *Report*, ECS-LFCS-86-2, Edinburgh University.

Heckerman D. (1986) : "Probabilistic Interpretation for MYCIN's Certainty Factors", A AAVV: *Uncertainty in Artificial Intelligence*, (L.N. Kanal i J.F. Lemmer Eds), North Holland, Amsterdam, pp. 167-196.

Herrman J., Bruck R., Temme K. (1988): *ALICE-1 An alternative IC CAD Environment*, Technical Report, University of Dortmund.

Johnson R. (1986) : "Independence and Bayesian Updating Methods (Research Note)", *Artificial Intelligence*, 29, 2, pp. 217-222.

Kassirer J., Gorry G. (1978) : "Clinical Problem Solving: A Behavioral Analysis", *Annals of Internal Medicine*, 89, 2, pp. 245-255.

López de Mántaras R., Sanz F., Sierra C., Verdaguer A. (1987) : "MILORD+PNEUMON-IA: Un outil et une application en médecine", a AAVV: *Colloque Intelligence Artificiel et santé*, Toulouse, pp 45-54.

López de Mántaras R., Agustí J., Sierra C. (1987) : "The management of Uncertainty in the MILORD System", A AAVV: *Proceedings of 2nd International Fuzzy Systems Association Congress*, Tokyo, Vol 1, pp. 294-297.

MACSYMA (1974) : "The MACSYMA reference manual", The MATHLAB Group, MIT.

- Maes P. (1988) : "Issues in Computational Reflection", a AAVV: *Meta-level Architectures and Reflection*, (Maes P. i Nardi D. eds), Elsevier Science Publishers B.V., Amsterdam, pp. 21-35.
- Maes P., Nardi D. (1988) : *Meta-Level Architectures and Reflection*, Elsevier Science Publishers B.V., Amsterdam.
- McCarthy J. (1980) : "Circumscription A form of Non-monotonic Reasoning", *Artificial Intelligence*, 13, pp. 27-39.
- Michel D., Botti G., Fieschi M., Joubert M., Casanova P., San Marco J. (1987) : "Validation d'une base de connaissance destinée aux médecins généralistes dans l'aide à la thérapeutique du diabète: étude en aveugle", a AAVV: *Proceedings of the 7th International Symposium on Expert Systems and Applications*, Avignon, vol 2, pp. 139-152.
- Miller D. A. (1986) : "A Theory of Modules for Logic Programming", a AAVV *Proceedings of 1986 IEEE Symp. on Logic Programming*.
- Mittal S., Ckandrasekaran B., Sticklen J. (1984) : "Patrec: A Knowledge-Directed Database for a Diagnostic Expert System", *IEEE computer*, September, pp. 51-58.
- O'Keefe R., (1985) : "Towards an Algebra for Constructing Logic Programs", a AAVV: *Proceedings of 1985 IEEE Symp. on Logic Programming*, pp 152-160
- Patil R. S. (1981) : "Causal Representation of Patient Illness for Electrolyte and Acid-Base Diagnosis", *Technical Report*, TR-267, MIT Laboratory for Computer Science, Cambridge, Mass., Octobre.
- Patil R. S. (1987) : "A case Study on Evolution of System Building Expertise: Medical Diagnosis", a AAVV: *AI in the 1980's and Beyond: An MIT Survey*, (Grimson W. E. L. i Patil R. S. eds), Cambridge, MIT Press.
- Pitrat J. (1987) : "Le futur de l'intelligence artificielle. Les métaconnaissances", *Revue d'intelligence artificielle*, 1, 4, pp. 69-88.

- Plaza E. (1987) : *EAR\* : Sistema d'ajut a l'adquisició i estructuració de coneixements*, Tesi doctoral en informàtica per la Universitat Politècnica de Catalunya.
- Pople H. E. Jr. (1982) : "Heuristic Methods for Imposing Structure on Ill-Structured Problems: The Structuring of Medical Diagnostics", a AAVV: *Artificial Intelligence in Medicine*, (Szolovits P ed), Westview Press, Boulder Colorado, pp. 119-190.
- Porter J., Kingsland L., Lindberg D., Shah I., Beuge J., Hazelwood S., Kay D., Homma M., Akizuki M., Takano M., Sharp G. (1988) : "The AI/RHEUM Knowledge-based Computer Consultant System in Rheumatology", *Arthritis and Rheumatism*, 31 pp. 219-226.
- Quinlan J.R. (1986) : "Induction of Decision Trees", *Machine Learning*, 1, pp. 740-744.
- Reiter (1980) : "A Logic for Default Reasoning", *Artificial Intelligence*, 13, pp. 81-132.
- Rubio F., Sanz R., Camacho E., Ollero A. (1987) : "A rule based inference method for supervision of self-tuning controllers", a AAVV: *Proceedings IFAC International Symposium on Low Cost Automation*, pp. 43-48, València.
- Samaan, M. (1985) : "Station de travail et systemes experts en supervision", *Rapport DEA d'Automatique*, LAG Grenoble.
- Sannella D., Wallen L. A. (1987) : "A Calculus for the Construction of Modular Prolog Programs", a AAVV: *Proceedings of 87 IEEE Symp. on Logic Programming*, pp 368-378.
- Sanoff S., Wellstead P. (1985) : "Expert Identification and Control", a AAVV: *Proceedings of IFAC Symposium on Identification and Systems Parameter Estimation*, pp. 1046-1051, Arlington.
- Sanz R. (1987) : *Diseño de controladores autoajustables. Nivel supervisor experto*, Tesi doctoral presentada a l'Escola tècnica superior de ingenieros industrials, Univ. Santiago de Compostela.

- Sanz R., Aguilar J., Sierra C , Godo L., Ollero A. (1988) : "Adaptative Control with a Supervisor Level using a Rule Based Inference System with Approximate Reasoning", a AAVV: *actes del congrés IMACS-88*, París.
- Shafer G. (1979) : "Allocations of Probability", *Ann. Probability*, 7.
- Shortliffe E. H. (1976) : *Computer Based Medical Consultations: MYCIN*, American Elsevier, New-York.
- Shriver B., Wegner P. (Eds.) (1987) : *Research directions in Object-Oriented Programming*, MIT press.
- Soula G., Thirion X., Guliana J., Navez I., Vialettes B., San Marco J., Ledoray V. (1987) : "Une validation multicentrique du système exoert PROTIS", a AAVV: *Proceedings of the 7th International Symposium on Expert Systems and Applications*, Avignon, vol 1, pp. 239-255.
- Sticklen J., Chandrasekaran B., Josephson J. R. (1985) : "Control Issues in Classificatory Diagnosis", a AAVV: *proceedings of IJCAI-85*, pp. 300-306.
- Sticklen J., Smith J. W., Chandrasekaran B., Josephson J.R. (1987) : "Modularity of Domain Knowledge", *International Journal of Expert Systems*, Vol 1, 1, pp. 1-15 .
- Szolovits P. (1985) : "Types of Knowledge as Bases for Reasoning in Medical AI Programs", a AAVV: *Artificial Intelligence in Medicine*, (Szolovits P ed), Westview Press, Boulder Colorado, pp. 119-190.
- Szolovits P., Patil R. S., Schwartz W. B. (1988) : "Artificial Intelligence in Medical Diagnosis", *Annals of Internal Medicine*, 108, pp. 80-87.
- Temme K. (1987) : "CHARM - A Knowledge based System for VLSI Chip-Architectures: A System's Perspective", a AAVV: *actes del congrés CompEuro*, Hamburg.
- Trillas E. (1979) : "Sobre funciones de Negación en la Teoría de Conjuntos difusos", *Stochastica*, Vol 3, 1, pp. 47-60.

- Valverde L., García P., Godo L. (1987) : "Técnicas de razonamiento aproximado", a AAVV: *Inteligencia Artificial: Conceptos, técnicas y aplicaciones*, Marcombo, Barcelona, pp. 93-100.
- Valverde L., Trillas E. (1985) : "On Modus Ponens in Fuzzy Logic", a AAVV: *Proceedings of 15th International Symposium on Multiple-Valued Logic*, Kingston, Canada, pp. 294-301.
- Van de Velde, W. (1988) : *Learning from Experience*, Ph. D. dissertation of Vrije Universiteit Brussel.
- Verdaguer A. (1989) : *PNEUMON-IA : Desenvolupament d'un sistema expert d'ajuda al diagnòstic mèdic*, Tesi doctoral en medicina per la Universitat Autònoma de Barcelona (en preparació).
- Verdaguer A., Patak A., Sanz F., Sierra C., López de Mántaras R. (1989) : "Validation du systeme expert PNEUMON-IA", a AAVV: *actes del congrés IA-SANTE*, Toulouse, (en premsa).
- Weiss S. M., Kulikovski C. A., Amarel S., Safir A. (1978) : "A model-based method for computer-aided medical decision-making", *Artificial Intelligence*, Vol 11, 1-2, pp. 145-172.
- Zadeh L. A. (1983) : "The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems", *Fuzzy Sets and Systems*, 11, pp. 199-277.
- Zadeh L.A. (1979) : "A theory of Approximate Reasoning", *Machine Intelligence*, 9, pp. 149-194.

