

DEUSTUKO UNIBERTSITATEA

Informatika Fakultatea

MAMUR : Mugapenen Betetzean oinarritutako
tresna modular multimaila.

Pikaza Atxa Juan Manuel-ek

aurkeztutako

DOKTOREGO TESIA

Bilbo, 1992

DEUSTUKO UNIBERTSITATEA

Informatika Fakultatea

MAMUR : Mugapenen Betetzean oinarritutako tresna modular multimaila.

Pikaza Atxa Juan Manuel-ek aurkeztutako DOKTOREGO TESIA
Del Moral Bueno Anselmo Doktoreak ZUZENDUA.

Zuzendaria

Doktoregaia

1992.eko Azaroa

Lourdesi

ESKERRAK

Nire esker zintzoa

Anselmo del Moral, nire zuzendariari, informatika arlo honetan sartu nauelako, eman dizkidan aholku zuzenengatik, eta behar izan ditudan baliabide guztiak eduki ditzadan arduratzeagatik.

Luis Aldamiz-Etxebarria-ri, CETODIA Sistema aditua medikuntzan erabiltzen diren arrazonamendu prozesuez hornitzeko elkarrekin eman ditugun etengabeko erreflexio eta lan ordu ugariengatik. Haren laguntza gabe ezin izango zen tesi hau egin.

Ulises Cortés-i, bere lagunduteko erakutsi duen prestaera bereziagatik, eta bere eritzi kritikoagatik tesi honen azken garapenean hainbesteko eragina eduki duena.

Enric Plaza-ri, nire ikerketarenganeko bere iruzkin eta iradokizunengatik.

David Diaz-i, bere laguntasunagatik CETODIA datu-base desberdinekin konektatzeko.

Informatika Fakultateko Kalkulu-zentruko Zuzendaritza eta langillei euren laguntza gogatsuagatik.

Inma Arrieta eta Amaia Albizuri-ri, hauen laguntasun estimagaitzagatik arlo honen terminologia Euskerara moldatzeko.

Nire etxetar eta lagun guztiei, urte guzti hauetan zehar animatu nautelako.

Eta azkenez, nire emazteari eta nire gurasoei, ez dutelako inoiz zalantzarik izan.

LABURDURAK

LABURDURA	KONTZEPTUA
AA	Adimen Artifiziala
EOS	Erregeletan Oinarritutako Sistemak
EGB	Egintza-basea
ERB	Erregela-basea
ES	Erabakitzeko Sistemak
HLAJ	Hedakuntza Lokala eta Atzera Jotzearen konbinazioa
IM	Inferentzi Motorea
JB	Jakintza-basea
JI	Jakintza Iturria
JL	Jakintzaren Lorpena
JOS	Jakintzan Oinarritutako Sistemak
MEB	Metaerregelen basea
MBA	Mugapen Betetze Arazoak
MOA	Mugapenetan Oinarritutako Arrazonamendua
OOL	Objektuetara Orientatuko Lengoia
OOP	Objektuetara orientatutako Programazioa
POP	Prozeduretara Orientatutako Programazioa
PS	Produkzio Sistemak
SA	Sistema Adituak

AURKIBIDEA

LABURPENA

ABSTRACT

ESKERRAK

LABURDURAK

1. KAPITULUA: TESIAREN AURKEZPENA	1
1.1.- MOTIBAZIOAK.	1
1.2.- EBATZI BEHAR DIREN ARAZOEN PLANTEAMENDUA. . .	2
1.3.- MAMUR SISTEMAREN EZAUGARRIAK.	3
1.4.- EKARPEN NAGUSIAK.	5
1.5.- TESIAREN KAPITULUEN BANAKETA.	6
2. KAPITULUA: LABURPEN HISTORIKOA ETA GAUR EGUNGO EGOERA.	8
2.1.- JAKINTZA.	8
2.2.- JAKINTZAREN ERERREPRESENTAZIORAKO FORMALISMOAK.	14
2.2.1.- Egoeren espazioa.	14
2.2.2.- Errepresentazio eskema deklaragarriak. . .	14
2.2.3.- Prozeduren erreprezetazio eskemak.	18
2.2.4.- Errepresentazio eskema mixtoak.	22
2.3.- JAKINTZAREN MANIPULAZIORAKO FOKATZEAK.	28
2.3.1.- Prozeduretara orientatutako programazio. .	29
2.3.2.- Objektuetara Orientatutako Programazioa. .	29
2.4.- ADIMEN ARTIFIZIALARI DAGOKION KONTROLA. . . .	35
2.4.1.- Erreflexioa.	38
2.4.2.- Introspekzioa.	41
2.4.3.- Mugapenak.	44

2.4.4.- Multimaila arkitekturak.	48
2.5 .- KONTROLERAKO METAJAKINTZAREN INPLEMENTAZIOA. .	50
2.5.1.- Kontrola PS-etan: Metaerregelak.	50
2.5.2.- Kontrola JB modularizatueta.	58
2.5.3.- Kontrola arazozen ebazpenerako eredu koopera- tiboetan.	63
2.5.4.- Kontrola denbora errealeko JOS-etan. . . .	69
2.6 .- IKASKUNTZAREN GARRANTZIA KONTROLEAN.	70
2.7 .- BIGARREN BELAUNALDIKO SA-TAKO KONTROLA. . . .	73
2.8 .- ONDORIOAK.	76
 3. KAPITULUA: JAKINTZAREN ERREPRESENTAZIOA MAMUR-EN. .	77
3.1.- SARRERA.	77
3.2.- JAKINTZA MOTAK.	77
3.3.- ERREPRESENTAZIO FORMALISMOAK.	78
3.3.1.- Produkzio sistema ideiaren integrazioa Frame-aren idearekin.	79
3.3.2.- Objektu desberdinen deklaraziorako funtzioak.	81
3.4.- JAKINTZAREN KOMPILAZIOA.	85
3.4.1.- Sarrera.	85
3.4.2.- Jakintzaren konpilazioa MAMUR-en.	91
 4. KAPITULUA: ZIHURGABETASUNA-REN MANEIOA.	94
4.1.- SARRERA.	94
4.2.- ZIHURGABETASUNAREN TRATAMENDUA MAMUR-EN. .	97
 5. KAPITULUA: JAKINTZAREN MANIPULAZIOA.	103
5.1.- SARRERA.	103
5.2.- IRAGAZKETA EDO MATCHING.	103
5.3.- RETE ALGORITMOAREN INPLEMENTAZIOA MAMUR-EN. .	107
5.4.- MAMUR-EN ARRAZONAMENDU PROZESUA.	109
5.5.- ARRAZONAMENDUAREN AZALPENA MAMUR-EN.	112
5.6.- SARRERA-IRTEERA INTERFACEA.	113

6. KAPITULUA: KONTROLA.	114
6.1.- SARRERA.	114
6.2.- MUGAPENETAN OINARRITUTAKO KONTROLA.	115
6.2.1.- Definizioak.	115
6.2.2.- Mugapenak plangintzan.	119
6.2.3.- Mugapenen Betetzea.	127
6.2.4.- Arazo Errazen ideia: Mugapenen Betetzeen kasua.	134
6.3.- KONTROL EGITURA MAMUR-EN.	138
6.3.1.- Arkitektura modular multmaila.	139
6.3.2.- Arrazonamendu metodoa.	142
6.3.3.- MAMUR-en kontrol arkitektura.	144
6.3.4.- Kontrol zikloa MAMUR-en.	149
6.3.5.- Metaerregelak MAMUR-en.	152
6.3.6.- Plangintza jerarkikoa.	153
6.3.7.- Mugapenen errepresentazioa erregelen bidez	158
6.3.8.- Kontrola Arazo Errazen ebazpenaren bidez.	162
6.3.9.- Konpilazioaren ondorioa MAMUR-en eraginkortasunaren hobetzean.	162
7. KAPITULUA: CETODIA: MAMUR-EN APLIKAZIO BAT.	164
7.1.- SARRERA.	164
7.2.- ARAZO MEDIKOA: ZETOAZIDOSI DIABETIKOA.	164
7.2.1.- Clinika.	166
7.2.2.- Tratamendu eredua.	168
7.3.- CETODIA-REN SOLUZIO INFORMATIKOA.	172
7.3.1.- CETODIA sistema adituaren justifikazioa.	174
7.3.2.- Arkitektura.	175
7.3.3.- Sarrera-irteera interfacea.	181
7.3.4.- Emaitzak.	184
8. KAPITULUA: ONDORIOAK ETA ETORKIZUNERAKO JARRAIPIDEAK	188
8.1.- ONDORIOAK.	188
8.2.- ETORKIZUNERAKO BIDEAK.	190
BIBLIOGRAFIA.	192

1. KAPITULUA: TESIAREN AURKEZ- PENA

1.1.- MOTIBAZIOAK.

Adimen Artifizialaren (AA) teknikak ahalegin ikertzaile handi baten ondorioz azken hamarkadan izan duten aurrerakuntza kontsideragarria dela eta, eta aplikazio garrantzitsuekin bukatu dena, batez ere Sistema Adituen (SA) alorrean, eta esesten diren arazoen zailtasun gero eta handiagoaren aurrean honako honen beharrezana dagoela ikusi da:

- * Egituraketa handiagoa hauen arkitekturetan, non kontrolari buruzko jakintza eta dominioaren ganeko jakintza bereizki eta argi errepresentatuak egon daitezen sistemen mantentzea eta berrikuntza errazteko.
- * Kontrolari buruzko jakintzaren tratamendu hobeagoa helburuetara azkarrago iristeko eta sistemaren iharduketa (performance) gradua gehitzeko.

Tesi honekin puntu biak azpimarratu nahi ditugu horrela MAMUR tresnak Jakintzan Oinarritutako Sistemak (JOS) eraikitzeke ahalbidea eskeini dezan era modular eta deklaragarrian.

1.2.- EBATZI BEHAR DIREN ARAZOEN PLANTEA-MENDUA.

JOS bat eraikitzeke orduan arazo batzu aurki ditzakegu:

- * Giza adituaren lana egin eta modu eraginkorrean arrazoitzen duen tresna edo programa bat eduki beharko da helburuak ahalik eta azkarren lortzeko.
- * Jakintzaren Lorpena (JL), bai teorikoa bai enpirikoa, JOS-aren garapeneko arazo nagusietariko bat da horregatik tresnak jakintza-ingenieriaren lana erraztu behar du.
- * Ez da beti egoten kasu konkretu baterako jakintza errepresentazio eta manipulazio aproposa haizutuko duen tresna bat. Honek tresna berezi bat eraikitzaera eraman dezake emaitza eraginkorragoa lortzeko, nahiz ta horrek ez luke suposatu beharko sistema berri baten kodifikazio osoa, Objektuetara Orientatutako Programa-zioaren (OOP) ideiari jarraituz aurretik idatzitako sistema baten zatiak erabil ahal izango direlako. Urrunago hel zitekeen Modularizazioa eta Bigarren Belaunaldiko JOS-en tesiak praktikan jarritz, hauek beste sistema batzuen modulo osoak konpartitzea haizutuko lukete, behar besteko orokortasunarekin eraikiz gero.

Beraz, oso litekeena da JOS bat eraikitzeke, kasurako tresna aproposak edukitzea pentsatu bahar izatea eta ondorioz jadanik badagoen tresna bat gutxienez aldarazi behar izatea, hemendik ateratzen dena da oso komenigarria izango dela tresna hauen eraikuntzarako kodea konpartitu eta sistema berriaren idazkera osoa ekiditea haizutuko duen metodologia erabiltzea. Frogatu da OOP-k eta Modularizazioak helburu honi laguntzen diotela.

1.3.- MAMUR SISTEMAREN EZAUGARRIAK.

MAMUR tresnaren ahalbideen ideia argia egin dezagun, esan dezakegu, nagusiki: produkzio sistema bat dela, honen arkitektura objektuetara orientatuta dagoela eta sailkapen¹ atazen ebazpenean espezializatuta aurreranzko eta atzeranzko arrazonamenduarekin eta aldagaiz egin daitekeen iragazketarekin RETE algoritmoari jarraituz, mugapenetan oinarritutako egitura modular multimaila batekin, eta termino linguistikoak erabiliz zihurtasun balio gisa.

MAMUR izena euskal mitologiatik dator eta potere handien jabe diren jeinu txiki batzu identifikatzen ditu, hauen lankortasun eta haien jabeak eskatu dien edozein lan egiteko ahalmenagatik nabaritzen dira, eskatutako lan hori zailena izanik ere.

Honegatik bereizten da:

- * Jakintzaren errepresentazioa produkzio sistemaren (PS) ideia frame-en ideiarekin integratuz, gaur eguneko errepresentazio deklaragarri eta prozeduralen arteko eztabaida gainditzeko joerari jarraituz [WINO75] [FIKE85] [SIER89].
- * OOP, hemen hala erregelak nola egintzak objektuek osatzen dituzte.
- * Iragazketa (matching) RETE algoritmoaren inplementazio hobetu batez egiten da, bai aurreranzko kateaketan bai atzeranzkoan. Iragazketa aldagaien erabilera haizutuz, zeinek erregelei orokortasun handiago batez gain premisen baldintza desberdinak erlazionatzeko ahalbidea ematen dieten.
- * Termino linguistikoen bidez adierazitako zihurgabetasunaren tratamendua haizutzen du.
- * Arkitektura multimailak jakintzaren errepresentazio modularrari bide ematen dio, JB-ren mantentzea eta

¹ Soluzioen bilaketa a priori ezaguna eta finitua den espazio baten.

eguneratzea erraztuz gain, hauen existentzia ahalbideratzen duelarik:

- Dominio maila bat, edo ataza zehatz eta behin betikoak egiten dituzten moduloak nahi diren helburuak lortzeko.
 - Metamaila bat, era jerarkikoan jarritako modulo batzuez osatua, zeinen jakintzak mugatu edo zehaztu egiten duen bilaketa espazioa aurrez esarritako plan bati jarraituz eta kasuaren soluzioa lortzeko beharrezko estrategiak sortuz. Plana maila altueneko moduloan dago eta honen erregelak eragile zeregina dute. Estrategiak zehazten dituzten moduloak bete behar diren mugapen gisa jarduten duten erregelek osatzen dituzte, hauen ondorioak Jakintza-basearen (JB) errepresentazio osoan zehar hedatuko dira, kasu bakoitzerako dominio zatirik aipagarriena erabiltzeko eta era egokienean.
- * JB konpilatzeko aukera karga azkarragodun LISP kode exekutagarria lortuz.
 - * Jarraitutako arrazonamenduaren azalpena eskeintzen du helburu zehatz bat lortzeko.
 - * Bada interface bat emaitzen aurkezpenerako leiho, kolore eta sagua erabiltzea errazten duena.

Tresna hau erabili da CETODIA sistema aditua eraikitzeke, Zetoazidosi Diabetikoaren tratamendua umeengan egiteko. Sistema honen garapena Kurtsoaren Azkeneko Egitasmo gisa hasi zen 88-89 kurtsoan zehar LABEIN-en emandako Adimen Artifizialari buruzko Kurtsoan, lehenengo prototipo bat lortu zelarik PERSONAL CONSULTANT PLUS tresna erabiliz. Behin kurtsoa bukatu eta gero eta egitazmoa guztiz egingarria zela ikusirik lanari ekin genion bigarren prototipo bat lortu arte, aurreko tresnaren ahalbide guztiak erabiliz. Kontrolaren aldea eta JB-ren modularitatea hobetzeko asmoz MAMUR egitea erabaki genuen.

MAMUR Unibertsitate honetan garatu da VAXLISP lengoaiaren 2.2 bertsioa erabiliz, COMMON LISP-aren dialektoa, lehenengo bertsio batera heldu arte. Azken bertsioa COMMON LISP-en beste dialekto bat erabiliz garatu da, GCLISP bere 3.1 bertsioan.

1.4.- EKARPEN NAGUSIAK.

Erregeletan oinarritutako sistema batean iragazketa denborak oso portzentai handia darama, % 90-ekoa ere izan ahal delarik, prozesuaren denbora osoari dagokionez, horregatik haren hobetzeak sistemaren errendimenduarengan eragin handia du. Hori dela eta RETE algoritmoa [FORG82] erabili da, bere eraginkortasunagatik gehien erabili dena [GUPT89], implementazio hobetu bat lortuz RETE zuhaitzari informazio gehigarria erantsiz beharrezkoak ez diren bilaketak deuseztuz honek dakarren denbora aurrezpenarekin.

RETE zuhaitzaren sorkuntzak denbora gehiago beharko du baina behin baino ez da sortuko JB-aren konpilazio aldian. JB konpilatu bat edukitzeak karga prozesua askoz ere azkarragoa izango dela esan nahi du.

Modularitatea errazteko zuhaitz bat sortzen da dominioaren espazio mugatu edo azpiarazo bati dagozkien jakintza errepresentatzen duten modulo edo erregela multzo bakoitzeko. Honekin batera zuhaitz guztiak zuhaitz orokor batean elkartzen dira modulo batean ateratzen dena bestetariko edozein eguneratu ahal dezan.

Iharduera plana MAMUR-en egoera espazio baten bidez errerepresentatuko da, goreneko mailaren moduloan, modulo honetako erregelek eragile zeregina beteko dute.

Estrategien sorkuntza MAMUR-eko egitura multimailaren metamailan kokatutako modulo jerarkian errepresentaturiko jakintzatik abiatuta egingo da, egitura multimaila hau garatuko den SA-k eskatzen dituen beste mailarekin osatua egongo da, Mugapenetan Oinarritutako Arrazonamendua erabiliz, non modulo hauen erregela bakoitzak mugapen baten zeregina beteko duen. Bilaketa espazioa mugatzea haizutuko duten mugapen maila desberdinak egon daitezke kasurako beharrezkoa den jakintza erabiltzeko. Mugapen bakoitzak jarraikoa egin ahal du: egintza berri bat sortu, beste mugapen modulo bat aktibatu (aurreranzko kateaketarekin prozesatuko dena), edo dominioaren jakintza daramaten moduloak aktibatu (atzeranzko kateaketaz prozesatuz).

Erregela guztiak, edozein modulo edo mailakoak izanik ere egitura bera daukate eta RETE zuhaitzean errepresentatuak egongo dira, interprete berberak prozesatuak izango direlarik.

1.5.- TESIAREN KAPITULUEN BANAKETA.

Tesi hau zortzi kapituluetan banatu da.

Lehenengo kapitulu honetan tesiaren aurkezpena egin da. Bigarren kapitulu laburpen historikoa egiten da eta SA-k eraikitze tekniken eta kontrolerako erabilitako metodoen gaurko egoera azaltzen du. Hirugarren kapitulu MAMUR-en eraikuntzan erabilitako jakintza errepresentazio formalismoaren xehetasunak ematen dira, baita JB-aren konpilaziorako jarraitutako prozedura ere. Laugarren kapitulu zihurgabetasunaren errepresentaziorako termino linguistikoen bidez erabilitako metodoa aurkezten da. Bostgarren kapitulu jakintzaren manipulaziorako erabilitako teknikei buruzkoa da, hau da, arrazonamenduan erabilitako kateaketa mota aldeztu aurretik iragazitako (RETE algoritmoa jarraituz) erregelak aktibatze. Eraitza eta azalpenen aurkezpenarako interfacea ere aurkeztu da. Seigarren kapituluaren barruan

helburuak ahalik eta azkarren lortzeko estrategia desberdinak azaltzen dira, hau da **metajakintzaren** erabilera mugapenen bidez errepresentatua arrazonamenduaren kontrolerako JB-aren erregelak eta metaerregelak osatzen zuhaitzaren iñausketaren bidez, kasu konkretu baterako baliozkoak ez diren adarrak deuseztatuz. Zazpigarren kapituloko edukiak MAMUR sistemaren alderdi guztien baterako funtzionamendua erakusten du, haren portaera eraginkorra CETODIA SA-aren eraikeran egiaztatuz. Azkenik zortzigarren kapituloan lanaren ondorioak erakusten dira eta etorkizuneko ikerketa bideak.

2. KAPITULUA: LABURPEN HISTO- RIKOA ETA GAUR EGUNGO EGOERA.

2.1.- JAKINTZA.

Adimen Artifizialaren arazoen ebazpenerako gaur egun erabiltzen diren tekniken berritasuna arazoaren dominioaren informazio espezifikoaren kantitatearen erabileran datza. Informazio hau ez da geroago interpretatuko diren datuen masa batez osatzen soilik, sistema informatiko konbentzionaletan bezala, baizik eta dominioaren aspektu batzuen errepresentazio sinbolikoak erte barne hartzen ditu Levesque-k "siniskera" (belief) izenarekin identifikatu zuena eta ikertzaile gehienak jakintza izenaz deitzeko bat etorri direna. Hala ere erabilitako informazio moten kopurua mugatu beharra dago, soluzioen espazioa txikiagoa izan dadin, eta leherketa konbinatorioa kontrolpean gera dadin [LEVE86] teknika hauek Jakintzan Oinarrituriko Sistemak (JOS) deitukoak sortu dituzte.

Sistemak erakusten duen jakintzak datuen egitura ezagun batzuetan errepresentatua egon behar du; datuen egitura hauen manipulazio esplizitoak sistemak adierazten duen jarrera zehazten du. Mota honetako datu egituren kolekzioa, jakintzan oinarrituta dagoen edozein sistemak duena, jakintza-basea (JB) bezala ezagutzen da, eta errepresentatze eskema batekin edo batzuekin eraikita egon daiteke, sistema konbentzionalak ez bezala.

Jakintza-base batek mota askotako informazioa eduki behar du:

- a) objektuei buruzkoa.
- b) Prozesuei buruzkoa.
- c) Inguruneari dagokiona.
- d) Errepresentatzeko zaila den informazioa: sen ona, motibazioak, Kausalitatea, helburuak eta abar.

Hipotesi honetatik ateratzen da jakintzan oinarritutako sistema baten esentziazko ezaugarria jakintza asplizitoen base baten existetzia dela, base honen gainean inplizitoki bere barnean duen informazioa ateratzen duten arrazonamendu mota desberdinak aplikatzen dira. Ikuspuntu honetatik JB-a errealitate bat errepresentatzen duen sinbolo multzo bat da, eta makina edo gizaki batek interpretatua izan daitekeena. AA arloan ematen den garrantzi handiko arazoa hauexen aukeraketa da:

- Dominioaren jakintza berezi mota bat edo batzu eta hau deskribatzeko primitibak.
- Errepresentatzeko formalismoa.
- Arazoa ebazteko proposak izan daitezcan arrazonamendu mekanismo bat edo batzu.

Egitura sinbolikoen erabilera ez da soil soilik AA-k egiten duena; datu-baseen kudeaketa bezalako arloetan ere erabiltzen da, eta programazio lengoaietan eta datu egituretan ere bai. Hala eta guztiz ere aurreko hipotesiaren azpian datzan metodologiak utzi egiten digu AA-ren barrutian jakintzaren errepresentazioari buruz egiten den ikerketa beste esparru batzuetan garatzen den ikerketarengandik desberdintzeko [BRAC85].

Dominio partikularretako informazio konplexuen egituren errepresentazioaren formalismo on batek hurrengo propietate hauek izan behar ditu:[RICH91]

- a) Egokitasun errepresentazionala, dominioan beharrezkoak diren jakintza mota guztiak errepresentatzeko trebetasuna.
- b) Egokitasun inferentziala, errepresentazio egiturak manipulatzeko trebetasuna, horrela aurrekoetatik inferituak diren jakintza barriei dagozkien egitura berriak sortzen dituztelarik.
- c) Eraginkortasun inferentziala, METAJAKINTZA izeneko errepresentazio egiturari informazio gehigarria eransteko ahalmena, hau konputuak hobetzeko inferentzi mekanismoen atentzioa zentratzeko erabil daiteke.
- d) Lorpen eraginkortasuna, informazio berria lortzeko ahalmena. Idealki sistemak bera bakarrik gai izan beharko litzateke informazio berriaren lorpena eta geroagoko errepresentazioa kontrolatzeko.

Lehenengo biak errepresentazioari buruzko aipamena egiten dute eta hurrengo biak sistemak egiturez egiten duen erabilerari buruzkoa.

Jakintzaren errepresentazioan bi osagai nagusi daude [MCCA69]:

- 1) Bat epistemologiko motakoa, errepresentazioari berari dagokiona, zein munduko egintzak daude erabilgarri?, nola errepresenta daitezke egintza hauek memorian?, eta, zein erregelak uzten digute legezko ondorioak ateratzen egintza hauek abiapuntu direla?.
- 2) Heuristiko motako beste bat, behin errepresentatua dagoen jakintzari ematen zaion tratamenduaari dagokiona, hau da, une bakoitzean erabakitzen duen eta arazoak ebazten dituen mekanismoa aipatzen du, erre-presentazioan dagoen informazioan oinarrituta.

Jakintzaren errepresentazioaren osagai heuristikoak joera bi sortu ditu: eskema deklaragarriak eta eskema prozeduralak.

Lehendabizikoetan, jakintza gero emango zaien bere erabiliaren-gandik aparte errepresentatzen da , eta kontrol egokia lortzen da bai xede orokorreko estrategien bidez bai informazio gehiagoren gehipenen bidez (kontrol informazioa) xede orokorreko estrategiak egiten duen jakintza deklaragarrien erabilera zuzentzen duena, aplikazio arloan. Eskema prozeduraletan, ordea, errepresentatutako jakintza nola erabili behar denari buruzko informazioa berarekin darama.

Formalismo desberdinak honelaxe klasifika litezke:

- 1) Egoeren espazioa.
- 2) Deklaragarriak.
 - 2.1) Eskema logikoak.
 - 2.2) Sare semantikoak.
- 3) Prozeduradunak.
 - 3.1) Representazio prozedurala.
 - 3.2) Produkzio sistemak.
- 4) Mixtoak.
 - 4.1) Frame-ak eta Script-ak.
 - 4.2) Analogikoak edo Zuzenak.

Formalismo hauek ez dute elkar baztertzen, baizik eta errepresentatu behar diren egiturak ulertzeko dauden modu desberdinak harrotzen dituzte. Beste alde batetik nahiz eta formalismo bakoitzak abantailak eskeintzen dituen dominio berezi batzuetan, gaurko joera formalismo desberdinen abantailak egitura bakar batean integratzea da, honegatik hauetariko hiru formalismo JOS-etan jakintzaren paradigma euskarriak bihurtu dira: eskema logikoak, produkzio sistemak eta frame-ak.

Hemen ez dira sartu errepresentazio lengoiak: PROLOG [BRAT86], PRUF [ZADE78], KRIPTON [BRAC83], KRL [BOBR77], KEE [FIKE85], OPS5 [BROW85], eta abar bezalakoak.

JOS baten errepresentazio formalismo egokia erabakitzailea da, horretarako kontutan eduki beharko da:

- * Adierazkortasuna eta konputabilitateen arteko oreka ona.
- * Errepresentatu behar den jakintza motak: teorikoak, enpirikoak eta abarrekoak.
- * Dominioari eta kontrolari dagokien jakintza banatzea..

Jakintza modu egokian errepresentatua izateaz gain, hau ahalik eta era egokienean manipulatu behar da helburuak ahalik eta era azkarrenean eta seguruenean lortzeko, horretarako fokatze bi daude:

- * Funtzionala edo prozeduratara orientatutako programazioa (POP).
- * Objektuetara orientatutakoa edo objektuetara orientatutako programazioa (OOP).

Fokatze hauetako bakoitza sistema baten eraikitzerako metodologia bat bezala erabil daiteke.

Jakintzaren errepresentazio eta manipulazio era behin zehaztuz gero komenigarria da sistema kontrol egitura batez hornitzea, egitura honek beharrezko jakintza estrategikoa eduki behar du kasu bakoitzerako komenigarriagoa den arazoari buruzko jakintzaren zatia erabiltzeko, eta horrela helburuak azkarrago lortu ahal izateko, sistemari jokaera adimentsua emanaz. Jakintzari buruzko jakintza honi METAJAKINTZA izena ematen zaio, [HAYR83]-en 7. kapituluan adierazten den definizio orokorrari jarraituz, honetan ezaten da "meta x" "x x-i buruz" dela, honekin era berean beste hauen definizioa ere ezarri daiteke: metateoria, metalengoia, metaarrazonamendua, etabar.

Jakintza adierazpenerako lengoia bat erabili behar da eta metajakintzarako metalengoai bat. Harekin asmo hauek lortu nahi dira:

- dominioa eta kontrolaren jakintzaen arteko bereiztea.
- bien errepresentazio deklagarria.

Objektu lengoia eta metalengoiaren arteko bereizte garbia logikoek ezarri zuten formalki lehenaldiz arrazonamendu matematikoaren oinarrien ikasketan (Tarski 1923-1938, Gödel 1930). Aiello eta Weyhrauch-en [AIEL75] arabera algebrari buruzko liburu bat irakurtzean konturatzen gara erakusten diren teoremen gehiengoa benetan metateoremak direla, teorema bat demostratzeko beste teorema batetan adieraztutako jakintzara jo behar dugu (objektumaila) demonstratu behar den teoremaren arauen arabera, hau metateorema bat bihurtu da beste teorema batzuei aplikatzen zaien teorema, metateorema honek esaten du zer egin jadanik ezagunak diren beste teorema batzuekin, honela egintza sinpleak abiapuntu hartuz gero egintza konplexuagoak frogatzera iritsiko gara.

Hemen azpimarratu behar dugu nahiz ta ordenagailuek segunduko milloika inferentzia egin ahal izan (MIPS-en neurritua) ez eraginkorra izan daitezkeela urrats sinple ta uniforme bidez eratorpen konplexuak egin nahi badira, leherketa konbinatorioaren erruz. Horregatik dagokion metateorema erabili beharko da objektu-mailako frogak murrizteko.

Frogapen Automatikoaren arloen eremuan lehenengoz erakutsi zen metateoriarenganako interesa AA-ren arlo guztietara zabaldu zen, horien artean Jakintza Erreprezentaziora, Arazo Ebazpena (Problem Solving), Sistema Adituetara, eta Datu-base Deduktiboe-tara (Deductive Databases).

2.2.- JAKINTZAREN EREERREPRESENTAZIORAKO FORMALISMOAK.

Jarraian formalismo bakoitzaren abantailak eta eragozpenak berrikusiko ditugu.

2.2.1.- Egoeren espazioa.

Ez da, berez, jakintzaren errepresentazioaren eskema bat, arazoen egoera bakoitzean eman ahal diren alternatibengatik errepresentatzen du arazoa.

Arazoaren transizio guztiak aztertzen saiatzen da, honek leherketa konbinatoriora eramaten du. Hartutako soluzioa egoera bakoitzari asoziatuta dauden alternatiben kopurua mugatzea izan da, helburua lortzeko hoberenak direnak aukeratzen saiatuz; horretarako programak arrazonatu beharko du sistemaren barnean kodetuta dagoen arazoari buruzko informazio ugariarekin jakintza-ren errepresentazio eskemaren baten bitartez. Lehen jakintza errepresentazio eskemak, beste arrazoi batzuen artean, egoera espazioetako leherketa konbinatoriaren ondorioak kontrolatzeko baharrizanetik sortzen dira.

2.2.2.- Errerepresentazio eskema deklaragarriak.

Eskema deklaragarrietan dominio partikular bati buruzko informazioaren gehiengoa egintza estatikoen multzo bat bezala errepresentatzen da, multzo hau prozedura edo metodo orokor batzuekin doa edozein egintza multzo manipulatzeko. Egintzak proposamen baliogarriak edo dominio bateko gertaerak dira. Honela

esplizitu geratzen da errepresentazioaren osagai epistemologikoa eta heuristikoren arteko bereiztea, hau da, dominio konkretu bat deskribatzen duten egintzen multzoaren artean eta egintza horiek manipulatzeko dituen prozeduren multzoaren artean. Lan sesio batean zehar sistemak prozedura orokorrak dominioaren egintza bereziei aplikatzen dizkio inferentziak egiteko. Bere ezaugarriak hauexek dira:

- Egintza bakoitza behin baino ez da gordetzen eta bere esanahia bere erabilerarengandik independientea da.
- Egintza berriak eranstea oso erreza da, jadanik badauden egintzak eta prozedurak aldatzeko beharrik gabe.

ESKEMA LOGIKOAK.

Logika errepresentazio mekanismo bat bezala erabiltzearen ideia Lull, Leibniz, Frege eta beste batzuk iragarri zuten, nahiz eta beste era batera; eta AA-ri erantsia Robinson-en [ROBI65] EBAZPEN printzipioa automatizatu ondoren. Ordutik ona oinarritzko garrantzia izan du eta etengabeko eztabaida iturri bihurtu da.

Eskema hauetan egintzak sistema logikoren bateko formula logikoak bezala aurkezten dira (Lehen ordeneko logika, Multibaliotua, Tenporala, Fuzzy, etab.

Logika motako errepresentazio eskemak arazo batzuen alde epistemologikoa miatzeko oso aproposak dira, jarraiko arrazoi hauengatik:

- 1) Naturaltasuna eta sinpletasuna.
- 2) Zehaztasuna, adierazpen baten esanahiaren mugapen zehatza metodo logiko deduktiboen bidez.
- 3) Malgutasuna, JB-aren aldarazte erreza eta funtsezkoa, formularen sarrera edo deuseztearen bidez.

- 4) Modularitatea.
- 5) Egokitasun adierazkorra, jakintza partzialaren erre-presentazio oso maila handia onartzen du.
- 6) Jakintza eta metajakintza integratzea haizutzen du.

Errepresentazioa eta tratamenduaren arteko desberdintasun hau da lehen ordeneko logiketan oinarrituta dauden eskemen eragozpenik nagusiena, ezin dutelako alde heuristikoa bete betean konpondu arrazoi hauexegatik:

- 1) Antolaketa metodoen eza JB-aren egintzen artean.
- 2) Primitibak hain sinpleak direnez ez dute uzten mundu errealaaren egitura konplexuak erreprestatzen.
- 3) Heuristikak espresatzeko zailtasunak. Hau programazio logiko leguaiekin konpontzen da edozein item deklarati-boaren interpretazio prozedurala egiten dutelako.
- 4) Funtsa eta osotasuna ez dira beti desiragarriak konputazio kostu handiagatik.

Laburtuz logika jakintza aztertzeke egokia da baina ez erreprestatzeko, hala ere badira autore batzuk arazoa ez dagoela logikan baizik eta arazoak sortzen dituen egoeren espazioa murrizteko informazio faltan dagoela argumentatzen dutenak [MOOR82] [KOWA79].

Errepresentazio eskema ez logikoak egokitasun heuristikoaren gehipena ematen dute egokitasun epistemologikoa galtzeaz. Orokorrean arrazonamendu ez monotonoak egiten dituzte. Inferentziak zerbait frogatzeko ahalmen edo ezintazunean oinarrituta daude kontzeptu hauek erabiliz [WINO80]:

- Mundu itxiaren hipotesia.
- Inferentziak aktibatzen diren ordena.
- Ondoriorren bat frogatzeko baliabide mugatuak esleitzen zaizkien inferentzi prozesu baten hutsa.

Eskema ez logikoen bidez espresatutako edozein jakintza item logikoaren bidez errepresenta daiteke (logika ez beti klasikoa). Honek semantika homogeneo bat bermatzen du formalismo hauetarako nahiz eta logikaren bezain mugatua ez izan [HAYE79] [WINO80] [MORA87].

SARE SEMANTIKOAK.

Inferentziak egiten diren ordena sarearen NODOekin konektatzen duten ARKU mota desberdinen existentsiak erregulatzen ditu; arku mota batzuetan oinarritutako inferentziak burutu baino lehen lehentasun handiagoa duten motako arkuek parte hartzen duten inferentziak burutzea egiten saiatzen da. Orden finko honen existentzia da desberdintzen dituen sare semantikoaren inferentzi mekanismoak ohizko mekanismo logikoengandik, eta arrazonamendu metodo hauen ez osotasunean datza sare semantikoaren gaitasuna modu funtsezkoan operatzeko ezosoko informazioarekin (adibidez, propietateen herentzian salbuespenak). Sistema batzuk, HAUTAPEN FUNTZIOAK deritzoten prozeduren jakintza heuristikoa ere aprobetxatzen dute, funtzio hauek aholkatzen dute sareko zein elementu konparatu lehendabizi eta nola konparatu hautatutako elementua. Hautapen funtzio hauek sare semantikoan zatidurak sortzen dituzte, elementuen bilakuntza exhaustiboa ekidinez. Honela, sistemak era zuzenean jokatzen du, hau ezaugarri estimagarria da informazioa jakintza-basean oso haundia izatera heltzen denean.

Semantika formal baten eza eta terminologia standard baten eza arkuak etiketatzeko gaingidituta geratu zen Ritchie-n [ritc83] lanarekin. Hala ere, JB handien manipulazio zaila sare semantikoaren arazo nagusietariko bat izaten kjarraitzen du.

2.2.3.- Prozeduren errepresetazio eskemak.

Jakintza metodo edo prozedura multzo bat bezala errepresen-
tatzen da, honekin dominio bat deskribatu daitekeen. Eskema hauek
hauen errepresentazioa errazten dute:

- 1) Gauzak nola egiteari buruzko informazioa.
- 2) Eskema deklaragarrien gehiengoari egokitzen ez zaiz-
kion informazioak. Adibidez, defektuzko arrazonamendua
edo arrazonamendu probabilistikoa.
- 3) Erregela heuristikoa, soluzio bilaketa espazioa
mugatzen.

EREPRESENTAZIO PROZEDURALA.

Lehenengo sistema prozeduralen abantaila arazoaren ebazpene-
rako estrategia edo inferentzia espezialitatutako prozedurek
zuzenduak izatean zegoen. Honela eskema deklaragarrietako tipikoa
den jokaera ez eraginkorra gaingitzen zuten JB-en erabilera eta
bilaketa exhaustiboa ekiditen zutelako. Baina errepresentazio mota
honen berezko izateak bere eragozpen iturria sortu zuen.

- Sistemen konplexutasuna haundiagotzean eguneratze eta
ulermen zailtasuna hazi egiten da.
- Sistemen gehiengoak erabiltzen dituzten inferentzi
erregelen berezitasunagatik ez dira ez osokoak ez
funtsezkoak, ezaugarriak hauek batzutan hobe izaten
dela sakrifikatzea eraginkortasun konputazional
handiago baten mesederako.

Prozedural motako jakintzaren errepresentazio eskemak
sailkatu daitezke [MYLO80] kontzeptu hauengatik:

- 1) Prozeduren aktibazio mekanismoak eta
- 2) Eskema batek eskeiniko kontrol egiturak.

Aktibazio mekanismoak erabiltzen dira zein egoeretan aktibatu behar den prozedura adierazteko (Pattern-Directed Invocation). Kontrol nozioak prozedura batetik bestearenganako kontrolaren transferentziaren kudeaketari dagokio.

Gaur egun saiatu behar gara sistema deklaratiboen modularitate eta prozeduralen soluzio bilkuntza konpaginatzeko, azken hauetan kontrol informazioaren errepresentazioa hobetzeaz gain.

PRODUKZIO SISTEMAK.

Produkzio Sistemak (PS) hauexek osatzen dute: Erregela Basea (ERB), Egintza Basea (EGB), Inferentzi Motorra (IM) edo bereiztutako interpretea.

PS-k Erregeletan Oinarritutako Sistemak (EOS) sortarazi dituzte, hauen artean nabararazi daiteke PROLOG programazio logikoko lengoaia.

Erregela aproposa erabaki eta aktibatzeke IM erabiltzen duen hautaketa-exekuzio zikloan iragazketa prozesua da baliabide konputazionalen gehiengoa erabiltzen duena. Honela, mota honetako errepresentazioen egokitasun heuristikoen gehipen batek ERB-k eta EGB-k datu egitura landuagoak bihurtzea eragiten du une bakoitzean zein erregela aplikatu behar den errazago erabakitzeke. Horretarako Forgy [FORG82], Woods [WOOD83], Quinlan [QUIN85] eta beste batzuk proposatzen dute erregelen baldintzen aldeak antolatzea taxonomi egitura batzuetan, eta taxonomi egitura hauetan sistemak ezagutzen dituen objektu eta egoera guztiak errepresentatuta daude. Taxonomi hauen ezaugarri bat informazioa aplikagarritasun maila orokorrean gorde daitekeela bait da, informazio honetara zeharkako era batez heldu daiteke kontzeptu berezienetatik beharrezkoa denean. Eraginkortasuna hobetzeko teknika jakintza egitura faktorizatua erabiltzea da (factored knowledge structure) non erregela desberdinetan agertzen diren kontzeptu berdinak behin baino frogatu ez daitezen biltzen dira.

Egitura hauekin erregela asko frogatu daitezke, banan banan kontutan hartu gabe.

Abantaila batzuei egotzi zaizkio erregelatan oinarritutako fokatzeari:

- a) Modularitatea.
- b) Uniformetasuna, interpretazioa errazten duena.
- c) Aldaketarako erreztasuna, modularitateagatik ematen dena.
- d) Beste dominio baterako egokitasuna produkzio erregelak soil-soilik aldatuz, baina bakarrik bilatze estrategia arazoaren dominio berriarentzako egokia izanez gero.
- e) Naturaltasuna. Adituek erregela heuristikoak erabiltzeko joera dute euren arazoaren burutze teknikak adierazteko, erregela hauek produkzio erregelak bezala kodetu litezke. Honek iradokitzen du produkzio erregelak sistema adituen eraikitzerako metodo egokiak izan litezkeela.
- f) Zihurra ez den jakintza egokitzeko aukera premisei, ondorioei eta erregelari berari jarritako koefizienteen bitartez.
- g) Azalpena. Produkzio sistemak euren ondorioak azal ditzakete. Horretarako egintza basearen aldaketak eta bere ondorioak erregistratu egin behar dira.
- h) Jarrera adimentsua. Egintza basea edo kontestua aldatzen den bakoitzean sistemaren portaera aldatzen da.
- i) Von Neumann-en arkitekturaren ondoan desberdinak diren arkitektura berriak erabiltzeko aukera ordenagailuetan paralelismoa uztatzeko.

Jarraiko desabantaila hauek aukeratu dira:

- a) Erregelen bidez adierazitako jakintza ez da beti beste edozein eratan adierazitakoa baino errezago ulertzen, hau da jakintza kontestual guztia erregela bakoitzean

baldintza aldean esplizitoki aldarrikatua izan behar duelako.

- b) Erregelen hautaketa eta bilaketa estrategia ez dira normalean lan erreza izaten.
- c) Produkzio sistema gehienek ez dituzte erregelak onartzen era honetan:

*Edozein X erako (X) baldin bada
orduan A(X)*

Erregela hauek hartzen dituzten sistemek *ad hoc* moduan egiten dute (Honen esepzio bat PROLOG da).

- d) Erregeletan oinarritutako sistema gehienak ez dira ondo definitutako semantika batean oinarritzen (PROLOG da berriz esepzioa). Honek arrazoitu ezin den eta batzutan aurreazalezina den jarrera bat sortzen du.
- e) Sistema gehienetan, bilaketa estrategia kodean sartuta dago. Eta honek aplikazio esparrua mugatu egiten du. Hau murriztu daiteke bilaketa strategiaren ezaugarriak esplizitoak izan daitezen eginez eta erabiltzaileari ezaugarri horiek hautatzeko aukera emanez.
- f) Eraginkortasun eza. Modularitatea eta uniformitate handiak exekuzioaren kostua (overhead) gehitu egiten dute. Adibidez ezin daiteke zikloa baino arrazonamendu pausu luzeagorik eman, beharrezkoa eta posible izan daitekeenean.
- g) Opazitatea. Kontrolaren jarioa jarraitzea zaila da arazoen ebazpenean. Egoera-ekintza jakintza modu naturalean adierazi arren, ez da gauza bera gertatzen jakintza algoritmikoarekin, horri erregelen independentziak eta euren uniformitateak laguntzen diote.
- h) Unibertsalitate eza. Erregelak bere kontestu zuzenean baino ez dira baliagarriak.
- i) Erregela multzo baten osotasun eta funtsaren egiaztapenaren metodologia baten eza. Arlo hau ikerketa foko eraginkor bat da bere benetako erabilpenetan duen inpaktoagatik [DOYL82].

Produkzio sistemak EOS-k sortu dituzte, eta hauen artean PROLOG programazio logikorako lengoaia nabarmentzea merezi du.

Aurrerago produkzio erregeletan oinarritutako fokatzea beste jakintzaren prozesamendu metodo batzuekin nola erlazionatzen den ikusiko dugu.

2.2.4.- Errepresentazio eskema mixtoak.

Errepresentazio batzu eta beste batzuen erabileraren aldekoen arteko eztabaidak AA-ren munduan eztabaida garrantzitsua sortu zuen eta honek jakintzaren errepresentazioaren garrantzia, arlo honetako lanetan, ezartzen lagundu zuen. Estabaida hau ez dago itxita oraindik, baizik eta joera integratzaile batek gainditu du zeren eta dominiorik gehien jakintza errepresentatzeko eskema mota biak bait dira beharrezkoak. Horrela, praktikan, errepresentazio gehienak erabiltzen dute, bai joera bien konbinazioa bai frame-ak eta script-ak bezalako errepresentazio eskemak, jadanik informazio mota biak erantsita daramatenak.

Errepresentazio deklaragarriak prozeduralaren ondoan hurrengo abantaila hauek dituzte [WINO75]:

- 1) Malgutasuna eta ekonomia. Egintza behin baino ez da errepresentatzen eta erabiltzen duten prozedurek kasu bakoitzean gehiago komeni den eran egingo dute.
- 2) Ulergarritasuna eta modularitatea. Errepresentazio eskema deklaragarri baten sinpletasunak JB-aren ulermena eta bere eguneratzea modularitatea errazten du.

Prozedurazko errepresentazioak, ordea, uzten dute:

- 1) Dominioaren jakintza heuristiko berezia erabiltzen.

- 2) METAJAKINTZA erabiltzeko aukera beste jakintza nola eta noiz erabili adierazteko. Posiblea da eskema deklaragarrietan, baina oso zaila arrazonamendu partikular batean kontestutik kanpo.

FRAME-ak ETA SCRIPT-ak.

Gehien garatu eta erabili diren errepresentazio eskemak dira. Informazio prozedurala eta deklaragarria dituzte barne. Objektu prototipikoa edo egoeraren bati buruzko jakintzak errepresentatzen dituzten datu egitura konplexuak dira, zeinei honi buruzko informazioa gehitzen zaien:

- nola interpretatu eta erabili bere edukia;
- bere espektatibak;
- espektatibak huts egiten duten kasuak nola tratatu.

Formalismo biak inferentzia edo oroimena errazteko eta arreta zuzenbideratzeko jakintzaren errepresentazio eskema bat antolatzeke metodoei dagozkie.

Frame-en erabilera, Minsky-k 1975.ean proposatukoa, abantaila hauek eskeintzen ditu:

- Objektu eta egoera berriei buruzko informazioa inferitzeko erabilgarriak dira, haiei arreta exhaustiboa jarri gabe. Arrazoi hauengatik:
 - * Objektu bat frame bati dagokiola behin konprobatuz gero bere slot guztien informazioak objektua aipatzen dutela ulertzen da.
 - * Objektuarekin harremanetan dauden beste frame batzuk dauzkaten slot-ak badaude, beste slot-ak bere osagarrien identifikazioa zuzentzeko erabil daitezke.

- Objektu baten errepresentazio naturala haizutzen dute identifikatzen duten ezaugarrien definizioaren bidez.
- Jakintzaren jerarkizazioa haizutzen dute.
- Defektuzko balioak erabiltzeko aukera.
- Berezkotasun orokorrak errepresenta daitezke.
- Bai jakintza prozedurala bai deklaragarria erabil daitezke.
- Arrazonamendu analogikoa errazten dute.

Jarraiko eragozpenak ikusten dira:

- Ez dago frame-en teoria formal bat, horregatik funtsaren egiaztapena eta inferentzia ez dira ondo mugatutako semantika baten oinarritzen.
- Zihurgabetasunezko arrazonamendua *ad hoc*-eko adierazpenetan oinarritzen da.

Script-ak Schank eta Abelson-ek garatuta [SCHA75] testuinguru berezi batean gertaera sekuentziak errepresentatzeko bereziki diseinatu diren egiturak dira. Izan daitezke defektuzkoak, arruntak, apartekoak edo erratuak. Script bat, frame bat bezala, slot multzo bat da, slot hauek eskema informazio berezigarria errepresentatzen dituzte, eta gainera slot bakoitzaren balioak erabakitzen duten egintza serie bat.

Sekuentzia bakoitzeko egiturak sendoki elkarkonektatuta daude, sekuentzia bateko slot baten balioak beste sekuentzietako slot-engan eragina du.

Frame bat eta script baten arteko desberdintasuna azken honen espezializazioan datza.

Frame edo script-en sistema batzuetan alderdi prozedurala (slot-ei lotutako aktibazio prozedurak) mekanismo nagusia da inferentzia prozedura gidatzeko.

Frame-ei orientatutako jakintzaren errepresentazio lengoai batean bada inferentzia multzo bat automatikoki exekutatzen dena eta sistemaren jakintza esplizitua haundiagotzen haizutzen duena. Inferentzi hauek frame-en eta taxonomien berezkotasun estrukturaletan oinarrituta daude [FIKE85]. Inferentzia metodoen artean honako hauek aipa daitezke [HAYE79a]:

- 1) Instantziazioa: kontzeptu bat errepresentatzen duen frame bat eman eta gero, frame honen instantzia bat sortarazten saiatzen gara bere slot-ei balio egokiak esleituz lehenago ikusitako mekanismoak erabiliz (herentzia, prozeduren erabilera, etab...).
- 2) Erizpidetasuna (criteriality): kontzeptu bat errepre-sentatzen duen frame baten slot guztientzako balioak aurki badaitezke, orduan kontzeptu horren instantzia bat esistitzen deneko zihurtasuna dago.
- 3) Iragazketa (matching): kontzeptu baten instantzia bat eman eta gero, baliteke beste kontzeptu desberdin baten instantzia ere izatea.

Orain arte ikusi da frame-ak gaitasun handia dutela objektuak deskribatzeko, zeinei buruz arrazonatuko duten, eta automatikoki exekutatzen dutela objektuen deskripzioei buruzko inferentzia multzo bat. Gainera, frame-ak produkzio erregelak errepresentatzeko erabil daitezke, erregela hauek JOS eraikuntza prozesu batean zehar adituak hornituko ditu [FIKE85]. Erregela bakoitza frame bat bezala errepresentatuta egotean, erregelak klaseetan erraz bildu daitezke eta hauen deskripzioak erabiltzaileak edo/eta diseinatzaileak nahi dituen beste slot zein atributu arbitrario izan ditzake (erabakitako erregelaren erregistroa, arazoak zeinetan erabilgarria den, erabiltzailearen formatuko erregelak, etab.).

Sistema baten produkzio erregelen kopurua hazten den neurrian zailagoa egiten da erregelen eta berauen portaeraren arteko interrelazioa ulertzea. Egoera honetan desiragarria da erregelak modulo txikietan eta erraz maneiatu daitezkeenetan

antolatzea. Frame-etan oinarritutako errepresentazio lengoi batek haizutzen du erregelak klase batean biltzea eta klase hau asoziatzea objektuen (zeinen gainean erregelak aritzen diren) klasea errepresentatzen duen frame-rekin. Erregelak talde gisa deitzen dira sistemak diagnostiko ataza betetzen duen bakoitzean klasearen edozein objektu partikilarraren gainean. Erregelak klaseetan biltzeko era honek sistemaren metaerregelen multzoa osatzen du.

Horrela, frame motako errepresentazio eskema baten jakintza prozeduralaren zati bat errepresentatzeko produkzio erregelak integratzean, errepresentazio teknika bien abantailak konbinatzen dituen sistema bat lortzen da. Alde honetan Godo et al-en lana nabarmena da [GODO89].

Erregela bakoitzeko barne errepresentazioak aplikatzen den kontestuari buruzko informazioa dauka, bere izena, zein modulotan agertzen den, arrazonamendu estrategien azalpen eraginkorragoaren garapena haizutu duen kontrol informazioa, etab.

Egintza baten barne informazioak kontestuari buruzko informazioa dauka, bere izena, zein modulotan erabiltzen den, baldintza gisa agertzen den erregelak, egintza bakoitzaren ebaluazio eraginkorragoa haizutzen duen kontrol informazioa, etab.

Frame-ak bezalako egituretan oinarritutako beste antolaketa mota bat, kategoriak errepresentatzen haizutzen duten sumarioek edo laburpenek osatzen dute. Egitura hauek objektu sinpleen taldeketa errazten dute entitate abstraktoen inguruan, batzutan prototipoak deitutakoak, hauek objektu bat kategorietako kidea den ala ez erabakitzea haizutzen duen mekanismo baten lagunduta daude ordenagailuan [PLA287].

ERREPRESENTAZIO ESKEMA ANALOGIKOAK EDO ZUZENAK.

Errepresentazio analogikoa edo zuzena Sloman-ek [SLOM71] definitu zuen jakintzaren errepresentazio eskema bat bezala, eta errepresentazio eskema honetan errepresentazioaren zatien arteko harremanak eta berezkotasunak deskribatzen dituzte, berezkotasunak eta harremanak errepresentatua den konfigurazio konplexu baten zatien artean, errepresentazioaren egiturak errepresentatua denaren egiturari buruzko informazioa ematen du.

Ikerketa arlo batzuk eskatutako jakintzaren errepresentazio zuzenak iradokitzen dute, dominio horietako arazoen batzuk ebazteko. Baina errepresentazio zuzen berau, oso erabilgarria arazo batzuetan, ez da egokia dominio beraren barruko beste arazo mugatu batzuk ebazteko. Bigarren talde hau ebazteko naturala da dominioaren errepresentazio zuzen bigarren baten erabilera [FURB84].

Eraren batean, errepresentazio bi hauek (edo gehiago) haien artean trukagarriak dira: eskema bat sakon ezagutuko balitz, bestea eraiki ahal izango litzateke. Honela, dominioaren jakintzaren alde aurretik pentsatutako erabilera da eskema bat bestearen ganetik gomendatzen duena.

Eskema hauen abantailen artean seinalatu daitezke:

- 1) Errepresentazio mota honen eguneratutasuna, dominio errealeko aldaketak isladatzeko, erraza da egintza bat aldaraztean zerbait harreman inplizitoki oraingotuak geratu ahal dira eta.
- 2) Soluzioen bilaketa errazten dute mugapenak errealitatean errepresentazioari aplikatzen zaizkion transformazio motatan mugapenen bidez errepresentatuak dira eta. Honela baliogarriak ez diren estrategiak berehala atzera botatzen dira.
- 3) Beste eskema batzuetan, egoera baten errepresentazio berberak zenbait eredu onartu ahal ditu (hau da,

dominio errealararen egoera batzuk eskema beraren bidez errepresenta daitezke, egoera hauek desberdintzen dituen aldeak ez bait daude errepresentazioan isladatuak). Alderantziz, jakintza zuzenaren errepresentazio eskema bat exhaustiboagoa eta bereziagoa da eta honekin eredu gutxiago onartzen ditu eta arazoen ebazpena eraginkorragoa egiten du.

Eragozpenen artean nabarmen daitezke:

- 1) Eskema berezienganako joera berak eraginkortasun eza aurkezten du arazoen ebazpenaren zerbait prozesutako orokortasunaren aurrean.
- 2) Informazio ez osoren zenbait mota zailak dira erreprestatzen.

2.3.- JAKINTZAREN MANIPULAZIORAKO FOKATZEAK.

Software-a informazioaren manipulazioa deskribatzen duen informazio dela esan daiteke. Fokatze bik jarrai daitezke software sistemen edo software bidezko informazioaren manipulaziorako tresnen eraikitzean:

- 1) Fokatze funtzionala edo prozeduretara orientatutako programazio (POP).
- 2) Objektuetara orientatutako fokatzeara edo objektuetara orientatutako programazio (OOP).

2.3.1.- Prozeduretara orientatutako programazio.

Tradizioz ulertu izan da software sistema bat honela osatzen dela:

- Informazioen bat errepresentatzen duten datuen kolekzioa eta
- Datuak manipulatzeko dituzten prozedura multzo bat.

Prozedurak software-aren batasuna osatzen du.

Sistemak bere helburuak lortzen ditu jarraiko prozedurei dei eginez sistema honek erabili behar dituen hasierako datuekin batera. Honela datuek errepresentatutako entitate batzuen artean harremanak ezartzea lortzen da.

Fokatze honetan sortzen den arazo bat datuak eta prozedurak independienteak izango balira bezala tratatzen direla da, izan ere ez direnean. Prozedurak zuzen funtziona dezan beti datuak eta prozedurak modu egokian aukeratu beharko dira, bestela prozedurak manipulatzeko dituen datuak espero direnetatik guztiz desberdiank izan daitezke; gainera, hautaketa hau beti programatzaileak egin beharko du.

2.3.2.- Objektuetara Orientatutako Programazioa.

Programazio teknika berri hau agertu zen batez ere Smalltalk sistemaren garapenarekin Xerox Learning Research Group-ek eginda Allan Kay-k zuzenduta.

Aurreko entitate mota bien ordean informazioa eta bere manipulazioa era independentean errepresentatzen dutena, objektuetara orientatutako sistema batek entitate mota bakar bat du, objektua bera, biak errepresentatzen dituena. Objektuak

datuak bezala manipula daitezke. Beraz, prozedurak bezala objektuek manipulazioak ere deskribatzen dituzte. Informazioa manipulatzeko da informazioa errepresentatzen duen objektuari mezuak bidaltzearen bidez.

Objektu bat informazio pakete bat da, eta informazio hau manipulatzeko deskripzio batzuk.

Mezua objektu baten manipulazio era baten espezifikazioa da.

Objektu batek mezu bat jasotzen duenean nola automanipulatu erabakitzen du, objektu honi mezuaren HARTZAILEA deitzen zaio. Mezu batek izen sinboliko bat dauka barnean, nahi den manipulazio mota deskribatzen duena HAUTATZAILE izenekoa. Mezuen ezaugarri erabakiorra hautatzailea izen bat baino ez dela, programatzaileak egin nahi duena seinalatzen duena, baina ez nola egin behar den. Mezuaren hartzaileak oraingo manipulazioa nola exekutatu behar deneko deskripzioa dauka. Objektura orientatutako sistema baten programatzaileak mezu bat bidaltzen du manipulazio bati deitzeko, prozedura bati deitu beharrean. Mezu batek manipulazioa izendatzen du baina ez du egin behar dena zehazten, hartzaileak modu desberdinetan interpreta bait dezake mezuak zer egin behar duen sinalatzen duena dena; prozedura bat ere bere izenetik deitu eta identifikatzen da, baina prozedura bakar bat baino ez dago izen bakoitzerako, horregatik izenak prozedura zehatza eta honek egin behar duena espezifikatzen du.

Hautatzaileaz gain, mezu batek manipulazioan parte hatzen duten beste objektu batzu eduki ditzake, mezuaren ARGUMENTUAK deitutakoak dira sarrera datu bezala erabiliko dituenak mezuak aktibatutako prozedurak.

Objektuaren informazioaren manipulazio motaren deskripzioa mezu mota bati erantzuteko METODO izeneko entitate prozedural bat da. Metodo bat, prozedura bat bezala, prozesadore batek exekutatu behar dituen ekintzen sekuentziaren deskripzioa da. Hala ere, prozedurak ez bezala, metodo batek ezin dio beste metodo bati

zuzenean deitu, horren ordeztu mezu bat bidali behar du. Garrazitsua da metodoak ezin daitezkeela objektetatik banandu. Mezu bat bidaltzen denean hartzaileak erabakitzen du exekutatu behar den metodoa mezuaren hautatzaileen bidez.

Objektu batek erantzun ahal dion mezu multzoari bere PROTOKOLOA da. Objektu baten kanpoko itxura bere protokoloa baino ez da, barneko itxura prozeduretara orientatutako sistema baten antzekoa da. Objektu batek badauka atributu multzo bat beste objektuei buruzko informazioarekin, objektu hauek aldagai pribatuak deitutakoak [ROBS81], eta metodo multzo bat jasotzen diren mezuekin zer egin behar den deskribatzen dutenak. Aldagai pribatuak datuak bezala funtzionatzen dute eta metodoak prozedurak bezala. Datu bat prozeduraren arteko desberdinketa hau objektuaren barnean dago hertsiki aurkituta.

Metodoak, beste prozedura batzuek bezala, zuzenean manipulatzen dituzten datuen formatua ezagutu behar dute. Metodo batek manipula ditzakeen datuen zati bat bere objektuaren aldagai pribatuen balioak dira.

Mezu bat objektu bati bidali izan ahal zaio berari buruzko zerbait aurkitzeko, eta bere balioa mezuak itzulita izango da.

Objektu bat kapsula bat bezala funtzionatzeari, zerbitzu batzuk eskeiniz nola implementatzen diran ikusi ahal ez izanik, teknikoki ENKAPSUKAZIO [COX 87] izendatzen zaio.

Pascoe-ren arabera [PASC86] legoai batek objektuetara orientatutako programazioari eutsi ziezaion jarraiko ezaugarri hauek erakutsi beharko ditu:

- 1) Enkapsilazioa.
- 2) Datu abstrakzioa, enkapsulazioa erabiltzeko modu bat dena zeinekin programatzaileak definitu ahal duen datu abstraktu mota bat barne errepresentazio batek gehi datuen manipulaziorako eta atzipenerako prozedura

multzo batek osatzen dutena, hau da, objektu baten deskripzio orokorra (KLASE) objektu berarengandik (INSTANTZIA) banatzeko aukera.

- 3) Batasun edo lotura dinamikoak (Dinamic Binding), hautatzaile bera erabiltzeko aukera objektu desberdinei bidalitako mezuetan emaitza desberdinak itzultzen dituelarik hartzaileak aplikatutako prozeduraren arabera (polimorfismoa).
- 4) Jerarkikoki nagusigoak direnen informazioaren herentzia.

Klase bat objektu bat edo antzerako gehiagoen deskripzioa bezala defini daiteke.

Instantzia bat klase mugatu baten egindako deskripzio bati dagokion objektu bat da. Edozein objektu klaseren bateko instantzia da. Instantzia bakoitzak klase bateko beste instantzietatik desberdintzen dituen informazioa dauka, informazio hau klaseko aldagai pribatuen azpimultzo bat da eta aldagai instantziatuak osatzen dute. Klase bateko instantzia guztiak aldagai instantziatu kopuru bera daukate, baina balio desberdinekin. Objektu baten software-a bere klasean aurkitzen da, klase bateko instantzia guztiak metodo bera daukate mezu mota mugatu bati erantzuteko, instantzia batzuen arteko desberdintasuna dagozkien aldagai instantziatuengandik lortutako emaitza desberdina izango da. Objektu batzuen aldagai pribatu batzu klase batetako instantzia guztiak konpartitzen dute, eta klasearen aldagai izendatzen dira bertan kokaturik daudelako.

Programatzaileak, sistema berri baten garapenean, beharrezko klaseak sortzen ditu haien atributuekin aldagaiak eta metodoak dituztela. Metodo orokor edo aldagai bat hobeto deskribatzeko atributu bakoitzeko objektu bat sortu daiteke, hauen atributu berriak, FAZETA izendatuak, jatorrizko atributoari informazio desberdianak esleitu ahal ziezaion, hauek bezalakoak:

- bere balioa

- defektuz sortutako balioa
- prozedurak edo metodoak, hauek exekutatuko dira batez ere:
 - * Slot-aren balioa lortzeko
 - * Slot-ari informazio berria gehitzen zaionean
 - * Slot-aren informazioa deuseztatzen denean
- ahalezko balioaren rango bat
- galdera bat, erabiltzaileari datua eskatzeko
- etab.

Klase bat objektu bat da, hortaz, edozein objektu bezala manipulatuak izan ahalko dira mezuen bitartez, eta beharrezkoak dituen metodoak izango ditu:

- Sistema funtzionatu erazteko,
- Klase berrriak sortzeko,
- Instantzia berriak sortzeko, etab.

Herentziak objektuen artean informazioa konpartitzea haizutzen duen mekanismoa da, erredundantziak ekidinez. Sistema-ren batzu objektu guztien arteko herentzia aukera eskeintzen dute, baina asko daude klase artean baino onartzen ez dutenak. Azpiklase batek bere superklasetik heredatzen du. Azpiklase bati jarraikoaldaketa hauek egin ahal zaizkio:

- atributuak gehitu.
- superklaseak erabili ahal dituen mezu batzuetarako metodo berriak.
- mezu berrientzako metodoak, superklasearen protokoloa-renak ez direnak.
- klase aldagaien gehiketa.

Herentziak objektu klase berriak sortzea haizutzen du jadanik badagoen klasea eta klase berriaren arteko desberdintasunak espezifikatzen aldi bakoitzean, hasieratik hasi beharrean, kodearen zati handia berrrerabiliz.

OOP-ren erabilera POP-ren parean abantaila hauek eskeintzen ditu:

- Enkapsulazioak eta datuen abstrakzioak fidagarritasuna inkrementatzen dute eta laguntzen dute espezifikazio errepresentazionala eta prozedurala inplementazioan banatzen.
- Lotura dinamikoek malgutasuna inkrementatzen dute objektu klase berrien gehipena haizutuz dagoen kodeketa aldatu beharrik gabe.
- herentziak lotura dinamikoekin batera kodea berrerabilia izatea haizutzen du, honek programatzailearen produktibitatea gehitzen du eta mantenimendua erraztu.

Badaude ezaugarri batzuk zerbait autorek eragozpenak bezala ikusten dituztenak:

- Estabaidatuena lotura dinamikoen mekanismoaren exekuzio denbora kostua da. Mezuaren bidaltze batek prozedura bati dei zuzen batek baino denbora gehiago behar du, baina estudio batzuk erakutsi dute ondo inplementatuko mezu bidaltze mekanismo batekin kostu hau funtzio bati egiten zaion dei bat baino 1,75 aldiz handiagoa dela gutxigorabera [COX 83] [COX 84], hala ere exekuzio abiaduren desberdintasunak prozeduratara orientatutako lengoaien artean eta objektuetara orientatutako lengoaien artean ez dira oso esanguratsuak. Hau sinesgarriagoa da kontutan izanik mezuen bidaltzeak funtzio gehiago betetzen dituela funtzioari egindako sinple batek baino. Aplikazio handietan mezuen bidaltzearen funtzionalitateak aplikazioa baliokide konbentzional bat baino azkarragoa izatea lor dezake:
 - * Kodeketaren uniformitate eta modularitateagatik.
 - * Lotura dinamikoen itxura egiteko kodeketa konplexuaren behar izan eza.
- Inplementazio konplexuagoa.

- Programatzailearen aldetik prestakuntza handiagoa.

Bukatu baino lehen OOP-ko kontzeptu garrantzizko bi aipatu behar dira:

- Era askotako herentzia edo azpliklase batek superklase bat baino gehiagotatik heredatzeko aukera, kodearen konpartizioa inkrementatzen duena baina superklaseen arteko gatazkak sortzen dituen.
- Memoriaren kudeaketa automatikoa, oso erabilgarria da erreferentzia zenbaketa (reference counting) eta zaborren bilketa (garbage collection) programatzaileei haizutzen bait diete objektuek behar duten leku liberazioari buruzko xehetasunak kontutan ez eduki behar izatea, aplikazioa argiagoa egiten delarik eta sistema osoa fidagarriagoa delarik.

Beraz, bukatzeko, OOP-k abantaila handiak eskeintzen dizkio software-aren mantenu eta produkzioari; garapen denbora laburtzen du eta kode konpartizioaren maila handiagotzen du. Honek software sistema konplexuen eraikitzerako teknologia egokia bilakatzea egiten du.

2.4.- ADIMEN ARTIFIZIALARI DAGOKION KONTROLA.

Metajakintza edo kontrolaren jakintza AA-ean erabiltzen da era askotara:

- * Inferentzia kontrolatu soluziora heltzeko pausu posibleen ikerketa-zuhaitza murrizteko eta eraginkortasuna gehitzeko:

- Soluzioa lortzeko behar den jakintza edo erregelen aurkitzea eta aukeratzea gidatuz.
- Erregela eta dominioari buruzko informazioa ematen metaerregela estrategikoek beren eritziak egiteko.
- Azalpen gaitasuna handitzeko erregelak justifikatu.
- Erregela berrietan akatsak aurkitzen lagundu, JB-aren zabalera errazteko.
- Eta abar.
- * Dedukzio sistema automatizatueta frogarako estrategiak asmatu.
- * Jakintza errepresentatzen duten lengoaien adierzpenera gehitu. Adibidez, eskemek eta frame-ek objektuen errepresentazioari buruzko jakintza adierazten dute, funtzio errepresentazioari buruzko "funtzioaren txantiloak" (function templates), eta inferentzi erregelei buruzko erregela ereduak. Eskemek edo frame-ek eta erregela ereduak trebetasunaren pilaketa haizutzen dute, transferentzia zuzenaren bidez, giza adituarengandik sistemaren JB-ra. Eskemek eta frame-ek funtzio-txantiloekin batera mekanismo bat sortzen dute JB-aren mantentzearen aspektu batzu jasateko.

Hemendik aurrera lehenengo puntuan oinarrituko gara, ikerketa ahalegin handiko objektua bait da, puntu honek garrantzi handia du AA-ean programen adimen-gradua gehitzeko gaitasuna duelako.

Arazorik handienetarikoa AA-ren programazioan, kontrolaren espezifikazioaren zailtasuna da. Von Neumann ordenagailuek instrukzioak sekuentzialki exekutatzen dituzte, programadoreak kontrola programan espezifikatzen du. AA-ean produkzio-ordenagailu izenekoek programadoreak idatzitako erregelak exekutatzen dituzte, baina kontrola inferentzia mekanismoaren pean dago. Dena dela, kontrola programatua ahal izatea zilegi da.

Azken baten metajakintza beharrezkoa da AA programa baten jakintzaren erabilera gidatzeko eta saturazio arazoa konpontzen saiatzeko.

Saturazio arazo honetarako bi erantzun posible dago:

- 1.- Jakintza iturria hobeto enfokatu, erregela berezi batzuekin, beste estrategien bidez berreskuratzeko orduan leherketa konbinatorioa ekiditeko.
- 2.- Saturazio arazoa onartu eta estrategia bat bilatu sistema hori gidatzeko (saturazioa egon arren). Dirudienez saturazioa ekidinezina da eta gainera AA-ren arazoak txarto-egituratuak deituak dira, ez bait dago konponbide algoritmikorik; honek ez-determinismo gradu bat egongo dela esan nahi du, beraz saturazio posibilitatea ere. JB-aren tamainua hazten bada saturazio hau ere haziko da. Estrategia hau izan beharko litzateke: Erabili, "zati" bat baino gehiago aplikatu ahal denean, jarraian zein "zati" (chunk) erabiliko dugun jakiteko. Sistemaren errendimenduak erabaki honetan erabilitako adimenaren eragin handia edukiko du. Metajakintza esplizitoa, eskurakoia, ondo antolatua, eta orokortasun egokian kontsideratua izan behar du.

Puntu honetan AA-k normalean erabiltzen dituen kontrol arkitektura batzu nabarmendu behar ditugu:

- * Arkitektura erreflexiboak.
- * Mugapenetan (constraints) oinarrituriko arkitekturak.
- * Multimaila arkitekturak.

Lehenengoak erreflexio kontzeptuan oinarriturik daude, hau da hipotesirik egokiena aukeratzeko gaitasunean, orain tratatzen ari den hipotesia alferrikakoa bihurtu ondoren (honen frogan lortutako informazioa kontutan harturik), kontzeptu hau ez da

agertzen programazio lengoai klasikoetan eta beharrezkoa da gaur egungo AA-ren lengoai batzuen arkitekturak ulertzeko.

Azpimarratzekoa da introspeziozko arrazonamendua ere erreflexio mekanismoaren partikularizazio bat bezala.

Bigarrenek mugapen (constraint) kontzeptua dute oinarri, lortu nahi diren helburuak lortzen saiatzen dira, kasu honetan erabilgarria den jakintza bakarrik erabiliz, erabilgarriak diren informazioak betetzen dituztelako jarritako edo sortutako mugapenak, konputazionalki eraginkorra den helburua lortzerakoan atzera jotzea (backtracking) eta erreflexioa ekidinez.

Multimaila arkitekturak jakintza-ingeniriaren lehen ahalegina da lehen aipatutako ideiak praktikan jartzeko, oso erabilgarria izan dela erakutsi du.

2.4.1.- Erreflexioa.

Erreflexio hitza logikan erabilia izan da teoria eta metateoriaren arteko erlazioak adierazteko. Sistema konputazional erreflexiboa da bera bere domonioaren parte denean.

Sistema konputazional bat dominio zehatz batean aritzen da, beraz dominioaren errepresentazio bat du eta honetan lan egiten du. Bere dominioarekin kausalki erlazionatuta dagoela esaten da, sistemak daukan dominioaren errepresentazioak, dominioa aldatzean, funtzesko eragina duenean. Honela dominioaren errerepresentazioan gertatzen diren aldaketak beste aldaketa batzu sorteraizten dituzte dominio berean, horregatik:

- Sistemak berari buruzko errepresentazio egoki bat du beti.
- Sistemaren egoera eta konputazioa beti daude ados bere errepresentazioarekin.

Erreflexioa teknika guztiz praktikoa da, eta gaurko sistema gehienek erabiltzen dute, arazketarako informazioa gordetzeko, pausuz pausu ebaluazio erraztasunak emateko, interfaceak , eta abar. Kontrolaren ikuspuntutik sistema erreflexiboek, une batean sistemaren egoeraren arabera, errealitatetik hurbil, jarraian zein konputu egin behar den erabakiak definitzen haizutzen dute, hau da, kontrolari buruzko arrazonamendua.

Programzio lengoaiaren ikuspegitik, lengoai batek arkitektura erreflexiboa izango du baldintza hauek betetzen dituenean [MAES88]:

- 1) Lengoaiaren interpretatzaileak, exekuzio denboran bere aspektu zehatz batzuen errepresentazioa ematen du.
- 2) Interpretatzaileak bera eta bere errepresentazioaren arteko lotura kausala bermatzen du.

Sistema erreflexiboek ematen duten kontrol motak programek bi zati dituztela erakusten du: bata objektu aldea eta bestea alde erreflexiboa. Objektu mailaren eragina dominioari buruzko arazoak ebaztea da eta erreflexibo mailarena, objektuari buruzkoak, alde hau programatzaileak programatu ahal du.

Arkitektura erreflexiboen adibideak aurki daitezke: lengoai prozeduraletan, logikoetan oinarritutakoetan, eta objektuetara orientatutakoetan.

Prozeduraleen artean 3-LISP [SMIT82] nabarmendu ahal dugu. Honek funtzio erreflexiboen kontzeptua sartzen du, besteak bezalako funtzioak dira, gertatzen ari den konputazioari buruzko kalkuluak egiten dituela salbu etsiz.

Lengoai logikoek metateoria kontzeptua erabiltzen dute, FOL-en [WEYH80] kasuan bezala.

Hauek dira arkitektura erreflexiboaz egin diren erregelatan oinarritutako lengoaien adibide batzu: TEIRESIAS [DAVI82], SOAR

[RESE88], MILORD [SIER89]. Guztiek metaerregela kontzeptua hartzen dute, dominioari buruzko erregeletan aritzen diren erregelak dira. Zein erregela diren erabilgarriak, une batean, zehazteko.

OOL-en artean 3-KRS [MAES88] aipatu ahal dugu. Honetan, objektu bakoitza, objektuaren elementu batzu errepresentatzen dituen metaobjektu bati dago lotuta. Une zehatz batean objektu batek exekuzioa eten ahal du eta konputazio erreflexibo bat egin metaobjektuaren bidez.

Lengoai erreflexiboak bi eratan inplementa daitezke:

- 1) Erreflexio prozedurala. Sistemak berak daukan errepresentazioa, programak exekutatzeko legoaiaren interpretatzaileak ere erabiltzen duen errepresentazioa da, beraz ez da agertzen konexio kausalaren arazoa. Bere errepresentazioak bezala bi helburu ditu:
 - konputazio erreflexiborako datu gisa balio izate, eta horretarako sistemari buruzko JB on bat eskaini behar du.
 - eta era batera sistema inplementatzeko erabilia izatea, orduan eraginkorra izan behar du. Erabiltzeko zaila izan daiteke eskakizun hauek askotan kontraesankorrak dira eta.
- 2) Erreflexio deklaratioa. Programatzaileagandik hurbil kokatzen den errepresentazio bat sortuz, eta beste bat interpretatzailearentzat. Kasu honetan konexio kausala interpretatzaileak seguratu behar du.

Erreflexioa gertatzen den unea kontutan harturik, horrela izan daiteke:

- 1) Explizitua. Erreflexioa bakarrik gertatzen da programak explizitoki agintzen duenean, 3-LISP eta FOL-en adibideetan bezala.

- 2) Implizitoa. Interpretatzaileak sistematikoki aktibatzen du kalkulo erreflexiboa, TEIRESIAS, SOAR eta MILORD-en bezala.

Arkitektura erreflexiboek arkitektura tradizionalak baino kontrolerako mekanismo sofistikatuagoak definitzeko aukera ematen dute, orduan AA programa guztiek erreflezio ezaugarriak eduki behar dituzte eta honek, behar bada, programa tradizionaletatik esentzialki ezberdiduko ditu.

2.4.2.- Introspektzioa.

Arkitektura introspektziboak arkitektura erreflexiboen erakoak dira. Ezagutzen edo uste dugunari buruzko arrazonamenduan enfasia ipintzeagatik bereizten dira. Une batean eduki ez zen informazioaren lorpena kezka-iturria izan da beti JOS-en kontrolean. Horretarako heuristikak definitzen ziren, inferentzia posibleen bilaketa espazioa eraginkorki kontrolatzeko. Kontrolaren jakintza bi motatan bana daiteke:

- 1) Munduarekiko erlazioaren jakintza edo zein dominiori buruzko jakintza behar dugun jakiteko kontrola, non lortu, eta abar.
- 2) Jakintza nola aplikatu behar deneko jakintza, hau izan da AA-n gehien erabili izan den kontrolaren jakintza, sistemen eragiginkortasunean duen eragin zuzenagatik. Metaerregelak jakintza mota honetako adierazpen bat sortzen dute EOS-etarako.

Lehenengo motako jakintzan oinarriturik, introspektziozko arrazonamendua munduarekin duen harremana eta jakintza beraren edukiei buruzko arrazonamendua dela esan dezakegu [KONO88]. Bere helburua jakintza ezberdinen motak ulertzea da, uniformeki adieraziz; beraz, konparazioa ere posible eginez. Jakintza mota bakoitzak ez-monotonizitatea ezaugarri inportantea du erkide.

Ikerketa bide honek emaitza batzu lortu ditu, eta Reiter-en defektuzko teoriak sistema introspektibo zehatz baten formalki baliokideak direla erakusten dute [KONO88], era batera sistema introspektiboek daukatela: zirkunskripzio, eta erregelen hutsari dagozkion Prolog-eko ukapenarekin erlazioaren bat frogatu da.

Mundu itxiaren hipotesia erraz errepresenta daiteke honelako esaldi batez:

Ez batut $\emptyset$ ezagutzen, orduan $\neg\emptyset$ egia da

$\emptyset$ munduko sententzia zehatz bat izanez. Munduaren eta sistemaren jakintzaren gradua-ren arteko erlazioa sinalatzen du.

Defektuzko arrazonamenduaren kasurako, sistema introspektiboek beren errepresentazioa ere haizutzen dute. Horrela adibidez adierazpen honek:

Ugaztunek tipikoki lau hanka dute.

kontrako informaziorik ez badago, defektuzko arrazonamendu batek ugaztun zehatz batek lau hanka duela sines-erasiko ligukela erakusten du. Hau teknika introspektiboei ezker erraz adieraz daiteke adierazpen honekin:

X ugaztun bat bada eta nik ez dakit X-k lau hanka ez dituen orduan X-k lau hanka ditu.

Introspektzioa era sinple edo era konplexuaz egin daiteke.

INTROSPEKTZIO SINPLEA.

Munduari lotutako JB baten eskena klasikoari (galderak jasotzen eta erantzunak ematen dituen), meta-JB bat gehitzen zaio, eta honetan galdera-erantzuna operazioak gertatzen dira.

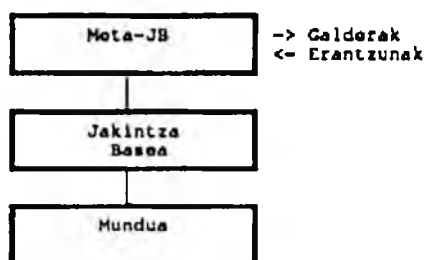


Fig. 2.1.- Introspektzio sinplea.

Meta-JB honek, lengoia baten bidez, JB-aren galdera-arantzun azalgarriak deskribatzen ditu. Lengoai sinpleenetariko bat $L\phi$ motako exprezioak dituen izan liteke, ϕ JB-aren elementu bat da eta $L\phi$ -k ϕ exprezioa JB-an sartuta dagoela esan nahi du. Galdera honi erantzuteko meta-JB-ak ϕ galdera efektiboki egin diezaike JB-ari eta JB-aren erantzunaren arabera

afirmatiboki edo negatiboki erantzun. JB-ak meta-JB-aren jokabidea guztiz mugatzen du. Dena dela sistemari arrazonamendu ahalmen berriak eman ahal dizkioten meta-JB-etan pentsatu ahal da, horrela JB baten jokabideari buruzko informazio gehiago edukitzen haizutuko diote. Honek frogen existentziari buruzko arrazonamendua posible egingo luke JB-an, zuzenean kontsultatu beharrean, edo $B(f(x))$ ("f", funtzio talde baten barruan bat izanez) motako predikatu baterako frogatuta dagoela aurkituko luke. Agian sistemarik ezagunena FOL [WEYH80] da.

JB-etan teknika introspektiboen bidez egindako hurbilketa kontrol deklarati boen osagaien definiziotik ezberdindu behar da. Nahiz eta biak beren exprezioak errepresentatzen saiatu, azken honek bilaketa eraginkortasunarekin zerikusi handiagoa du JB-an atera izan ahal den ala ez denarekin baino.

INTROSPEKTZIO KONPLEXUA.

Lehen ikusitakoa baino introspektzio mota interesgarriagoa da hau eta meta-JB-ak mundua eta JB eta munduaren arteko erlazioa ikusten duenean gertatzen da. Adibide honetan ikus dezakegunez:

$$\neg L\phi \Rightarrow \neg \phi$$

ϕ sententzia atomikoa da. Hauxe da exprezioak adierazten duena: ϕ atomo bat ez badago JB-an sartuta bere ukapena egiazkoa da. Lehenengo exprezioak JB-ari egiten diola erreferentzia ikus

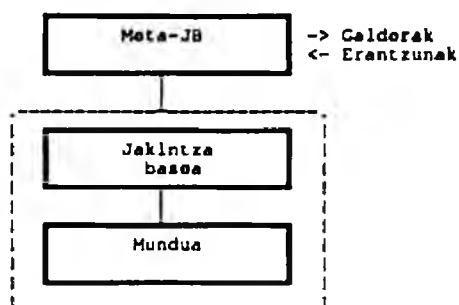


Fig. 2.2.- Introspezio konplexua.

dezakegu, baina bigarrenak erre-presentatu nahi den munduari egi-ten dio erreferentzia. Arrazona-mendu mota honetako izaera ez mo-tonikoa begi bistan dago.

2.4.3.- Mugapenak.

Montanari-ren definizioari segituz esan genezake, mugapen bat aldagai multzo bat eta aldagai hauen arteko erlazio bat dela [MONT74]. Sistema bat orokorki deskribatu ahal izateko mugapenak sare batean, edo mugapen grafo batean, kokatzen dira. Bilaketa espazioa murriztea eta kasuari dagokion jakintza bakarrik erabiltzea, atzera kotzea ekidinez edo egotziz, haizutzen dute; horrela eraginkortasun konputazionala hobetuz.

Literaturan sarrien aipatu den hurbilketa mugapenen lehengotariko sistema gisa Waltz-ena da [WALT72]. Hala ere, lehendabiziko lanak hirurogeigarrenen amaieran garatu ziren. R. Fikes-ek, adibidez, mugapenak erabili zituen beren sisteman prozedura gisa aurkeztutako arazoak ebazteko [FIKE70].

D. L. Waltz-ek (eta baita Stallman, Sussman [STAL77] y Stefik [STEF81a] [STEF81b] ere) mugapenak erabili ditu aplikazio berezi batean. 1980 arte ez da mugapen teoria bat eta mugapenen lengoai baterako interprete baten inplementazioa agertzen; une honetan G. L. Steele-ek [STEE80] mugapenetan oinarritutako programazio lengoai bat garatu zuen. Lengoi honek laguntza desberdinak eskeintzen ditu mugapenen definizio eta maneiorako, baina eragozpen bat dauka: ekuazio algebraikoetan oinarritutako mugapenak baino errepresentatu ezin izatea [GÜSG89b]. Beraz,

beren aplikagarritasuna laburra da AA-ren aplikazio askotan errepresentatutako eta prozesatutako mugapenak sinbolikoak bait dira. Hala ere, ekarpen historiko garrantzitsua da mugapen maneiurako.

Hirugarren kasu gisa, SA-ren implementazioan erabilitako lan ingurunetan integratuak izan ziren mugapenak berrikusiko ditugu. Honelako ingurunetan mugapenen formalismoak ez dira isolaturik inplementatu, baizik eta erabili behar dira beste formalismo batzuekin batera, erregelak edo frame-ak bezalakoak. Konbinazio hauen abantaila bat beste formalismo batzuetara aldatu ahal izatea da beartasunak somatzen direnean mugapen formalismoan.

Kontutan izatea merezi dute honako hauen lanek: E. C. Freuder-ena [FREU78], zeinek era bitar eta extensionalki² emandako mugapen sareek garatu zituen eta hauek orden altuagoko mugapenen sintetizazioz betetzen dira mugapen oinarritzko multzotik; eta Fikes-ena [FIKE70] mugapenen manipulazioen metodoetan datzana.

[STEE85]-n kontrol estrategia malguago bat sartzen da. Kontrol egitura inplizitua eduki beharrean, L. Steels-ek erabiltzaileak espezifika dizakeen kontrol egitura proposatzen du mugapenak espezialista edo aholkulari gisa ikusirik. Mugapen batek egoera berezi bati buruzko aholkua eman dezake, baina ezin dezake egoera zuzenean aldatu. Alderantziz, aholku baten ebaluazioa zuzendari (manager) batek egingo du, zeinen portaera erabiltzaileak defini dezake. Zuzendariek zeintzu mugapen kontsulta daitezkeen ere erabakitzen dute. Zuzendarien aholkulari gisa funtzionatzen dute goragoko maila batean. Orduan sistema orokorra "aditu komunikatzaileen jerarkikoki egiteratutako elkartea" [GÜSG89b] bezala ikus daiteke.

² Era exhaustibo eta osoan adieraziz mugapen posible guztiak.

Mugapen asko ez dira baliozkoak kontestu guztietan, klase honetako mugapenak mugapen ahulak edo mugapen heuristikoak deituak izan dira.

Erlazio nabaria dago mugapenen eta programazio logikoaren artean. Predikatuek haien argumentuen arteko aerlazioak adierazten dituzteneko egintzan dago oinarrituta. Hortaz, hauen aldagaien argumentuen ganean inposatutako mugapenak bezala ikus daitezke.

Argitasuna mantentzeko, programatu behar den ataza aspektu partzialak bakarrik deskribatu beharko lituzke mugapen batek. Orduan aldagaia konpartituz mugapen multzo batez osaturiko mugapen sare batek deskribatuko du ataza orokorra.

Mugapen sareak betetzeko oso erabilera handia duen teknika hedakuntza lokala da; mugapen batek kalkulatzeko dituen emaitzak (adibidez bere aldagaiek betetzen dituzten mugapenak) sareko ondoko mugapenen artean zabalduak izango dira. Hedakuntza lokalaren emaitza funtsezkotasun lokala da (adibidez aldagaientzako balio posibleen multzoa). Funtsezkotasun lokalak edo arku funtsezkotasunak [MACK84] hauxe esan nahi du: saretik mugapen bat eta bere aldagai baten estaldura hartzen badute, beste aldagaien estalduretan balioak aurki ditzakete (balio hauen tuplek mugapenak betetzen badute).

Sareko mugapen guztiak betetzen dituen aldagaiei balio ematea litzateke soluzio orokorki funtsezkoa. Mugapen-sare baten aldagai guztiak behin bakarrik lortzen badira, edo ez badago ziklorik sarean, aldagaiaren estaldura lokalki funtsezkoa eta mugapen-sarearen soluzio orokorki funtsezkoa berdina dira. Orduan, mugapenaren aplikazio askotarako, nahikoa da funtsezkotasun lokalaren kalkuloarekin. Hala ere, batzuetan, beharrezkoak dira mekanismo batzu funtsezkotasun orokorra lortzeko.

Mugapenak plangintzarako erabiltzen dira batez ere, eta interpreta daitezke honela [STEF81a]:

- Errregelen deusezteia.
- Deskripzio partzialak eta konpromezuak plangintzaren birfinketa ikuspegitik.
- Menpeko arazoen arteko elkarrekintzak adierazteko komunikabidea.

Eraginkortasun konputazionala erarik sinpleenetan hobetzeko helburuaz erabiltzen dira, jakintza deklaragarriaren bitartez:

- Mugapenak errepresentatuz sarean, zuhaitz itxuraz ahal bada, plangintza eta JB-ak jerarkizatuko dituen.
- Premiagabeko arrazonamenduei dagozkien sarearen atalak kontsideratu gabe, okerrak eta atzera kotze beharra ekiditeko erabaki hauek atzeratzen dira ahal dan bestean, honelako hurbilketak sortuz: Constraint Posting [STEF81a] eta Delayed Evaluation [ZWEB89].
- Atzera jotzea ekidinez edo deuseztuz [KENG89] [DECH88] [PURD83] [HARA80] [DOYL79] [BITN75], bere kostu konputazional altuagatik; erreflexioarekin kontraesanean.

Mugapenei buruzko hiru operazio bereiz ditzakegu arazoen erabilkuntza prozesuan jokutzen dituzten zeregin ezberdinak kontutan harturik:

- * Mugapenen formulazio (Constraint Formulation) edo betetzeko mugapen berrien adizioa xehetasun txikiagotik handiagora, hau da, zehazgabeki formulatutako arazo batekin hasi, gero eta zuzenago espezifikatu bihurtuko dena soluzioaren prozesuan zehar. Honen kontra, mugapen tradizionalen betetzean mugapen zenbaki tinko bat dago, aurretik zlagutua ("off-line" sistemak [DAVI87]).
- * Mugapenen hedakuntza (Constraint Propagación) edo mugapen berrien adizioa dauden mugapenetan finkaturik edo mugapen bat betetzen duten baloreek eta betetze

honen ondorioak gainerako sarean hedatzea. Ikusi [DAVI87] [GÜSG88].

- * Mugapenen Betetzea (Constraint Satisfaction) aldagai batentzako bere mugapenak beteko dituen baloreen kokatze-operazioa da. Aplikazio bereziengantik independente den lengoai bat aurkitu nahi da, eta era batera, iragazi teknika batzu erabiltzen haizu dezala. Teknika hauek izan nahi dute konputazionalki eraginkorrak konbinatori leherketa ekidinez eta iragazi balore multzoak hedatuz, euren artean Cartesiar biderketa kalkulatu gabe [GÜSG89a] [KENG89].

Sistema ugari egin dira Mugapenetan Oinarrituriko Arrazonamenduarekin (ROA): MOLGEN [STEF81a][STEF81b], R1 [MCDE82], ENVISION [KLEE84], SPAM [MCDE84] [DAVI87], CONSAT [GÜSG89b], eta abar.

2.4.4.- Multimaila arkitekturak.

Ohizko kontrol modu bat osatzen dute AA-ean erreflexio edo mugapen maila ezberdinekin gauzatuta, maila bakoitza beheko mailek bete behar dituen kontrol-politika mugatzen arduratzen da.

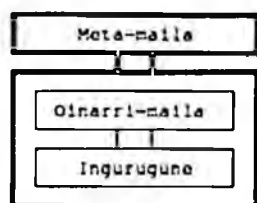


Fig. 2.3.- Meta-maila eta Oinarri-mailaren arteko erlatzioa.

AA programa gehienek atazak bi kategorietan bana daitezke:

- * Metamaila.
- * Oinarri-maila, objektu-maila edo jakintzaren iturria (KS) izeneko.

Oinarri-mailako egintzak elkarrekin exekutatzen direnean programaren helburuak betetzen dituztenak dira. Metamailako egintzak momentu zehatz baten zein oinarri-mailako egintzak exekutatu behar diren erabakitzen dituztenak dira. Adibidez,

blokeen munduan robot baten besoaren mugimendu fisiko guztiak oinarri-mailako egintzak izan litezke, eta plangintza guztia metamaila izan liteke. Dedukzio automatikoko sistema baten inferentzia pausuak oinarri-mailakoak izan litezke eta inferentzia pausuak exekutatzeko ordena erabaki behar duen egintza metamailakoa.

Bereizketa hau argi edukitzeko oinarri-maila eta metamaila agente bereziak bezala ikusi behar ditugu, 2.3 irudian ikus dezakegun bezala. Oinarri-mailak emandako ingurunea kontsultatu eta aldatu egiten du. Metamailak ingurune zabalagoan operatzen du (ingurune honetan oinarri-maila barne delarik). Bere ataza oinarri-maila astertzea da, egintza ideala zein den erabakitzea (exekutatu ondoren helburuak eraginkorki lortu ditzan) eta azkenik, egintza burutu ondoren, oinarri-maila aldatzea. Metamaila ere beste mailez osatua egon liteke, eta maila nagusiak menpeko mailak hartu behar duen egintza seinala lezake.

Multimaila egituraren, ideia nagusia lengoaiaren deklaragarriaren erabilera da, portaera deskribatzeko. Lengoaia honen barruan oinarri-mailaren egoerari buruzko sententziak, egintzak eta helburuak idatz ditzakegu; eta metamailak sententzia hauekin arrazonatu ahal du oinarri-mailak exekutatu behar duen egintza idealaren erabakian.

Kontrolaren lengoaiaren ezaugarri garrantzitsu bat portaeraren espezifikazio partziala haizutzen duela da. Honek sistemaren garapen inkrementala haizutuko du, batez ere kontrolaren sententziei dagokionez. Beste abantail bat AA egitura ezberdinen integrazioa da, adibidez "demons", OOP eta dedukzio kontrolatua portaeraren espezifikazio partziala adierazteko. Lengoaiaren ausentsiaz, beste erabaki arbitrarioak *ad hoc* hartu behar izango dituzte, egitura baten inplementazioan integrazioa ezinbidetuz.

Multimaila egituraren ideia ez da berria. McCarthy-k horrelako ideia bat deskribatu zuen 1959. urtean, ondoren beste ikertzaile batzuk jarraitu dute prozesu hau: Brown, Kowalski,

Davis, Doyle, Gallaire, Genesereth, Lasserre, Hayes, Sandewall, B. Smith, Warren, Weyhrauch eta abar. Hala ere arrazonamendu introspektibo explizitoak oso aplikazio serio gutxi aurkitu ditu.

2.5 .- KONTROLERAKO METAJAKINTZAREN INPLE- MENTAZIOA.

Metajakintza era ezberdinetan inplementa daiteke, orden historiko bat jarraituz, paradigma hauek hartu ahal ditugu kontutan:

- Kontrola PS-etan: Metaerregelak.
- Kontrola JB modulariztuetan.
- Kontrola arazoen ebazpenaren eredu kooperatiboetan.
- Kontrola denbora errealeko JOS-etan.

2.5.1.- Kontrola PS-etan: Metaerregelak.

Metaerregelak jakintzaren erabilera gidatzeko erabiltzen dira, estrategien adierazpenerako baliabide bat eskainiz.

Metaerregelez transmititutako informazioaren izaerari dagokionez puntu batzu hartu behar ditugu kontutan:

- * Beste erregela batzuei buruzko zerbait argitutzen duten erregelak dira. Jakintzari buruzko jakintza honek metaerregela bihurtzen ditu.
- * Metaerregelak oinarri-maila edo objektu-maila erregeletarako arrazonamendu ez zehatz garaituaren eredua nola erabili erakusten dute. Baina bi ideia hauek,

metajakintza eta arrazonamendu ez zehatza independien-
teak dira.

- * Metaerregela baten ondorioak erregela bateko **erabilgarritasunari** egin ahal dio erreferentzia eta ez bere **balioari** (zihurtasun-faktore nagusiagoa duten errege-lak exekutatuko dira lehengo). Nabaritu behar da, **erabilgarritasunari** buruzko informazioa oso ezberdina dela oinarri-mailako erregeletan premisako aurre-baldintzez erabakitua den **baliotik**. Metaerregela baten informazioa oinarri-maila erregeletatik ezberdina da eta oinarri-maila erregela batek daitekeen erabilgarritasunari buruzko abisu bat da.

METAEERREGELEN SINTAXIA.

TEIRESIAS-en, honek eragiten dituen beste sistema eta tresna batzuetan bezala, PERSONAL CONSULTANT PLUS bezala, metaerregilek erabilgarritasuna edo ordena partziala seinalatzeko balio dute, ikus 2.4 eta 2.5 irudiak.

```

BA

      A botetzen bada eta
      Badaude arauak X alpatzen {oz} dutenak
      | premisetan |
      | endorlootan |
      | biotan      |

ORDUAN

      | Bein betiko   alforrikakoak izango dira
      | Seguraski     " " "
      | .....
      | Seguraski     boroizki erabilgarriak izango dira
      | Bein betiko   " " "
  
```

2.4 Irudia.- Erabilgarritasuna adierazten duten meta-arauen formatoa.

Ikusten denez, metaerregelen sintaxia oinarri-mailaren erregelak bezalakoa da. Honek kodeketa uniformeak ematen dio sistemari maila guztietan eta oinarri-mailan dauden mekanismo erabilgarri guztiak erabiltzea haizutzen du, zihurtasun faktoreak sententzia ez zehatzetarako bezala, konjuntzioetarako, disjuntzioetarako, eta abar.

BA			
A eta B batetzen badira eta			
Badaude arauak X alpatzen (ez) dutenak		premisatan	
		ondorioetan	
		biatan	
ORDUAN			
Bein botiko		orabil beharrekoak dira X alpatzen (ez) dutenak	premisatan
Seguraski			ondorioetan
.....			biatan
Posibleki			
	baino	lehenago	
		geroago	

2.5 Irudia.- Orden partziala adierazten duten meta-arauen formatoa.

METAERREGELEN FUNTZIONAMENDUA.

Sistemak L zerrenda bat berreskuratzen du, oraingo helbururako lortzeko erregela nabarmen guztiekin. Baina dei egiten saiatu baino lehen ea metaerregelarik dagoen helburu horri esleitua egiaztatzen du. Horrela bada, lehenengo oinarri-mailarako diseinatu zen dei mekanismo standarra erabiliz dei egiten da. Haren egintzaren ondorioz L zerrendako erregelatan desiratu daitekeen erabilgarritasun edo orden relatiboari buruzko ondorio batzu lortu ahal ditugu. Azken batean, berantolatzen da edo laburtzen da L (zihurtasunez erregela alferrikakoa ikusten bada), ondorioa interpretatzileari pasatuz.

Edozein beste maila onartzen da, bakoitzak jakintzaren erabilera menpeko mailan zuzentzeko balio du. Helburu bati aplikagarriak diren erregelen L zerrenda berezkiuratz gero, kateaketa bai aurrerantz bai atzerantz izanez, L mozteko edo berordenatzeko ea lehen orden erabilgarriko metaerregelen bigarren zerrenda L' dagoen egiaztatzen du. Baina dei egin baino lehen ea bigarren orden erabilgarriagoko metaerregelen zerrendarik dagoen L' ordenatzeko edo berordenatzeko eta abar.

Errekurtsioa goi-ordeneko erregela multzo bat aurkitzen ez denean gelditzen da, lortutako azken mailako lehen metaerregela aplikagarria exekutatzuz. Horrela menpeko ordenaren metaerregelen

zerrenda aldatuko da, eta horrela elkarren segidan oinarri-mailako L zerrenda moztu eta berordenatu arte.

Esandako guztia TEIRESIAS programak burutzen du MYCIN-en, funtzio batzuen bidez (THEREARE, MENTIONS, eta abar.).

Adibidez:

(THEREARE RULES (MENTIONS OBJECT PREMISE RECESION) LIST1)

TEIRESIAS-ek L zerrendako erregela bakoitzera heltzen da MENTIONS-en bidez eta ea OBJECT objektuak RECESION balorea hartzen duen premisan egiaztatzen du, baiezko kasuan erregelako identifikatiboa LIST1 zerrendara pasatuz. Iturburu kodearen azterketa zuzenaren teknika honek *edukinari erreferentzia* izena du eta esplizitoena eta eskurakoiena da, txekiatuz gainera erregelen edukiari buruz arrazonatzen haizutzen du.

Beste sistema batzuetan antzeko funtzioak ere badira.

Aurreranzko kateaketarekin (datuaren bidez zuzenduta) eta atzeranzkoan (helburuaren bidez zuzenduta) erabili ahal izan arren gainera, "means-ends" edo baliabide-helburua (diferentzia-
ren bidez zuzenduta) analisisian oinarritutako sistemetan ere erabil daitezke, laburtu beharreko diferentzia bat emanez eta eragile zenbaki nabarmen bat diferentzia horretarako. Diferentzia bakoitzarekin elkartuak metaerregelak eduki ahal ditugu, eragile nabarmenak baldintza aldakorren menpean nola ordenatu behar diren informatuz.

Lehenago oinarritzko kontzeptu eta metaerregelen mekanismoez hitz egin dugu. Orain ideia hauen kontribuzio zentralak ebaluatu eta aztertuko ditugu:

- * Metaerregelen erabilera deiaren kontrolerako bidea da.
- * Posibilitatea ematen dute kontrolari buruzko arrazonamendua egiteko.

DEIAREN KONTROLA.

Sistemak konparatzeko momentuan kontutan hartu behar dira puntu hauek:

- * Indexazio-maila.- Oinarri-maila erregelak hautatzen dituen funtzioaren kokapena.
 - Interpretatzaile-an integratuak, GPS, STRIPS, QA3, HEARSAY-II, eta abar.
 - Helburu ezpezifikoeekin elkartutako hautapen lokalen funtzioak, honek ez du sortzen beste helburuentzako bilakerarako kosturik; adibidez hauetan bezala: PLANNER, CONNIVER, QA4, TEIRESIAS, eta abar.
- * Uniformetasuna.- Errepresentazio beraren erabilera bai dominioaren oinarri-mailan (erregelak) bai metamailan, QA3 eta TEIRESIAS-en bezala.
- * Eskuragarritasuna.- Informazio estrategikoa kodedutua programa mailako objektu bat bezala (PLANNER, CONNIVER, QA4, TEIRESIAS) eta eskuragarria programa bera-tik. Bestela interpretatzailean zopatu da.
- * Explizitasuna.- Ahalbide batzu daude: estrategian erabilitako erizpidearen argitasunetik, azken ondo-rioaren seinalera bakarrik, azken hau ere oso argia ez izanik.

TEIRESIAS-en, eta beste ondoko sistemetan haren eraginare-kin:

- * Errepresentazioa explizito eta kokatua da (metaerrege-letan).
- * Arazoen konponketaren etapan deiaren kontrola birfin- ketaren bidez egiten da, honek errendimendua handitzen haizutzen du jarraian zein jakintza zati deitu behar duenari buruzko metajakintza gehituz. Kontuan hartu behar da adimenaren definizio zati batek informazioa- ren erabilera egokia barne ematen duela.

Expresibitatea ere handitzen du goi-mailako hizkuntza eskaintzen bait du.

Metaerregelak gehitu edo ezaba daitezke.

Estrategiek exekutatzeko erregelik egokiena seinalatzen dute, aurkeztutakoen artean kasurik hoberena nagusituko da, segurtasun faktoreak kontutan hartuta.

KONTROLAREN GAURKO EGOERAREN MUGAK EDO BEHARRAK.

Hauek ikus ditzakegu:

- * Estrategiek oraingo erregela talde aplikagarrienaren zerrendan bakarrik jokatzen dute. Beraz ezin daitezke inplementa "pentsamenduaren lerro" baten kontrolaren estrategiak.
- * Kontrola iñausketa eta berordenazio gisa, mekanismo nahiko poteretsua ez izatea gerta daiteke; beste plan konplexuagoen beharra egon daiteke, baldintzasko sententziadunak.
- * Primitibaren hiztegi zabala, birfinketa eta oinarri-mailako informazioa (ez bakarrik ordena eta erabilgarritasuna) deskribatzen duen metajakintza adieraz dezan.
- * Oinarrizko kontrolaren zikloa erregela baten deia da, eta deiaren kontrola erregela aplikagarrien zerrendaren aldaketan datza. Izan daiteke IM mota bakoitzak bere konputazioaren oinarrizko zikloa edukitzea, beraz bere ikuspegi propioa edukiko du kontrolatua izango den konputazio baten elementu primitiboari buruz.
- * Orain arte erabilitako metajakintza dominioaren menpeko da, baina ingeniariartzaren ikuspegitik gutxienez baliogarria dirudi (eta azken azterketek horretara jotzen dute) dominioarengandik independente den metajakintza bat edukitzea "gune" bat eraikitzeke, edo modulo batzuk, erabilgarriak dominio desberdinetan.

- * Desiragarri litzateke mekanismo bat edukitzea jarraian lortu behar den menpeko helburua zein den explizitoki seinala dezan, barrutik egindako mekanismo finko bat izan gabe TEIRESIAS-en eta beste batzuetan bezala.

DEIAREN KONTROLARI BURUZKO ARRAZONAMENDUA.

Erregela aplikagarrien zerrendaren birfinketa arazoen ebazpenerako lan bat bezala ikus daiteke, bere JB propioarekin, bere heuristikekin eta abar. Hau da programak bere kontrolari buruz arrazonatu ahal du.

Hau posible da bi ideia hauengatik:

- * Kontrolaren jakintza oinarri-maila edo dominioaren jakintzatik aldendu da, metaerregeletan explizitoki errepresentatuz.
- * Kontrolaren jakintza eskurakoia programa beragatik. Metaerregelak euren artean aztertu ahal dira.

Jakintzaren errepresentazio uniforme maila guztietan eta IM berbera maila guztietarako erabiltzeak ere laguntzen du.

Kontrolari buruzko arrazonamenduak lagunduko digu hurrengo exekutatu behar den erregela era azkar eta adimentsuan aurkitzea eta erabilgarria izango da gitxienez hiru egoera hauetan:

- * Kontrol egituren saturazioa dei kostu handiarekin oinarritzko jakintza potentzial larregi dagoela eta, honek haizutu edo aholkatzen du dei hobe batean baliabideak inbertitzea.
- * Oinarritzko jakintzaren deia aldi kopuru handiegi batean.
- * Denbora errealean lan egitea.

Bestela bilaketa espazioa honako modu honetariko batean korritu beharko genieke:

- * Estentsiboki bilatzen.
- * Dominioari buruzko jakintzak erabiliz era hautakorragoan bilatzeko. Ez da beti egia abiadura handiagoa eta hautakizun txikiagoa emanik edo alderantziz errendimendu hobearekin lortzen denik.

Kontrolari buruzko jakintza sententzia independente kopuru batzuekin eranstean da, sententzia hauek bakoitzak bere nabaritasuna dakar oinarritzko jakintzaren erabilera egokiari buruz, hau da arrazonamendua oinarritzko mailan bezala egiten da zerbait pausutako inferentzia kate baten bidez. Metaerregelen barrutiak gaitasun hauek eskeintzen ditu:

- * Era banatuan eta argian espezifikatzen du arrazonamendua egiten duen jakintza.
- * Erizpide hauek era murriztuan eta eskugarrian eranstean dira.
- * JB-aren kontrol informazioaren mailetakoa banaketa erabilgarria da, eta hauek IM-etik bereizteak abantaila hauek eskeintzen ditu:
 - Eraikuntzarako erraztasuna.
 - Aldaketa inkrementalak haizutzen ditu.
 - Hazkuntza inkrementala onartzen du.
 - Errepresentagarriagoa da.
 - Gardentasuna inkrementatzen du.
 - Sistemaren portaera oinarritzen deneko informazioa eskuragarria egiten da.

MUGAK KONTROLARI BURUZKO ARRAZONAMENDUAN.

Kontrolari buruzko arrazonamenduan ere mugak aurki daitezke. Honako hauek seinala ditzakegu:

- * Mekanismo inferentzial berberaren erabilerak metamailan, berarekin darama, abantailez gain, mekanismoaren muga intrinsekoko. Oinarritzko mailan adieraztzeko

erraza dena meta mailan erraza da eta zaila dena zaila da. Orduan galde genezake ea kontrol informazioa desberdina den errepresentazio eskakizunei dagokienez edo "beste dominio" bat den errepresentagarria dena erabilgarri dauden teknikekin, izen haundiendun autoreek iragartzen dutenaren arabera [DAVI80] hazken hau iradoki zitekeen, baina hau arazo irekia da.

- * Nabaritasun gatazkatsuarekin tratatzeko metodologia zenbaki terminoetara murriz daiteke (zihurtasun faktoreak) oinarritzko mailan, hau ez nahikoa izan daiteke meta mailan.
- * Ahultasun potentzial bat ere badago jakintzaren antolakuntza eta erabilerarako barruti baten proposamenen, soluzio baten espezifikazioaz gain ataza partikular bat espezifikatzen du. Era bat iradoki dugu deiaren kontrolerako antolakuntza eta erabileari buruz pentsatzeko, ez zein informazio izan behar den. Informazioaren espezifikazio ataza kontsideragarria da eta barrutiaren erabilgarritasunaren arabera da.

Desiragarria litzateke inferentzia mekanismoa erabiliz erabiliko den berreskuratze mota ateratzea (datuek zuzendua, helburuak zuzendua, edo diferentziak zuzendua) eta erregela bat dagoen kontestualekin bat etorritz aurreranzko, atzeranzko kateaketa etab aplikatu, edukia aipatzen duten erregelak erabiliz. Edukiari zuzendutako deia deitu zitzaiokeen eta kasu bakoitzerako kontrol egiturarik onena hautatzea haizutuko luke.

2.5.2.- Kontrola JB modularizatuetan.

Erregelen kopurua izugarri hazten bada JB modularizatzea behar dugu; irakurgarritasun , fidagarritasun, aldarazteko erraztasun, eta egiaztatagarritasun , etab ; handiago batez gain.

Modularizazio metodo desberdinak erabili dira, ia denak ad hoc soluzio baten bila, baina azkenaldian ikerkuntza ahalegin handiak egiten ari dira sistema desberdinetan erabilgarria den eta sistema desberdinen baterako moduloak sortzen dituen metodologia ezartzeko, Breuker eta Wielinga-ren KADS metodologia [BREU85] [PLAZ89a] bezala eta orokorrean bigarren belaunaldiko SA-etan.

Ikusi genuen TEIRESIAS-en JB bi aldeetan zatitua zegoela:

- * Metaerregelen basea (MEB).
- * Erregelen basea (ERB).

Modularizazioarekin JB-ak egitura konplexuagoa lortzen du eta kontrola era zailagoan egiten da. Sistema modularreko adibide bat MILORD [SIER89] da.

KONTROLA KADS METODOLOGIAREN ARABERA.

KADS metodologia jakintzaren lortzerako metodoa da, honek barruti deskripzionala eskeintzen du txantiloekin (txantilo hauen multzoari Interpretazio Eredua deitzen zaie) ataza generikoentzat edo atazaren osagaientzat, barruti honekin JB bat eraiki ahal izango da adituengandik lortutako informaziotik abiatuz erabilgarria izan nahian sistema desberdinentzat eta jakintza-ingeniriaren lana egituratu nahian.

KADS-ek premisa bi finkatzen ditu, honako hau esaten dutenak:

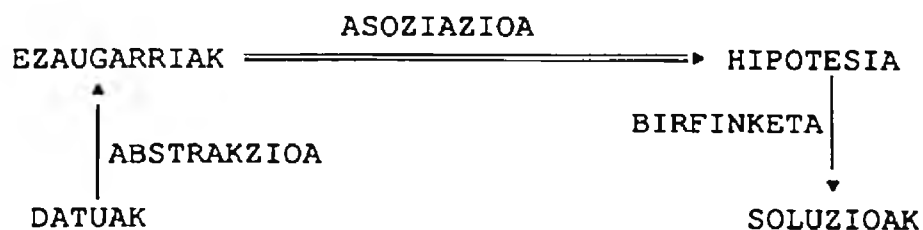
- * Posible eta erabilgarria da jakintza mota generiko desberdinen artean desberdintzea arrazonamendu prozesuan bete ahal izango duten zereginen arabera. Adibidez erregelak eta metaerregelak.

- * Jakintza mota hauek geruzetan antolatu daitezke, elkarrekintza mugatuarekin.

Geruzak, zeinetan informazioa egituratzen den honako hauek dira:

- * Dominioaren maila.- Dominioaren kontzeptuak ditu, erlazioak (is-a, consist-of) eta egitura konplexuak (prozesu edo gailuen ereduak).
- * Maila inferentziala.- Metaklase eta jakintza iturritz osatuta dago.
 - Metaklaseak.- Kontzeptuen zeregina arrazonamenduan deskribatzen dute, adibidez "adina" kontzeptua galde daiteke, inferentzia baten euskarri gisa erabil daiteke eta inoiz ez da izango gaisotasun baten agerpidea. Metaklaseen jerarkiak daude.
 - Jakintza iturriak.- Dominio erlazio bat nola erabil daitekeen iragartzen duten deskripzioak dira, hau da, zein inferentzia mota egin daiteke. Jakintza estrategikoa eusten diote dominioaren jakintzaren gainean.

Inferentzia egiturak metaklase eta jakintza iturri sare bat erlazionatzen ditu, zein inferentzia egin daiteke dominio batean iragarritz (egitura estrategikoa).



- * Ataza maila.- Arazoen ebazpen estrategia dauka Ataza bat (MILORD sistemaren modulo baten baliokidea da) jakintza iturrien antolakuntza multzo bat da

helburu bat lortzeko asmoarekin, erabaki egiten du nola edo noiz egiten diren inferentzia mota batzu. Egintza guztiak ataza mailan jakintza iturriei eta metaklasei dagozkie. Atazak jerarkietan antola daitezke orokorretatik bereziagoetara. Helburuaren arabera ataza talde bat erabil daiteke lortzeko.

Maila hau helburu zuhaitz batek errepresentatzen du eta dagokion deskonposazio helburu baten atazetan, atazen erregelen bidez.

- * Maila estrategikoa.- Planak egin eta atazen exekuzioa kontrolatu eta ganikusteko jakintzak dauzka. Malgutasuna lortzeko batzutan estrategia finkoa ez izatea beharko da, horregatik helburuen zuhaitza dinamikoki sortzen duen planifikatzaile bat beharko da arazoen ebazpenei metodo orokorrei buruzko jakintzak erabiliz (baliobide-helburu analisia, sailkapen heuristikoa) bere jokaera egokitzeke.

KONTROLA MILORD SISTEMAN.

Arkitektura multimailadun sistema bat da, bost mailako jakintza mota posiblekin: Diskurtsoaren unibertsoa, Asoziatibo eta Erlazionala, Hipotetikoa, eta Heuristikoa. Hauetariko bakoitza dominioaren mailan errepresentatzen dute honako hauek, urrenez urren: Egintzak, Erregelak, Moduloak, Estrategiak, eta Iharduketa planak; kontrolaren maila mota desberdinetako metaerregelen lau mailaz osatuta dagoelarik. Maila bakoitzak era independientean diseinatutako portaera du, honek SA-ren erai-kuntza errazten du. Honen modularitateak maila baten diseinu osoa errepresentatu ahal izatea bestengan eraginik izan gabe egiten du.

Sistema honetan ERB moduloez osatuta dago. Modulo batek helburu bat du, erregela batzu eta erregelak kontrolatuko dituzten metaerregela batzu. Modulo bakoitzak azpihelburu bat du, kunplitu behar dena azken helburua emateko, eta beste moduloene-

takotik desberdian izan daitekeen estrategia bat; egon daiteke modulo bat datuak hartzeko beste bat emaitzen erakuspenerako, etab. Helburu bera duten moduloak daude baina sarrera datu desberdinekin.

MEB-an ebatzi behar den arazo bakoitzeko hasierako moduloa, azken moduloa eta aurrerantza kateatutako tarteko moduloak iragartzen da. Modulo baten exekuzioa MEB-eko metaerregelek seinalatzen duten estrategia aldarazi dezake.

Kontrol maila desberdinen eta jakintzen errepresentazioaren arteko bereizketa honek JB-aren diseinu teknika desberdinak haizutzen ditu:

- JB-aren beherantzako diseinua haizutzen dute. Kontrol maila altuenetatik hasita.
- Arazoen ebazpen metodo desberdinak definitzen haizutzen du, maila hipotetikoan adierazitako jakintzaren menpean daudelako. Desberdindu ahal direlarik, adibidez, sailkapen sinplea eta sailkapen heuristikoa [CLAN85].

ganera eredu arkitektoniko honek sistemak honelakoak izatea egiten du: luzagarriak, erraz arazgarriak eta azalpen argidunak.

Kontrolaren multimaila arkitekturaren diseinuan eragin handiena izan duen alderdietariko bat sistemaren portaera gizaki aditu batek jarraiko zukeen estrategien ahalik eta antzerakoena izatea eragiten saiatzen da.

Metaerregela lau mota ditu:

- JB-aren erregelen aplikagarritasunaren egiaztapenerako.
- Oinarrizko estrategien eraikuntzarako.
- Salbuespenetarako.
- Bukapenerako.

Bakoitzak azpiko mailaren kontrolean du eragina, erreflexio mota inplizitoa da, interpreteak kudeatuta, eta beraren errepresentazioa desberdina da programatzailearen eta interpretearen ikuspuntutik, interpreteak errepresentazio bien arteko lotura kausala seguratzeko duelarik.

2.5.3.- Kontrola arazoaren ebazpenerako eredu kooperatiboetan.

Modularizazio beharritzen eredu kooperatiboak sotu zituen eta zehazkiago Harbel Arkitekturak.

Harbel arkitekturai buruzko lehenengo ideiak Newell-engandik sortu ziren 1962.ean eta harbel ereduak, gaur egun ulertzen den bezala, lehenengoz HEARSAY-II sisteman agertzen da (1971 - 1976) Simon-en ideien arabera, Reddy y Erman-ek inplementatuak. Geroagoko aplikazioak HASP, CRYSLIS, OPM, etab bezalakoak 1975-1980 aldikoak; HEARSAY-III, AGE, BB1, etab bezalakoak 1977-1984 aldikoiak; TRICERO, DVMT, etab bezalakoak 1981-1985 aldikoak; eta BB*, GBB, etab 1985-1987 aldikoak; aldaketa asko sartu dituzte.

Arkitektura mota honek honako ezaugarri hauek dituzten arazo zailak ebatzi nahi ditu:

- * Jakintza heterogeneoaren erabilera.
- * Bilaketa espazio handiaren beharritana.
- * Datu ez oso fidagarriak eta ez osoak.
- * Ebazpen prozesu modelizagarria prozesu kooperatibo bat bezala, non espezialista independenteak parte hartzen bait dute, arazoaren alderdi konkretuak ebazteko gai direnak.
- * Arrazonamendu hipotetikoaren beharritana eta estrategia desberdinen eta arrazonamendu metodoena.

Harbel arkitekturak diseinu modularra eta egitaratua haizutzen dute software tresnak aportatzeaz gain eredu kontzeptualak inplementatzeko sistema konplexuetan integratuz.

Jakintza arazoaren zatien ebazpenean espezialista zeregina betetzen duten moduloetan egituratzen da, modulo bakoitzari Jakintza Iturria (JI) deitzen zaio eta asoziatua izan dezake:

- * Jakintzaren errepresentazio formalismoa.
- * Jakintzaren interpretazio mekanismoa.

Arazoak era kooperatiboan ebatziko dira JI bakoitzaren parte hartze inkrementalaren bidez.

Ji desberdinen arteko komunikazio mekanismoa erkide memoria baten bidez egiten da, **harbela**, non JI bakoitzak hau egin dezake:

- * Haren azpiarazoa ebazteko beharrezko datuak irakurri.
- * Egindako prozesuaren emaitzak anotatu.

Aipatutako osagai bi hauetatik, JI eta harbela, eta PS-ak ez bezala alderdi hauek nabarmen ditzakegu:

- * Jakintzaren modularitatea.
- * Komunikaziorako baterako erkide.- Objektu guztiak (dominioaren datuak, hipotesiak, helburuak, etab) abstrakzio mailetan elkartzen dira harbelean.
- * Agente desberdinen srteko kooperazioa.

Baina hiru osagai gehiago dago, harbel arkitekturaren bereizgarriak:

- * MONITOREA.- Arazoenganko egoerari soluzio berriak eman diezaizkien JI-ak hautatzen ditu. Ezagutzen du haren aurrebaldintzak bete zezakeen JI eta harbeleko aldake-

ten arteko erlazioa. Gainera JI-en aplikagarritasuna egiazta dezake haren aurrebaldintzak benetan konplitzen badira, kasu batzuetan hau Erabaki Sistemak (ES) egiten du.

- * ERABAKI SISTEMA (SCHEDULER).- Sarrera gisa exekutatu behar diren ataza multzoa jasotzen du, hauek orokorrean JI-en exekuzioa da parametro bezala pasatzen diren harbelean objektu zehatz batzuen gainean. ES honen helburua sarrera ataza multzotik ataza bat hautatu eta Exekutatzailleari pasatzea da, horretarako JI bat lortzea haizutzen dion hautaketa erizpide multzo bat aplikatzen du.
- * EXEKUTATZAILEA.- Aurreko osagaiak lortutako JI exekutatzen du.

Funtzionamendu ziklo orokorra jarraiko hau da:

- 1.- Harbelean aldaketak ematen dira JI-en iharduketaren ondorioz.
- 2.- Aldaketek dagozkien JIek parte har diezaten eragin diezaketen datu berriak dakartzate. Aldaketak harbelean JI-en deia zuzentzen dute baina ez dute bermatzen Monitoreak deitutako JI-k parte har diezaketen, egiaztatu beharko bait da uneari aplikagarriak diren eta egokiena aukeratu.
- 3.- JI baten hautaketa, ES-ren aldetik, soluziorantz aurreratzen eragin dezaten aldaketak egiteko. Honako erizpide hauek erabil ahal izanik:
 - JI-aren deia sortu duten objektu moten arabera.
 - JI-ak sortutako objektu moten arabera.
 - JI-aren beste ezaugarri batzuen arabera; tipologia, lehentasuna, exekuzio maiztasuna, etab.
- 4.- Hautatutako JI-aren exekuzioa, Exekutatzaillearen aldetik, zeinek aldaketa berriak eragingo dituen

harbelean, JI berrien aktuazioa haizutuz. Prozesua bukatuko da:

- Jakintza nahikoa -JI- ez daudenean arazoa transformatzen jarraitzeko.
- Soluzioa lortu denean.

Aurretik ikusitakoaz kontrol oinarritzko zikloa honetara mugatzen da:

- * Harbelean gertatzen diren aldaketak begiratu.
- * Aldaketetik abiatuz aldaketek sartzen dituzten datuak agertzean aktibatzen diren JI-ak hautatu, hau da JI-ak, zeinen aurrebaldintza betetzen den.
- * JI hautatu eta exekutatu.

Lehenengo pausu biak Monitoreak egiten ditu.

Kontrolaren arazoan zentratuz, arkitektura hau erabil dezakegu bera ebazteko.

Kontrolaren arazoaren ebazpenean, sistema batek erabakitzen du, inplizitu edo esplizituki, zeintzu arazo saiatuko da ebazten, zein jakintza beharko duen eta zeintzu arazoaren ebazpen metodo eta estrategia aplikatuko dituen. Erabakiko du nola ebaluatuko dituen arazo alternatiboaren soluzioak, nola jakingo duen arazo berezi bat ebatzita dagoela eta zein zirkunstantzia pean eten egingo duen arazoaren edo azpiarazoaren hautaketa. Kontrolaren arazoaren ebazpenean sistema batek bere portaera kognitiboa zehazten du.

Kontrolaren arazoa giza adimenaren era berdinean ebazteko AA sistemak denbora errealeko planifikazio arazo bat bezala tratatu beharko lukete [HAYE85b], horrekin era adimentsuan inplementatuko litzateke kontrolaren arazoaren ebazpena lortzeko, gutxienez, portaera helburu hauek:

- 1.- Kontrolaren arazoa ebazten duten kontrol erabakiak esplizitoak egin.

- 2.- Erabaki zein ekintza exekutatu, zeintzu ekintza diren desiragarri eta zeintzu diren egingarri buruzko erabaki independenteak errekonsiliatuz.
- 3.- Garau tamainu aldakorreko kontrol heuristikak hartu, hau da, ekintza berezi edo orokorrak prozesuaren unearen arabera egiten dituzten heuristikak hartu.
- 4.- Arazoaren ebazpenean eguneko egoeran erabilgarriak diren egintza atributuetariko edozeinen ganean fokatzen duten kontrol heuristikak hartu.
- 5.- Kontrol heuristika indibidualak hartu, gorde eta gaitzetsi arazoaren ebazpeneko egoera dinamikoen erantzunez.
- 6.- Erabaki nola integratu garrantzi aldakorreko kontrol multipleko heuristikak.
- 7.- Dinamikoki planifikatu ekintza estrategikoen sekuentziak.
- 8.- Dominio eta kontrol ekintzen lehentasun erlatiboari buruz arrazonatu.

Harbel arkitektura erabiliz implementatutako sistema baten kontrolaren arazoa era desberdinetan ebatzi daiteke. B. Hayes-Roth-en arabera [HAYE85b] OPM-en implementatutako harbel kontrol arkitekturak aurreko ezaugarri guztiak kunplitzen ditu, hau ez da beste kontrol arkitektura batzuekin gertatzen:

- * HEARSAY-II-ko erabaki sistema sofistikatudun arkitektura.
- * HASP-en soluziora fokatutako arkitektura.
- * MOLGEN-en implementatutako metamaila arkitektura.

Harbel kontrol arkitektura honek ezaugarri garrantzizkoak ditu:

- * Esplizitoki errepresentatzen ditu dominioaren eta kontrolaren arazoak, jakintza eta soluzioak.
- * Dominioaren eta kontrolaren arazoen ebazpena oinarriko kontrol ziklo sinple batean integratzen ditu.

- * Beraren jakintzaren eta portaeraren errepresentazioak artikulatu, interpretatu eta aldarazten ditu.
- * Arazoen ebazpenerako jakintza, haren aplikazioa jakintza honi, eta haren oinarrizko kontrol zikloa arazoen ebazpenen egoera dinamikoei egokitzen die.
- * Ezaugarri hauek harbel arkitektura uniforme batera sartzen ditu, zeinetan dominioaren eta kontrolaren arazoen ebazpenaren portaera karakteristikoki inkrementalak eta oportunistikoak bait dira.

Labur bilduz esan dezakegu harbel kontroleko arkitekturak puntako malgutasuna ematen duela sistemaren portaeran, honako hau egin dizakelarik:

- * Estrategi prozedural hertsiei jarraitu.
- * Bata bestean atzeko fase estrategikoen inplementazioa koordinatu arazoen egoera berezietan.
- * Eten, jarraitu edo bukatu hartutako estrategiak.
- * Batera aplikagarriak diren estrategien azpimultzo aldakorak eta estrategikoak ez diren heuristikak hartu.
- * Erregela aldakorak erabili hartutako heuristikak integratzeko eta egiteko dauden arazoen ebazpenerako ekintzen artean aukeratu.

Dena batuz harbel kontroleko sistemak kontrol plan konplexuak egokitzen dituzte, zeinen estrategiak, heuristikak eta integrazio eta programazio (scheduling) erregelak behin eta berriz alda daitezkeen arazoaren ebazpenean zehar. Gainera sistemaren portaera goberenatzten duten parametro guztiak sistemaren kontrolpean jartzen dute.

Harbel kontroleko arkitekturaren eragozpenik handiena **konputazional eta gordetzeko kostu handia** da, honek eragiten du errendimendu handiko edo azken aplikazioko sistema eraiki nahi badugu, harbel kontroleko arkitektura desegokia izango da.

Bestalde, harbel kontroleko arkitekturak ingurune oso malgua eta modularra eskeintzen du garapen esperimentaleko aplikazio sistementzako errendimendu altuko ingurune batean berinplementazioaren aurrekoa.

2.5.4.- Kontrola denbora errealeko JOS-etan.

Denbora errealeko sistemek erantzun bat bermatzeko trebetasuna daukate aurrez definitutako eta mugatutako denbora baten barruan.

Denborak mugatutako arrazonamenduak AA-eko teknika gahienak behar ditu eta diseinu metodo bereziak konplexu (ARTIFACT, RESCU, QUIC, etab bezalako adibide isolatu apur batzu daude) eta interesgarria erresultatuz aplikazio askorako.

Kategoria bi daude:

- * Muga biguneko sistemak.- Agindutako mugen artean kokatutako exekuzio denbora ertaina eduki beharko luketen sistemak dira. Adibidez gizon-makina elkarrekintza sistemak.
- * Muga gogorreko sistemak.- Agindutako mugen barruan kokatutako exekuzio denbora maximoa eduki behar duten sistemak dira. Adibidez segurtasun sistema kritikoak.

Denbora errealeko sistemak ezaugarri hauek dituzte:

- * Dinamismoa.- Mundu dinamiko bateko funtsezko deskripzioak mantentzen dituzte.
- * Memoria.- Inferitutako datuen eta neurtutako edo jasotako balioen iraunkortasuna.
- * Zihurgabetasuna.- Aleatorietateagatik zarata dela eta xehetasun gabetasunagatik doitasun fdelta dela eta.

- * Baliabide mugatuak.- Lehentasunak, ES baten arabera eta defektuzko balioak.

Eta hurrengo eskakizun hauek:

- * Arrazonamendua denborarekin, denboraren errepresentazio esplizitoa.
- * Denborari buruzko arrazonamendua, lehentasunak jasatzen dituelarik eta ES erabiltzen duelarik.
- * Arrazonamendua zihurgabetasunarekin, datu sentsorialen balioespena eta jakintza enpirikoen errepresentazioa.
- * Arrazonamendua jakintza iturri desberdinekin, teoriakoa, enpirikoa eta diseinukoa.

Teorikoki komendagarria litzateke Harbel Arkitektura eta Objektuei Orientatutako Errepresentazioa [SHAR88] kontrolaren arazoa ebatziz aurreko atalean seinalatutako metodoetariko batekin, baina soluzio honen kostua aukera hau garapen esperimentaleko sistementzako baino baliagarria ez izatea eragiten du sistema berinplementatu behar delarik errendimendu altuko ingurune batean. Dena dela eta denbora errealean lan egiteak ere kontrolari buruzko arrazonamendu azkarrago eta zehatzagoa edukitzea behartzen du.

2.6 .- IKASKUNTZAREN GARRANTZIA KONTROLEAN.

Metajakintzaren erabilera eta errepresentaziorako paradigma egokiaren lortzea, aplikazio mugatu baterako betebeharrak zaila izan daiteke; baina "nola errepresentatu" mugatuz gero "zer errepresentatu" jakitea falta zaigu, hau da zein den ezinbesteko informazio hori jakitea falta zaigu, programak bere helburuak

modu eraginkor batean lor dezan eragiten duena. Hiru modu ezberdinez lor dezakegu:

- * Jakintzaren lorpena bibliografiara joz.
- * Adituaren jakintzaren lorpena Jakintzaren Lorpen metodo korapilatsuak erabiliz. Eragozpen hauekin: ezin izango dugula adituaren laguntasuna mugagabea izan, eta ezin izango dela bere jakintzaren % 100 atera ere. Mota honetako Jakintzaren Lorpenean sistemak berak lagundu liezaiguke, Davis-ek erabilitako metodoaren kasuan bezala [DAVI76], bere TEIRESIAS sisteman "Esperientziaren transferentzia elkareragillea...." deitutakoa.
- * Jakintzaren Lorpen Automatikoa edo Ikasskuntza Automatikoa, esperientziatik irtendako erregela orokorrak atera ditzan ikaskuntza modulu baten bidez, adituaren jokabidea simulatuz bere bizitzan.
- * Kontzeptu Berrien Sormena, aurreko bien arteko ideia bat da, honetan jakintzaren- ingeniariak hasierako eta azkeneko kontzeptuak ezarriz informazio estadistikora edo dokumentalera joz erregela batzu lortzen ditu kontzeptu askoren disjuntibak osatutako ondorioarekin, adituari galdetuz gero gerta daiteke disjuntzio hau kontzeptu berria izatea edo betarteko kontzeptua. Adituak ez badaki zer den disjuntzio hori, abandonatu egiten da. Kontzeptu berrien sormenak erregelen kopurua gutxitu erazten du eta azalpen potentzia haundiagotzen du [CUEN89]. Fu eta Buchanan-ek [FU 85] estadistiketan geometrikoki erdikoak diren puntuak bilatzen dituzte eta izan daiteke puntu hauek adituak identifikatutako kontzeptu berrri batekin batera etortzea. AM sistema [LENA82] aipa dezakegu ideia honen adibide klasiko gisa, honekin 300 kontzeptu berrri sortu bait ziren, eta hauetatik 25 nabarmenak.

Prozesuaren une bakoitzean zein jakintza erabili sistemak erabakitzeak, ahalik eta lasterren soluziora iristeko, era

"ezin txarragoa", etab. Reflexio ezaugarria du eta blokeen bidez ikasten du (chunking) [LAIR87].

- PRODIGY .- STRIPS-en prozedurari jarraitzen dio baina ikaskuntza eta metakontrolarekin. Kontrolaren jakintza ikaskuntzatik % 80-an lortu baehar da, ikasi beharreko hiru kontrol erregelak mota daude: Lehentasun Erregelak, Ukapen Erregelak, eta Hautaketa Erregelak [MINT87].

2.7 .- BIGARREN BELAUNALDIKO SA-TAKO KONTROLA.

Gaur egun SA-ak produkzio erregelen bidez errepresentatutako jakintza superfizialekotatik aurrera arrazonatzen dute, eta ezaugarriarik handiena erabilpen konkretuetan duen bere funtzionamendu ona da, produkzio erregelak dominioaren menpeko jakintza praktikoak errepresentatzen bait dituzte. AA-erako beste hurbilketa batzuk domonioarengandik independienteak diren jakintza teorikoen errepresentazio deklaragarria proposatzen dute axiomen bidez lengoai formal batean eta inferentzia mekanismo orokorren erabilera ebazpen printzipioa bezalakoak. Sistema konexionistek ere dominioaren menpekoak ez diren metodoak aholkatzen dituzte. Dominioaren menpekoak diren eta ez diren metodoen arteko desberdintasuna da lehengoetarako adimentsua izateak jakintza praktiko gehiago edukitzea esan nahi du eta, ordea, bigarrentarikoerako inferentzia mekanismo nagusiagoa izatea esan nahi du.

Bigarren Belaunaldiko SA-ek gaur eguneko SA-ekin lortutako esperientzia erabili nahi dute gizakien trebetasunaren izaera analizatuz arazoaren ebazpenen eta jakintzen lorpen eta bere errepresentazioen prozeduretan. Aurretiko ikerketak [CLAN85], [STEE87], Simmons 1988, etab., erakutsi dute aplikazio barietate

garrantzitsurako ataza orokorren multzo amankomuna dagoela, hauek bezalakoak: sailkapen, mugapen kontestuala, dedukzioa, kokapena, egiaztapena, egokitzea, generaltzea, etab. Adituek ataza hauek arazoen ebazpenerako beharrezkoak diren jakintzak errepresentatzen dituzten erduei lotutako ebazpen metodoen aplikazioaren bidez konpontzen dituzte. Atazarekin gertatzen den bezala dominioaren eredu multzo bat ere identifika daiteke eta dagozkien ebazpen metodo generikoak ere bai, hau da, aplikazio alor desberdinentzako berdinak direnak. Dominioaren ereduak informazio heuristikoz osotu behar dira ebazpen metodoek eraginkorki nola erabili behar dituztenari buruzko informazioa. Honako hauek osatutako multzoa:

- Atazak
- Ebazpen metodoak
- Dominio ereduak
- Informazio heuristikoa

jakintza sakona izendatzen dena osatzen dute, bai jakintza teorikoak bai praktikoak esplizitoki eta banatuta errepresentatzen bait ditu erregelen bidezko errepresentazio azalaren kontra, erregelek era konpilatuan eta nahastuan modelizatzen dituzte bai jakintza teorikoak bai praktikoak, erregela berberan aurretik seinalatutako ataza orokorretariko edozeini dagozkien klausulak, espezifikatutako atazaren ebazpen metodoak, eta honekin batera erregela horretan hartu behar den ekintzaren informazio berezi gehigarria duten klausula batzuk.

Jakintza sakonetan oinarritutako SA batean ebazpen metodoak, erregeletan inplizito egon beharrean, metaerregelen bidez esplizitatzen dira. Adibidez birfintzeko metodoa "mugapen kontestuala" atazarako 2.6 irudiko metaerregelen bidez inplimentatzen dira.

Atazaren egitura ere inplizitoki inplementatzen da meteerregelen bidez. Metaerregela mota biak, bai ebazpen metodoei orientatutakoak bai atazei orientatutakoak, dominio eredu

Ba

eginkizuna textuinguru rostriazio bada
kandidatu elementu batzuk badaude
elementuak birfinketak onartzen badituzte

ORDUAN

birfinketak elementuen ordeiz jarri.

2.6 Irudia.- Mataerregela.

desberdinak erabiltzen dituzte, kontestu jerarkiak bezalakoak, birfinketaz egindako mugapen kontestualerako, eta prototipoak bezalakoak, objektuaren identifikaziorako desberdinketaren bidez.

Aditutasuna konfiguratzen duen jakintza sakonaren errepresentazio esplizito hau jakintzaren egituraketa era garrantzitsua da. Egituraketa hau hain maila sakonetara hel daiteke non errepresentatzen ari den jakintzaren arloko oinarritzko printzipioari buruzko jakintzak esplizitatzea haizutzen dute.

Jakintzan sakontze honek gaur egungo SA-en mugak ganditzea haizutzen du, honako hauei buruzkoak:

- * Ahultasuna arazoen ebazpenean, ez bait dute jakintza sakona ager litekeen jakintza azalekoaaren falta errepresentatzeko.
- * Azalpenen txirotasuna.
- * Jakintza lortzeko zailtasuna, galdera bereziagoak eginez.
- * Erabiltzaielei galdetzen zaien orden desegokia.

Bigarren Belaunaldiko SA-ren eragozpena da konputazionalki EOS azalekoak haiek baino eraginkorragoak direla, hau jakintza sakona erregelen erara konpilatuz konpon daiteke eta jakintza sakona erabiliz konpilatutako erregelek nahikoak ez direnean.

Abantaila mantenimendua eta JB handien erabilera errazten duela da, erregelek jakintzaren egituraketaren parte bait dira eta ez dira jakintza unitate isolatuak.

Fokatze honek hauxe behar du:

- * Ataza orokor berrien identifikazioa, dominioaren eredua eta ebazpen metodoak.

- * Aurrerakuntzak ikasketa automatikoan azaleko jakintza eratortzeko jakintza sakonetik abiatuta.
- * Jakintza sakonaren hartzean oinarritutako metodologiak.

2.8 .- ONDORIOAK.

Metajakintzaren erabilpena eta errepresentaziorako paradigma batzu ikusi ditugu, eta ondorio honetara iritsi gara: metajakintzaren erabilpena ezinbestekoa dela erabilgarriak diren eta errendimendu eraginkorra eskeintzen duten sistemak eraikitzeke. Aplikazio erreal baterako paradigma egokiaren lortzeak zeregin zaila izaten da, honetaz ez dira bakarrik malgutasuna, modularitatea eta ganontzeko aspektu positiboak baloratu behar baizik eta kontutan izan behar da suposatzen duen kostu konputazionala eta biltegitratzekoak eta honek guztiz baliaezina izatea eragin dezake; horregatik gaureguneko sistemetan kontrol informazioa sistema bakoitzerako *ad hoc* inplementatzen da eta dominioaren menpekoa da. Baina sistemen erikuntza egituratzeko eta hauetan adimen maila handiago bat hartzeko, gaur eguneko ikerketen joera dominiotik independentea den kontrol informazioa erabiltzera doa eta honekin batera ikaskuntza automatikoari garrantzi handia emanez Bigarren Belaunaldiko SA-etan ikusi daitekenez.

3. KAPITULUA: JAKINTZAREN ERREPRESENTAZIOA MAMUR-EN.

3.1.- SARRERA.

Sistema baten erabiliko diren errepresentazio elementuak mugatzeko kontutan izan beharko dira errerepresentatuko diren jakintza motak [CLAN85] egokienak diren errepresentazio elementuak mugatzeko, bakoitzak aukera mugatu batzu bait ditu.

3.2.- JAKINTZA MOTAK.

Lehenengo zatiketa batean esango dugu SA guztietan izango dela:

- * Dominioaren jakintzak, emaitzak lortzeko eman beharreko pausuak iragartzen dituzten ekintzek, erregelek eta moduloek osatutakoak.
- * Kontrol jakintzak, dominioaren jakintzetan jarraitu beharreko biderik onena iragartzen duten (emaitzak lortzeko) metaerregela eta maila goreneko moduluak errerepresentatuak.

jakintza mota biak ekintza beraien ganean aritu ahal direlarik.

Jakintza errerepresentagarriak izan daitezke gainera:

- * Teoirikoak, eztabaidarik onartzen ez dituzten egi frogatuak errerepresentatzen dituzte eta eredu bidez

errepresentatzen dira, errealtatearen errepresentazio abstraktu edo sinpletuak direnak, nahiz eta SA-ren arloan gehienetan ez osokoak diren.

- * Enpirikoak, praktikan gertatzen diren egintzek erre-presentatzen dituzte.

Jakintza mota baten edo beste baten nagusitasunak sistemaren arkitektura zehazten du, arazo matematiko baten kasuan ia jakintza guztia teorikoa da eta sistemak **dedukzio automatikoko sistemak**-en izena hartzen dute SA-ren ordezt; azken hauetan ugariak dira esperientzia praktikoaren bidez hartutako jakintzak, kasu askotan ezosokoak edo zehazgabekoak direnak, horrela ezinezkoa eginez eredu formalen erabilera datuen faltagatik eta maila sakonean erabiltzen badira ekartzen duten eraginkortasun ezagatik, eredu teorikoekin batera. Jakintza enpirikoek jatorri bi izan ditzakete:

- * Heuristikoak .- Adituaren praktikaren kontzeptuen arteko asoziazioak.
- * Estadistikoak .- Kontzeptuen asoziazioen oarpen sarriak.

3.3.- ERREPRESENTAZIO FORMALISMOAK.

MAMUR erregeletan oinarritutako sistema (EOS) bat da horregatik egintzen baseaz (EGB), erregelen baseaz (ERB), eta inferentzia motorraz (IM) osatuta dago. Osagai bakoitzaren alderdi errepresentazionala MAMUR-en objektuetarako orientazioak eragiten du. Egintzak kalkulu proposizionalaren proposizioak dira edo balioen multzo finitua eduki dezaketeen aldagaiak, 0+ ordeneko logika izendatzen dena osatuz. Erregeletan logika difusoa zihurgabetasuna termino linguistikoaren bidez modelizatuz erabilia da.

3.3.1.- Produkzio sistema ideiaren integrazioa Frame-aren idearekin.

2.2.4 atalean seinalatzen den bezalaxe errepresentazio eskema deklaragarri eta prozeduralen artean zegoen eztabaida errepresentazio eskema mixtoek ganditu dute, gehituz kasu honetan Frame-en erabileraren abantailak (2.2.4.), honek OOP ahalbideratzen du, PS-renekin (2.2.3.); beren naturaltasun eta azalgarritasunarekin. Eskema deklaragarri eta prozeduralen integrazio honek oinarri metodologikoa nahikoa ematen du tamainu hazkorreko sistema adituen eraikuntzari aurre egiteko, eta mundu errealean aplikagarriak.

MAMUR-en JB oinarrean klase eta instantzien definizioez osatuta dago objektuetara orientatutako edozein sistematan ohizko den bezala. Sistemak aurretik definitutako klase batzu ditu, bakoitzaren menpean jartzen diren objektuei egitura berezia emango dizkiotenak, eta hauexek dira:

- * RAIZ.- Beste klase guztiak klase honen menpean daude, bai MAMUR-enak zein berarekin eraikiko den sistema adituarenak eta objektuetara orientatuko edozein lengoai bezala oinarritzko metodoak barne dauzka.
- * HECHOS .- Klase honen berezitasuna egintzak erreprezentatzen dituzten klaseen erroa izatea da, honengatik klase honetatik sortzen diren objektu guztiek berezitasun bat dute, oinarritzko fazeten multzo bat eduki ahal izatea (ikus 3.1. irudia), dominioaren informazioaren bat erreprezentatzen duen atributu bakoitzeko, atributua honako hauekin erlazionatzeko asmoz:
 - erabiltzailea.
 - moduluak eta erregelak, non erabilia izaten den.
 - metodo bat.
 - defektuzko balio bat, rango bat, etab.

Atributu bakoitzak fisikoki deklaritzen zaizkion fazetak bakarrik ditu. Fazeta berriak gehitu zaizkioke aplikazioaren beharrianen arabera.

FAZETA	ESANAHIA
Pregunta	Erabiltzaileari egingo zaizkion galderaren testua atributu honen balio eskatzeko.
Nombre-largo	Erabiltzaileak ezagutzen duen kontzeptu bat identifikatuko duen atributuaren izen osoa.
Módulos	Ekitza hua erabiliko den moduluen izenak.
Conclusión-de	Atributu hau ondorio aldean erabiltzen duten arauen izenak.
Premisa-de	" " premisa " " " " "
Valor	Atributu honi dagokion balioa. Balio bat edo gehiagokin osatutako zerrenda bat izango du baina.
Valor-certeza	Atributu honi dagozkion ziurtasun balioa. Hitzkuntz hitz bat ordezkatzeko duen zenbaki bat izango da.
Resp-esperadas	"Valor" fazetaren balio onargarriaren zerrenda.
Rango	"Valor" fazetaren balio onargarritzat joteko tartea.
Método	Automatikoki exekutatu den prozedura atributu honen balioa ateratzeko ezezaguna denean.
Defecto	Atributuak hartuko duen balioa "Valor" fazetak eta "metodo" fazetaren balioa NIL denean.
Deducible	Atributu hau ateragarria den edo ez adierazteko (T edo NIL).

3.1 Irudia.- Olinarrizko fazeten zerrenda.

- * **REGLAS** .- Erregela guztien klase erroa da, bai modulu maila desberdinena bai metaerregelena eta haietako bakoitzak definitzen du egitura.
- * **MOTOR** .- Metodoak eta beharrezko informazioa dauzka MAMUR-en IM-ren funtzionamendu zuzenerako.
- * **FUNCIONES** .- Eraikiko den SA-an erabili beharko diren erabiltzailearen funtzioen egitura definituko dute, eta hauek haren instantziak izango dira, eta edozein egintza edo erregelatik exekutatuak izan daitezkeenak.
- * **AVISOS** .- Erregeletatik aktibatzen diren abisu guztien egitura definitzeko.

Hasierako egintza bakoitza edo deduzitua Hasierako Ekintzen Basean eta EGB-en, MOTOR klasearen atributuak direnak, txantiloi honen bidez errepresentatuta geratuko dira:

((objektu atributua) balioa zihurtasun-balioa)

eta hemen zerrendaren lehenengo elementua da helburua.

Erregela bakoitza eta erregelen modulo bakoitza -edozein maila edo motatakoa izanda ere- barneko formato batetan errepre-sentatuta geratuko da, aurretik sortutakoa JB-aren konpilazio prozesuan zehar eta honen emaitza kargatu ondoren RETE zuhaitzean MOTOR klasean beste osagarritzko atributu batzuekin dedukzio prozesua konputazionalki eraginkorragoa izan daiten eragiteko.

3.2.2.- Objektu desberdinen deklaraziorako funtzioak.

Bai erregelak bai egintzak OOP-ei printzipioak jarraitzen dieten objektuak dira 2. kapituluan (2.3.1) ikusitako ezaugarrie-kin: enkapsulazioa, mezuen bidalketa, herentzia, etab.

Objektu abstraktu bat errepresentatzen duen entitate bat

```
(defclase klase meta-klasea ( (atributu1 balio1)
                              (atributu2 balio2)
                              .....))
```

3.2. irudia.- Klase baten definizioa.

klase bat bezala errepresen-tatuko da eta 3.2. irudiko komandoen bidez definituko da; non "klase" klase berria-ren izena da, eta "metakla-sea" jerarkikoki klase honen

menpean geratuko da, izan daitekeelarik MAMUR-en aldeztu aurretik definitutako klase bat, RAIZ, HECHOS, REGLAS, etab bezalaxe edo erabiltzaileak definitutako klase bat komando beronen bidez. Definitutako atributuak klasearen berezko atributuak dira iragarritako hasierako balioarekin, gainera klase berrian beste klaseekin jerarkikoki erlazionatzea hhaizutuko duten beste atributu batzu sortuko dira, herentzia eta etorkizuneko ins-tantziak ahalbideratzeko asmoz.

Objektu partikular eta zehatz bat instantzia bat bezala errepresentatuko da 3.3. irudiko komandoaren bidez, eta honekin instantzia berriaren izena espezifikatzen da, zein klasearen menpean den eta espezifikatutako klasearen berezko atributu

```
(definst instantzia klase ( (atributu1 balio1)
                             (atributu2 balio2)
                             .....))
```

guztien edo batzuen balio ezagunak edo aurreko klase batenak.

3.3. Irudia.- Instantzia baten definizioa.

REGLAS-en menpeko klaseak moduluak dira edo dominioaren zati mugatu bati dagokion jakintza duten erregela multzoak REGLAS klaseak dituen berezko atributuen egitura berarekin. Bere definiziorako komando hau erabiliko da:

```
(defmodulo modulua modulu-aita)
```

Modulu bakoitzeko instantziak erregelak dira, 3.4. komandoarekin definituko direnak, hemen premisak eta ondorioak bakoitza asoziazio zerrenda bat izango da azpizerrenda bakoitzak bete beharreko baldintza bat izango duelarik (premisetan) edo hartu beharreko ekintza bat (ondorioetan), jarraiko datu hauekin:

```
(defregla arauen-izena
           moduluaren-izena
           premisak
           ">"
           ondorioak
           [erabilgarritasuna]
           [arauen-zihurtasun-balioa]
           [dokumentazioa]
           )
```

3.4. Irudia.- Erregela baten definizioa.

- * aipatutako objektuaren izena.
- * dagokion atributuaren izena.
- * konparatzeko erabilia den funtzioa (premisal aldean) edo agintzeko eran (ondorioetan), ikus 3.5 irudia.
- * aurrez aipatutako atributuaren "valor" fazetaren balioarekin konparatzeko balioa (premisal aldean) edo esleitzeko (ondorioal aldean).
- * baldintzan espero den zihurtasun balioa, edo gehieneko ondorioan.

Gainera ondorioetan ager zitekeen, lehenago seinalatu dugunaren orde, E funtzioa ebaluatzeko:

- * Lisp adierazpen bat.
- * Funtzio berezi bat hauek bezalakoa:
 - ADELANTE.- Modulu bateko egintza berriak deduzitzen ditu aurreranzko kateaketa erabiliz.

- ATRAS .- Helburu bat deduzitzen du modulo baten atzeranzko kateaketa jarraituz.
- REORDENAR-REGLAS-MENCIONEN.- Gatazka multzoa berordenatzen du iragarritako objektu-atributu-balioa seinalatutako erregelen aldean agertzen bada erregelen erabilgarritasuna handiagotuz.
- PODA-REGLAS-MENCIONEN.- Erregela baliogabetzat jotzen du emandako objektu-atributu-balioa seinalatutako erregelen aldean agertzen bada.
- EFU .- Erabiltzailearen funtzio baten exekuzioa.
- PREGUNTAR .- Iragarritako objektu-atributuaren balioa zuzenean aldarazteko ahalbidea ematen du, erregela baten exekuzioan zehar.
- VER-AVISO .- Abisu ezagun bat ikustea eragiten du, gertatzen den momentuan edo SA-ri egindako kontsultaren emaitzarekin batera, errepikapenak ekidinez.
- INC-CERT.- Zihurtasuna inkrementatzen dio hurrengo termino linguistikoki objektu-atributu ezagun batentzako.
- DEC-CERT.- Dekrementatuko du.

* Ebaluatu beharreko erabiltzearen edozein beste funtzio.

iragazketa aldagaiak sartu ahal izanik, premisetan aurretik erreferentziatuak, funtzioaren paramentroak gisa.

Gainera opzionalki espezifikatu zitekeen, (0 - 100) erabilgarritasun balioa zihurtasun balio bat erregelen multzora-ko, eta karaktere-kate bat dokumentazio gisa.

Zihurtasun balioak linguistikoki adieraztuko dira alda daitezkeen balio rango bat eta esanahi batekin.

Bai premisetan bai ondorioetan iragazketa aldagaiak erabil daitezke, eta hauen izenak "?" karakterarekin hasiko dira, premisetan eta ondorioetan, balioari eta atributuari destinatutako posizioak betetuz erregelei orokortasu handiagoa emateko.

conocido	Eraguna
=	Berdina.
/-	Ez berdina.
>	Handiagoa.
<	Txikiagoa.
>=	Handiagoa edo berdina
<=	Txikiagoa edo berdina
>10>	Handiagoa % 10 baino gehiagotan.
>20>	Handiagoa % 20 baino gehiagotan.
/>10>	Ez handiagoa % 10 baino gehiagotan.
/>20>	Ez handiagoa % 20 baino gehiagotan.
><	Lehen zenbakia baino handiago eta bigarrena baino txikiago, premisen baldintza bateko balioen posizioaren barruan (parentesi artean adierazitakoa).
?	Atributuaren balioa itzultzen du.
->	Objektuak eta atributuak identifikatutako egintza deseuzten du.

3.5. Irudia.- Premisetan eta ondorioetan erabilgarriak diren funtzioak.

Ondorio baten barruan eta balioaren posizioan Lisp funtzio bat ager daiteke edo "(objektu atributu ?)"-ren bitartez adieraztutako atributu baten balioaren eskaera.

Badira komandoak metodo bat atributu bati esleitzeko, eta fazetak hasi edo aldarazteko ere bai.

```
(defmetodo objektu atributu gorputza)
(valor->faceta objektu atributu fazeta balio)
```

Objektuen artean mezuak igarotzeko bada komando bat iragarritako objektuari heltzen diona honetan espezifikatutako metodoa bilatuz eta gehitzen diren argumentuarekin exekutatzuz, bere formatua hurrengo hau da:

```
(enviar objektu metodo argumentol argumento2...)
```

metodoa seinalatutako objektuan ez badago heredatu egingo da.

Bakarrik egon daiteke aurreko bat klase bakoitzeko horregatik ez du herentzi multiplerik onartzen.

3.4.- JAKINTZAREN KOMPILAZIOA.

JB konpilatua izan daiteke kargaketa azkarragoa eta exekuzio eraginkorragoa lortzeko, bere egituraren eta ekintzekiko harremanen aurrekonputazioen bidez.

3.4.1.- Sarrera.

Nahiz eta Jakintzaren Konpilazioa terminoak erabilera handia izan literaturan arazoan ebazpenari, ereduari oinarritutako arrazonamenduari, eta programazio automatikoari dagokionez, bere esanahia oraindik ez dago argi eta oraintxu Jakintzaren Konpilazioari buruzko literaturan kontutuan hartzeko eztabaiden iturri izan da [GOEL91].

Eztabaida hau lan batzuekin hasi zen larogeigarren hamarkadaren harieran, eta hauek eredu kausalekin egiten den arrazonamenduaren arazo orokorra piztu zuten, MYCIN-en erabilitako jakintza heuristiko motarekin kontrastatuz. Hart eta Michie-n lanak aditzera ematen dute asoziazio heuristikoak arazoaren erabakia azkarragoa egiteko erabilgarriak diren bitartean, dominioaren eredu kausalak beharrezkoak direla arazo zailak erabakitzeko.

Geroago de Kleer, Chandrasekaran ta Mittal, eta Sembuga-moorthy ta Chandrasekaran-k aditzera ematen dute errepresentazio sakonagoak orokorragoak diren bitartean eta beraz potentzialki ataza batzuetarako erabilgarriak, jakintza konpilatua eraginkorragoa dela ataza bereziak erabiltzeko, adibidez diagnostikoa bezalakoa.

[DAVI89]-en, Davis-ek auzitan jarri zuen egitura funtzionalako ereduaren jakintza heuristikoaren konpilazioaren erabilgarritasuna, hauxe argumentatuz:

- * Arrazonamendu metodo baten eraginkortasuna erabiltzen duen edukiaren arabera izango da eta ez honen formaren arabera.
- * Egitura funtzionaleko ereduen asoziazio heuristikoak konpilatuz jakintzaren forma aldatu egiten da baina ez bere edukia.
- * Orduan, orokorrean, jakintza heuristiko konpilatuaren erabilera diagnosi atazak erabiltzeko ez da ereduen erabilera baino eraginkorragoa.
- * Hortaz, jakintzaren konpilazioa ez da itxaroten siren helburuetara iritsi.

Keller-ek Davis-en kritikari erantzun zion, AAAI-90-an [KELL89], haxe argumentatuz:

- * Ereduan oinarritutako arrazonamendua, orokorrean konputazionalki tratagaitza da.
- * Arrazonamendu metodo baten eraginkortasuna bien menpean dago erabiltzen duen jakintzaren eduki eta itxuraren menpean.
- * Edonola egitura funtzionaleko ereduen jakintza heuristikokoaren konpilazioak bai jakintzaren edukia bai itxura aldatzen ditu..
- * Diagnostiko atazen ebazpen heuristikak erabiliz konputazionalki eraginkorragoa izan daiteke, bereziki bereganatze erraztzaile batzuen pean.
- * Hortaz ereduen jakintzaren konpilazioa heuristiketara, batera esanguratsua eta erabilgarria da.

AAAI-90-an bertan, Davis-ek Keller-en tesia gezurtatu zuen jakintzaren konpilazioa baliogabea izanez gain inkoerentea zela aditzera emanez eta hau argudiatuz:

- * Diagnostiko atazak dira konputazionalki tratagaitzak direnak eta ez ereduan oinarritutako arrazonamendu metodoa.

- * Ereduan oinarritutako arrazonamenduaren kontextu barruan, sistemaren modelazioa da konputazionalki konplexua dena, ez ereduaren erabilera *per se*.
- * Huts sinplearen jasokunde peko ereduaren oinarritutako diagnosia konputazionalki nahiko eraginkorra da.
- * Huts multipleko ereduaren oinarritutako diagnosia ere tratatzen da, erregelatan oinarritutako arrazonamendua eraginkorra egiten duten jasokunde sinplifikatzaile berberak emanik.
- * Horregatik, egitura funtzionaleko ereduaren jakintza heuristikoen konpilazioak ez ditu benefizio bereizgarriak eskeintzen.

Gainera hau ere argudiatu zuen:

- * Jakintzaren itxurak oso esanahi txikia du metodo baten konplexutasun konputazionalan.
- * Egitura funtzionaleko ereduaren asoziazio heuristikoen konpilazioak jakintzaren edukinaren aldaketa txiki xamarra darama, hau emanez gero.
- * Edonola, ereduari dagozkien aplikazioen konpilazioak ez ditu tratatzen diagnostiko ataza zailek egiten dituzten faktoreak.
- * Horrenbestez, jakintzaren konpilazioak ez darama eraginkortasun konputazionalera.

Eztabaidaren erdigunea eraginkortasuna den arren, atzaren jakintza bereziaren lorpena konpilazioaren bitartez ere garrantzitsua dela dirudi.

Beste ikertzaile batzu ere jakintzaren konpilazioa estudiantzen aritu dira ikuspuntu desberdinetatik, hauen artean Bylander, Dietterich, eta Tong nabarerazi daitezke.

Planteatutako arazoaren garrantzia dela eta interesgarria da kontutuan izatea aurreko ikertzaile batzu geroago egin dituzten puntualizazioak.

Bylander-ek dioenez [BYLA91] kasu bi eman daitezke jakintza mailan.

- * Jatorrizko jakintzak jakintza konpilatua inplika dezala konpilatzaileak informazio berririk sortu behar ez duelarik.
- * Jatorrizko jakintzak jakintza konpilatua inplika ez dezala eta horren ondorioz konpilazio jakintza gehitu beharko delarik.

Izan ere jakintza konpilatuak ez du izango ezer berririk jadanik jatorrizko jakintzan eta konpilazio funtzioan inplizitoki dagoena. Konpilazio funtzioak jatorrizko jakintza kode eraginkor-rago bihurtuko du, baina ez da pentsatuko ataza tratagaitzak tratakori bihurtzea.

Chandrasekaran-ek [CHAN91] JOS-ei buruz literaturan egiten diren lan bereizpenak nabarerazi ditu: ereduaren oinarritutako arazoaren ebazpena erregeletan oinarritutakoen kontra, ereduak jakintza esperientziala edo asoziazionalaren kontra, jakintza sakona jakintza konputazionalaren kontra, eta edukina itxuraren kontra; eta baieztatzen du ataza edo metodo baterako beharrezko jakintza zuzenki gertu badago, sistemak jakintza konpilatua izango duela ("bildutakoa"-ren zentzuan), bestela (kontrako kasuan) jakintza sakonetik ateratakoa izan beharko du. Gainera, diagnostikoaren arrazoiaren eredu kausalaren eredu konpilatua-goekin konbinatzeko aukera seinalatzen du.

Dietterich-ek [DIET91] espezifikazioa eta inplementazioaren artean dagoen banaketa edo hutsunea azpimarratzen du, seinalatuz AA-ren helburua metodologiak garatzea dela sistema eraginkorrak edo adimentsuak eraikitzeke, eta horregatik beharrezkoa izango da espezifikazioa eta exekuzioaren arkitekturaren arteko hutsunea (hau badagoela onartuz gero) ganditzea transformazio metodo automatiko baten bitartez: jakintzaren konpilazioa. Espezifikazioak era inkrementalean biltzen badira, erregeletan erreprestatuta, eta era interpretatiboan exekutatzen badira arazoaren

erabaki eraginkorra lortzeko azalpen erabilgarriekin; ez da aurretik esandako hutsunea egongo eta konpilazioa ez da beharrezkoa izango. Aplikazio handiagoetan mugitzen garen bezain laster hasiko da agertzen hutsunea edo banapena, hiru arrazoik eraginda:

- * Dominio batzuetan erraza da desiatutako sistemaren funtzionamendua espezifikatzea interpretatiboki exekutatu den modu oso ez eraginkor batean.
- * Badira dominioak non ezinezko den espezifikazioak era interpretatiboan exekutatzea somagarriak ez diren kantitateen terminoetan adieraztuta bait daude.
- * Espezifikazioak kontraesanak edo erizpide gatazkatsuak eduki ditzakete eta hauek hantzemanak izan beharko dute eta erabakiak exekuzioaren aurretik.

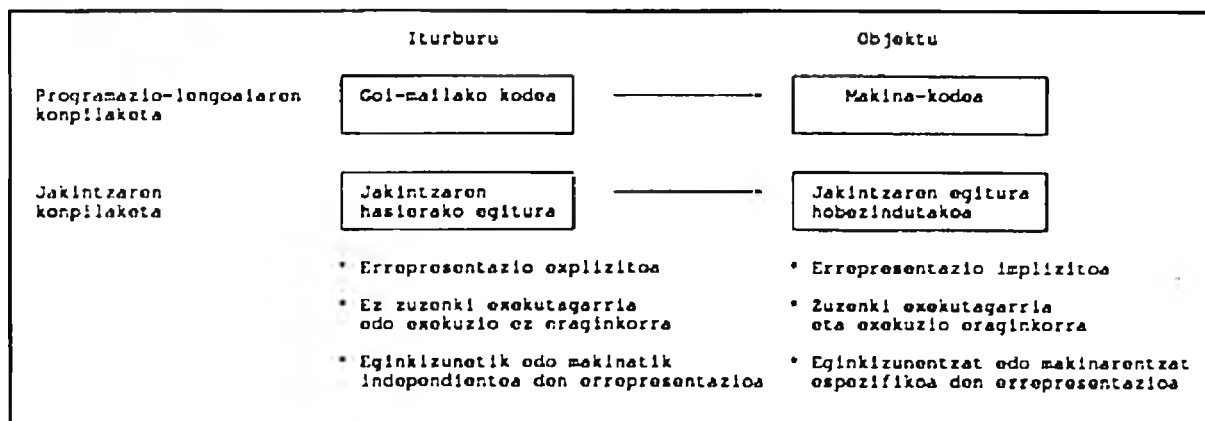
Keller-ek [KELL91], definizio sinple eta unibersalki onartu baten gabetasunen aurrean, hiru alternatiba proposatzen ditu literaturan erabilitakoen artean.

- * Jakintzaren konpilazioa aldaketa prozesua da era deklaragarri batetik jakintzaren errepresentazioaren era prozedural batera.
- * Jakintzaren konpilazioa jakintza esplizitoaren erre-presentazio ez eraginkor batetik errepresentazio inplizito eta eraginkor batera pasatzeko transformazio automatikoa da.
- * Jakintzaren konpilazioa maila altuetako espezifikazio-tatik JOS sortzeko prozesua da.

Definizio hauek elkarrekiko ezaugarri bat dute: hau da, existitzen den software-aren sistemak automatikoki beregituratzen dituen prozesu bat dela, sistema berriak eginez, eta hauek erakusten dute:

- eraginkortasunean eta erabilgarritasunean gehikuntza bat
- aldaketa errepresentazio mailan
- arrazonamendua murriztea, eta

- esplizitudea eta gardentasuna gutxitu.



3.6. Irudia.- Gol-mailako lengoai baten konpilaketa eta jakintzaren konpilaketaren artean dagoen analogia.

Edukia jagonez transformazioa hauen bitartez egiten da:

- Sinplifikazioa
- Makro-eragileen erabilketa
- Aurrekonputazioa, edo
- Eboluzio partziala.

Transformazioa, edukia aldaraziz, honen bidez egiten da:

- Atari aplikazioak,
- Kualitatezko transformazioa,
- Kostu handiko informazioaren egozketa, edo
- Hurbiltze funtzionala.

Tong-en [TONG91] ikuspuntutik jakintzaren konpilazioa hobeto karakterizatzen da hiru mailetako jerarkia baten terminoetan, bakoitzean sistema era desberdin batean deskribatuz:

- * Jakintza mailan ataza espezifikatzen da exekutatzeko erabiliko den metodoa iragarri gabe, hau iragarritz: arazoaren mota, soluzio onargarriak zehazteko erizpi-dea, eta beharko den jakintzaren dominioa.

- * Funtzional mailan sistema osagai funtzionalez deskribatzen da, sortzaileak, egiaztatzaileak, eta partxeatzaileak edo patchers. Arkitektura batzuetan konfiguratzeko dira bilaketa algoritmo batzuei errepresentatuz.
- * Programa mailan sistema deskribatzen da programazio konbenzional baten terminoetan Lisp, Prolog, edo C bezalakoetan.

Jakintzaren konpilazioak ataza baten jakintza mailaren espezifikazioa maila funtzionaleko bilakuntza algoritmora konbertitzen du aurreko mailako jakintza zuzen inplementatuz. Behin maila funtzionalaren deskripzioa eraikiz gero, programa mailaren deskripzioa sor dezaketen automatikozko programazio tresnak daude.

Jakintzaren konpilazioa lehenengo mailatik bigarrenenerako pasolari legokio gehiago eta komenigarria litzateke lehenengo mailan erabili ahal izatea espresatzeko, ulertzeko, arazteko, aldarazteko, mantentzeko, baita sistema exekutatzeko ere.

3.4.2.- Jakintzaren konpilazioa MAMUR-en.

JB eraikiko da MAMUR-en Objektuetara Orientatutako Lengoaiaren komandoen bidez, horrela klaseak, instantziak, moduluak, erregelak, metaerregelak, eta metodoak definitu ahal dira, fazetei balioak sartu, etab., komando batzuen bitartez. Lehenengo maila hau, [BYLA91] [TONG91]-etan aipamena egiten den jakintza mailaren parekidea, era interpretatiboan exekuta daiteke baina era ez eraginkorrean hauen kostuagatik:

- Sortzea, osatzea, eta objektu guztiak elkar-arremantzea, eta
- RETE zuhaitza sortzea eta bere osagarritzko atributuak MOTOR objektuan.

Konpilazioarekin bigarren maila batera iristen da, Tong-en maila funtzionalaren parekidea, aurreko mailaren aurrekonputazioaren bidez bere edukia jagonez, egitura eta instantziak sorterazteko Lisp funtzioekin errepresentatuta geratuko dena eta fitxategi multzo baten dauden balioak dagozkion atributuei esleitzuz. Maila honetan sortuko da, gainera, informazio gehigarria eraginkortasuna gehituko duena lehenengo mailatik heuristikak ateraz sistemaren kontrolari buruzko jakintza sorteraziz (ikus 6.3.9), jarraikoekin osatuko dena:

- * Modulu bakoitzaren erro nodoa.
- * Metaerregelen metamaila.
- * Egintza bakoitzen ateragarritasunari buruzko jakintza.
- * Sarbide zuzena egintza baten egiaztapena hazten den RETE zuhaitzaren nodora.

Programa mailarako pasoa automatikoki egingo da [TONG91] Lisp konpiladorearen bidez, karga azkarragoko objektu fitxategi multzo bat lortuz.

KONPILAZIO PROZESUA.

MAMUR-en objektuetara orientatuko egiturak erraztu egiten du, jatorrizko JB kargatu ondoren, fitxategi multzo bat lortzea diskoan sortutako objektu bakoitza bere atributu eta fazeta bakoitzaren edukinarekin esplizitoki definitzen duten Lisp instrukzio kopuru txiki batekin, hauxe da, fitxategiek gordetzen dituzte JB originalaren komandoen aurrekonputazioaren emaitzazko objektuak zuzenean sortu ahal dituzten Lisp instrukzioak.

JB-ren konpiladoreak urrats hauei jarraitzen die:

- 1) JB memorian kargatzen du, erregelen RETE zuhaitza eta eraginkortasuna hobetzeko aurretik espezifikatutako informazio gehigarri guztia sortuz. Kargaren emaitza objektu berrien multzo bat, egintzak eta erregelak,

eta MAMUR-eko RAIZ klaseko eta MOTOR klaseko atributu batzuen aldaraztea da.

- 2) Fitxategi nagusi bat eta fitxategi laguntzaile multzo bat sortzen du (bata egintzari dagokion objektu bakoitzeko eta bestea modulu baten objektuen multzoko) Lisp funtzioak edukiz honetarako:
 - fitxategi laguntzaileak kargatzeko.
 - objektu berrien egituren eraikuntza eta definizio kasu bakoitzeko Lisp instrukzio bakar bat erabiliz.
 - balio berrien esleipena aldarazitako atributuei.
- 3) Lisp iturburu-kodea transformatuko du, sortutako fitxategi guztiena, objektu-kodean.

Aurrerantzean fitxategi nagusia kargatzea nahikoa izango da, Lisp objektu-kodean, karga abiadura hobetuz.

Hau posible egiteko eta formatoa dela eta, zeinetan geratzen diren JB-ren hasierako kargak sortutako objektuen atributuen balioak, batez ere lambda espresioen kasuan:

```
#<SCANNED lambda espresioa> edo  
#<LISP::SCANNED lambda espresioa>
```

beharrezkoa da:

- Lisp-aren Irakurketa Taula edo Readtable aldarazi konpilazio prozesuan zehar #< makro-karaktereeekin ez jakina egin dezan, hauek ezinezkoa egiten bait dute Lisp-ak atribuen balioak irakurtzea.
- Lisp objektuaren SCANNED edo LISP::SCANNED hitzei ez jakina egin.

4. KAPITULUA: ZIHURGABETASUNA- REN MANEIUA.

4.1.- SARRERA.

SA-en JB-etan erabilia izan den informazioa normalean zihurgabea denez gero jarraiko arrazoi edo arrazoiengatik [LOPE90]: ez osoa izateagatik, ez guztiz fidagarria, kontraesane-koa, edo zehazgabekoa; errepresentatzeko orduan informazio honen zihurtasunaren gradua adierazi beharko du adituarentzako era argi eta ezagunean. Horregatik, erabiltzaileak berak zehaztuko dituen termino linguistikoak erabiltzen dira, sistema askotan erabilita-ko zenbakizko balioen ordean, MYCIN-etik [BUCH84] hona, arazo hauek ekiditeko:

- zehaztasun behartua
- erabiltzailearen eguneroko hizkuntzatik urruntzea
- elkarrengandiko askatasun eta eskusio hipotesia
- aditu eta erabiltzaileen arteko funtsik eza zenbakizko balioak ematean
- hedakuntzaren kostu konputazionala

Arrazonamendu posibilistikoaren eredua logika difusioan oinarritutakoa [ZADE75] [DUBO85a] uhertasunaren errepresentazioa haizutzen du azpimultzo difusoen bidez.

Oinarritzko egintzak $[X \text{ A da}]$ eran errepresentatzen dira, hemen X aldagai bat da U unibertsoan eta A X -i aplikagarria den termino linguistikoa, X -ek har ditzakeen balioak mugatzen dituen, zeren eta $[X \text{ A da}]$. Termino hau U unibertsoaren azpimultzo difusioan errepresentatuta dago.

Egintza baten zihurgabetasunaren errepresentazioa $[X \text{ A da}]$ posibilitate distribuzioaren funtzio baten bitartez mugatuta dator, A-ren azpimultzo difusoaren kide-funtzio edo ezaugarri-funtzioarekin identifikatzen dena, hau da posibilitate distribuzioaren funtzioa honelako funtzioa da:

$$\Pi_A: U \rightarrow [0, 1] \text{ hau } \Pi_A(u) = \mu_A(u) \text{ izanda betetzen badu}$$

hemen μ_A , A azpimultzo difusoaren ezaugarri-funtzioa da.

Π_A posibilitate distribuzioaren funtzioa $u \in U$ bakoitzari X-ek u balioa hartzeko duen aukera lotzen dio, zeren eta $[X \text{ A da}]$ eta.

Egintza konposatuak oinarrizko egintzen konjuntzio, disjuntzio eta ezeztapenen bidez errepresentatzen dira. Egintza konposatu baten posibilitate distribuzioa osatzen dituzten oinarrizko egintzen posibilitate distribuzioen bitartez kalkulatuzen dira:

$$\Pi_{A \wedge B}(u) = T(\Pi_A(u), \Pi_B(u))$$

$$\Pi_{A \vee B}(u) = S(\Pi_A(u), \Pi_B(u))$$

$$\Pi_{\neg A}(u) = N(\Pi_A(u))$$

funtzio hauek arau triangularren edo T-arauen izenak hartzen dituzte, koarau triangularrak edo T-koarauak, eta ezeztapen gogorrak urrenez-urren [LOPE90], hauen ezaugarriak 4.1 irudian ikus daitezke.

4.2 irudiak T-arau eta T-koarau arruntenak erakusten ditu dagokien ordenarekin batera [GODO89]. MAMUR-en (T_1, S_1) , (T_2, S_2) , eta (T_3, S_3) pareak baino ez dira kontutan hartzen Bonissonen emaitza esperimentalak jarraituz, behar beste desberdinak diren emaitzak sortzen bait ditu bedèratzitik gorakoa ez den termino multzo baterako.

T-arauak	T-koarauak	ezeztapen gogorrak
$T: [0,1] \times [0,1] \rightarrow [0,1]$	$S: [0,1] \times [0,1] \rightarrow [0,1]$	$N: [0,1] \rightarrow [0,1]$
$T(0,0) = 0$	$S(1,1) = 1$	$N(N(x)) = x$
$T(1,p) = T(p,1) = p$	$S(0,p) = S(p,0) = p$	$N(x) \leq N(y) \text{ si } x \geq y$
$T(p,q) = T(q,p)$	$S(p,q) = S(q,p)$	
$T(p,q) \leq T(r,s) \text{ si } p \leq r \text{ y } q \leq s$	$S(p,q) \leq S(r,s) \text{ si } p \leq r \text{ y } q \leq s$	
$T(p, T(q,r)) = T(T(p,q), r)$	$S(p, S(q,r)) = S(S(p,q), r)$	

4.1 Irudia.- T-arauean, T-koarauean, eta ezeztapen gogorren adibideak.

T-arauak	T-koarauak
$T_0(x,y) = \begin{cases} \min(x,y), & \max(x,y) = 1 \text{ bada} \\ 0, & \text{bestela} \end{cases}$	$S_0(x,y) = \begin{cases} \max(x,y), & \min(x,y) = 0 \text{ bada} \\ 1, & \text{bestela} \end{cases}$
$T_1(x,y) = \max(0, x + y - 1)$	$S_1(x,y) = \min(1, x + y)$
$T_{1,3}(x,y) = xy / [2 - (x + y - 1)]$	$S_{1,3}(x,y) = (x + y) / (1 + xy)$
$T_2(x,y) = xy$	$S_2(x,y) = x + y - xy$
$T_{2,5}(x,y) = xy / (x + y - xy)$	$S_{2,5}(x,y) = (x + y - 2xy) / (1 - xy)$
$T_3(x,y) = \min(x,y)$	$S_3(x,y) = \max(x,y)$

Honela ordenatuta daude:

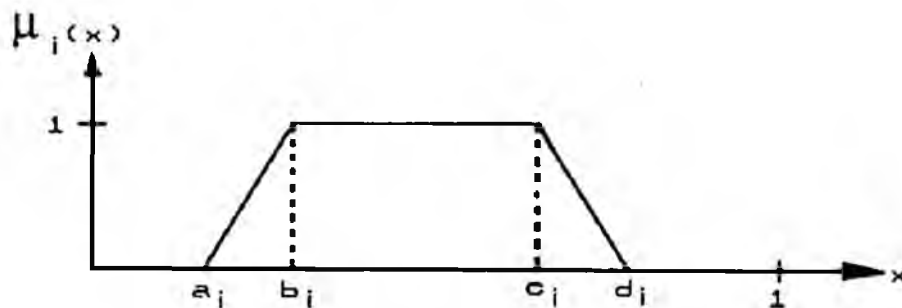
$$T_0 \leq T_1 \leq T_{1,3} \leq T_2 \leq T_{2,5} \leq T_3$$

$$S_3 \leq S_{2,5} \leq S_2 \leq S_{1,3} \leq S_1 \leq S_0$$

4.2 Irudia.- T-arau eta T-koarau arruntenak.

4.2.- ZIHURGABETASUNAREN TRATAMENDUA MAMUR-EN.

MAMUR-en inplementatu da tarte difusoetan oinarritutako hurbilketa, logika difusoan funtsatutako arrazonamendu posibilistikoaren ereduari jarraituz [ZADE75] [DUBO85a], MILORD sisteman erabilia bezalakoa [SIER89] [LOPE90], hemen balio linguistiko bakoitza tarte difuso batek errepresentatzen du, edo zehatzago, $[0,1]$ tarteko azpimultzoetako multzo difuso bateko kide-funtzio batek, Bonissone-ren lana jarraituz [BONI79]. Hau da, τ_i zihurtasun bakoitzari $\mu_i(x) \in [0,1]$ kide-funtzioa dagokio non $x \in [0,1]$.



4.3 irudia.- Funtzio trapezoidala.

Arrazoi konputazionalak direla eta, kide-funtzio bakoitza lau parametrok $\tau_i = (a_i, b_i, c_i, d_i)$ deskribatutako funtzio trapezoidal batek errepresentatzen du. $[b_i, c_i]$ azpitartea (non $\mu_i(x) = 1$) tarte difusoaren gunea da eta $[a_i, d_i]$ azpitartea euskarria. Gunea puntu batera murrizten bada tarte difusoa zenbaki difuso bihurtzen da.

Errepresentazio honen interpretazio semantikoa zihurtasuna adierazten duen termino linguistikoaren egokitasunean oinarritzen da, zenbaki balioei dagokienez gunean eta euskarrian, horregatik honako arazo hauek aurki daitezke:

- * Sarritan adituek ezin dezakete balio linguistikoaren esanahia definitu zenbaki balio eskala baten bidez.
- * Aditu desberdinak desakordioan egon daitezke balio linguistiko baten edo batzuen errepresentazioarekin.

Hala ere, alderdi positiboak oso interesgarriak dira:

- * Eraginkortasun konputazionala, egiazko-taularen aurrekonputazioagatik zihurgabetasunaren hedakuntzarako.
- * Hedakuntza berezia lortzeko aukera termino multzo partikular batentzako, adituak aukeratu ahal duena. Errepresentazio baikorra eta ezkorra batera haizutuz, euskarria era ezkorrean definituz eta gunea era baikorrean, taula desberdinak sortuz T-arauak eta T-koarauak definitutako konjuntzio, disjuntzio eta inplikazio eragiketarako eta balio desberdinak sortuz β hurbilketa parametroarentzat [SIER89].
- * Beste aukera baterako urrats bat da Etiketa linguistiko Difusoaren erabileran datzana [GODO89], edo zihurgabetasunaren beste errepresentazio era batzuen erabileran termino linguistikoaren bidez terminoekin eurekin eragitea haizu dezana azpian datzan zenbakien esanahiaren ordez.

4.4 irudian ikus daitezke bederatzi termino linguistikook dagokien balioarekin, apostotzat edukiak PNEUMONIA [SIER89] sistema adituaren eraikuntzarako ikerketetan medikuntza aplikazioarako eta MAMUR-en erabiliak defektuz.

Zenb. Terminoak	Trapezioko balioak
1 Imposible	(0,00 , 0,00 , 0,00 , 0,00)
2 Muy poco posible	(0,00 , 0,00 , 0,05 , 0,08)
3 Ligeroamente posible	(0,05 , 0,07 , 0,14 , 0,17)
4 Moderadamente posible	(0,10 , 0,15 , 0,35 , 0,45)
5 Posible	(0,25 , 0,35 , 0,55 , 0,65)
6 Bastante posible	(0,45 , 0,55 , 0,75 , 0,85)
7 Muy posible	(0,65 , 0,75 , 1,00 , 1,00)
8 Casi seguro	(0,95 , 0,98 , 1,00 , 1,00)
9 Seguro	(1,00 , 1,00 , 1,00 , 1,00)

4.4 irudia.- Defektuz hartutako trapezioen zenbakizko balioak.

Termino linguistiko hauek premisen, ondorioen, eta erregelen zihurtasun gradua adierazteko erabiliko dira, arrazonamenduan kontutan izanik zihurtasun gradu egokia esleitzeko

parte hartu duten, premisekin, ondorioekin, eta erregelekin bat etorritz sortutako egintza berriei.

Zihurtasun linguistiko balio hauek hedatu eta konbinatzeko dedukzio prozesuan zehar, beharrezkoa da haien gainean T-arauak eta T-koarauak definitutako konjuntzio, disjuntzio, eta inplikazio eragiketak exekutatzeari. Hauen artetik MAMUR-ek defektuz honako hauek erabilko ditu:

- * T3 konjuntzioetarako, eta
- * S3 disjuntzioetarako.

Aritmetika difusoaren arauak erabiltzen dira [DUB085b] (4.5 irudia) $I = (a, b, c, d)$ eta $I' = (a', b', c', d')$ tarte difusoei eragiteko.

Baina termino linguistiko bien trapezioen balioekin eragitearen emaitzak ez dituenetz beti emango aurretik definitutako beste termino linguistiko baten trapezioaren balioak, emaitza hurbilenari hurbildu beharko zaio [BONI79], eta hau bi eratara egin daiteke:

$$\begin{aligned} I \div I' &= (a \div a', b \div b', c \div c', d \div d') \\ I - I' &= (a - a', b - b', c - c', d - d') \\ I \times I' &= (a \times a', b \times b', c \times c', d \times d') \\ I / I' &= (d / a', c / b', b / c', a / d') \\ \min(I, I') &= (\min(a, a'), \min(b, b'), \min(c, c'), \min(d, d')) \\ \max(I, I') &= (\max(a, a'), \max(b, b'), \max(c, c'), \max(d, d')) \end{aligned}$$

4.5 irudia.- Aritmetika difusoaren arauak.

- * Eragiketa bakoitzaren osteko hurbilketa.
- * Hurbilketa bakar bat egintza bat deduzitzeko beharrezkoa den azken eragiketaren ostean.

Lehenengo aukerak hurbilketa errakuntza gehitzeko desabantaila du erregelen kateatzean zehar; hala ere, bere abantail nagusia behin adituak balio linguistikoa definitu eta gero konjuntzio, disjuntzio eta inplikazio eragiketak batera berkonputatu daitezkeela zihurtasun linguistikoko balioen guztientzako eta emaitzak gorde daitezkeela egiazko taulen gisa erabiliko diren matrizeetan. Zihurtasunaren hedakuntza eta konbinaziorako MAMUR-en prozasu deduktiboan zehar, konpilazio prozesuan aurretik sortuak

eta adituak onartuak izanak diren taula hauek kontsultatzeko bakarrik erabiliko dira.

Kalkulatutako balioen eta termino linguistiko bakoitzaren arteko distantzia kalkulatzeko multzo difusoen parametro batzuk erabil daitezke, eta hauen artean nabarmentzen dira: Potentzia, Entropia, Lehenengo momentua, eta Soslaia MAMUR-en, terminoen ordena eta solapamentu ezaren ezaugarriengatik, nahikoa da Potentzia eta Lehen momentoarekin 4.6 irudian ikus daitezke erabilitako formulak Potentzia, Lehen Momentua, eta distantziaren kalkulurako [SIER89]. 4.7 irudian konjuntzio eta disjuntzio orokorreko matrizeak erakusten dira, defektuz erabiltzen direnak.

Potentzia:

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

Lehen momentua:

$$M(A) = \left\{ \begin{array}{ll} \frac{1}{3} \frac{(c^2 + d^2 + cd) - (a^2 + b^2 + ab)}{(c + d) - (a + b)} & c + d \neq a + b \text{ bada} \\ a & \text{bestela} \end{array} \right\}$$

Distantzia:

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

4.6 irudia.- Potentzia, Lehen momentua, eta distantziaren formulak.

Azkenean IM-ak matrize bi eduki beharko du:

- * Konjuntzio matrize bat premisen baldintzaren eta erregelaren zihurtasun balioaren hedatutako zihurtasun balioa lortzeko.
- * Disjuntzio matrize bat ateratako egintzaren zihurtasun balioa lortzeko hedatutakoarekin, aurreko matrizen bidez lortua, eta ondorioaren zihurtasun balioarekin batera.

Konjuntzio matriza
(T_3 con $\beta = 0.5$)

Disjuntzio matriza
(S_3 con $\beta = 0.5$)

1 2 3 4 5 6 7 8 9

1 2 3 4 5 6 7 8 9

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

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

1 *Imposible*
 2 *Muy poco posible*
 3 *Ligeramente posible*
 4 *Moderadamente posi.*
 5 *Posible*
 6 *Bastante posible*
 7 *Muy posible*
 8 *Casi seguro*
 9 *Seguro*

4.7 Irudia.- Sortutako matrizeak.

JB-an termino linguistikoak eta dagozkien balioak definitu daitezke funtzio hauen bidez:

```
(defterm-ling ((imposible 0 0 0 0)
               .....
               (seguro 1 1 1 1)
               ))
```

Aholkugarria da termino linguistikoen kopurua 7 ± 2 izan daiten Miller eta Beyth-Marom-ek egindako estudio psikologikoaren arabera.

Zihurtasun hedakuntzarako matrizeak DEFMAT-CERT funtzioaren bidez deklaratzeko dira parametro hauekin:

- * Erabili behar den T-araua (T_1 , T_2 , o T_3) edo T-koaraua (S_1 , S_2 , o S_3).

- * Distantzia koefiziente β -ren balioa, Sierraren lanaren arabera [SIER89] $[0.2, 0.5]$ rangoan kokatutako balioek izan behar duena.

$$(\text{defmat-cert } \begin{Bmatrix} T_i \\ S_i \end{Bmatrix} \beta)$$

5. KAPITULUA: JAKINTZAREN MANIPULAZIOA.

5.1.- SARRERA.

OOP-ak ekarten dituen abantailak, eta detektatuko oztopoen gauditzea dela eta (ikus 2.3.2), JOS-aren eraikuntzan masiboki erabiliak izan dira, horregatik sistemaren tamaina eta zailtasunaren gehipenak ia ezinbestekoa egiten du haren erabilpena.

Behin jakintzaren manipulaziorako fokatzea finkatuz gero eta berarekin bat etorriz jakintzak errepresentatuz gero; inferentzia mekanismorik onena hautatu beharra dago, sistemaren arrazonamendua eraginkorra izan dadin lortzeko; eta gainera, prozesuan jarraitutako urratsen azalpen argia erraz dezan.

5.2.- IRAGAZKETA EDO MATCHING.

Iragazketa etapa hau da denbora gehien erabiltzen duena EOS baten exekuzioan, % 90 inguruan [GUPT89], horregatik beharrezkoa da iragazketa algoritmo eraginkor bat edukitzea, zenekin PS prozesuen moteltasuna ekidin. Edozein iragazketa algoritmoaren helburu nagusia erregela guztien ezkerreko aldean ebaluazioa egitea eta gatazka multzoa osatzen duten instantziazioak lortzea da.

Kasu honetan Forgy-ren RETE algoritmoa [FORG82] erabili da aurreko helburua era eraginkorrean lortzen duelako erregela eta/edo EGB-ko egintza askorekin, honako arrazoi hauengatik:

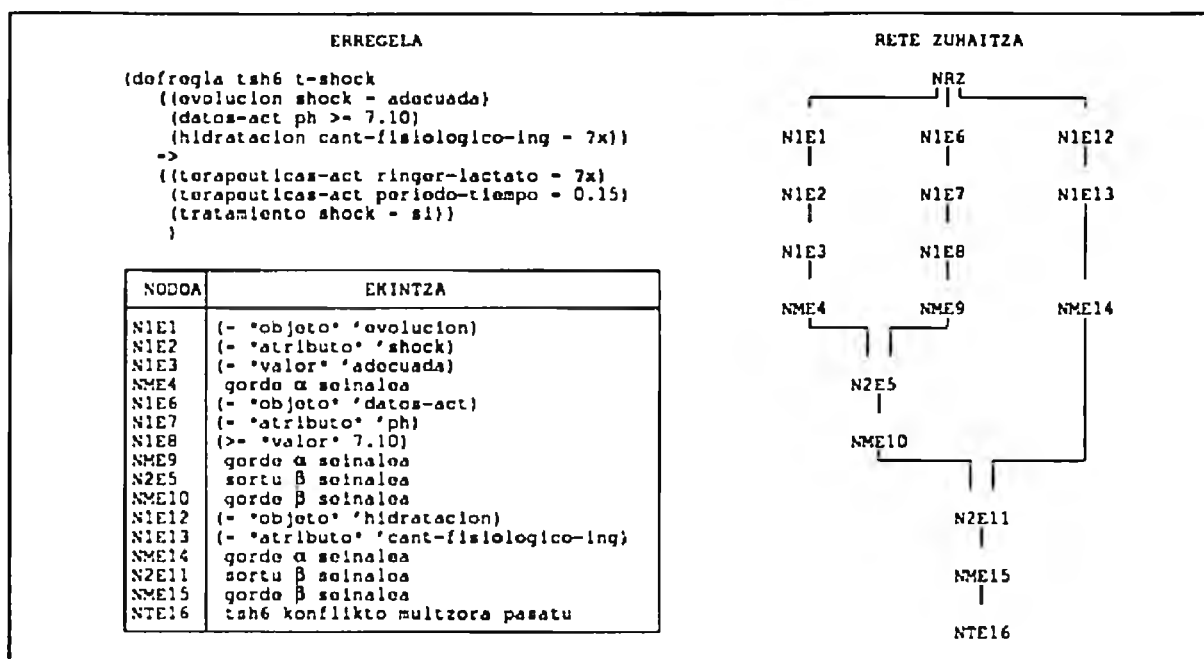
- EGB-ko aldaketak baino ez ditu prozesatu behar.
- Adierazpen arruntak ebaluatzen dituzten egiaztapenak konpartitzen ditu.

horregatik izan da inplementatua lengoai edo tresna gehienetan, eta Gupta, Forgy eta Newell-ek egindako estudio baten arabera [GUPT89] oso eraginkorra dela uste da uniprozesadoreei dagokionez eta prozesadore paraleloi dagokienez ere bai. Geroago beste algoritmo batzu agertu dira, beragandik eratorriak, eta hauen artean aipamen berezia merezi duen TREAT 1987an Miranker-ek aurkeztua [MIRA90].

RETE algoritmoak zuhaitz konpilatua erabiltzen du datuen jarioaldirako premisei edo erregelen ezkerretako aldeek osatzen dute, RETE zuhaitza izendatzen dena. 5.1 irudiak erregela bati dagokion zuhaitza erakusten du. Zuhaitzaren nodoek iragazketa prozesuan zehar exekutatu behar diren eragiketa abstraktoak errepresentatzen dituzte eta interpretazio garaian interpretatzen dituzte interpreteak edo IM-ak. Nodo artean pasatzen diren objektuak seinaleak (tokens) izena dute eta bakoitzak osagai bi ditu:

- Seinalea zuhaitzari gehitu edo ezabatu behar zaion iragortzen duen eragiketa etiketa (tag) bat, hala ere erregela baten gehiketa edo ezabaketa gatazka multzoan.
- Egintza zerrenda bat, zerrendaren elementu bakoitza erregelaren baldintza bakoitzarekin parekide delarik eta orden beratan.

Seinale bakoitzak baldintza bat errepresentatzen du, premisena edo erregela baten ezkerretako aldeena, eta honek egiaztatua izan behar du dagokion erregelaren iragazketa lortzeko. Seinaleak RETE zuhaitzaren erro nodotik bidaltzen dira, beherantz, EGB-an aldaketaren bat gertatzen denean eta zuhaitza jarraituko dute bere test-a bete ezin daitekeen nodo batera iritsi arte edo azkenerarte. Erregelen hautaketa-ekintza



5.1 Irudia.- Tsh6 erregelaren RETE zuhaitza.

α seinaleak:	
formatoa	seinaleak
(nombre-token tag (obj. atrib. valor val-cert.))	(ALF1 (+ (EVOLUCION SHOCK ADECUADA 9))) (ALF2 (+ (DATOS-ACT PH 7.14 9))) (ALF3 (+ (HIDRATAACION CANT-FISIOLOGICO-ING 100 9)))

β seinaleak:	
formatoa	seinaleak
(nombre-token- β tag ((obj-1 atrib-1 valor-1 val-cert-1) (obj-n atrib-n valor-n val-cert-n)) valor-corteza-promisas)	(BET4 (+ ((EVOLUCION SHOCK ADECUADA 9) (DATOS-ACT PH 7.14 9)) 9)) (BET5 (+ ((EVOLUCION SHOCK ADECUADA 9) (DATOS-ACT PH 7.14 9) (HIDRATAACION CANT-FISIOLOGICO-ING 100 9)) 9))

5.2 Irudia.- Tsh6 erregelaren α eta β seinaleak.

zikloaren hasieran seinaleak momentuan gehitu zaion edo EGB-tik ezabatua izan den egintza (α seinalea) bakar bakarrik dauka. Seinalearen jarrialdian zehar zuhaitzaren bidez egintza berriak gahituko zaizkio, zerrendari, zeinek parakatuko diren dagozkien baldintzekin (β seinalea).

Zuhaitzean lau nodo mota ager daitezke:

- * Erro nodoa: Zuhaitzaren goiko aldean kokatuta dagoen nodoa da, EGB-aren aldaketak zuhaitzan zehar hedatzen dituen.
- * Sarrera bateko nodoak [FORG82] edo konstanteen egiaztapen nodoak [QUIN85][GUPT89]: Hauen bidez pasatzen diren α seinaleak dagozkien erregelaren baldintzaren konstanteak betetzen dituzten egiaztatze erabiliak dira. Nodo hauek RETE zuhaitzaren goiko aldean agertzen dira eta konpartituak dira baldin eta ezkerreko alde batzuk nodo berdin berdinak eskatzen badituzte, nodo bikoiztuak eraiki beharrean.
- * Memoria nodoak: Nodo hauek ziklo batzuen iragazketaren emaitza gordetzen dute egoera bat gisa. Hauetariko egoera bakoitzak asoziatutako erregelaren ezker aldeko baldintza bat edo batzu kunplitzen dituen seinale zerrenda bat errepresentatzen dute. Dituen seinaleen arabera izan daitezke:

- α memoria nodoak: erregelaren baldintza bat kunplitzen duten seinaleak dituzte.
- β memoria nodoak: baldintza batzu batera kunplitzen dituzten seinaleekin.

Nodo mota biek balio dute heltzen diren seinaleen kopiak gordetzeko edo ezabatzeko etiketaren zeinuaren ("+" edo "-") arabera, lehenengo kasuan seinalea zuhaitzean hurrengo nodoetara pasatuko da.

- * Sarrera biko nodoak: Nodo hauek α edo β seinale biren baterako betetzea egiaztatzen dute erregela beraren baldintza elementu bi (edo gehiago) betetzen duen zati instanziario bat osatzen duen β sinale berri bat sortuz. Sarrera biak memoria nodo bitik datoz. Sarre-retariko baten memoria nodotik datorren seinale bakoitza kontrako sarrerari konektatutako memoria nodoaren seinale guztiekin konparatuko da. Aldagai lotura (binding) sendoak dituzten seinale pare guztiak, eta β seinaleak sortuz, bidaliko dira sarrera biko nodoari jarraitzen dion memoria nodora, AND nodo bat bada; NOT nodo bat bada pasatuko seinalea eskuine-

ko sarreran seinale sendorik ez baldin bada. Eskuine-tako sarrera beti α motako memoria nodo bati konektatuta dago.

- * Nodo terminalak: Nodo bat dago erregela bakoitzari asoziatua eta RETE zuhaitzaren beheko aldean dago. Seinale bat mota honetako nodo batera iristen bada erregela bete egiten dela esan nahi du eta gatazka multzoan txertatua edo honetatik ezabatua izan behar du.

Egiaztapen berdinen exekuzioa behin eta berriro errepika ez dadin ekiditeko, RETE algoritmoak gordeko du egintza ezagunen iragazketa egoera baten gisa memoria nodoaren barruan. Honela, EGBn oraintxungo erregelaren aktibazioa edo desarrak sartzen dituzten aldaketak bakarrik prozesatuko dira ziklo bakoitzean. EGBko aldaketa bakoitza RETE zuhaitzaren bidez iragaziko da gordetako egoera zuhaitzean barruan eguneratuz. Zuhaitzaren helburua gatazka multzoko aldaketen espezifikazioa da.

5.3.- RETE ALGORITMOAREN INPLEMENTAZIOA MAMUR-EN.

MAMUR-en ezaugarriak lgo kapituluaren erakutsiak ikusita (1.3), RETE algoritmoa ez da bakarrik iragazketa eraginkor bat lortzeko erabili baizik eta gainera:

- * JB-aren garaketa modularrerako oinarria da; bai dominioaren mailaren moduluetan eta metamailarenetan bai aurreko modulu bakoitzaren metaerregelen moduluetan.
- * JB-aren konpilazioa ahalbideratzen du.
- * Erregelen erabilera mugapen gisa ahalbideratzen du behin zuhaitza bananduz gero. Gainera zuhaitz honek

RETE zuhaitzaren sorkuntzarako eta erabilerarako prozedurak MAMUR-en:

- * Behin JB kargatuz gero erregela guztien premisen egitura dituen RERE zuhaitza sortu beharko da.
- * Baldintza batek iragazketa aldagaiak baditu, atributuaren eta balioaren posizioetan agertu ahal izango dira soil-soilik, hauek ez dute nodorik sortutako RETE zuhaitzan eta baldintza horren iragazketan zehar atributo edo balio posible guztiak hartuko dira kontutan objektu horren berezko atributuentzat dago-kion bi sarreratako nodoan. Iragazketa aldagaiak hartzen dituzten balioak erregela horrentzat baino ez dira baliogarriak.
- * Behin RETE zuhaitza sortuta exekuzioa has daiteke hasierako egintza ezagunetatik hasita zuhaitza korrituz EGB-en egintza gehitu edo ezabatu bakoitzeko, aurrerantz edo atzerantz kateaketa motaren arabera.
- * MAMUR-en modulo baten erregelak erabil daitezke une jakin baten, aurrerantz edo atzerantz doan kateaketaz prozesatuz.
- * Konpilazioak honako hauen barneko errepresentazioa haizutzen du:
 - JB-an dauden objektu guztiena.
 - RETE zuhaitzaren off-line sorkuntzatik ateratako informazioa duten atributuena.

5.4.- MAMUR-EN ARRAZONAMENDU PROZESUA.

Arrazonamendu prozesua egin ahal izango da bai era interpretatuan bai era konpilatuan.

JB-aren barruan daude era deklaragarrian hala dominioaren jakintza nola metajakintza, hau da, sistemak bere funtzionamendu zuzenerako beharko duen jakintza guztia [PIKA92c].

Prozesua mailarik altueneko moduloa edo planifikazio moduloaren exekuzioaren eskaerarekin hasiko da (aurreranzko kateaketarekin), zeinek bere aldetik gainerontzeko moduloak eskatu erazoko dituen hasierako ekintzen, eta sortuko direnen, balioen arabera era deklaragarrian iragarritako kateatze mota egokiaz.

Metamailako moduloak eta metaerregelen moduloak aurreranzko kateaketaz exekutatu dira eta dominioaren moduloak atzeranzko kateaketaz. Moduloen maila multipleen exekuzio prozesuaren kontrola 6. kapituluan zehazki azalduko diren mugapenetan oinarrituta dago.

Aurreranzko kateaketa JB-an komando honen bidez eragingo da:

(adelante { moduloaren izena })
RAIZ
NIL

eta RETE zuhaitza aurrerantz korritzea suposatzen du honetatik hasita:

- iragarritako moduloari dagokion erro nodoa, edo
- zuhaitza osoaren erro nodoa, iragarritako moduloa RAIZ edo NIL bada.

egintza berri bat sortzen den bakoitzean gatazka multzoari erregela berria gehitu ahal zaiolarik. Erregela bakoitza baten baino gehiagotan exekutatu ahal da, premisetako baldintza bat beteko duen seinale bat dagoen bitartean. Hasierako egintzak dagozkien objektuetatik hasiko dira jarraiko orden honetan:

- 1) Balio fazetaren edukia.

- 2) Esleitutako metodoa exekutatzuz.
- 3) Defektuzko balioa harturik.
- 4) Aurreko objektuetatik heredatuz.

Exekutatu daitezkeen erregela bat baino gehiago ba daude gatazka multzoan, dagokien ordenean aktibatuz joango dira kontzeptu hauen arabera:

- 1) Erabilgarritasuna
- 2) Espezifikizitatea
- 3) Zihurtasuna.

Atzerantzko kateaketa komando honen bidez lortuko da:

(atras moduloa 1-helburua 2-helburua n-helburua)

honek suposatzen du desiatutako helburu (objektu bat eta atributo bat dituen zerrenda) bakoitzaren dedukzioa egiten ahalegintzea honako prozedura hau jarraituz eta adierazten den ordena errespetatuz:

- 1) Hasierako egintzen basean ea erabilgarri dagoen zihurtatuz.
- 2) JB konsultatuz, lehenago deduzitu balitz ez balitz.
- 3) Helburu honi esleitutako metodoa exekutatzuz.
- 4) Defektuzko balioa hartuz.
- 5) Herentziaz.
- 6) Erregela bat aktibatuz, aurreko kontzeptuaren arabera aukeratua, seinalatutako moduloan helburua lortu dezaketen erregelen artean.
- 7) Erabiltzaileari galdetuz, dagokien fazetan galdera bat baldin badago.

Erregela bat aktibatzeak RETE zuhaitza atzerantz korritzea suposatzen du hautatuko erregelaren nodo terminaletik hasita. Premisen baldintzak betetzen badira ondorioak edo erregelaren eskuinetako aldeak aktibatuko dira, RETE zuhaitza eguneratuko duten egintza berriak sortuz. Premisetako baldintzaren bat

betetzen ez bada prozedura berberari jarraituz atera beharko den helburu berria sortuko da.

5.5.- ARRAZONAMENDUAREN AZALPENA MAMUR-EN.

Lortutako helburu bakoitza, bai aurreranzko kateaketan bai atzeranzkoan, asoziazio zerrenda batean gordeko da, eta honetan helburu bakoitzeko edo objektuaren eta atributoaren izenak eratutako zerrenda bakoitzeko, honako informazio hau biltzen da:

1) Nola lortu zen:

- Nondik atera zen erregela osoa.
- Datu ezaguna bada, hasierako egintza.
- Erabiltzaileari galdetua izan bada, ateragarria ez izatearren baina bai galdegarria.

2) Bere balioa.

3) Dagokion zihurtasun balioa.

Behin prozesu deduktiboa bukatuz gero erabiltzaileak nahi duen helburuaren azalpena eska dezake jarraikoa pantailan lortuz: balioa, zihurtasun balioa (termino linguistiko batek adierazia), iragarritako helburua deduzitu izan zen era, eta bera lortzeko deduzituak izan behar izan diren helburu parzial guztiak hasierako egintzetara edo erabiltzaileak prozesu deduktiboa eragitean emandako hasierako datuetara heldu arte.

Ateratzen den informazioak ekidingo du sistema "kutxa beltz" batek bezala jokatzearabiltzailearentzat eta jakintzaren ingeniariari jarraitutako arrazonamenduaren baliogarritasuna edo ez ezagutzen haizutuko dio.

Edozein SA-k azalpen modulo bat eduki behar du eta erabiltzailearekin erlazionatzen diren sistemaren zati guztiak

bezala aurkezpen atsegina izan behar du, horretarako sarrera-irteera interfacea erabiliz.

5.6.- SARRERA-IRTEERA INTERFACEA.

Erabiltzaile eta SA-ren arteko harremana atsegina eta erraza izan dadien haizutzen duten funtzio multzoa da, pantailaren ahalmen guztiak erabiliz koloreei eta informazioaren distribuzioari leihotetan dagokionez, eta saguaren erabileraren bidez aukera egokiaren hautaketarako erabiltzaileari aurkeztu ahal zaizkion menu ugarien artean.

Menuak Pop-up motakoak dira saguarentzat zentzudun alorren definizio batzuetarako egitura bat izanik, bai pantailaren goiko barran bai menu zabalduetan, hauek zutabe bat baino gehiago erabili ahal izanik. Menu zabalduak okupatzen duten pantailaren alorrak memorizatuko dira hasierako egoerara itzuli ahal izateko.

Zentzudun alor bakoitzeko espezifika daiteke:

- Zentzudun testua.
- Dokumentazio lerroaren informazioa.
- Zein ekintza hartu behar den aukera hau hautatzen bada.
- Tokia pantailan.
- Zentzudun espazioaren neurriak.

Datuen aurkezpena leiho zentral biren bidez egingo da, lehenengoak lerro baten bigarrenetan erakusten den informazioaren datu identifikatzaileak edukiz. Gainera beheko leiho bat edukiko da, lerro bitakoa, abisuak eta errakuntza mesuak erakusteko.

6. KAPITULUA: KONTROLA.

6.1.- SARRERA.

MAMUR-en kontrol egituran honako hau lortu nahi da:

- 1) Eraginkortasun konputazionala, sailkapen arazo konplexuei aurre eman ahal izateko emaitza bete beteekin. Mugapenen betetzean oinarritutako kontrol estrategia jerarkiko baten bidez lortzen da. Estrategia hau dominioaren jakintzaren erabilerak zuzentzen duen metamaila batek dauka.
- 2) Sinpletasuna eta malgutasuna metamailaren jakintzaren errepresentazioan, zeinen erregelek mugapenak errepresentatzen bait dituzte. Erregela hauek jerarkikoki moduloetan biltzen dira; mailarik altuenetakoak exekutatu beharreko plana edo ataza dauka, eta gainerontzekoek bilatzekoespazioa mugatzen duten azpiplan edo azpiataza batzu kasu bakoitzerako beharrezkoa den jakintza erabiltzeko. Esperientziak CETODIA sisteman erakutsi digu adituak era berdinean jokatzen duela.
- 3) JB-aren eraikuntza eta mantenua erraztu.

6.2.- MUGAPENETAN OINARRITUTAKO KONTROLA.

2. kapituluari kontrol arkitektura desberdinak estudiantu ziren (2.4), sistema honetan mugapenetan oinarritutako arkitektura erabili da eraginkortasun konputazional handiagoa eta kontrol mekanismo errazago eta gardenagoa lortzeko jakintza- ingenieria- rentzat, ondorengo SA-aren mantenimenduan pentsatuz.

6.2.1.- Definizioak.

Mugapenak era desberdinetan defini daitezke. Hemen jakintzaren errepresentazio formalismo batean pentsatuz egingo dugu, formalismo hauek oinarritzko elementuak ditu, zeinekin gero egitura handiak eraikiko diren. AA-ren ikuspuntutik, mugapenak interesgarriak dira sare eta konputazio lokalarentzat.

MUGAPENAK.

Mugapen bat era abstrakto batean honela defini daiteke:

Mugapen bat aldagai multzo bat eta aldagai honen gaineko erlazio bat da.

Mugapenen laguntzarekin errepresentatu beharreko erlazioak oso konplexuak izan daitezke. Mugapen baten barruan errerepresentatu ahal badira ere, mugapen hau laster opakoa egingo da tamainua dela eta. Mugapen batek, horregatik, kontutan hartutako sistema-ren aspektu partzialak baino ez lituzke deskribatu beharko, adibidez garden eta erraz ulertzekoak diren aspektuak.

MUGAPEN SAREAK.

Sistema bat, multzo bat bezala, ezin du gutxi gora behera alkarren ondoan dauden mugapen indibidualaren multzo batek deskribatu. Mugapen indibidualaren interkonexioa lortuko da mugapen aldagaien identifikazioaren bidez. Mugapen independienteen kolekzio baten ordeztu, mugapen sare bat lortuko dugu, honela definitzen dena:

Mugapen sare bat konpartitutako mugapen aldagaien multzo batek osatzen du.

Mugapen sarea terminoa Montanari, Mackworth, eta Freuder-ek [MONT74] [MACK85] [FREU82] erabilitako **mugapen grafoarekin** bat dator.

FUNTSEZKOTASUN LOKALA ETA OROKORRA.

Mugapen sareak betetzeko sarritan erabilitako teknika bat hedapen lokala da: mugapen batek kalkulaturako emaitzak haren auzoko mugapenei hedatuak dira sarean. Hedakuntza lokalaren emaitza estaldura (covering) edo balio onargarrien rangoa da, lokalki funtsezkotasuna duen aldagai batena. Funtsezkotasun lokalaren kontzeptua deskribatzeko lehenengo balioen tupla edo aldagai kopuru bera duen multzo bati esleigarriak diren balio multzo batek betetako mugapen bat definituko dugu.

Definizioa: (Mugapenen Betetzea).

Izan daiten n aldagai kopurua mugapen sare batean, $(d_1, \dots, d_n)$ balio sinpleen tupla bat sarearen aldagaientzat, eta C mugapen bat i_1 -garren, i_2 -garren, ..., eta i_1 -garren sareko aldagaiari dagozkiona ($1 \leq n$). C $(d_1, \dots, d_n)$ -k betea da, baldin eta kunplitzen badu:

$(d_{i_1}, \dots, d_{i_n})$ C -ren erlazioko elementua da.

Orain defini dezakegu lokalki funtsezko aldagaiaren estaldura honako era honetan:

Izan daiten n aldagaien kopurua mugapen sare batetan $\{C_1, \dots, C_m\}$ mugapen multzo batek emana, eta $v_1, \dots, v_n$ balio multzoa sarearen aldagaiarentzat. $(v_1, \dots, v_n)$ tupla lokalki funtsezko aldagaien estaldura da mugapen sarearentzat, baldin eta jarraiko hau kunplitzen ba du:

$$\begin{aligned} &\forall i \in \{1, \dots, n\} \forall d_i \in v_i \forall C \in \{C_1, \dots, C_m\} \\ &\exists d_1 \in v_1, \dots, d_{i-1} \in v_{i-1}, d_{i+1} \in v_{i+1}, \dots, d_n \in v_n : \\ &C(d_1, \dots, d_n)\text{-k betetzen du.} \end{aligned}$$

Nahiz eta hedakuntza lokala oso teknika erabilia den [GÜSG89b], beste teknika batzu garatu dira, hauetariko batzu aldagai orokorki funtsezkoen estaldura kalkulatzeko, zeinek honela definitzen bait dira:

$(v_1, \dots, v_n)$ tupla bat aldagai orokorki funtsezkoen estalki bat da $\{C_1, \dots, C_m\}$ mugapen multzo batek emandako mugapen sare batentzako, baldin eta jarraiko hau kunplitzen bada:

$$\begin{aligned} &\forall i \in \{1, \dots, n\} \forall d_i \in v_i \\ &\exists d_1 \in v_1, \dots, d_{i-1} \in v_{i-1}, d_{i+1} \in v_{i+1}, \dots, d_n \in v_n \\ &\forall C \in \{C_1, \dots, C_m\} : \\ &C(d_1, \dots, d_n)\text{-k betetzen du.} \end{aligned}$$

Lokalki funtsezko estaldura eta orokorki konsistentea dena berdinak izango dira baldin eta sareko aldagai guztiak behin baino zehaztuak ez badira [FREU82].

Intuitiboki, mugapen sare baten soluzio bat balio sinpleen tupla bat da, zeinek mugapenak betetzen dituzten.

Puntu honetan interesgarria da Montanari, Mackworth, eta Freuderen teoriak berrikustea, funtsezkotasun lokalari eta konputazionalki eraginkorrak diren mugapen sareen lortzeari

buruzkoak zuhaitz itxuran egituratuak bezalakoak, atzera jotzearen behar izanik ez dutelako.

Freuderen arabera [FREU82], **mugapen sare ordenatu** bat nodoak linealki ordenatuak dituen da atzera joz bilaketa algoritmoek exekutatu aldagaien esleipenen sekuentzia isladatzeko. Nodo baten **zabaltasuna** nodo honetatik aurreko nodoetara norabidetzen diren arkuen kopurua da, **ordenazio baten zabaltasuna** bere nodo guztien zabaltasun gehienezkoa da, eta **sare baten zabaltasuna** sarearen ordenazio guztien zabaltasun gutxienezkoa da. Freuderrek, algoritmo eraginkor bat proposatu zuen sare baten zabaltasuna eta zabaltasun honi dagokion ordenazioa lortzeko. Gainera, erakutsi zuen mugapen sare bat zuhaitz bat dela bakarrik baldin eta zabaltasuna 1 bada.

Montanari [MONT74] eta Mackworth-ek [MACK77] mugapenen arteko funtsezkotasan lokal mota bi sartu dituzte **arkuen funtsezkotasuna** eta **bideen funtsezkotasuna** deitutakoak. Haien definizioak hartzen dute sarea norabidetua dagoela, adibidez, mugapen simetriko bakoitza norabidetutako arku bik erreprezentatzen dute.

Izan dadin D_i V_i aldagaien dominioa; $R_{ij}(x,y)$ -k esan nahi du R_{ij} mugapen esplizitoak (x,y) haizutzen duela V_i eta V_j -ren artean.

Mackworth-en arabera, (V_i, V_j) norabidetutako arkua funtsezko arkua da (**arc consistent**) edozein balio $x \in D_i$ -rako eta $y \in D_j$ balioa baldin badago horrela $R_{ij}(x,y)$ izanik.

Montanariren arabera, m luzariko bide bat $(i_0, i_1, \dots, i_m)$ nodoetan zehar funtsezko bide bat da (**path consistent**) baldin eta edozein balio $x \in D_{i_0}$ e eta $y \in D_{i_m}$ $R_{i_0 i_m}(x,y)$ izanik, honelako balio zekuentzia bat badago

$$z_1 \in D_{i_1}, \dots, z_{m-1} \in D_{i_{m-1}}$$

$$R_{i_0 i_1}(x, z_1) \text{ y } R_{i_1 i_2}(z_1, z_2) \text{ y } \dots \text{ y } R_{i_{m-1} i_m}(z_{m-1}, y) \text{ izanik}$$

$R_{i_0 i_m}$ erlazio unibertsala ere izan daiteke, bikote posible guztiak haizutuz.

Mugapen sare bat arku (bide) funtsezkoa da baldin eta honen arku (bide) norabidetu bakoitza arku (bide) funtsezkoa bada. **Arkuen funtsezkotasunera** heltzeak aldagai jakin batzuen dominioen balio jakin batzu deuseztatzea esan nahi du, ondorengo arkua funtsezko arkua izango delarik, soluzio multzo orokor bera errepresentatzen den bitartean. **Bideen funtsezkotasunera** heltzeko mugapen lokalek hasieran haizutu zituzten balio pare jakin batzu deuseztuak izan beharko lirateke. Montanari eta Mackworth-ek denbora polinomikoko algoritmoak proposatu dituzte **arkuen funtsezkotasunera** eta **bideen funtsezkotasunera** iristeko. [MACK85]-an erakutsi da **arkuen funtsezkotasuna** $O(ek^3)$ -ean lortua izan daitekeela aldiz **bideen funtsezkotasuna** $O(n^3k^5)$ -an lortua izan daiteke, non n aldagaien kopurua bait da, k aldagai posibleen kopurua bait da, eta e puntaren edo adarren kopurua bait da.

6.2.2.- Mugapenak plangintzan.

Mugapenak plangintzan ere erabiliak izan dira, plangintza sistema ezagunenak, mugapenak erabiltzen dituztenen barruan, honako hauek dira [DRUM89]: Fikes-en REF-ARF, Stefik-en MOLGEN [STEF81a], Wilkins-en SIPE [WILK83] [WILK84], Sridharan eta Bresina-ren PLANX10-D, ISIS-II, eta Currie ta Tate-ren O-PLAN.

PLANGINTZA AAN ETA BERE ERREPRESENTAZIOA.

Plangintza sistema bat AAn plan baten sorkuntzaz arduratzen da, zein espezifikatutako arazo baten soluzio posible bat den. Sortutako planak aplikazio dominio bakoitzerako sistmari sarreraren emandako eragileen eskema (operator schemata) batez osatuta egongo da.

Arazo bat hasierako egoera bategatik eta helburu egoera baten deskripzioagatik karakterizatzen da. Hasierako egoeraren deskripzioak plangintza sistemari munduaren gaurko egoera iragartzen dio. Helburu egoeraren deskripzioak (helburua) behin plana exekutatzuz gero munduak nola geratu behar duen iragartzen du. Helburuak egoera baten alderdi nabarmenak baino ez ditu eskeintzen eta ez da beharrezkoa xehetasun guztiak eman ditzan. Helburua betetzen duen edozein egoera da onargarria.

Eragileen eskemak jazokuntzak ezaugarritzen ditu (ekintzak ere deituak) hauen aurrebaldintza eta ondorioen terminoetan deskribatuz. Planak eragile eskema hauetatik hasita eraikitzen dira, zeintzuk hainbat erataraz antolatuak edo errepresentatuak izan daitezkeen planen barruan. Eskema terminoak esan nahi du jazokuntza deskripzio bakoitzak konstanteek errerepresentatu dezaketen aldagaik dituela, jazokuntza indibidualak eta bereziak deskribatzen dituzten eragile instantziak eratortzeko.

AA-ko plangintzari buruzko literatura guztian Fikes eta Nilson-en STRIP-ak [FIKE72a] [FIKE72b] erabili dira extensiboki eragile gisa. STRip-ek egintza bat deskribatzen dute hiru zatiren bidez: aurrbaldintza formula bat (zeinek eragilea aplikatua izan baino lehenago kunplitu behar diren egintzak ematen dituen), gehipen zerrenda bat, eta deuseztatze zerrenda bat. Gehipen eta deuseztatze zerrendak batera erabiltzen dira ekintza baten izaera simulatzeko. Eragile baten aplikazioak esan nahi du gehipen zerrenda eta deuseztatze zerrendaren ganeko jokaera egoera berri bat ekoizteko. Egoera berria deuseztatze zerrendan emandako

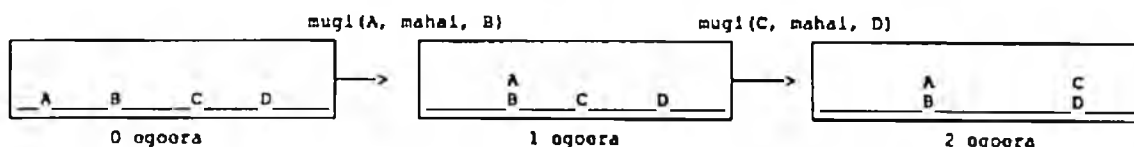
formula guztien lehenengo ezabapenagatik eta geroagoko formula guztien eransketagatik gehipen zerrendan sortzen da.

Plan bat eragileen kolekzio antolatua da eta arazo jakin baten soluzioa da arazoaren hasierako egoeran aplikagarria bada, eta plan honen exekuzioaren ondoren helburua konplitzen bada. Plan bat aplikagarria izateak esan nahi du lehenengo exekutatu behar den eragile bat dagoela eta honen aurrebaldintza guztiak hasierako egoeran konplitzen direla. Eragileen behin eta berrizko aplikazioa erdiko egoeren deskripzioak ekoizten ditu. Sarrera eta irteeraren konexio prozedurak izen batzu hartzen ditu: planaren sorkuntza, sintesi, eta eraikuntza. Esaten da planifikatzaile bat dominiotik independientea dela errepresentazio lengoaia eta sorkuntza algoritmoak aplikazio zentzudun ugaritasun handiarekin erabil badaitezke.

Errepresentaziorako formalismo bi erabiltzen dira: Egoeren Esspazioa eta Ekintzen Ordenazioa.

Egoeren Espazioen bidezko errepresentazioa, ikus 2. kapitulua, estentsiboki erabiltzen da AA-rentzako plangintzan emandako arazo baten soluzioaren ezagupena errazten duelako. Edozein egoeren espazioa erraz errepresentatua izan daiteke egitura grafiko norabidetua bezala, non nodoek egoerak deskribatzen dituzten eta arkuek egoera bat beste baten transformatzen

Eragilea: Mugl(objektua, iturburu, helburu)



6.1 irudia.- Plan bat egoera espazioaren arauera.

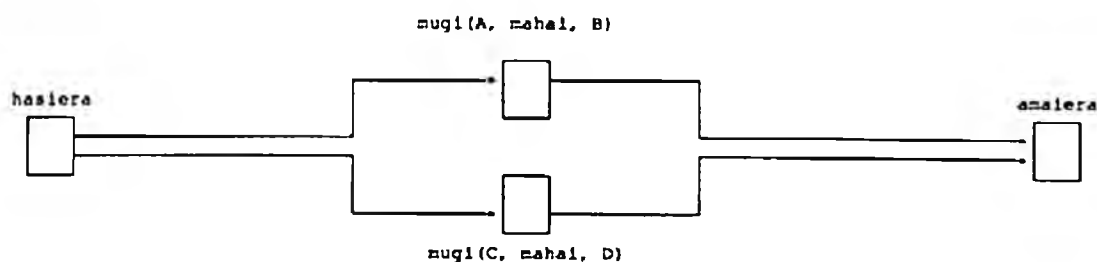
duten eragileen aplikazioa (ikus 6.1 irudia). Arazo baten soluzioa grafikoan zeharreko bide batetik erator daiteke, zein hasierako nodoetariko batean hasi eta helburua betetzen duten nodoetariko batean bukatzen den.

Beste planen ohizko errepresentazioa ekintzak exekutatuak izan daitezkeen ordenean mugapenak dituen ekintza multzo bat da. Adibidez, sarritan komenigarria da plan bat ekintza zerrenda sinple bat bezala errepresentatzea. Ekintzak zerrendan agertzen diren ordenean exekutatzeko dira, egoera espazioarekiko desberdintasunez ekintza eta ekintzaren gaineko ordenu erlazioarekin osatzen direlarik (ikus 6.2 irudia). Lanskyk (1985) errepresenta-



6.2 irudia.- Ekintzen ordenazioen bidezko plana (guztiz ordenatuta).

zio mota honi *portaerazko plangintza* (behavioral planning) deitu dio aplikatuak izango diren ekintzen gainean gehiago fokatzeko duten zentzuan eta gutxiago ekintzengan eragina duten baldintzen gainean. Sacerdoti-ren NOAH sistemak (1977) ekintzen ordenazioaren errepresentazioa erabiltzen du, sare prozedurala deitu zana [SACE75] eta honela definitu zena: *xeheetasun maila aldakorretako ekintzen sare bat, parzialki ordenatuak dauden denbora sekuentzien jerarkia baten barruan egitaratua*. Sare prozedural bat egitura grafiko norabidetu baten gisa errepresentatu daiteke ordenagailu batean, hemen nodoek ekintzak deskribatzen dituzte, eta arkuek orden erlazioak. Planeatutako ekintzei buruzko ordena osorik espezifikatu ez denean, gerta daiteke ekintza batzu beste guztiei dagokienez desordenatuak egotea, plan hauek *partzialki ordenatutako planen* izena hartzen dute eta eraikitzen dituzten sistemak *partzialki ordenatutako planifikatzaileak* (partially ordered planners) deituko dira (ikus 6.3 irudia).

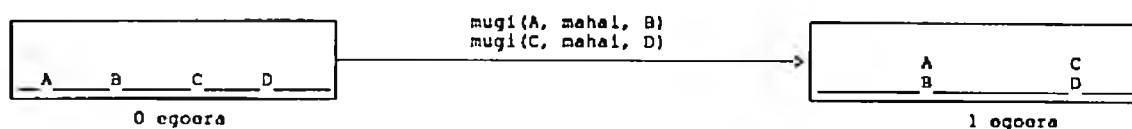


6.3 irudia.- Ekintzen ordenazioen bidezko plana (partzialki ordenatuta).

NOAH eta bere ondorengoak, NONLIN (Tate, 1977), DEVISER I (Vere, 1981), SIPE, O-PLAN, TWEAK (Chapman, 1985), FORBIN (Miller et al. 1985), eta DEVISER (Vere, 1985) bezalakoak; partzialki ordenatutako planak erabiltzen dituzte.

Ekintzen ordenazioen errepresentazio baten, egoerak ez daude esplizitoki errepresentatuak, honek zaila egin dezake partzialki ordenatutako planei buruzko arrazonamendua, hori gertatu arren sarritan erabilgarriak dira. Plan batek ekintza serie bat ordenatu dezake soil soilki egoera berriak zein erdiko egoeraren antza duen espezifikatzeko beharizanik gabe, erdiko espazio guztien osoko espezifikazioa ekidinez egoeretan oinarritutako hurbilketaekin gertatzen den bezala. Sare prozedurala gaur egun sartzen da *konpromesu gutxiagoko* (least commitment) planen eraikuntza tekniken erabilera haizutzeko, honek esan nahi du desordenatutako ekintzen haizutzea ahalik eta gehien (honi Drummond eta Tate-k *plan paraleloak* deitzen diote [DRUM89] ekintza ordenatuen planei edo *plan serialei* kontraerriak). Ekintza paraleloak haizutzen ditu hauek denbora une berean gertatzeaz gain nahasten ez direnak direla duelarik.

Konpromesu gutxiagoko plangintza egoeren espazioetan ere eman daitezke, honek ekintza baten exekuzio ordenaren gaineko luzamendua haizutzen du (ikus 6.4 irudia). Horretarako egoera bi lotzen dituen arkuak edozein ordenetan exekutatuko diren eragile



6.4 irudia.- Egoera espazioen bidezko plana (konpromisu gutxiagokoa).

multzoa errepresentatuko du, horregatik planifikatzaileak ekintzen exekuzio desordenatua haizu dezakeen plan baten errepresentazioa behar du. Hau hala, egoeren espazioetan nola ekintzen ordenazioetan egin daiteke.

Plangintzagileak bilaketa espazioa kurritzen dute arazoa konpontzen duen plana aurkitzeko. Aztertu beharrko bilaketa espazioa definitzeko modu batzu daude:

- 1) *Bilaketa egoeren espazioan* (state-space search), nodoek egoera posibleen deskripzioak daukate.
- 2) *Plan partzialaren bilaketa* (partial-plan search), nodoek plan partzial posibleak daukate.

Helburua lortzeko mekanismoa definitzeko ere modu desberdin batzu daude partzialki ordenatutako planak analizatzeko zailtasuna dela eta. Charniak eta McDermott-ek diote:

[Ataza baten exekuzioaren ondoren kunplitzen diren baldintzak]... ataza egiten denean munduaren egoeraren menpean daudela orokorrean, eta egoera hau aldiz atazen menpean dago, zein partzialki baino ezagutzen ez den (Charniak eta McDermott, 1985, p.504) [DRUM89].

Jarraian MOLGEN sistema aurkeztuko dugu objektuen hautaketa konpromesu gutxiagoko tekniken bidez egiteko trebetasuna duen lehen planifikatzailea izateagatik mugapenetan oinarritutako sistema baten eman daitezkeen fase desberdinak erabiliz: mugapenen formaulzioa, hedakuntza ea betetzea. Ideia hauek geroago SIPE eta O-PLAN-en garatuko dira. Planak serialak dira eta egoeren espazioen bidez errepresentatzen dira plan partzialen bilaketaz egiten delarik azterketa.

MOLGEN: MUGAPENEDIKO PLANGINTZA JERARKIKOA DISEINURAKO "CONSTRAINT POSTING"-EN BIDEZ.

Planifikatzaile jerarkiko batek egoera abstraktuen deskribaketak sortzen ditu eta bere plangintza ataza azpiarazoetan zatitzen du egoera abstraktuak birfintzeko. Egoera abstraktoek gogarren garrantzitsuen gainean fokatzeari haizutzen diote, zailtasuna gutxituz, azpiarazoen alde aurreko banaketa eta espezialista desberdinei hauen esleipena ahalbideratuz. Plan-

gintza arazo praktiko gehienetan azpiarazoez intereragiten dute. Obserbazio hau zentrala da arazoen ebazpenean, partikularki plangintzan eta diseinuan. Ellarekintzak aurreratuak izan daitezkenean, lanaren zatiketa gida dezakete. Berandu eriditen direnak, eskatutako aldaketak zailak eta inkorporatzeko neketsuak izan daitezke. Eekarrekintzen maneiaren zailtasuna arazoen tamanoak eta konplexutasunak osatuta dago. Diseinu egitamu handietan ezusteko eekarrekintzak sarritan egitamuaren zuzentzaileen denboraren alderdi substantziala kontsumitzen dute.

Eekarrekintza hauek jasateko trebetasunik gabe, planifikatzaile jerarkikoak kasu idealizatuarekin baino ezin izango dute eraginkorki tratatu, non azpiarazoez independenteak diren eta bereiz erabaki daitezkeen.

Plangintza jerarkikoa bakarrik erabili ahal izango da arazoez honen detailez gogarpenak atzeratu ahal badaiteke. Abstrakzio prozedura plangintza erabakiak zati ditzake azpiarazoa independenteetan (*loosely-coupled*). Abstrakzioa plangintza jerarkikoaren oinarri gisa plangintza programazio gehienetan erabilia izan da Newell, Shaw eta Simon-en General Problem Solver (GPS)-tik. Plangintza jerarkikoa eta ez jerarkikoaren arteko eztabaidan normalean lehenengoa aukeratzen da [SACE74].

MOLGEN, aurreko plangintza jerarkikoko programetatik desberdina da, plan bati xehetasunak gehitzeko trebetasunean, mugapenen adizioaren bidez, honek ekiditen du dominioaren hiztegirako kritikatasun (*criticality*) orokorraren mailak esleitzeko saiaketaren arazoa (ABSTRIPSen bezala [SACE74]), eta isladatzen du plangintzako konpromesuen mugapen berri gisa egon daitezkeeneko perspektiba. Honek JB-aren hurbilketak errazten ditu atzera jotzerko (*backtracking*), konpromesuen plangintzan eratzeko arazoak aztertzen dituenak.

Mugapenekin arrazontatzeko MOLGENen trebetasunaren generaltasuna mugapenen betetzearen errekerimendu sinpleetatik dator. Mugapenen betetzeak mugapenak ebaluatzeko trebetasuna baino ez

du eskatzen. MOLGENek ahalezko soluzioak sortu ditzakeen bitartean, erraz egiaztatu dezake ea mugapen arbitrarioak bete diren. Egiaztatzeko mugapenen erabilera honek ez du kontutan hartzen sorkuntza gidatzeko erabiltzeko ideia ahaltsuena, aldez aurretik aplikatuz soluzio partzialak aurkitzeko.

Mugapenak aldez aurretik aplikatzeko MOLGENen trebetasuna mugapenen hedakuntzaren inplementazioaren menpean dago, honek ahultasun serio batzu ditu. MOLGENen mugapen hedakuntza eragileak hertsiki mugapenen iragazketasun sintaktikoetan oinarrituta daude. Zoritxarrez, MOLGENek ez du mugapenetan predikatu logikoen baliokidetasuna ezagutzeko gaitasuna. MOLGENek sortuak zituen mugapenak hedatu ditzakeen arren, ez du mugapen horien logikoki baliokideak diren aldakuntzak edo haren hiztegi mugatutik kanpoko mugapen arbitrarioak hedatzeko gaitasuna. Honek mugapenen erabilerarako MOLGENen trebetasunaren mugatze praktiko batera darama; bitartean, mugapen berri bat bete dezakeen plana sor dezake noiz edo behin, denbora gehiegi eraman diezaioke bete beteko plan bat proposatzeak baldin eta bakarrik berandu aplikatu ahal badu mugapena plangintza prozesuan.

Bigarren muga bat da MOLGENek ez duela mugapenik erabiltzen prozesuak deskribatzeko, ez du baliabide ahaltsurik prozesuak deskribatzeko. Ez da mugapenik garatu prozesu partzialen deskribapenerako.

Hirugarren muga bat metamugapenen ez erabilera da. Lehenengo ordeneko mugapenak planetan objektuei buruzkoak dira; metamugapenak planari edo plangintza prozesuari buruzkoak izan litezkeen. MOLGENen interpretea mailatan antolatuta dago, eta mugapenen manipulazioari buruzko jakintza bakar bakarrik agertzen da mailarik baxuenean, diseinu prozesuari buruzko mugapenen arrazonamendua jasatzeko.

"Constraint Posting" arazoen ebazpen tankera bat da, jakintza trinkoarena, mugapenak noiz formulatu eta nola hedatzeari buruzko jakintza substantziala eskatzen du. Mugapenen

formulazioari buruzko jakintzaren galerak ondorio seriogoak ditu hedakuntzari buruzko informazioaren galerak baino. Lehenengo kasuan eredu genetikoan huts egingo luke laborategian¹ eginezinak izango diren esperimentuak proposatiko lituzke; bigarren kasuan plangintzaren erabakien arteko interferentzia ez da eridengo atzera jotzeko lan extra asko egina egon arte, baina plan zuzen batera iritsi ahal izango da. Lehenengo jakintza mota JB-ari erraz inkorporatzeko zailtasunak iragartzen du jakintzaren lorpenari buruz gehiago ikertzearen beharizana.

MOLGEN-ek denboraren errepresentazio desegokia erabiltzen du objektu berberarentzat izen desberdinak erabiltzen bait ditu dagokien denbora unearen arabera eta ez du mugapenen betetzearen aldiei buruzko arrazonamendu esplizitoa errazten, eta ez du ahalezko mundu desberdinen erregistrorik mantentzen ere. Nahiz eta nahikoa den mugapenen betetzerako.

"Constraint Posting" mugapenen interpretazio desberdinak uztiatzen dituen plangintza jerarkikoari hubilketa bat da eraginkorki planifikatzeko. Planifikazio jerarkian zehar mugapenak formulatzen ditu konpromesu berriak eransteke eta hedatu egiten ditu alde aurretik erabiliak izan ahal izateko diseinu prozesuan interferitzen diren soluzioak deuseztatzeko. Jakintza intentsiboko arazoen ebazpenari hurbilketa bat ere bada. Hurbilketa honen rutinaren aplikaziorako eragozpen bat informazio hori ordenagailu batera transferitzeko baliabide eraginkorren falta da.

6.2.3.- Mugapenen Betetzea.

Mugapenei buruzko eragiketa mota honetan zentratuko gara, MAMUR-en erabilia izango bait da.

¹ MOLGEN-en helburua JOS bat oraikitzea da ingeniari genetikoel laguntzako laboratoriorako esperimentoen planifikazioan.

AA-ren ataza asko MBA gisa formulatuak izan daitezke, honek mugapen multzo bati lotutako aldagaiei balioen esleipena berarekin darama.

WALTZ-EN ALGORITMOA.

Waltz-en algoritmoa mugapenen betetzerako algoritmoa bezala har daiteke kontutan. Lotune bat poliedro baten irudian mugapentzat har daiteke. Lotune baten arista bakoitzari aldagai bat esleitzen zaio, aldagaiaren balioa aristaren etiketaren emaitza da. Ez bait dira errealitatean gertatzen aristen etiketa guztien konbinaziok, balioen esleipen guztiak aldagaiei ez zuten posibleak izan beharko. Hortaz, haizututako balio guztien definizioak mugapenen erlazioetan espezifikatuta daude. D. L. Waltz-ek egindako lehen esentziasko oharra, ahalezko balioen esleipenen kopurua txikiagoa dela magnitude orden batzuetan konbinatorioki ahalezkoak baino.

Lotune bakoitzari mugapenak esleitura izan eta gero, arista bera errepresentatzen duten aldagaiak identifikatzen dira (aldagaiak konpartituz arista berdinentzat). Orduan mugapen sare bat lortuko dugu. Waltz-ek aurkeztutako algoritmoak lokalki funtsezko balioen esleipenak erabakitzen ditu mugapen sarearen aldagaientzat. Hedakuntza lokalaren ideian oinarrituta dago. Algoritmoak soilik zuzenki aldameneko nodoak gogoratzen ditu nodo batenak diren mugapen aldagaien balioen rango mugatu nahi duenean. Balioen mugapenei buruzko informazioa lokalki pasatzen da. Xehetasunean algoritmoak jarraiko era honetan egiten du lan: Lotune baten ahalezko etiketak auzoko nodoen ahalezko etiketekin konparatuko dira, eta auzoko nodo baten etiketa batekin ere konpatibleak ez diren etiketak deuseztatuak dira. Prozedura berbera exekutatzen da eragindako lotune baten auzoko guztientzat, eta horrela elkarren segidan. Algoritmoa aldaketa gehiago gertatzen ez denean bukatzen da.

Honela, atzera jotzeko algoritmo standarraren leherketa konbinatorioa ekiditen da eta algoritmoak funtsezko etiketatzea CPU denbora segundu gutxitan aurkitzen du, atzera jotzeko algoritmoak orduak inbertitzen zituen bitartean. Hau izan zen, hain zuzen ere, historikoki lehenengotariko adibide bat non kontrol regimen argi batek leherketa konbinatorioak gaingitu ahal izan zituen. Partikularki, erakustrern du leherketa konbinatorioa ezin dela beti erabilia izan argumento gisa eskala handiko AA-ren sistemen eginkortasunaren kontra.

CONSAT: MUGAPENEN BETETZERAKO SISTEMA BAT.

Mugapenetan oinarritutako lengoaia da interprete batekin, lehenagoko hurbilketetan ez bezala balio posibleen multzoak hedatzen dituen (balio sinpleez gain) mugapen sarean zehar. Hedakuntza prozesuan zehar balio multzoak iragazten dira funtsezko azpimultzoak lortzeko, edo balio berriak kalkulatzeko dira emandako aldagai multzoko ez zehaztutako aldagaienezako.

Mugapenen erlazioak hainbat modutara deskriba daitezke. Erlazio finitoak extensionalki errepresentatuak izan daitezke tuplen bitartez, eta hemen tupla bakoitzak balio onargarriak errepresentatuko ditu. Erlazio infinituak intensionalki² klase bat edo balioen konbinazioa errepresentatzen duten patroiek errepresenta ditzakete. Mugapen intensionalen idazkera tuplen bidezko errepresentazioa bezalakoa da: Patroi baten i-garren adierazpenak mugapenen i-garren aldagaien balioa deskribatzen du. Mugapenen iragazketan (pattern matching) zehar errakuntzak ekiditeko, mugapen baten ebaluazioa ezabatua izan daiteke aldagaien oraingo balioaren arabera. Erlaziorik konplexuenak mugapen sareek eta mugapen sareen jerarkiek errepresentatu ditzakete. Kontrol estrategia batzu aurrikusi dira, eta hauek soluzio lokalki funtsezkoak eta batera orokorki funtsezkoak kalkulatzeko dituzte, azken hauek estrategia biren bidez:

² Era funtzionalan mugapen aldagaiak aipatuz. Haren ebaluaziotik lortutako balioak mugapena betotzen den zehaztoko omandakoekin konparatuko dira.

- 1) Estrategia lokal eta atzera jotzearen konbinazioa.
- 2) Indize errekurtsiboen bidez balioen arteko erlazioak mantentzeko.

Leherketa konbinatorioa murriztu ahal da mugapen baten deskribapenean erabilitako aldagai kopuruaren arabera.

CONSAT-ek dominioetik independientea den mugapen lengoaia izatearen aspektuak konbinatu nahi ditu arazoen ebazpen estrategia eraginkor batekin [GÜSG89a] mugapenen betetze sistema batean, G. L. Steele [STEE80] eta D. L. Waltz-en [WALT72] ideiak konbinatuz.

Mugapen bat honako hauek aktibatzen dutenean:

- 1) Hedakuntza kontrol unitateak.
- 2) Beste azpisistema batek.
- 3) Erabiltzaileak.

Haren balio batzu jadanik asoziatuak egon daitezke balio posibleen multzoekin. Sistemak lehenengo zerrenda hauen konbinazio guztiak kalkulatzeko dituzten mugatu gabeko aldagaiak UNCONSTRAINED balioarekin jarritik. Balioen konbinazio bakoitza mugapenaren elementuarekin iragaztua da honako modu honetara: Gogartzen den mugapenaren elementua tupla bat bada, honen balioak konbinazioan dagozkienekin konparatuko dira. Elementu hau patroia bat bada, honen baldintza (ematen bada) ebaluatu egingo da lehenengo. Baldintzak ebaluazio T balioa hartzen badu edo ez badago, orduan patroien gainontzekoa honela ebaluatuko da: patroia atomo eta adierazpen guztiak ebaluatu egingo dira eta kalkulatuak balioak emandako balio konbinazioari dagozkienekin konparatuko dira.

Kasu bietan, iragazketa prozesuak arrakasta izango du emandako konbinazio bakoitzeko haxe gertatzen bada:

- UNCONSTRAINED balioa hartzen badu, edo

- mugapenaren elementuari dagokion balioa berdina hartzen badu.

Iragazketa prozesuak arrakasta badu, kalkulaturako balioak emaitzako aldagai estaldurei eranstsiko dira. Itzuliko dira konbinazio guztiak mugapenei dagozkien elementu guztiekin iragaztuak izan eta gero.

Formalki, C mugapen baten ebaluazioaren emaitza honela denotatua egon daiteke: Izan dadin $v_1, \dots, v_n$ sarrerako aldagaien estaldura C mugapenerako eta R C -ren mugapen elementuaren multzoa (tuplak edo patroiak). Orduan, C -ren ebaluazioa izango da

$$pr_{1,\dots,n}((v_1 \times \dots \times v_n) \cap R_{inst})$$

non

- * $pr_{1,\dots,n}$ era txikitu bat gisa erabiltzen da $(pr_1, \dots, pr_n)$ projekzio funtzioen tuplarako, adibidez $pr_{1,2}(\{(a,b), (c,d)\}) = (\{a,c\}, \{b,d\})$, eta
- * R_{inst} C -ren erlazio instantziatua da R -tik lortua dena patroien ebaluazioen bidez $v_1 \times \dots \times v_n$ elementuei elkarren segidan aurkitutako aldagaiekin.

[GÜSG89a]-en lengoai honentzako konpiladore bat nola egin behar deneko zirriborroa egiten da, zeinek mugapenen deskripzioak hobetzen dituen definizio garaian leherketa konbinatorioa gutxitu ahal delarik mugapenen deskripzioan erabilitako aldagaien kopuruaren arabera.

Mugapen sareak CONSAT-en mugapen primitiboen multzoa edo maila baxuagoko mugapen sareak dituzte, zeinek aldagai konpartituekin konektatuak daude. Mugapen primitiboen sarrera/irteera portaera berbera dute, adibidez, mugapen sare bat ebaluatzen da aldagai estaldura batzuekin sarrera gisa, eta emaitza iragazitako aldagaien estalduren multzoa da.

Partikularki, mugapen sare batek badauzka: mugapenak, aldagai orokorrak, aldagai estaldurak, projekzioak mugapen

aldagai lokalen artean eta mugapen aldagai orokorren artean, eta interface bat. Interfacea hasierako balioekin asoziatuak izan daitezkeen aldagai orokorren azpimultzoa da.

Mugapen sareen definizioaren arabera, mugapen primitiboetako eraikuntza berberak erabil daitezke. Mugapen sareek mugapen primitiboek bezala lan egiten dutenez (hau sarrera/irteera portaeratik ikusita), mugapen jerarkiak eta mugapen sareak eraikiak izan daitezke mugapen sareei erreferentziaz soil soilik mugapen primitiboei baino lehen beste mugapen sare baten barruan. Metodo honen ondorioa da mugapen errekurtsiboen sareak defini daitezkeela, adibidez sareari dagokionez sarearen beraren barruan.

CONSAT-en kontrol estrategia standarra hedakuntza lokalean dago oinarrituta: erabiltzaileak mugapen sare bat aktibatzen duenean hasierako balioekin interfaceen aldagai batzuentzako, aldagai hauen gainean inposatuak dauden mugapenak ebaluatzen dira. Ebaluazioaren emaitza auzoko mugapenetara hedatzen da, eta horrela elkarren segidan. Aktibazio prozesua aldagaien aldaketa gehiagorik ez dagoenean amaitzen da. Aldagaiak mugapenak iragaziak direnez gero, hedakuntza prozesua beti geratzen da errekurtsiboak ez diren mugapen sareen kasuan. Interface aldagaien balioak hedakuntza lokalaren prozesuaren emaitza gisa itzutzen dira.

Hedakuntza lokalak funtsezkotasan lokala kalkulatzeko du, eta hau gutxiegia da kasu batzuetan. Orduan CONSAT-ek beste kontrol estrategia batzu eskeintzen ditu funtsezkotasan orokorra lortzeko, hauetariko bat hedakuntza lokala eta atzera jotzearen konbinazioa da (HLAJ). Atzera jotzearen emaitza bakarra ez bada HLAJ-k heuristika sinple batzuekin gidatutako hautaketa puntua kokatzen du. Orduan, hedakuntza lokala berriz hasten da, beste hautaketa puntu baten kokatzeak jarraitua, eta horrela elkarren segidan. Hedakuntza lokalean zehar funtsezkogabetasunen bat antzeman bada edo soluzio bakarra aurkitu, baina beste soluzio batzu eskatu badira (erabiltzaileak muga ditzake kalkulatu

beharreko soluzioen kopurua), atzerantz jotzen da hautaketa puntu batera eta beste hautaketa puntu bat kokatzen da. Algoritmoa gelditzen da behar beste soluzio kalkulatu denean edo hautaketa puntu gehiago ezin daitezkenean kokatu.

HLAJ-z gain, CONSAT-ek beste kontrol estrategia bat eskeintzen du, hedakuntza lokalean oinarrituta eta indizeak erabiliz hedakuntza prozesuan zehar balioen arteko erlazioak mantentzeko.

Partikularrean, CONSAT-ek dominio arbitrarioak jasan ditzake; adibidez, mugapen aldagaiak edozein balio klaserekin asoziatuak izan daitezke. Balio posibleen multzoak, balio sinpleez gain, mugapen sareen bidez heda dezakete. Orduan CONSAT aplika daiteke:

- * emandako aldagai multzoaren mugagabeko aldagaienezako balioak kalkulatzeko,
- * balioen funtsezkotasuna egiaztatzeke, edo
- * balio posibleen multzoak iragazitako, adibidez multzo hauetako balio funtsezkogabeak deuseztatzeke.

CONSAT hainbat aplikaziotan erabili izan da: adibidez; diagnosi prozesurako sistema batean, hemen mugapenak aplikatzen dira erlazio funtzionalak mantentzeko unitate fisikoetan, eta dominio musikalean, non koralean armonizazioa mugapenen hedakuntzaz jasaten bait da.

Gainera, BABYLON [GÜSG87] jakintza hibridoaren errepresentazio sistemara CONSATen eransketa mugapenak erabiltzeko, errege-lak, frameak, eta PROLOG-ekin batera; erabilgariagoa egin du.

6.2.4.- Arazo Errazen ideia: Mugapenen Betetzeen kasua.

Minsky-ren ustez [MINS63], arazoen ebazpenean giza trebetasunaren osagarri garrantzitsua arazo zailen soluzioa gaidatzeko arazo errazen ebazpenari buruzko jakintza erabiltzeko trebetasunean datza. Bakarrik lan apur batzu AA-n saiatu dira gaitasun hau erabiltzen [SACE74] [CARB83]. Gaschnig [GASC79], Guida eta Somalvico, Pearl [PEAR83], eta Dechter eta Pearl-k [DECH85] [DECH88] iradokitzen dute arazo errazei buruzko jakintza heuristikien aurkitze mekanikorako erabilia izan litekeela. Honekin bat etorritz litekeena izan beharko litzateke arazo zail baten errepresentazioaren manipulazioa arazo erraz bat bihurtu arte, arazo erraza ebatzi, eta soluzioa bilaketa prozesua gidatzeko erabili arazo originalean.

Eskema honen implementazioak hiru pausu behar ditu:

- 1) Sinplifikazioa, horretarako *a priori* erizpide bat eduki beharko da, zeinen bidez ezagutuko den noiz duen arazo batek soluzio erraza.
- 2) Soluzioa.
- 3) Gomendioaren sorketa.

Aplikatuiko dizkiegu ideia hauek mugapenen betetzearen atazei MAMUR-en erabiltzen direnak bait dira, nahiz eta era berezian. Mugapenen Betetzearen arazoak (MBA) hiekin daramatzate balioen esleipena mugapen multzo bati lotutako aldagaiei. Normalean atzera jotzeak duen bilaketa bertsioaren batekin ebazten dira, zeinek bilaketa denbora esponentziala errekeritzen bait du.

Orokorrean arazo bat erraza dela esaten da haren errepresentazioak soluzioa denbora polinomialean haizutzen duenean. Hortaz, atzera jotzeko algoritmoekin tratatzen badugu batez ere, MBA bat erraza dela kontsideratuko dugu atzera jotzeko prozedurarik gabe ebatzia izan ahal bada. Normalean, atzera jotzeko algoritmo batek

orden zehaztu baten instantziatu ditu aldagaiak, eta hurrengo aldagai bakoitzerako alde aurreko esleipenerekin funtsezkoa den balioa aukeratzen du. Ez badu aurkitzen, alde aurreko aldagairaino doa atzeraka, aldagai horrentzat esleipen berri bat lortzen saiatuz, eta honetatik aurrera jarraituz. Algorimoa gelditzen da aldagai guztiei balioak esleitu zaizkenean edo balio esleigairik geratzen ez denean lehenengo aldagaientzat. Atzera jotze gabeko bilaketa bat atzera itzultzeko algoritmoa atzera jo izan gabe osatzea eragiten du, honek aldagaien kopuruarekin denbora linealeko soluzioa eragiten du.

Gure esposazioren zatirik handiena mugapen sareen kontzeptuan edo mugapen grafoetan [MACK77] oinarrituta daude honetan nodoek aldagaik errepresentatzen dituzte eta ez norabidetutako arkuek haien arteko mugapen baten izaera esplizittoa errepresentatzen dute. Freuder-ek [FREU82] mugapen sare batek atzera jotzerik gabeko MBA bat emateko beharrezko baldintzak identifikatu ditu; erakutsiz, adibidez, mugapen sareak zuhaitz egiturarekin eraiki daitezkeela baldintza horiek betetzeko, prozesu kantitate apur batekin. [DECH85] eta [DECH88]-etan aurreratu egiten da atzera jotzerik gabeko soluzioak emon ditzaketen mugapen sareen klaseaz eta ebazteko algoritmo erginkorren ikerketan.

Behin klase hauek identifikatu eta gero arazo baten sinpletze eskema baterako eredu gisa hautatuak izan daitezke; mugapenak hautakorki deuseztuak izan daitezke jatorrizko espezifikaziotik jatorrizko arazoa atzera jotze gabeko bihurtzeko. Arazo sinpletuak jatorrizko arazoaren menpean dagoen hautaketari buruzko gomendioak eman ditzake. Dechter-ek bere tesian "funtsezko soluzioen kopurua arazo sinplean" erabiltzea proposatzen du adierazle gisa lehentasunak ezartzeko jatorrizko arazoaren atzera jotzeko bilaketarako balioen esleipenetan; gainera, erakusten du adierazle hau kalkulatu izan daitekeela arazo erraz baten soluzio sinplea aurkitzeko arabakitako denborarekin konparagarria den denbora batean.

Freuder-ek honako teorema hau proposatu zuen [FREU82]:

- a) Mugapen sareak 1-eko zabaltasuna badu (adz: zuhaitza bada) eta funtsezko arkuduna bada, orduan atzera jotzerik gabeko soluzioak onartzen ditu.
- b) Mugapen sarearen zabalera 2 bada eta bide funtsezko-duuna ere bada, orduan atzera jotzerik gabeko soluzioak onartzen ditu.

Aurreko teoremak zuhaitz motako MBA-ak honela ebatziak izan daitezkeela iradokitzen du:

- 1) "arku funtsezkotetasuna" lortuz eta
- 2) sareari 1 zabalera edukitzea eragingo dion ordenean instantziatu aldagaiak.

Instantzia honek $O(ek)$ pausu eramaten digunez arazo osoa $O(ek^3)$ -an ebatzia izan daiteke eta horregatik zuhaitz motako MBA-ak errazak dira. Berezkotetasun hau baieztatzeko test-a erraz egiaztatzen da. Sare ezagun bat zuhaitza den ala ez baieztatzeko zuhaitza erregularraren algoritmo baten bidez $O(n^2)$ -en egin daiteke.

Garrantzitsua da MBA ezagun batek errepresentazio baliokide desberdinak eduki ditzakela argitzea, soluzio multzo bera onartzen duelarik. Errepresentazio bakoitzak ere mugapen sare desberdina izan dezake, hauetariko bat zuhaitza izan daiteke. Hala ere, ataza zaila izan daiteke MBA batek zuhaitzezko errepresentazio baliokidea duen egiaztatzea, eta errepresentazio hori aurkitzea [DECH85].

Teoremaren bigarren zatiak tentazio hau sortzen digu: 2 zabalera mugapen sare batek atzera jotzerik gabeko soluzioa onartu beharko lukeela bideko funtsezkotetasun algoritmo batetik pasatu ondoren. Kasu honetan, hala ere, bideko funtsezkotetasun algoritmoak sareari arkuak gehi diezaioke eta zabalera 2en ganetik gehiagotu. Hau sarritan gertatzen da algoritmoak aldagai pare baten balio pareak deuseztatzen dituenean, aldagai pareak hasieran mugapen unibertsalaz zeudelarik erlazionatuta (konexio

arkurik izan gabe haien artean) eta sarritan gertatzen da bideko funtsezkotasuneko algoritmo baten bidez mugapen sare osoa errepresentatzen duela. Gerta daiteke, beraz, ezin atera izatea inolako probetxurik MBA batek 2 zabalerako mugapen sarea edukitzeatik ez bada oraindik funtsezko bidekoa. Dechter eta Pearl-en [DECH85] arabera ez da segurua 2 zabalera nahikoa izan dedin NP-osotasuna ezintzeko.

[DECH85]-n arku eta bide funtsezkotasunaren definizio ahulagoak ematen dira, zeinek nahiko ere badira atzera jotze gabeko soluzioak bermatzeko, baina abantaila birekin Montanari eta Mackworth-ek definitutakoen gainean:

- 1) Eraginkorragoki lortuak izan daitezke.
- 2) Arku gutxiago gehitzen diote mugapen sareari, horrela sarearen zabalera gordez arazo handiagoen klaseetan.

Funtsean, sinpletzeko alderdirako: Dechter-ek [DECH85] arazo errazak ezagutzeko erizpidea ideiatu du azpian datzaten mugapen sareetan oinarriturik, Montanari, Mackworth eta Freuder-en lanari jarraituz. Erizpide hauetan aurkitzen ditugun ezaugarriak arazo zailen sinpletzerako helburu gisa erabil daitezke haren mugapenen batzu deuseztuz. Norabidegarritasunaren sarrera arku eta bide funtsezkotasun nozioetan arazo ezaugarri errazen klaseak luzatzen uzten digute zuhaitzak baino harago, 2 zabalerako arazoak sartzeko.

Soluzio alderdirako: norabidegarritasuna erabiliz algoritmo hobetuak idea daitezke sinpletutako arazoen ebazpenerako eta hauen hoberetasuna erakusteko. Zehatz mehatz, Dechter-ek erakusten du zuhaitzan egituratutako arazoak $O(nk^2)$ pausutan ebatziak izan daitezke eta 2 erregularreko zabalerako arazoak $O(n^3k^3)$ pausuetan.

Gomendioen sorketaren alderako Dechter-ek erakutsi du [DECH85] gomendio erauzketarako metodo sinplea arazo errazetatik, atzera jotze algoritmoari laguntzeko balioen esleipenaren menpeko

opzioen artean erabakitzeke dagoena. Metodoak mugapenen betetzearen atzetik geratzen den zatia hurbiltzea darama berarekin zuhaitzean egituratutako arazo baten bidez, eta menpeko esleipen bakoitzarekin funtsezko soluzioaren kopurua. Kalkulu hauek eraginkorki lor daitezke eta adierazle bezala erabil daitezke opzio bakoitzak eskeintzen duen etorkizuna estimatzeko.

Dechter-ek, bere doktorego tesian, atzera jotzeko "RBT" algoritmo regularrekin eta atzera jotzeko "ART" algoritmo aholkotuarekin esperimentatzen du, aholku ahul bat eraginkortasun maila altura heltzeko nahikoa zeneko ondorioara iritsiz, aholku zehatz batek sorkuntza denbora handiegia darama eta.

6.3.- KONTROL EGITURA MAMUR-EN.

MAMUR-en ematen den soluzioak tresna baten eraikuntzarako, erraztasun eta eraginkortasun konputazionala lgo kapituluan finkatutako ezaugarriekin nagusitzen du.

Aurrerantzean MAMUR-en ezaugarriak arakutsiko ditugu beste sistema batzu aipatuz, adibidez: MILORD [SIER89], CONSAT [GÜSG89b], MOLGEN [STEF81a] [STEF81b] SIPE [WILK83] [WILK84], eta Currie-ren O-PLAN .

MAMUR-ren kontrolari dagozkion ezaugarriak nagusienak honako hauek dira:

- Arrazonamendu mekanismo bakar bat erabiltzen da edozein jakintza motatako.
- Kontrola era deklaragarrian errepresentatuko da erregelen bidez, egitura berdin berdinekin jakintzaren maila edo mota edozein delarik [PIKA92c].

- Plan bat ezarri eta kasuaren menpeko estrategiak sortzen dira soluziora heltzeko.

TEIRESIAS eta ondorengo beste sistema batzuen eraginari jarraituz. Estrategia kontzeptua ebatzi behar den kasuaren alderdi zehatzerako jakintza egokia duen moduloaren hautaketan eta erabil behar den arrazonamendu motan datza.

Kontrol alderdi desberdinak ikertzeko honako kontzeptu hauek nabaritutako ditugu:

- Arkitektura modular multimaila.
- Arrazonamendu metodoa.
- MAMUR-en kontrolaren arkitektura.
- MAMUR-en kontrolaren zikloa.
- Plangintza Jerarkikoa.
- Mugapenen Betetzea bilaketa espazioa murrizteko.
- Arazo Errazen ideia eraginkortasunaren hobetzean.
- Metajakintza maila desberdinak osatutako kontrol egitura.
- Eraginkortasunaren hobetzea JB-aren konpilazioaren bidez.

6.3.1.- Arkitektura modular multimaila.

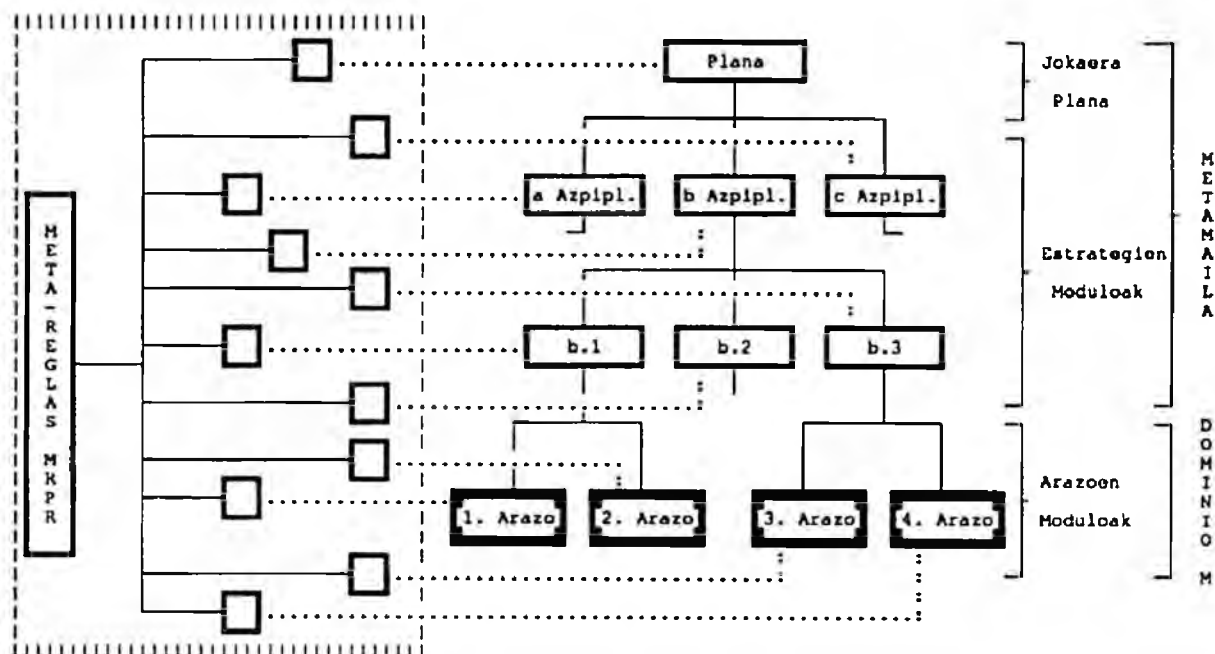
MAMUR-en multimaila arkitekturak dominioaren espazio zehatzaren jakintza errepresentatzen diduzten erregelak bildu nahi ditu, KEE-ren eta beste batzuen "erregelen-multzoak" ("rule-sets") ideia gandituz, edozein sistema modularrean gertatzen den bezala. MAMUR-en moduloak "espezialistak" bezala egiten dute lan atazaren ebazpenean, [STIC87]-ren "knowledge-groups"-ak bezala, horregatik erabilgarritasun zehatza dute, nahiz eta, moduloen kopurua handiagotu daitekeen edukin orokorragoa emanik kodearen errepikapena ekiditeko eta eredu baten bererabilerara ere iritsi, dominioaren menpekoak izan arren (mailarik altueneko moduloa izan

ezik, zeinek jokaera plana daukan, dominiotik independentea izan daitekeena). Aldiz, MILORD-en moduloak abstrakzio funtzionalak gisa ulertzen dira, modulo hauen bererabilitasuna bilatuz ataza desberdinetan eta dominio desberdinetan ere, horregatik jakintza orokorra izan behar dute. Modulo generiko bat egitura berdina daukaten moduloen abstrakzioa da. Honela moduloaren kodea behin baino ez da idatziko eta funtsezkotasun ezak ekidingo dira. Kasu honetan ahalegin handiagoa izango da beharrezkoa jakintza-ingenieriaren aldetik. Gaur eguneko jarraibidea jakintza sakonaren errepresentaziorantza joateko era da, hau da, Bigarren Belaunaldiko SA-etara.

Modularizaziorako arrazoi behinena JB-aren diseinu, egiaztatze, eta mantenimenduaren zailtasuna murrizteko beharria izan zen, hau nabaria egin zen CETODIA sistemaren eraikuntzan zehar. MILORD-en gainera, modulo bakoitzerako izen desberdinen espazio bat dago, honela albo-ondorioek eragindako elkarrekintzak desagertu daitezten. MILORD-en hau moduloen lengoaia erabileraz lortzen da, zeinekin JB-aren maila estrukturala definitu den haxe espezikatuz: erabiliko diren egintzen *multzoak*, inportatzen direnak, esportatzen direnak, eta maila honetako *espezifikazio* bakoitzean beharrezkoak diren moduloak (eta azpimoduloak). Azken hau ez da esistitzen MAMUR-en, hemen edozein aldagai edozein moduloaren esanetara dago guztiz gaurkotutako informazio batekin albo-ondorioak erraz kontrolagarriak direlarik alde zurretik definitutako objektuen atributuak erabiltzeagatik. Ez dago informazioaren ezkutaketarik (information hiding) eta moduloen arteko elkarrekintza amankomunak diren egintzen multzoaren bidez egiten da.

MAMUR-en arkitekturak JB-aren eraikuntza modularra haizutzen du modulo hauek era jerarkikoan banatu ahal dituelarik maila desberdinetan, azpiko mailaren moduloak berehalako bere ganeko mailaren moduloko erregela baten aginduz prozesatuko direnak, kateaketa edozein mota aukeratu ahal izanik.

Metaerregelen moduloen metamaila



6.5 Irudia.- MAMUR-ek interpretatutako JB baten ager litzakeen moduloak.

Azpireneko mailako moduloak erregela kopururik handiena izango dute helburu finko bat edo batzuekin dominioaren espazio mugatu baten jakintzaren errepresentaziorako erabiliko dira. Modulo hauek prozesatuko dira, normalean, atzeranzko kateaketaz horrela helburu zehatz baten lorpenerako beharrezkoak diren erregelak bakar bakarrik korritzeko.

Goiko mailek askoz ere erregela kopuru txikiagoa izango dute eta aurreranzko kateaketaz prozesatuko dira aukera guztiak ahitzeko Mugapenetan Oinarritutako Arrazonamenduari jarraituz. Modulo hauek estrategiak dituzte, erregelen bidez errepresenta-tuak, kasu bakoitzean interesatzen diren moduloak bakar bakarrik korritzeko.

Mailarik goreneko moduloak arazoaren edozein kasu partikula-rrer prozesamendurako jarraitu behar den plana dauka, arazoa errepresentatutako dominioaren jakintzaz ebatziko dena.

Modulo bakoitzak, dagokion mailaz aparte, metaerregela multzoa eduki dezake, honetarako:

- 1) Haren erregele batzuen inausketa edo deusezteia oraingo arazoaren ebazpenerako baliogabeak direlako usteagatik.
- 2) Haren erregelen berordenazioa bilatutako helburura heltzeko aukera gehiago dutenak lehenengo egiaztatzenko.

Modulo baten metaerregelek beste metaerregelen moduloekin batera osatuko duten metamailako beste modulo baten barruan egongo dira. Metaerregela hauek JB-an definitu ahal izango dira era deklaragarrian. Modulo batean jakintza erabili nahi den guztietan IM-ak modulo horrentzako metaerregelen moduloaren izatea egiaztatuko du, aurreranzko kateaketaz exekutatu baiezkoa izatekotan. Metaerregelen modulo baten aktibazioak eragina duen erregelen "utilidad" fazeta aldaraziko du, honekin haren lehentasuna gatazka multzoan aldaraziko da .

Egitura modular hau RETE algoritmoaren egiten den inplementazioan oinarrituta dago, modulo bateko erregelengana heldu ahal izanik bakarrik RETE zuhaitzan modulo bakoitzeko erro nodo partikularren ezistentziagatik. Aurretik esandakoak balio du bai aurrenranzko kateaketarentzat bai atzeranzkoarentzat. Esperientzia praktikoak erakutsi digu errepresentazio era hau adituaren pentsatzeko erara hurbiltzen dela, eta honek JB-ren eraikuntza, mantenimendua eta berriztea errazten du. OOP-ak ere laguntzen dio lan honi.

6.3.2.- Arrazonamendu metodoa.

Erabilitako arrazonamendu metodoak kontrol mota bat zehazten du. Clancey-ren arabera [CLAN85] SA mota bi daude:

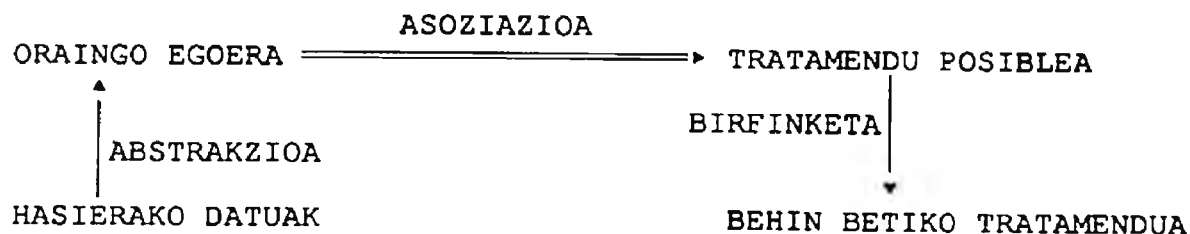
- * Analitikoak, hemen datu ezagunen multzo batetik soluzio bat edo batzu ateratzen dira, batzutan zihurgabetasunez, aurretik ezarritako soluzioen multzotik; honako metodo hauek erabiliz:
 - Sailkapen sinplea, kasu bateko datuak edo haren abstrakzioak parekidetzen dira zuzenean soluzioaren ezaugarriekin.
 - Datuen abstrakzioa, datuak soluzioen ezaugarriak dira.
 - Sailkapen heuristikoa, kasuaren datuen arteko eta soluzioen ezaugarrien arteko parekidetasun heuristikoeak egitea haizutzen dute, asoziatzen dira zuzenean jerarkia desberdinetako elementuak. Adibidez, klinika konstanteak gaisotasun posibleekin. Arazoa gero eta gehiago mugatzen eta arrazonamendu prozesua fokatzen duten maila desberdinak ezartzen dira.
- * Sintetikoak, hauetan soluzioa elementu ezagun batzuetatik eta aurretik ezarritako eragile batzuetatik abiatuz eraikitzen da, adz. AM [LENA82].

Ondo egituratutako arazoen rango zabal bat (diagnostiko, plangintza, etab.) sailkapen heuristikoa metodoez ebazten dira. MAMUR arazo mota hauen ebazpenerako egindako tresna da, horregatik komenigarria da metodo honen erabilera. Kasu hauetan datuak aurretik jarritako soluzio multzo batekin erlazionatzen duen inferentzia bereziaren egitura edukitzen da, abstrakzioz, asoziazio heuristikoz, eta birfinketaz; erabilitako errepresentazioetik zerikusirik ez duen arazo bat ebazteko beharrezko jakintza adieraziz.

MAMUR-ekin lortu nahi da JB-an jakintzaren errepresentazio erraz bat erabiliz ondo egituratutako arazoak ebaztea haizutzen duen tresna bat. Objektu batzuetatik abiatzen da, objektu hauen atributuek abiatze datuak dituzte (egintza bat objektu, atributu, balio, eta zihurtasun balio batek errepresentatua

geratzen da), eta nahi da soluzio objektuek hasierako egintzetatik eta JB-an utzitako jakintzetatik ateratzen den informazioa hartzea. Soluzio honetara heltzeko objektu desberdinen atributoak parakidetzen dira adituaren jakinmtza teorikoa eta enpirikoa errepresentatzen duten erregeletan oinarrituta.

CETODIA SA-n, hurrengo kapituluan zehazki ikusiko dena, pazientearen datuetatik eta haren datu analitikoetatik abiatzen da, era berean haren egoera jasan ditzaketen arazo posiblei dagokionez, denbora mugatu batean Arreta Intentsiboen Unitateko (AIU) tratamendu porozesuen barruan. Hauetatik ateratzen da arazoen egoera berria eta haren eboluzioa segidan oraingo denbora unean emon behar den tratamendua asoziatu ahal delarik. Lortutako tratamendua laboratorioek komertzializatzen dituzten produktuen ezaugarriei eta Erizaintzaren lan metodoari egokitu behar da, emaitza ahalik eta merke eta erraza izan daiten.



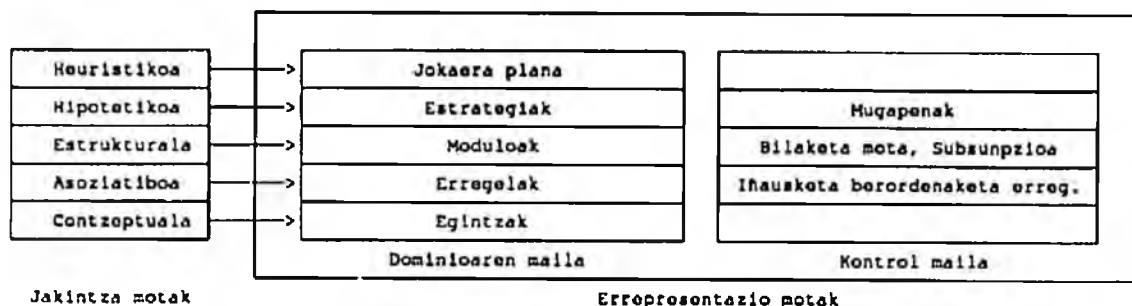
6.6 Irudia.- CETODIA-n erabilitako arrazonamendu metodoen eskema.

MAMUR-en ezaugarriak bereziki orientatuak daude mota honetako analisiak posible egiteko.

6.3.3.- MAMUR-en kontrol arkitektura.

Kontrolaren ikuspuntutik, eta MILORD-entzako [SIER89]-n esposatutako ordenari jarraituz, kontrolaren arkitektura bi dimentsiotan zatituko dugu:

- a) Jakintza moten dimentsioa.
- a) Errepresentazio elementuen dimentsioa, non honekin batera dominio maila eta kontrol maila bereiztuko ditugun.



6.7 Irudia .- MAMUR-en kontrol arkitekturaren eskema.

Jarraian jakintza mota bakoitza ikertuko dugu haietariko bakoitzari esleitutako kontrol motarekin.

MAILA KONTZEPTUALA

MILORD-en bezala, maila egonkorra da harengan eragina duten kontrol elementurik gabe, egintzak sortu edo aldarazten dituen ikasketa sistema induktiboa ezartzen ez bada.

MAILA ASOZIATIBOA

Erregelen bidez erlazionatzen dira egintzak. Kontrol mekanismoa hauen bidez dago errepresentatuta: erregelen aplikagarritasuna era esplizitioan zehazten duten metaerregelen bidez, eta kontrol inplizito baten bidez; TEIRESIAS, MILORD, etabarren ideari jarraituz. Metaerregelek erreferentzia egiten dioten moduloaren erregelen itausketa edo berordenaketa eragingo dute, jakintza mota kontutan izan gabe, erregela hauen "utilidad" atributuaren edukiera aldaraziz, dagozkien funtzioaren aplikazioaren bidez. Inplizitoki IM-ak modulo bateko erregelak

ebaluatuko ditu, honen arabera: erabilgarritasuna, zihurtasun gradua, espezifizitatea, eta idazketa ordena.

MAILA ESTRUKTURALA

Helburu zehatz batzu dituzten moduloetan biltzen ditu erregelak, ataza baten ebazpenerako behar den jakintza guztia biltzeko (ikus 6.3.1-en MILORD-i dagokionez dituen desberdintasunak) [STIC87]. Modulo hauek sailkapen prozesuaren egitura errepresentatzen dute soluzio partikularrak dituelarik. Ezin izango lirateke hartu [SIER89]-n egituratuko modulotzat hartzen dena bezalakoa, ez bait dago informazioaren ezkutaketarik ezta abstrakzio funtzionalik, eta nahiz eta goiko moduloetatik informazioa hereda daitekeen, gehiago da egintzen eskuragarritasuna de la eta herentzia mekanismoaren existentzia dela eta baino. Kontrol mekanismoak modulo baterako ebaluazio prozesu desberdinak zehazteko ezartzen dira: Bilaketa mota, zihurtasun gutxieneko balioa egintza bat kontutan hartzeko, subsunpzioa, etab. MAMUR-en gaurko bertsioan bilaketa mota, edo *modus ponens*-aren aplikazioa erregelen kateaketa motaren arabera, era esplizitoan espezifikatzen da eta gainontzeko puntuak era implizitoan, honako era honetan:

- Bigarren termino linguistikoa hartzen da, oraingo JB-rako definituen artetik, zihurtasun balio gutxieneko-tik abiatuz. MYCIN-en 0.2 zihurtasun faktorea (CF) da.
- Subsumpzio ondorioa lortzeko hauek lehenengo dispartzen dira: zihurtasun balio handiena duten erregelak, erregela espezifikoenak (baldintza kopuru handiagoa premisetan), eta idazketa ordena jarraituz. Honela jakintza ez osoa erabili ahal izango da beti espezifikoena erabiliko delako garantiarekin.

MILORD-en ere egin daiteke, era esplizitoan parametroen (bilaketa motarako eta zihurtasun atarirako) eta kontzeptuaren arteko jabetasun erlazioen bidez, kontzeptuak "mre" metaerregelen bidez ezarrita dauden maila hipotetikoan (subsunpziorako).

MAILA HIPOTETIKOA

Ebazpen estrategiak edo soluzio hipotesien ezarrera aipatzen ditu, garrantzi izugarrizkoa sistemari portaera adimentsua emateko. MAMUR-en unitate estrategikoa moduloa da, edo jakintza zatia (chunk), zeinei buruz fokatu behar den arazoaren gaurko egoerari egokiena izateagatik. Errepresentazio elementuak mugapen batzuen betetzearen ondorioz modulo baten prozesamendu osoa eragiten duten funtzioak dira. Mugapenen errepresentaziorako edozein beste jakintza mota bezalako formatoa duten erregelak erabiltzen dira, 2.5.1 atalean nabaritzen dena jarraituz kontrol informazioa "beste dominio" [DAVI80] bat dela kontsidera daitekeenari buruzkoa. Deitutako moduloak mugapen berriak edo dominioaren jakintzari dagozkien erregelak eduki ditzake arazo zehatz baten ebazpenerako. Ebatzi behar den kasuaren alderdi zehatzak aipatzen dute mugapenek estrategia zehazteko. Maila hau osatzen dute maila heuristikok dauden iharduketa planen ondorengo azpiplanak errepresentatzen dituzten modulo jerarkia batzuek. Eragile (edo mugapen) zeregina duten erregelen aplikagarritasuna zehazten duten metaerregelak egon daitezke, maila asoziatiboan esan den bezala, maila honetako modulo bakoitzaren barnean. Hemen arrazonamendu metodo desberdian inplementatu daitezke: sailkapen heuristikoa, sailketa sinplea, datuen abstrakzioa, eta [SIER89]-n seinalatuak bezalako beste batzuek. Bai Stefik-ek [STEF81a] bai Sierra-k [SIER89] propugnatzen dituzten joerari jarraituz jerarkia garbiak erabilki dira estrategiek zehazteko orduan, helburu honekin:

- Argitasunez erakutsi zein den sortutako estrategia egintza nabarmen batetik abiatuz.
- Metaerregelen aplikazioa kontrolatu adituari mugapen zehatz batzuei nabarmentasun handiagoa emateko aukera emanik, zein egintza aipatzen duen arabera.
- Mugapenen leherketa konbinatorioa ekidin.

honekin mugapen egituraketa handiagoa lortzen da, eta adituari haizutuko dio horietako bakoitzean gehiago konzentratzea guztien funtzionamendu orokorragatik arduratu gabe. Subsumpzioa estrategiei ere aplikatuko zaie.

MILORD-en kasuan errepresentazio elementuak metaerregela espezifikoko batzu ("mre" metaerregelak) sortutako moduloen identifikatzaileen zerrendak dira. Metaerregela hauek dominioaren egintzak moduloekin asoziatzeko balio dute; bisitatu, deuseztu edo tratatu behar diren moduloak esplizitoki definituz; estrategiaz aldatu ahal delarik arazoaren egoera aldatu bezain laster, sistemari exijitzen zaien eekarrekintza dela eta, maila heuristikoaz geratzen delarik eman daitezkeen gatazken ebazpena. Badira ere metaerregela batzu salbuespenen antzematerako ("mrx" metaerregelak). Erregelenak bezalako algoritmo batekin ebaluatuko dira. Aldez, MAMUR-en beti RETE algoritmoa erabiliko da erregelak errepresentatzen duten jakintza motatik independienteki.

MAILA HEURISTIKOA.

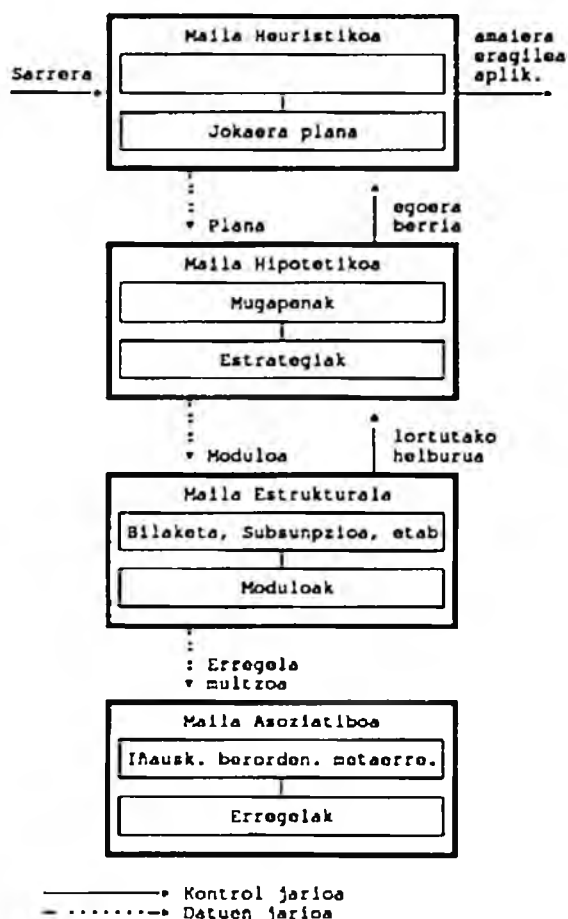
Gaur eguneko MAMUR-en bertsioan, era deklaragarrian espezifikatzen den iharduketa plana dauka, egoeren espazio baten bidez errepresentatua modulo orokorrenen barruan. Eragileek (erregelak) egoera (objektu zehatz batzuen egoera) batetik bestera igarotzea eragiten dute, edo prozesuaren amaiera soluzio egoerara heldu delako. Estrategiek bakarrik iragartzen dute soluzio desberdinak analizatu behar diren ordena, horregatik estrategiak koordinatu eta arazoaren soluzio orokorra lortzeko estrategiak prozesatu behar diren ordena marka ditzan beste geruza batek egon behar du. Aurreko mailak bezala, arazoen ebazpen metodoak ere hautatzen ditu, baina aurreko mailan soluzioen espazioaren barneko azpiespazioak hautatzen dira oraingo mailak dadgokion azpiespazioan bilaketa egin dezan. Maila honetako eragileen aplikagarritasuna zehazteko ere erabil daitezke metaerregelak. MILORD-en maila honetan, maila hipotetikoari buruzko informazioa baino ez da erabiltzen, horregatik dominiotik independientea da. Metaerregela berezi batzu erabiltzen dira planen sorkuntzarako ("mrp" metaerregelak) "mre" metaerregelak sortutako estrategia guztiak konbinatzeko enkargatzen dena. Plan bat frame batek errepresentatua geratzen da, hauen slot-ak iragarriko dute: planen zihurtasun gradua,

azaldutako unitateak, deuseztuko moduloak, bisitatutako moduloak, estrategiak definitu duten "mre" metaerregelen multzoa, planaren justifikazioa dominioaren informazioaren bidez, eta estrategien konbinatzaileak edo "mrp" metaerregelak. Amaiera metaerregelak ere badaude ("mra" metaerregelak) adituak prozesua bukatutzat emateko definitzen duen kontrolari buruzko jakintzarekin.

Jakintza motaren maila desberdinetan banaketak eta deklaragarritasuinak haizutzen dute; JB-ren beharrazko diseinua, arrazonamendu metodo desberdiank, eta sistemaren portaera sinesgarriagoa. Gainera, Además, jakintza-ingeniariari abantaila hauek eskeintzen dizkio: Luzapengarritasuna, arazketagarritasuna, eta azalgarritasuna.

6.3.4.- Kontrol zikloa MAMUR-en.

Lehenago aurkeztutako mailak elkarekiten dute, JOS-en exekuzio prozesuan zehar, 8.8 irudiak iragartzen duen bezala (ikus 6.9 irudia, MILORD-ekin konparaketak ezartzeko). Maila heuristikoa esplizitoki adierazitako plan bat dago, hau gauza honetaz arduratzen da: Sarrera datuen hartzea, hasierako egoeraren egiaztatzea, eta geroagoko egoeren sorkuntza azken egoera antzeman arte. Ez dago metajakintza bat planen sorkuntzarako, MILORD-en gertatzen den bezala. Behin eragile baten aplikagarritasuna egiaztatzen denean, eragile honek egoera zehatz batera heltzeko exekutatu behar den azpiplana iragarriko du. Hautatzen da azpiplan hau eta maila hipotetikora jaisten da. Maila honetan modulo jerarkia bateri dagozkion erregelak mugapenen zereginean erabiltzen dira eta arazoen gaurko egoeraren informazioarekin bete ondoren, kasu honetarako beharko den jakintza fokatuko dute eta jarraian aplikatuak izango diren moduloak zehaztu. Haretariko lehenengoa hartuz maila estruktruralera jeisten da, eta hemen jarraiko huek erabiliz: moduloaren izenari asoziatutako bilaketa motaren parametroa, eta sisteman inplizitoki dauden kontrol informazio beste mota batzu, egintza



6.8 irudia.- Kontrol zikloa MAMUR-en.

algoritmoaren aktibazioarako haren erro nodotik (NRZ), eta honekin sortutako objektu-atributu-balioari dagozkien memoria nodoak eguneratzen dira (RETE arbolan) gero aktibatzekeko erregela berri bat ere gordetzera (iragazitako erregelen agendan) heldu ahal delarik. Erregela hau exekuzioan dagoen moduloarena edo beste edozeinena izan zitekeen, beraz, modulo batean erabakitako guztia gardena da modulo guztientzako.

Maila asoziatiboak bere lana bukatzen du bilatutako helburura heltzen denean (atzeranzko kateaketa) edo erregela aktibagarri gehiago ez dagoenean (aurreranzko kateaketa), prozesuan dagoen moduloan. Une honetan bukatzen da maila hipotetikoak seinalatutako estrategiaren moduloaren prozesua, prozesu bera errepikatuz kurtsoan dagoen estrategiaren hurrengo

bat kontutan hartzeko zihurtasun maila gutxienezkoa bezala, subsuntzio era bezala, etab; haren ebaluazioa hasten da. Jarraian, jadanik maila asoziatiboan modulo honen inhausketa edo berordenazio metaerregelak aplikatuko dira, baldin badaude, aurreranzko kateaketaz. Moduloaren erregela aktibagarrirei RETE algoritmoa aplikatuko zaie, kasu honetan erabilgarriak direnak iragazteko, zeinek iragazitako erregelen agendan geratuko bait dira. Agendaren erregelak maila estrukturaletik jasotako erizpideen arabera ordenatuko dira, egintza bat edo batzu sortuz lehentasunezkoenengoa aktibatuz. Egintza berriak seinale (token) gisa egingo dute RETE

kontrolak, eraginkortasun handia lortzea haizutzen dute.

6.3.5.- Metaerregelak MAMUR-en.

Premisa desberdinetatik abiatuz helburu berberak ateratzen dituzten erregela asko egon daiteskeela eta, egiaztatzen diren ordena ere garrantzitsua da, prozesuaren abiadura handiagotzeko, horregatik arazoaren ebazpena nola dagoen arabera interesgarria izan daiteke modulo bateko erregelen ordenazioa modu zehatz baten edo haietarikoren bat kontutan ez hartzea. MAMUR-ek modulo bakoitzari esleigarriak diren metaerregelen multzoa erabiltzeko aukera ematen du, dagoen mailaz aparte (ikusi 6.5 irudia), multzo hauek aurreranzko kateaketaz prozesatuko dira modulo horretako erregelen inhausketa edo berordenaziorako. Modulo baten prozesamendu funtzioa exekutatzen denean, haren erregelen iragazketa hasi baino lehen egiaztatzen da modulo horretarako metaerregelen modulo bat dagoen, errepresentatzen duen objektuaren "meta-reglas" atributuaren bidez. Metaerregela hauek ere gainontzeko erregelen egitura bera dute, eta honek, egitura modular multimaillarekin batera, kontrola deklaragarria izatea eragiten du argitasuna gehitzeko eta SA-ren eraikuntza eta mantenimendua sinpletuz.

Metaerregela bakoitzak, gainontzeko erregelek bezala, erregelen modulotik aipatzen den modulo baten menpeko objektua osatzen dute. Metaerregelen moduloaren izena erregelen moduloaren izenarekin gehi "MRPR" aurrizkiarekin egiten da. Metaerregelen moduloak IM-aren META-REGLAS-MRPR klasearen menpean egongo dira. Metaerregela baten definizioa JB-an 6.10 irudiko funtzioaren bidez egingo da, honek esan nahi du

```
(def-meta-regla
  metaerregelaren izena
  eragindutako erregela moduloen izena
  premisak
  =>
  ondorioak
  [erabilgarritasuna]
  [metaerregelaren-zihurtasun-balioa]
  [dokumentazioa]
)
```

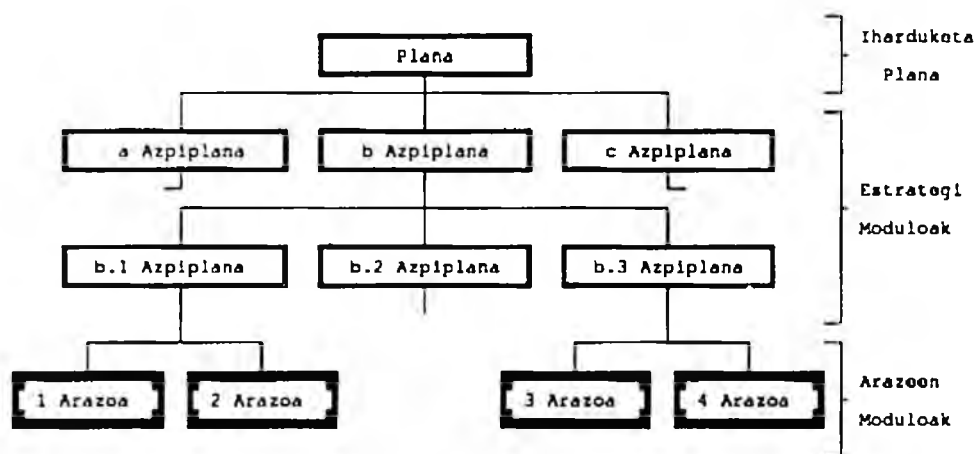
6.10 irudia.- Metaerregelak definitzeko funtzioa.

META-REGLAS-MRPR-ren menpeko klase bat sortuko dela MRPR-<moduloaren izena> izenarekin, ez bada existitzen. Metaerregelen modulo honi instantzia bat erantsiz metaerregelen izenarekin.

Metaerregelek erregelen egitura bera dute, RETE zuhaitzean errepresentatuta egongo dira, eta ere berean interpretatuko ditu IM-k.

6.3.6.- Plangintza jerarkikoa.

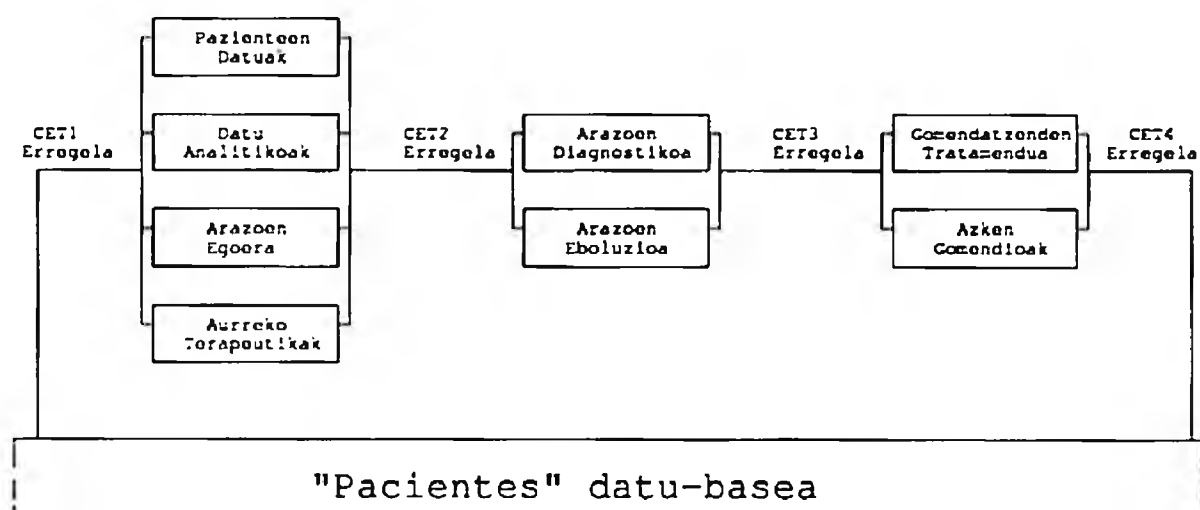
MAMUR-en ez da plan bat sortzen baizik eta esplizitoki seinalatutako bat exekutatu egiten da. Plana azpiarazoetan zatitzen du, aurretik MOLGEN-entzako adierazitako arrazoi



6.11 irudia.- MAMUR-en JB bateko moduloen jerarkia.

berberengatik, eta haien arteko eekarrekintza seguratzen du. Arazoaren xehetasunen kontutan hartzea diferi daitekeen kasuetan erabili ahal izango da, hau posiblea da sailkapen arazorik gehienetan eta CETODIA-n partikularrean. Mugapenak dituzten moduloen maila desberdian egotea, haien gainean eragina izan dezaketen metaerregelez gain, beste mugapen batzuen ganean eragina duten mugapenak erabiltzea haizutzen dute. Metamugapenen erabilera ezin daiteke eman MOLGEN-en.

Estrategiak ezartzen dituzten moduloek jerarkiak osatzen dituzte (6.11 irudia), zeintzuen erregelak mugapenen zeregina egiten dute, mugapen hauek bete beharko lirateke azpiko moduloetan errepresentatutako jakintza erabiltzeko. Ez dute bakarrik zein modulo erabili esaten baizik eta nola ere bai. Modulo bakoitzak plan edo ataza bat dauka, esplizitoki espezifikatua JB-aren garapenean zehar, egoeren espazioak bidez errepresentatua.



6.12 irudia.- CETODIA SA-ren mailarik altuenean exekutatzeko plana.

JB-an definitutako objektu desberdian *egoerak* osatzen dituzte; eta *erregelak*, zeinek ahalbideratzen duten egoera berrien sorketa soluzio egoerara iritsi arte, *eragileak* osatzen dituzte. Mailarik altuenak emaitzen lortzerako exekutatu behar den plana dauka; 6.12 irudian ikus daitekeenez (CETODIA-ren mailarik altuenei dagokiona) hau plan ez osoa da, zeinetan eragileek seinalatzen duten zer egin behar den baina ez nola egin behar den, azken hau azpiko mailen moduloetan espezifikatzen da. Azpiko maila bakoitzak goiko mailaren etapa edo azpiarazo baten ebazpenerako azpiplan edo azpiataza bat osatzen dute; honek kontsiderazio garrantzitsuak fokatzea haizutzen du, honela zailtasuna gutxituz eta azpiarazoen aurretiko banaketa ahalbideratuz [STEF81a].

```

(defregla cat2 cetodia
  ((datos-entrada recibidos = s1))
  =>
  ((E (adelanto diagnostico-p))))
;;.....
(defregla tra21 tratamientos
  ((problemas-ant edema-cerebral = s1)
   (problemas-act edema-cerebral = no)
   (terapias-ant intubacion = s1))
  =>
  ((E (atras t-edema-cerebral
        (tratamiento edema-cerebral)))))
;;.....
(defregla tra28 tratamientos
  ((datos-act tiempo >< (12 24))
   (potasemia aumento-de-potasio /= s1))
  =>
  ((potasemia concentracion-de-potasio = 30)
   (fosfatemia aumento-de-fosforo = s1)))
;;.....
(defregla tra31 tratamientos
  ((potasemia concentracion-de-potasio = ?x)
   (potasemia aporte-total-potasio = ?atp)
   (fosfatemia aumento-de-fosforo /= s1))
  =>
  ((terapias-act clk = (E (/ ?atp 2.1)))))
;;.....
(defregla th115 t-hidratacion
  ((hidratacion perdidas-previas = ?z)
   (hidratacion mantenimiento = ?y))
  =>
  ((E (vor-aviso cal-hidricos))
   (E (preguntar hidratacion perdidas-previas))
   (E (vor-aviso mantenimiento ?y))))

```

6.13 Irudia.- Planaren eragile edo erregela mota desberdina.

Azpiarazoen eekarrekintza RETE algoritmoaren egiten den implementazioaren funtsezkotasunaren mekanismoak ebazten du

Arazoari soluzioa emango dien eragile multzo antolatu batek osatzen du plana, arazoaren hasierako egoerari aplikagarria bada eta helburua lortzen badu exekuzioaren ondoren; beraz, lehenengo eragileak egiaztatu baherko du ea datuak edo hasierako egintzak espero direnekin bat datozen eta ez gaudela SA-k ebatzi dezakeen kasu-

tik kanpo geratzen den kasu baten. Egoera batetik bestera pasatzeko eragile (erregela) baten baino gehiagoen aktibazioa behar daiteke, baita kasu konkretu baten haietariko bat beharrezkoa ez izatea. Aktibazio ordena eragileen aurrebaldintzetan parte hartzen duten aldagaien eta eragile hauen erabilgarritasuna (metaerregelen bidez aldaruzgarria) egungo egoeraren arabera izango da; partzialki ordenatutako planak eraiki ahal direlarik, eta konpromesu gutxiagokoak ere. Eragileek sortarazi dezakete: azpiko mailako moduloaren erabilera, beste eragile batzuen aurrebaldintzak konplitzeko beharrezko egintza berrien ateratzea egoera baten abisu azalgarri baten igorpena, erabiltzaileari galdera zuzena egintza ezagun edo ateratako baten balioa aldarazteko, eta egoera berria errepresentatzen duten objektuen atributuei balioen esleipena (ikus 6.13 irudia); hau da, MAMUR-ek erabili ahal dituen mota guztietako erregelak izan daitezke. Modulo hauek aurreranzko kateaketaz exekutatu dira egoera bakoitzerako eragileen baldintza guztiak egiaztatze.

Bilaketa espazioaren kontrolerako erabili dira, orain arte garatutako sistemetan, hauek bezalako oinarritzko metodo batzu:

- * Baliabide Helburu Analisia (Means-ends Analysis), GPS-an (Ernst eta Newell, 1969) sartua eta geroagoko sistema askotan erabilia.
- * Konpromesu Gutxiagokoren bidezko Bilaketaren Murriztea (Search Reduction through Least Commitment) NOAH-rena.
- * Bilaketa Sakona Atzera Jotzearekin (Depth-first Backtracking), PROLOG-en bezala.
- * Azau Bilaketa (Beam Search), ISIS-II-en bezala.
- * Atzera Jotzerekin heuristika lokalak emandako aukera hoberena bidezko Bilaketa (One-then-Best Backtracking). NONLIN (Tate, 1977).
- * Menpekotasunak Zuzendutako Bilaketa (Dependency-Directed Search), MOLGEN-en eta beste batzuetan erabilia.
- * Bilaketa Oportunistikoa (Opportunistic Search), MOLGEN-en eta harbel sistemetan.
- * Metamaila Plangintza (Meta-level Planning), MOLGEN-en.
- * Plangintza Banatua (Distributed Planning), harbel sistemena.

Gainera Bilaketa Espazioaren Iñauusketa Orokorren metodo batzu erabili dira (General Pruning of the Search Space) [DRUM89], hauek bezalakoak:

- * "Lehentasun handiagoko" helburuak kontsideratu lehen-bizi planifikatzaile jerarkikoan, ABSTRIPS-en [SACE73] bezala.
- * Elkarrekintzak era adimentsuan antzeman eta zuzendu, NOAH-n [SACE75] bezala.
- * Egoera edo plan ezagun, ezinezkoak, edo egoeraren edo eguneko planaren erregelen batekin kontrajarriak atzera bota, WARPLAN-en (Warren, 1973) [DRUM89] *impossible* sententziak bezala, *dominioaren mugapenak* [ALLE83b]

eta koherentzia iragankorra-ren Drummond eta Currie, 1989) [DRUM89] teknika bezala.

- * Errekurtsoaren erabileraren maila mugatuz, denbora mugapenak ekintzen gainean, etab. NONLIN, DEVISER, eta SIPE-n bezala [WILK83][WILK84][DRUM89].
- * Eragileen aurrebaldintzak idatziz (typed preconditions) NONLIN-en aurkeztua eta O-PLAN eta SIPE-n estentsiboki erabilia.

Kontrola bilaketa espazioan, MAMUR-en, mugapen zeregina egiten duten eragileen aktibazioagatik sortarazitako erabakia lortzean oinarrituta dago. Moduloen jerarkia eta abstrakzio mailen erabilera eginez bilaketa estriktoak jarraitzen du mailez atzera jotzeko ahalbiderik gabe. Eragileen aurrebaldintzak (metamailaren erregelen premisak) esplizitoki idazten dira, NONLIN, SIPE, eta O-PLAN-en estentsiboki erabilitako soluzioak.

Kontrol era honek honako abantaila hauek eskeintzen dizkigu:

- 1) Aurrebaldintzen edo premisen idazkera esplizitoak JB-aren eraikuntza errazten du.
- 2) Atzera jotzerik ez egoteak eraginkortasuna areagotzen du.
- 3) Azpiko mailetak moduloetara heltzen denerako egiaztatuta dira haren erregelak aktibatzekeko baharrezko premisa batzu.
- 4) Premisen kopurua murriztu daiteke azpiko mailetak erregeletan, hauek goiko mailetan aurretik mugatutako jakintza erabiltzen dutelako, aurreko puntuan seinaltutakoaren arabera.
- 5) Modulo baten beste edozeinetan ateratako egintzak erabiltzeko ahalbideak ere areagotzen du eraginkortasuna.

Nabarmendu behar dira ere, horri buruz, jarraiko eragozpenak:

- 1) Modulo bateko erabilera eksklusiboko aldagaiak beste modulo batzuetatik eskuragarriak dira, eta hau ez da beti desiragarria.
- 2) Ez bada nahi premisa errepikaturik egotea azpiko mailletako erregeletan, premisa hauek erregelen ezaugarri nagusienetarikoa galduko dute: modularitatea; goiko mailletakoenaren (erregelen) menpekotasuna dela eta. Edonola ere, modularitatea preserbatu nahi bada, premisa batzu berridatziz, RETE algoritmoak behin baino egiazta ez daitezzen egingo du.

Denboraren errepresentazioa oso elementala da, objektu desberdinak sortu behar bait dira denbora une desberdinetan kontzeptu baten egoera errepresentatzeko. Arrazonamenduan zehar, haietariko bakoitzera esplizitoki heldu daitezke, nahiz eta ahalbidea egon hauetara ere heltzeko:

- azken balio ezagunera.
- aurreko denbora guztien balioera.
- denbora mugatu bati dagozkion baliora, etab.

Zerbait berdintsua gertatzen da MOLGEN-en eta beste batzuetan.

6.3.7.- Mugapenen errepresentazioa erregelen bidez

Aldez aurretik esan denez, egintzek bete behar dituzten mugapenak, bai hasierakoak bai hauetatik abiatuta ateratakoak, kasuaren jakintza aplikagarria eskuratzeko erregelen bidez errepresentatzen dira jakintzaren gainontzekoa bezala. IM-ak erregela honek dominioarenak bezala interpretatuko ditu RETE algoritmoa aplikatuz. Hortaz, ez da behar inolako prozedura berezirik kontrolari buruzko jakintza errepresentatzeko, nahikoa da moduloak erregelekin era jerarkikoan esplizitoki definitzea, L. Steels-ek [STEE85] aurkeztutako estrategiaren berdin bati jarraituz. Mugapen bakoitza (emandako definizioaren arabera)

erlazionatutako aldagaien multzoa da. Premisa hauetako bakoitza, era berean (ikus 3. kapitulua) objektu, atributu, konparazio funtzio, balio, eta zihurtasun balio batek osatua dago.

RETE ZUHAITZA MUGAPEN SARE GISA.

Erregelek mugapen zeregina egiten badute erlazionatzen dituzten RETE zuhaitza mugapen sare bihurtzen da, mugapen desberdinen aldagaiak konpartitzen bait dira. Sare hau ez da 6.2.2 atalean definitutakoaren berdina, hau da, ez dago bakarrik aldagai konpartituak errepresentatzen duten nodoez eta erlazionatzen dituzten arkuez osatuta. Hala ere, mugapen sare baten helburua mugapenak onartzen dituzten balioak iragaztekoa ere bada [GUSG88], RETE zuhaitzak erregela aplikagarriak zehazteko erabiltzen duen era berdintsuan. Azken kasu honetan, eraginkortasun konputazionala handiena bilatuz ez dira aldagaiak (premisak) bakarrik konpartitzen baizik eta premisa bakoitzaren osagarriak¹ ere konpartitu daitezke (baita dominioaren erregelekin ere). Gainera, ebaluazio partzialaren memorizaziorako nodo batzu dituzte; premisen osagarrien betetzearen egiaztatzea, premisa osoarena, edo premisa bat baino gehiagoren elkar betetze ez errepikatzeko.

RETE zuhaitzaren erabilera mugapen sare gisa abantaila handia da haren zuhitz egiturak mugapenak atzera jotze gabe ebaluatzea haizutzen duelako [FREU82][DECH85], ondoriozko porozesuen aurrezkiaz. Behin RETE zuhaitza sortuz gero arku eta bidearen funtsezkotasuna, lehenago definituak, bermatuta geratzen dira. Laugarren kapituluan iragartzen denez zuhaitz hau JB-aren konpilazioan zehar sortzen da, behin bakarrik, gero sistemarekin batera kargatzen delarik.

¹ Zehatz-mehatz objektu, atributua eta balioa. Hainetarikoa bakoitzorako nodo bat egongo da, iragazketa aldagai batek errepresentatuta ez baldin badago.

FUNTSEZKOTASUN LOKALA ETA OROKORRA BATERA.

Funtsezkotasun lokala eta orokorra batera lortzen dira helburu bakoitza behin baino kalkulatzeko ez delako eta ez da aldarazten (monotonikoa) [FREU82]. Iragazten diren egintzak ondoko nodoetan eragina dute zuhaitzak duen ordenaren arabera. RETE algoritmoaren funtzionamendu berak mantentzen du funtsezkotasun lokala eta orokorra beti edozein egintza berrik RETE zuhaitza egunera dezan eginik.

DESBERDINTASUNAK BESTE SISTEMA BATZUEKIN.

CONSAT-i buruz desberdintasun hauek eza zitezkeen:

- a) MAMUR-ek hedatzen ditu bakar-bakarrik RETE algoritmoaren terminologian seinale (token) bezala ezagutzen dena eta egintza ezagun edo ateratakoa errepresentatzen duen zerrenda bat eratzen duena osagai hauekin: objektu, atributu, balio, eta zihurtasun balio. Ezin ditu balio multzoak hedatu.
- b) Mugapen erlazioak era hauetan egin daitezke; [GÜSG89a]-an CONSAT-entzako erabilitako terminologia erabiliz, beren berdintasunagatik:
 - Erlazio finitoak objektu baten atributuaren balio batekin edo balio rango batekin izatearekin konparatzen duten erregelek errepresentatuko dituzte.
 - Erlazio infinituak erregelen bidez, atributu baten edukia aurretik ezarritako metodo baten aplikazioaren emaitzarekin, edo beste atributo baten edukiarekin (bere balioa, beste mugapen baten ondorio aldeko funtzio batekin kalkulatu izan dena; CONSAT-en "patroi"-aren baliakidea izango litzatekena) konparatuz.
 - Erlazio konplexuagoak mugapenen beste modulo batzuetan dauden mugapen sareek errepresentatuko dituzte.

- c) Iragazketa prozesuak arrakasta badu kalkulaturako balioak EGB-ri eta RETE zuhaitzeko memori nodoei eranstsiko zaizkie.
- d) MAMUR-en mugapen sarearen edukia RETE zuhaitza algoritmorako zuhaitza izateak zehaztua dator. Aldiz, CONSAT-en hauekin osatuta dago: mugapenak, aldagai orokorrak, aldagaien estaldurak aldagai lokal eta orokorren arteko proiektzioak, eta interfacea. Badira, gainera, mugapen sare primitiboak eta mugapen sare errekurtsiboak eraikitzeke aukera ere.
- e) Kontrol standarreko estrategia hedakuntza lokalean oinarrituta dago. CONSAT eta beste askotan bezala, eta antzeko funtzionamendu batekin.
- f) Hedakuntza lokala orokorrarekin batera dator eta ez da beharrezkoa atzera jotzea. CONSAT-en kontrol estrategia bi eskeintzen dira funtsezkotasan orokorra lortzeko.
- g) Mugapen aldagaiak bakarrik asoziatu daitezke balio zehatzekin (Booleanoak edo ez) edo balioen rangoekin. CONSAT-en edozein balio klaseekin.

CONSAT aplikagarritasun orokorragoduna da, MAMUR-ek gehiago bilatzen du eraginkortasuna.

Zuhaitzaren aurrekonputazioaren eta norabidegarritasunaren abantailak nabari geratzen dira beste lan batzuetan ere [DECH89].

Mugapenen Betetze sistemetan MAMUR-en egineko bertsioan soportatzen ez diren kontzeptu batzu agertzen dira:

- Mugapenen Betetze Partziala [FREU89], ahal duen soluziorik onena lortzeko edo soluzio nahikoa MBA-ak soluzio zuzena ez duenean.
- Mugapen Tenporalak, LOGOS-en [MENG91] eta beste batzuetan [ZWEB89].
- Mugapenen Lasaigarritasuna, mugapen ezagun batzuk leuntzeko kasu zehatzean aplikagarriak diren estrategia deklaragarrien bidez [MENG91].

- Mugapenen bidez ez espezifikatutako objektuen Deskribapen Partzialak, SIPE-n erabilitakoak, dominioengandik independentea den plangintza errazteko.
- Etab.

6.3.8.- Kontrola Arazo Errazen ebazpenaren bidez.

Lehenago definitu den bezala [DECH85] arazo bat erraza dela pentsatzen da haren errepresentazioak denbora polinomialean soluzio bat haizutzen duenengan. Hortaz, atzera jotzeko algoritmoekin tratatuz, esan dezakegu MBA bat erraza dela atzera jotze gabe ebaluatu izan ahal denean. Sistema honetan kasu bati aplikagarria den jakintza zehazten duen mugapen egitura osoa aurretik sortutako zuhaitza batek errepresentatzen du, horregatik sistemak jakintza honetara iritsi ahal izango da era eraginkorrean arazo "erraz" baten ebazpenagatik.

MAMUR-en kontrol estrategia standarra hedakuntza lokalean oinarrituta dago: Mugapen sare bat aktibatzen denean hasierako balioekin aldagai batzuentzat, mugapenak ebaluatzen dira eta emaitza alboko mugapenetara hedatzen da, eta horrela beti. Prozesua bukatzen da aldagaien balio berririk ez dagoenean. Aurreko prozesuan zehar beharrezkoak izan daitezen dominioaren moduloak aktibatuz joango dira dagozkien mugapenek sortutako aholkua jarraituz, atzera jotzerik egin gabe; hau da, arazo "erraz" baten ebazpenaren bidez.

6.3.9.- Konpilazioaren ondorioa MAMUR-en eraginkortasunaren hobetzean.

IM-k jakintza gehigarria du nahi den helburua azkarrago lortzeko. Adibidez atzeranzko kateaketan prozesua gelditzen da

helburua zihurtasun osoz zehazten den bezain laster, helburu batzu egiteko geratu arren erregeletan iragazketa aldagaiak erabiltzearen ondorioz. Jakintza hau, beste batzuekin batera, inplizitoki dago IM-an. Hala eta guztiz ere, MAMUR-en eraginkortasunean eragin handien duen jakintza JB-ren konpilazioaren bidez lortzen da. Prozesu honen zehar, karga azkarragoko JB lortzeaz gain, RETE zuhaitza orokorra sortzen da erregela guztietarako eta informazio gehigarri osoa, jarraikoa haizutzen duena:

- * Modulo bakoitzerako erro nodo bat izatea, honek MAMUR-en modularitatea ahalbideratzen du.
- * Metaerregelen moduloak dituen metamaila osatzea.
- * Egintza baten ateragarritasuna ezagutzea HECHOS klaseko objektu bakoitzak JB-an definitutako atributu bakitzean "deducible" fazetaren sorketaren bidez, fazeta honek iragarriko dio ateragarria den JB-an eskuragarri dagoen jakintzarekin edo ez. Informazio hau bereziki erabilgarria izango da atzeranzko kateaketan beharrianik gabe ez sortzeko eguneko helburua zehazten duen iragazketan zehar atera beharreko helburu berririk. JB-an parte har dezaketen egintza guztien ateragarritasunaren zehaztapena kostu konputazional bistakoa, baina bakarrik egingo da konputazio fase honetan zehar.
- * IM-aren eskuetan jartzea egintza batena den objektu izen bakoitzari dagokion sarrera bateko nodoarekin nodoarekin asoziazio zerrenda bat. Informazio honek lagunduko du nodoa zuzenean zehazten, nodo honen bidez hasi beharko da RETE zuhaitzaren egiaztapena edo eguneratzea, hasierako egintza edo sortutakoari dagokien seinale edo token batentzako; hau ezinbestekoa izango da praktikan JB erreal baten RETE zuhaitzak tamainu oso handia izango duelako.

Guzti hau bakarrik egingo da konpilazioan zehar eta ondorengo JOS-aren exekuzio eraginkorra lortuko du.

7. KAPITULUA: CETODIA: MAMUR- EN APLIKAZIO BAT.

7.1.- SARRERA.

Tresna hau aplikatua izan da CETODIA SA eraikitzeke (SA Zetoazidosi Diabetikoaren tratamendurako umeetan). Sistema honen garapena LABEIN-en hasi zen 1989.ean, lehen prototipo bat PERSONAL CONSULTANT PLUS tresna erabiliz lortzen zelarik.

Luis Aldamiz-Etxebarriari bereziki agertu behar diot nire esker ona kapitulu hau egiteko eman didan laguntzagatik.

7.2.- ARAZO MEDIKOA: ZETOAZIDOSI DIABETI- KOA.

Diabete Mellitus odolean glukosaren gehitzeak (gluzemia) karakterizatzen duen Sindromea da, la sangre (Glucemia), eta I motako Diabetea edo Haur Diabetea izendatzen da insulina jarioa benetan murriztuta dagoenean bai baraurik bai eragingarri desberdinei erantzunez.

Haren debuta, normalean, hogeitamar urteak baino lehenago ematen da, maiztasun gehienez hamabi eta hamasei urteen bitartean agertzen delarik, eta haren prebalentzia⁴ milako 3,5ean estimatzen da. Hasiera klinikoa aldiunekoa izaten da.

Gure mendearen hasieran, insulina eskasia latza zuen diabetiko baten bizi itxaropena diagnostikotik hasi eta urte baterakoa zen. Paziente mota honek XVII menderako Inghaterran "The Pissing Evile" bezala deskribatutakoaren sintoma eta zeinu kliniko karakteristikoak zituen, etengabeko egarria, gosea, poliuria bortitza, nekea, eta argaltzea. Glukosuria zetonuriak lagunduta zeinu diagnostikoa zela pentsatzen zen. Gaisotasunaren ibilbidea laburra eta atzeraezinezkoa zen, koma zetoazidotikoan eta heriotzean bukatzen zelarik.

Hogeigarren hamarkadaren hasieran Throntoko Unibertsitateak intsulinaren aurkitzeak eta haren aplikazioa giza-medikuntzan, erabat aldarazi zuten paziente hauen iragorpena eta laster kontsientzia eduki zen nolako arazoa zen kontrol egokirako beharrezkoak diren insulina maiztasun eta dosiak erabakitzea eta haietariko konplikazio batzu akiditzea. Gaur egun paziente hauen manei terapeutikoa ez dago guztiz ebatzia eta ahalegin ikertzaille sakonaren gaia jarraitzen du izaten, horrela medikuntza lan ugaritan hauen kontrola hobetzeko algoritmoak eta aholkuak erakusten dira, bai egoera basal edo konpentsatutan bai **zetoazidosia** bezalako deskonpentszio aldiunekoaetan. Honela ulertzen da paziente hauen maneiak arlo honetako espezialisten eskuetan egon beharko duela.

⁴ Gaur egungo eragindako gaixo kopurua inguruko bizilagunen artean.

7.2.1.- Clinika.

Zetoazidosi diabetikoan gertatzen diren jasokuntzak azaltzeko, intsulinak duen oinarritzko zeregina berehalako hiru printzipioetan metabolismoan gogoratu behar da.

INTSULINA

Areak isuritako hormona bat da, honen oinarritzko zeregina organismora iristen diren baliabide energetiko guztien probetxamenduari laguntzea da (hau da karbohidratoen, proteinen eta gantzen), zeinen kontra aritzen den.

Karbohidratoak elikagaietan almidoi edo polisakarido txikiagoen eran aurkitzen dira. Hestera heltzen direnean bihurtuak izaten dira, liseri-entzimei ezker, monosakaridoak deitutako molekula txikiagoetan. Honetatik garrantzitsuena glukosa da, zein liseri-tutuan zehar odolera pasatzen den eta hemendik, intsulinaren ondorioz organismoaren zelula desberdinetara sartzen da, non energia iturri gisa artzen den. Helburu honetarako beharrezkoa ez den glukosa giblean bilduko da molekula kate luzea eratuz Glukogeno deitua. Ganera, intsulinak glukosa gantz bihurtzeari haizutzen dio.

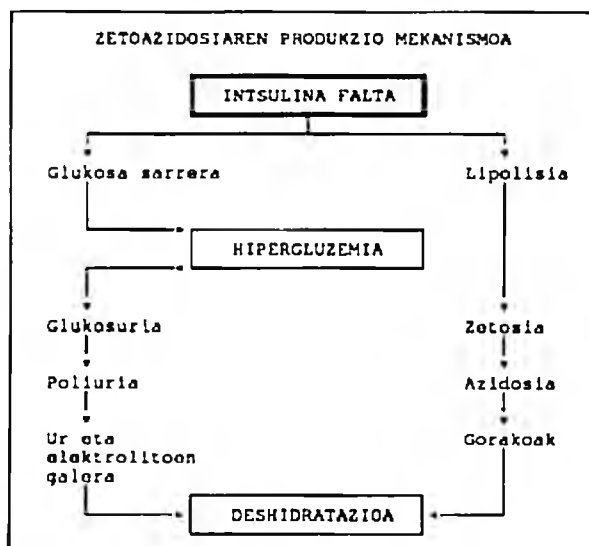
Proteinak aminoazidoak deitutako elementu txikiagoez osatuta daude. Hestera heltzen direnean hidrolizatuak dira haien absortzioari laguntzeko. Intsulinak zelulak aminoazido hauek erabiltzeari laguntzen dio eta ondorioz proteina sintesiari.

Gantzak triglizerido deitutako elementu txikiagoez osotuta daude. Odolean hestearen absortzioaren ondoren sartzen dira eta hauexetan deskonposatzen dira: gantz-azidoak, eta glizerola. Gantz-azidoak zelulek ere erabil ditzakete energia gisa edo gantz-ehunean gantz etara gordeak izan daitezke. Prozesu honek lipogenesi izena hartzen du eta intsulinak lagunduta ere badago.

ZETOAZIDOSI DIABETIKOA.

Paziente diabetikoarengan sarrien ematen den deskonposizio aldiunekoa da eta debut-ean edo egoera berezietan intsulinarekin egindako tratamenduan zehar ematen da. Intsulina eskasia batek (bereizten da) du eta estres hormonak (Kortisol eta Katekolaminak) gehipen batek bereizten dute (7.1 irudia).

Se caracteriza por un déficit insulínico y un aumento de las hormonas de estrés (Cortisol y Catecolaminas) (Fig. 7.1).



7.1 irudia.- Zetoazidosiaren produkzio mekanismoa.

Intsulina faltak ehunek glukosa gutxiago erabiltzea eragingo du, honek glukosa odolean gehitzea motibatuko du (hipergluzemia). Organismoa bere eskasia energetikoa konpenzatzeko saiatzen da beste energia batzu mobilizatuz, honetarako haren gantz eta proteina erreserbak erabiltzen ditu (lipolisi, proteolisi).

Glukosaren gehiketa odolean giltzurrunak deusztu du (Glukosuria), eta honek diuresi garaia (Poliuria) eragiten du, ondoriozko elektrolitoen galerarekin (Sodio, Potasio, Kloro, Magnesio, Fosforo), uraren galerak egarraren gehipena sortarazten du (Polidipsia), eta hau ez da nahikoa galerak konpensatzeko, orduan deshidratazioa, hipotentsioa eta salbuezpenez shock-a.

Gertatzen den lipolisiak triglizeridoen eta gantz-azido libreen mailan gorakada bat sortuko du, hauek maila hepatikoan gorputz zetonikoen eraketa eragingo dute. Gorputz zetonikoak, azidoak direnak, odolean dagoen bikarbonatoarekin erreazionatzen dute, honen kontsumoa denbora laburrean egiten da, eta ondorioz azidozi metabolikoa sortzen da pH-aren eta bikarbonatoen (AB) gutxitze bategatik. Organismoa azidosi hau konpentsatzeko saiatzen da arnasketaren maiztasuna eta sakontasuna gehituz. Azidosia,

azkenik, beste fenomeno biren erantzulea da: zelulatik potasioen irteera (giltzurrunetik deusezten dena), eta goraken agerpenak; honek are gehiago laguntzen dio deshidratazioari.

Zetoazidosi diabetikoa duen paziente batek honako arazo hauek erakutsiko ditu ingresatzean: Hipergluzemia, Deshidratazioa, daitekeen Shock-a, galera elektrolitikoak, Azidosi Metabolikoa, eta Poliuria glukosuria eta zetonuriarekin.

Jarri behar zaien tratamendua ez da bakarrik bideratua egongo glukosa balio normalak lortzera baizik eta gainera hidratazioa, diuresia, azidosia eta galera elektrolitikoak normaldu beharko dira.

7.2.2.- Tratamendu eredua.

Tratamendu helburua intsulinaren administrazioa da, prozesu honen aldaketa fisiopatologikoen jatorria bait da, desoreka hidroelektrolitikoen zuzenketa, azidosiaren normalizazioa, eta zetonuriaren negatibizazioa, eta honekin batera honako hauek bezalako konplikazioak erakus ditzaten ekiditeko: hipogluzemia eta gurunetako edema.

PERFUNDITUKO DIREN LIKIDOEN BOLUMENA.

Zetoazidosi Diabetiko aldiunekoaren egoeran, ur eta elektrolitoen galerak 7.2 irudiak iragartzten dituenak izan daitezke.

ura.....	100 ml/kg.
Sodioa.....	7 tik 10 mEq/L. ra
potasioa....	5 tik 7 mEq/L. ra
Kloroa.....	5 tik 10 mEq/L. ra

7.2 irudia.- Galera teori-
koak.

Berhidratazio zuzena egiteko perfundituko diren likidoen beharritzenen kalkuloak aurreko galerak, ematen ari diren galerak, eta mantentzearenak kontutan izan beharko ditu (7.3 irudia).

- | | |
|---|--|
| 1) Mantenua..... | Eguneko 1.500 etik 1.800 ml/m ² ra. |
| 2) Aurretik gertatutako galeren berreskurapena... | Aurretik gertatutako galerak * pisu ezaguna edo teorikoa |
| 3) Egiten ari deneko galeren berreskurapena: | |
| - Polipnea dela eta..... | orduko 3 ml/kg. |
| - Poliuria dela eta..... | Diuresia > orduko 4 cc/Kg. |

7.3 irudia.- Perfundituko diren bolumenak.

Berhidratazio eredua astiro egin beharko da, osmolaritate plasmatikoa bapateko aldaketa ekiditeko, bi egunen zehar.

- Lehenengo eguna: Mantenimenduaren bolumena gehi aurreko galeren erdia. Ematen ari diren galeragatiko bolumena gehituz.
- Bigarren eguna : Lehenengoan bezala.

PERFUNDITUKO DIREN LIKIDOEN ERRITMOA ETA MOTAK.

Shock-a dagoela nabareraizten bada, zein hipobolemikoa izaten

Denbora	Mota	Erritmoa
0 - 1 ordu:	Suero fisiologikoa	20 ml/kg.
1 - 4 ordu:	ClNa % 0.45 ean	20 ml/kg.
4 - 6 ordu:	ClNa % 0.45 ean	(bolumen teoriko osoa lgo egunean / 2) - 40 ml/kg.
8 - 24 ordu:	ClNa % 0.3 ean	volumen teoriko total lgo egunean / 2 ml/kg.

7.4 irudia.- Likidoen mota eta erritmoak.

dena, alde aurreko tratamendu bereziaren protokoloa dago eta ondoren 7.4 irudiko ereduarekin jarraitzen da, Shock-a zuzentzeko administratutako bolumenak deskontatuz ezarritako likidoen kalkulutik (7.4 irudia)

GLUKOSA APORTAZIOA

Gluzemia plasmaticoari dagokionez 7.5 irudian seinalatutako apor-teak egiten dira.

Gluzemia	aportak
> 500 mg/dL	0
500 - 300 mg/dL	glukosa % 2.5 ean
300 - 200 mg/dL	glukosa % 5 ean
< 200 mg/dL	glukosa % 10 ean

7.5 irudia.- Glukosa apor-teak.

POTASIO APORTAZIOA.

Potasioa gehitzen da tratamenduan lehen ordua bukatutakoan, diuresi egonkorraren faltan ere bai, salbu eta: potasio plasmati-koa 5 mEq/L koa baino txikiagoa ez bada, edo hiperpotasemia seinu elektrikoak badaude. 7.6 irudiko kantitate eta kalitateak apor-tatuko dira.

Kantitate:	- 40 Meq/L lehenengo 6 orduetan
	- 30 Meq/L berhidratazioa bote arte, maila plasmatikoen kontrol pean eta E.C.G.
Kalitate:	- % 50, ClK % 15 ean (1 ml = 2.1 mEq) pean
	- % 50, fosfato bipotasiko (1 ml = 1.9 mEq) pean

7.6 irudia.- Potasioen kantitate eta kalitateak.

Hipopotasemiak protokolo berezia eskatzen du. Fosfatoaren administrazioak kalzemia zaintzea behar du.

AZIDOSIAREN ZUZENKETA.

Normalean ez da bikarbonatoa administratu behar, salbu eta: arteri pH-a 7.10 baino txikiagoa bada, eta/edo bikarbonato plasmatikoa 5 Meq/L baino handiagoa (bikarbonato 1 - 2 ml/kg. molar 1, 1 - 2 orduetan pasatzeko administratuz).

INTSULINOTERAPIA.

Intsulinaren aplikazioa egingo da, hasieran, preferenteki modu etengabea, perfisio ponpan, 0.1 - 0.2 UI/Kg/orduko hasierako dosi batekin.

Pazientearen ondorengo kontroletan, eta glukemia eta egoera metabolikoaren balioen arrazoiz, erritmoa aldaraziko da edo aldatuko da intsulina aplikaziora era etenkorra, azal azpitik (7.7 irudia).

■ Baldin eta intsulina bpc 4 ordu ondoren glukemia oz bada jaitxi 50 mg/dL-ko neurrian	→	orduko 0.2 U/kg-ra gehitu.
■ Baldin eta glukemia ≤ 300 mg/dL	→	orduko 0.05 edo 0.025 U/kg-ra gutxitu.
- Baldin eta g. zotoniko posit. glukemi ≤ 200 mg/dL-rokin	→	glukemia perfus. % 10-ora gehitu.
- Baldin eta g. zotoniko negatiboak	→	intsulinoterapia bpc kendu eta ETENKO-RRERA pasatu
		(0.2 - 0.5 U/kg/4-6 orduko azal az.).

7.7 irudia.- Kontsiderazioak intsulina etengabekoarekin.

Modu etenkorrean, maiztasuna eta beharrezko dosiak glukemia kurbaren portaerak eta perfil metabolikoak erregulattuko ditu.

Terapeutika honen aplikazioan zehar beste konplikazio kliniko batzu ager daitezke: Garaunetako Edema, Birikietako Edema, Aireztapen Hutsa, bigarren mailako Alterazio Ionikoak, Infekzio bererantzia; zeintzuk diagnostikatuak eta tratatuak izan beharko deten.

KONTROL ANALITIKOAK.

Glukemia 2 orduro, normalizatu arte.

Ioiak, gasometria, urea, calcioa, osmolaritatea; 4 orduro egonkortasunerarte.

Hematokrito, Hemoglobina, Proteina Osoak, eta urea; 12 orduro.

Glukosuria eta zetonuria; orduro normalizatu arte.

Fosforo eta magnesioa, 24 orduro.

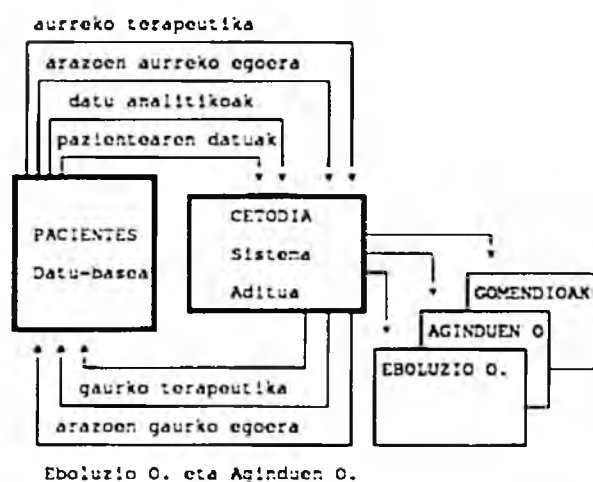
Eredu hau, Esta pauta, aldarazi ahal izango da pazientearen portaera metabolikoaren arabera.

7.3.- CETODIA-REN SOLUZIO INFORMATIKOA.

CETODIA MAMUR-en ezaugarritasunei osoki egokitzen zaien JOS bat da, bere ahalbide guztiak erabiltzera iritsi gabe erakuzten du MAMUR-en aplikagarritasuna sailkapen arazoetan [MORA92] haren konplexutasunei aurre eginez bere ezaugarriengatik: objektuetara orientazioa, zihurgabetasunaren tratamendua, modularitatea, eta deklaragarritasuna [PIKA92a] [PIKA92c].

Gai da paziente batek denbora une batean eskatzen duen tratamendua eskeintzeko eguneko egoeraren diagnostikoa eta haren eboluzioa ikusirik datu-base batetik hartzen diren aldagaietatik

abiatuta ateratakoak. Emaitezak: Eboluzio Orria, Aginduen Orria, eta sistemak iradokiko dituen gomendioak pantallaren bidez lortuko dira, eta aukeraz, impresoraren bidez, gainera datu-basea ere eguneratuz. Ospitaleko AIU-n (Arreta Intentsiboen Unitatea) kokatuta dagoen paziente bati dago-kion tratamendua zehazteko pazienteak jasaten dituen arazo edo konplikazio klinikoer ingresotik iragandako denbora-



7.8 Irudia.- CETODIA eta bere ingurunea.

ren arabera. Sistema emaitzak lortzeko gai izango da, informazio ez osoaren kasuan ere bai (aldagaien azken balio ezagunera joz); kasu askotan laborategiko analisiei itzaron behar zaielako, edo neurritzea zaila edo garestia delako. Adituak emaitzak aldarazi ahal ditu behin betiko emaitzak igorri baino lehenago eta pazienteen datu-basea eguneratu baino lehenago.

Honela seguratzeko da SA-ri egingo zaizkion urrengo kontsultak hasierako datu zuzen batzuetatik abiatuko direla; aldarazpen hau eman daiteke bai arrazonamendu egokiari jarraitu ez zaiola pentsatzeagatik bai espezifikatutako tratamendua behar zen zehaztasunez ebazpenaren ezintasunagatik. Pazienteen datuak datu-basean Erizaintzak sartuko ditu, aipatzen duten denbora ageraziz; denbora tarteak minutuetakoak izan daitezke tratamenduaren lehen etapetan, eta orduetakoa azken etapetan. Azken finean, CETODIA kontsulta sistema gisa aritzen da, erabiltzailearekiko ekarrekintza baxuarekin, tratamendu bat guztiz zehazten duena une horretan erabilgarri dagoen informazioarekin. Eboluzio Orriaz gain eta Aginduen Orriaz gain, pantallaren bidez honako hau erakusten du:

- Berehalako abisu batzu, prozesuan zehar antzematen doan egoeren arabera, zeinek haizutzen duten:
 - * Aurkitutako arazoa sailkatzea, litezkeen zergatiak iragarritz.
 - * Hartu beharko liratekeen ekintzei buruz adituari aholkuak ematea.
 - * Galdetzea, aldagairen baten balioa aldaraztea ere haizutuz
- gomendio batzu, kontsultaren azkenean, proposatutako tratamenduaren aplikazioan zehar jarraitzeko.

GMACS erabili dugu JB-a eraikitzeko, nagusiki kodeketa eta kasuen funtzio partikularrak errazteko, nahiz eta erabili zitezkeen edozein beste editore. Exekuziorako JB-aren bertsio konpilatua erabiliko da. Datu-basea eraikitzeko eta LISP-etik honetara sartzeko CLIPPER 5.1 erabili dugu.

Behin betiko tratamendua produktu komertzialei egokitu behar zaienez eta Erizaintza lan metodoei, sailkapen heuristikoko arrazonamendu metodoa erbili dugu (ikus 6.6 irudia).

Sistema honi buruzko lan luzea, hiru urtetakoa, Luis Aldamiz-Etxebarria-rekin batera MAMUR-en garapenerako ideia iturburu amaigabea izan da.

7.3.1.- CETODIA sistema adituaren justifikazioa.

Justifikatuta geratzen da SA baten eraikuntza Zetoazidosia-ren kontrol eta tratamendurako haxe kontutan badugu:

- adituen kopuru mugatua dagoela.
- aldiuneko konplikazioen eraginaren gutxitze batera gatozela,
- coma zetoazidotikoaren hilkortasunak garaia izaten jarraitzen duela (% 3 - 5), eta
- jakintza arlo mugatua dela.

Kasu honetan erraza da Zetoazidosi Diabetikoaren diagnostikoa, sintoma argiengatik, baina tratamendua bereziki delikatu da.

ZIHURGABETASUNAREN TRATAMENDUA

Zihugabetasunaren tratamendurako termino linguistiko berriak erabiltzen dira, MAMUR-en defektuz definituak.

Erabiltzen diren datuen gehiengoa analitikoa dela eta zihurgabetasunaren eragina tzikiagoa da; hala ere, denbora eta laborategiaren funtzionamenduarengatik, sarritan arazoaren eguneko egoera diagnostikatu beharra dago, haien eboluzioa ezagutu, eta

dagokien tratamendua zehaztu informazio ez osoarekin (aldagaiaren azken balio ezagunera joz).

Jakintzaren modularitatea eta deklaragarritasuna bai dominioarenak bai kontrolaren gainekoa, JB-aren erikuntza, mantentzea eta arazketa erraztu dute.

7.3.2.- Arkitektura.

MAILA KONTZEPTUALA.

Egintza guztiak bere objektu-atributu-balio-zihurtasun balioaren errepresentatzen dira. Objektuak klase bat aipa dezake edo klase horren instantzia izan daiteke, klase horrek geroago zerrendatuko den informazio mota bat dauka, dauden denbora bezain besteko instantziarekin (azken denboratik hasita), zeinei egokituko zaizkien gaizoaren datu eskuragarriak (lehenengo zazpientzako):

- 1) pazientearen datuak, instantzia bat.
- 2) datu analitikoak, gehienez hamar instasntzia.
- 3) arazo bakoitzaren egoera, instantzia bi.
- 4) arazo bakoitzaren eboluzioa, instantzia bi.
- 5) arazo bakoitzaren aldagai nagusia, instantzia bat.
- 6) arazo baten tratamenduaren notifikazioa, instantzia bat.
- 7) terapeutikak, instantzia bi.
- 8) honako hauen metodo partikularrak: arazo bakoitza, datuen sarrera, edo emaitzen ikuskatzea; klase desberdinetan.
- 9) erabiltzailearen funtzioak,
- 10) abisuak edo gomendioak, 36 instantzia.

383 egintzen gehiengo batetik abiatzen da, tratamenduaren unearen arabera. Gehienez 140 egintza atera daitezke, gaisoaren ezaugarrien arabera, era honetan banatuta:

- 16-k arazoaren egoerari dagokio.
- 16-k beraien eboluzioari.
- 36-k terapeutika aplikagarriei.
- 72 tarteko egintza era desberdinetan ateratakoak, erregela, metodo, galdera, eta abarren bidez.

Egintzen elkarkonexioa RETE zuhaitzean ezartzen du erlazio-natzen dituzten erregelen ezkerraldeen errepresentazioak.

MAILA ASOZIATIBOA.

Egintzak formato berdineko erregelen bidez erlazio-natzen dira, metaerregelak erabili ahal dituelarik bera-ren aplikagarritasuna zehazteko. Erregela bakoitza behin baino gehiagotan aktiba daiteke, baldin eta baldintzaren bat kumplitzen duen egintza berri bat sortzen bada, eta haren ezker eta eskuin

```

;; ..... METAERREGELAK .....
(defmeta-regla m-tin1 t-insulina ;
  ((terapeuticas-ant modo = continuo))
  ->
  ((E (poda-reglas-mencionen terapeuticas-ant modo discontinuo premisas))
   (E (poda-reglas-mencionen terapeuticas-ant modo desconocido premisas)))
  )
(defmeta-regla m-tin2 t-insulina ;
  ((terapeuticas-ant modo = discontinuo))
  ->
  ((E (poda-reglas-mencionen terapeuticas-ant modo continuo premisas))
   (E (poda-reglas-mencionen terapeuticas-ant modo desconocido premisas)))
  )
(defmeta-regla m-tin3 t-insulina ;
  ((glucemia diferencia-tiempo-modo-tiempo-actual < 4))
  ->
  ((E (poda-reglas-mencionen glucemia cambio-modo si conclusiones)))
  )

;; ..... ERREGELAK .....
(defregla tin1 t-insulina
  ((datos-act glucosa-p >= 300)
   (datos-act ph <= 7.25)
   (paciente-act peso = ?p)
   (terapeuticas-ant modo = desconocido))
  ->
  ((terapeuticas-act modo = continuo)
   (terapeuticas-act dosis-insulina-ord = 0.1) ; en u/kilo/hora
   (terapeuticas-act ritmo-insulina = (E (* ?p (* 0.1 5))))
   (terapeuticas-act tiempo-modo = (datos-act tiempo ?))
   (tratamiento glucemia = si))
  )
(defregla tin9 t-insulina
  ((datos-act tiempo > 0)
   (paciente-act peso = ?p)
   (terapeuticas-ant modo = continuo)
   (datos-act glucosa-p <= 300)
   (datos-act ph >= 7.30))
  ->
  ((terapeuticas-act modo = discontinuo)
   (terapeuticas-act dosis-insulina-ord = 0.25) ; en u/kilo
   (terapeuticas-act dosis-insulina-ord-total = (E (* 0.25 ?p)))
   (terapeuticas-act ritmo-insulina = 0))
  )

```

7.9 Irudia.- Metaerregelak eta erregelak zeinen gainean eragina duten.

alderdien edukien arabera 6.12 irudiak erakusten dituen erregela mota desberdinak idatz daitezke.

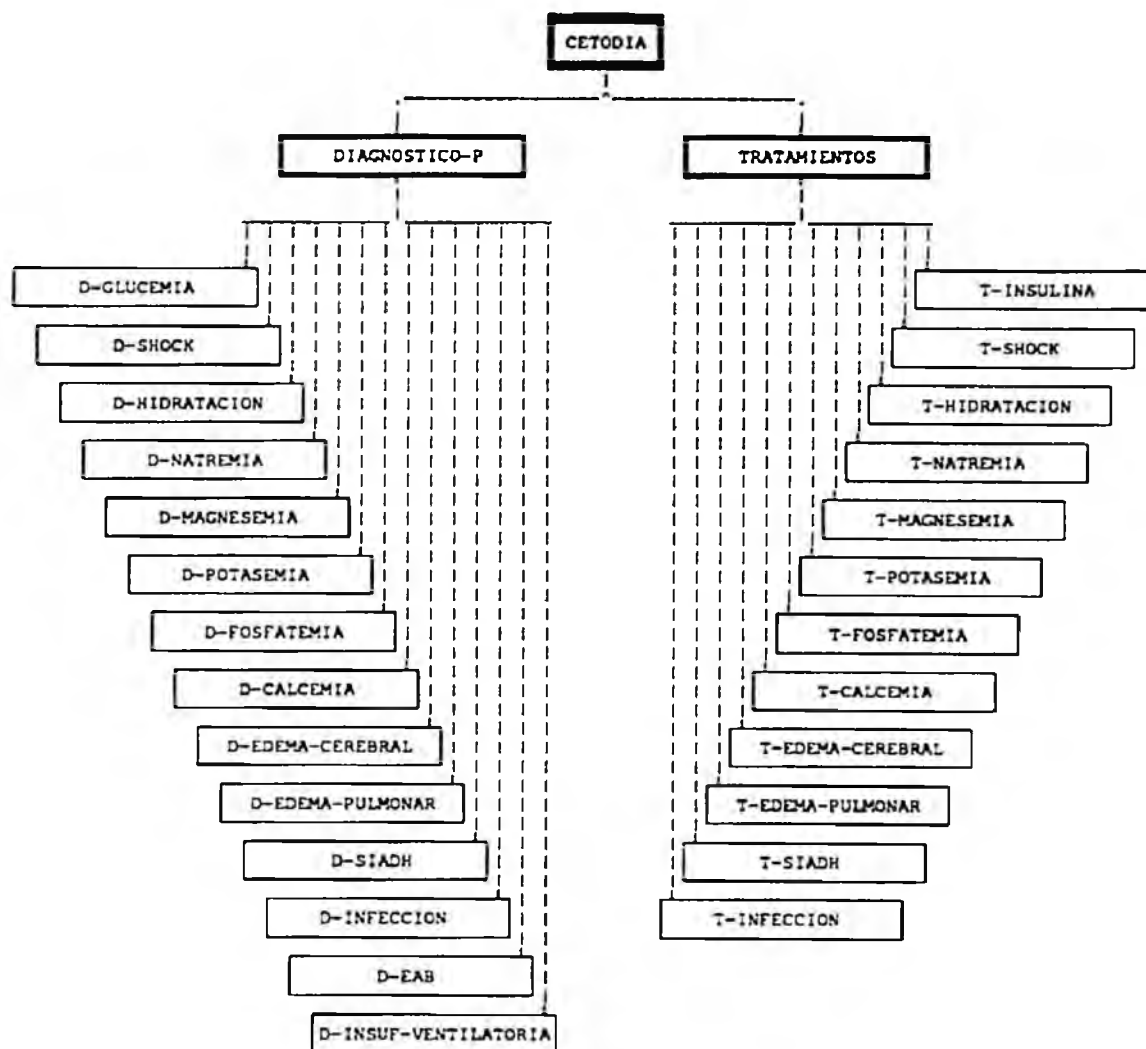
Zihurtasun baliorik espezifikatzen ez duenean "seguro" ulertzen da.

7.9 irudian "T-INSULINA" moduloan erabilitako metaerregelak erakusten dira (ikus 6.3.1 eta 6.3.5) eta erregeletariko bi, zeinen gainean aritzen den (ikus 3.2.2). Lehenengo biek "terapeuticas-ant" objektuaren "modo" atributua haren premisetan aipatzen duten erregelak inäusten dituzte, balio "desconocido", "continuo", edo "discontinuo" bada; kasuaren arabera. Hirugarrenak ondorioetan "si" balioarekin "glucemia" objektuaren "cambio-modo" atributoa aipatzen duten erregelak inäusten ditu. Honela, kasu bakoitzean intsulinoterapia egiten den moduari dagozkion erregelak baino ez dira aplikatuko (etengabea edo etenkorra) eta modu aldaketa 4 orduko estudiantuko da.

MAILA ESTRUKTURALA.

Erregela kopuru kontsideragarria dela eta, eta haren portaera eta interrelazioen ulermena errazteko, jakintzaren zatigarritasuna aprobetxatu da erregelak modulo txikietan eta maneiatzen erlatiboki errazak direnetan antolatzeke.

Moduloak hiru mailatan banatu ditugu (ikus 7.9 irudia), hauetan "CETODIA" moduloak kontsulta prozesu bakoitzean eman behar diren pausuak zehazten dituen ihardu keta plana dauka: Datuen hartzea, pazientearen egoeraren diagnostikoa, eta arazo bakoitzaren eboluzioa, tratamenduaren sorkuntza eta gomendioak, eta datu-basearen eguneratzea (ikus 6.11 irudia). Bigarren maila batean "DIAGNOSTICO-P" eta "TRATAMIENTOS" moduloak kokatzen dira, zeinek bete behar diren mugapenak dituzten, haren diagnostikorako eta tratamenduaren zehazpenerako aurkezten diren arazoen jakintza moduloetara sartzeko.



7.10 irudia.- CETODIA-ren moduloak.

Hirugarren maila batean diagnostikorako edo arazo bakoitzeko tratamendurako jakintza daukaten moduloak kokatzen dira.

"TRATAMIENTOS" moduloan tratamendua, aldez aurretik azpiko moduloetan ezarria, produktu komertzialen ezaugarriei eta Erizaintza lan metodoei egokitzen zaie, sailkapen heuristikoaren arrazonamendu metodoa segituz.

JB-a jerarkikoi banatutako 29 moduloetan dauden 310 erregelek osatzen dute. Gainera, 3 metaerregela ditu "T-INSULINA" moduloarentzat dagokion metaerregelen moduloan (ikus 6.3.5).

```

(defregla cet2 cetodia
  ((datos-entrada recibidos = s1))
  =>
  ((E (adelante diagnostico-p)))
  )

(defregla dia7 diagnostico-p
  ((datos-act osmo-p < 280)
   (datos-act sodio-p < 132)
   (datos-act urea-p < 15)
   (datos-act urea-p = 7x)
   (datos-act urea-p < 7x)
   (datos-act urea-o = 7y))
  =>
  ((E (atras d-siadh '(problemas-act siadh))))
  )

```

7.11 irudia.- Delak moduloetara.

Baina honen kontrol aldea aipatuz gero, zera seinalatu behar dugu (ikus 6.3.3):

- * Modulo bateko bilaketa mota deiaren unean finkatuko da.
- * Zihurtasun gutxienezko balioa, subsunpzioa, eta abar, era inplizitoan inplmentatzen da.

MAILA HIPOTETIKOA.

CETODIA-ak sortzen dituen estrategiek zehazten dituzte gaiso zetoazidotikoaren egoera diagnostikatzeko eta datu analitikoaren gaurko balioei eta eboluzioaren arabera dagokien tratamendua lortzeko beharrezkoa den jakintza erabiltzeko aplikatu behar diren dominioaren moduloak. Egintza ezagunek bete behar dituzte erregelen bidez errepresentatutako "DIAGNOSTICO-P" eta "TRATAMIENTOS"-etan dauden mugapenak. Mugapenetan oinarritutako kontrol egitura honek adituak eman behar dituen pausuak simulatzen ditu, gaisoaren egoera konkretu bati dagokion tratamendua zehazteko orduan, arrazonamenduan atzera jotzeren beharrianik gabe; pausu hauek hauexek dira:

- * Ingresoan intsulina faltak edo ondorengo intsulinoterapia desegokiak eragindako arazoen diagnostikoa; arazo bakoitzaren diagnostikorako jakintza modulo desberdin batean kokatuta dago.

- * Arazo bakoitzaren eboluzioaren zehaztapena aldeztatik sortutako tratamenduaren edo tratamenduen aplikazioaren ostean.
- * Tratamendu berriaren sorkuntza arazo bakoitzari dagokion moduloan dagoen jakintzarekin eta bere ondorengo egokitasuna produktu komertzialei eta Erizaintza lan metodoei.

Ateratako egintzak ondorengo edozeinetan erabil daitezke.

Modulo bakoitza porzesatuko da osorik deitua den bakoitzean.

MAILA HEURISTIKOA.

Maila honetan iharduketa plana era deklaragarrrian ezarri da, 6.11 irudian isladatzen den bezala, plan hau aurreranzko kateaketaz exekutatu da. Erregela bakoitza bere planaren eragile zereginetan eta behin bere aurrebaldintzak betez gero modulo baten aktibazioa sortarazi dezake, osorik prozesatuko dena beste eragile baten exekuzioaren saiketaren aurretik.

BATERAKO FUNTZIONAMENDUA.

6.3.4 eta 6.3.6-an iragsarritako prozesua jarraitzen du, honek "cet1" erregelaren aktibazioa suposatzen du haren lehenengo baldintzaren konplizteagatik eta bigarren baldintzaren atributoak metodo bat edukitzeagatik datuen kargarako datu-basetik aurrez adierazitako objektuetara. Datuak eduki bezain laster "CET2"-ek "DIAGNOSTICO-P" moduloaren jakintza prozesatuko du aurreranzko kateaketaz, haren jakintzak bilaketa espazioa murriztuko du kasuari aplikagarria zaion dominioaren (gaisoak aurkez ditzakeen arazoen besteko moduloetan bananduta eta atzeranzko kateaketaz prozesatuz) jakintza baino ez erabiltzeko. Behin arazoen egoera eta eboluzioa zehaztu direnean "CET3" erregela aktibatuko da, eta honek "TRATAMIENTOS" moduloa prozesatuko du, diagnostiko moduloa

bezalaxe, dagokion tratamendua zehazteko. Azken egoerara iritsiz gero "CET4" erregelak azken mezu bat emititzendu eta kontsultan zehar sortutako azken gomendio guztiak biltzen dira.

Emaitzak eduki ahal izango dira sarrera-irteerako interfacearen menuaren bidez, zeinetara sartu ahal izango den kontsulta bukatutakoan.

PLANAREN ERREGELAK

```
(defregla cet1 cetodia
  ((datos-entrada recibidos - no)
   (datos-entrada carga-hechos-iniciales - ?x))
  =>
  ((datos-entrada recibidos - si))
)
(defregla cet2 cetodia
  ((datos-entrada recibidos - si) )
  =>
  ((E (adelante diagnostico-p)))
)
(defregla cet3 cetodia
  ((problemas-act glucomia - ?g)
   (problemas-act hidratacion - ?h))
  =>
  ((E (adelante tratamientos)))
)
(defregla cet4 cetodia
  ((tratamiento glucomia - si)
   (tratamiento hidratacion - si))
  =>
  ((E (VER-AVISO mensaje-final))
   (resultados avisos-finales
    (E (enviar 'avisos
              'recop-avisos-f-emt)))))
)
```

7.12 irudia.- Jokaera plana.

GAURKO EGOERA

Behin JB-a idatzita eta konpilatuta dagoenez gero honen balidazio estadistikoa egiten ari gara Donostiako Arantzazuko Amaren Ospitaleak dituen historietatik abiatuz.

MAMUR-en gaurko bertsioan ez da ahalegin berezi bat egin LISP kode eraginkorrago bat lortzeko, ezta ere objektu-lengoai komertzial bat erabiltzen; hala ere kontsulta baterako minutu bateko erantzun-denbora gehienezkoa lortzen da (ordenagailu pertsonal batean 80486 mikroprozesadore batekin 33 MHZ-tara), ebatzi nahi den arazorako nahikoa dela uste dena.

7.3.3.- Sarrera-irteera interfacea.

MAMUR-en sarrera-irteera inteerfacea erabiliz, sistemaren erabilera erraz eta atsegina ahalbidera-tzen duten menuak definitu ditugu. Menuak pantaila batetan erakusten dira, une egokian erabili beharreko leihoen honako egitura honez:

- Pantaila osoa, 80 lerro eta 25 zutabe.
- Leiho nagusia, (2, 2) posiziotik (22, 78)-ra, aukera barra erakusteko eta dagozkion menu zabalduak.
- Lan leihoa, (5, 1) posiziotik (20, 80)-ra, honako hau erakusteko: abisuak, galderak, eta emaitzak.
- Errakuntza leihoa, (23, 1) posiziotik (24, 79)-ra okupatzen duena.

leiho bakoitzak aldarazi daitezkeen kolore batzuk ditu defektuz esleituta.

CETODIA-ren hasierako pantailaren ostean 7.13 irudiko menua agertzen da,

honetatik ahal izango da: DATu-ba-sera sartu hau kontsultatu edo eguneratzeko, edo SA-ri kontsulta edo lortutako emaitzen imprimaketa eskatu.

SISTEMA EXPERTO CETODIA			
Ayuda	Basa de Datos	Sistema Experto	Salir
	Pacientes Datos Problemas Terapéuticas	Consulta Imprimir	No Si
.....egoera lerroa.....			
mezuak galderak			

7.13 irudia.- Lehenengo menuari dagozkion aukera eta menu zabalgarriak.

"Consulta" aukera erabiltzen bada SA-ren prozesua hasiko da, zeinek abisuak erakutziko dituen lan leihoan eta galderak errakuntzen

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolución	Recomendaciones	Hoja Ordenes	Salir
	Ver Explicación	Ver Imprimir	Ver Explicación Modificar	No Si
			Salvar Salir	
.....egoera lerroa.....				
mezuak galderak				

7.14 irudia.- Lehenengo menuko "Consultar" aukerari dagozkion aukera eta menu zabalgarriak.

leihoan amaierako mezua arte, 7.14 irudiko menua agertuko den unean, eta honek haizutuko du: erakusketa, azalpena (7.15 eta 7.16 irudiak), kontzeptu aldarazgarrien aldarazketa (7.17 irudia), eta nahi izanez gero grabazioa; Eboluzio Orrian, Gomendioetan, eta Aginduen Orrian aurkeztutako emaitzena.

Kontzeptu baten aukeraketak hau aldarazteko siposatzen du errakuntza

leihoan gal-
dera baten
agerpena,
zeinek hai-
zutzen duen
gaurko balioa
erakustea eta
balio berri
bat sartzea.

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolución	Recomendaciones	Hoja Ordenes	Salir
	Ver Explicación	Problemas encont. Evolución prob. Eti. Edema Cereb. Eti. Edema Pul. Dosis Insul. Ord. Tot. aportes liq. Tot. perd. insen. Tot. Diuresis Balance hídrico Aporte teórico	Shock Glucemia Hídrat. Eab Natremia Magnesem. Potasemia Fosfatem.	Osmolarid. Calce mia Edema Car. Edema Pul. Insuf. ve. SIADH Coma Infección

7.15 Irudia.- Eboluzio Orriko kontzeptuak, zeinen azalpena eska daitekeen.

Bai honako hauen edukien erakusketak: orri biena, azken gomendioena,

edo hautatu-
tako kontzep-
tuaren azal-
penarena; lan
leihoaren
bidez egingo
da 7.21 iru-
dian ikusten
den bezala,
testua kutso-
rearen mugi-

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolución	Recomendaciones	Hoja Ordenes	Salir
		Dosis insul. ord. Dosis insul. NPH Antibiótico Dosis antibiótico Seroalbumina Fisiológico Ringer-lactato Glucosado - 50 % Glucosado - 10 % Glucosado - 5 % Glucosado 1/3	Glucosalino 1/5 ClNa ClK Fosfato bipotás. Periodo de tiempo Ritmo diuresis Ritmo periodo t. Ritmo perfusión Modo de perfusión Tipo ventilación Alimentación	Ver Explicación Modificar

7.16 Irudia.- Aginduen Orriko kontzeptuak, zeinen azalpena eska daitekeen.

mendu teklek in korritzeko aukera emanik irakuketa errazteko.

Erizain-tzak beste programa bat ere edukiko du, gaisoen datuak datu-basea sartze-ko, hauek gertatzen diren neurri-an.

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolución	Recomendaciones	Hoja Ordenes	Salir
Dosis insul. ord. Glucoselino 1/5 Dosis insul. NPH ClNa Antibiótico ClK Dosis antibiótico Fosfato bipotás. Seroalbumina Período de tiempo Fisiológico Ritmo diuresis Ringer-lactato Glucosado - 50 % Glucosado - 10 % Glucosado - 5 % Glucosado 1/3			Ver Explicación Modificar	

7.17 irudia.- Aginduen Orriko kontzeptu aldarazgarriak.

7.3.4.- Emaitzak.

Jarraian erakusten diren sarrera datuentzako: pazientearen datuak (7.18 irudia), eta datu analitikoak (7.19 irudia). CETODIA-ren portaera hauxe izango da:

DATOS DE PACIENTES	
Num. Historia:	237022 Fecha de Ingreso: 27-04-92
Apellido1:	LARRAZABAL
Apellido2:	UGARRIZA
Nombre:	UNAI
Edad:	7
Peso:	23
Talla:	120
Servicio:	PEDIATRIA
Cama:	126

7.18 irudia.- Gaisoaren datuak.

DATOS ANALITICOS			
N.Historia: 237022		TIEMPO: 0.00	UNAI LARRAZABAL UGARRIZA
Constantes Clinicas.		Valores Sanguineos.	
FC	100	Hematologia:	
FR	14	HID	40.0
TEMPERATURA	35.8	HB	14.0
TA / TS	135	Bioquimicos I.	
I TD	100	GLUCOSA	632
DIURESIS	0	UREA	88
TAM	111.6	PT	6.8
Valores Urinarios:		OSMOLARIDAD	328
GLUCOSURIA	4	SODIO	131
GETONURIA	4	POTASIO	3.7
SODIO	0	CLORO	101
POTASIO	0	CAP	15.7
UREA	0		
OSMOLARIDAD	0		
		Bioquimicos II.	
		FOSFORO	3.2
		MAGNESIO	0.0
		CALCIO	10.8
		Gasometria:	
		Ph	7.36
		PCO2	33.0
		EB	- 6.10
		AB	18.00
		PaO2	42.0

7.19 irudia.- Datu analitikoak.

- Adituaren iradokizunen erakusketa kontsulta prozesuan sortuak (7.20 irudia).

SISTEMA EXPERTO CETODIA			
Ayuda	Base de Datos	Sistema Experto	Salir
	Pacientes Datos Problemas Terapéuticas	Consulta Imprimir	No SI
-> Se ha logrado determinar el tratamiento correspondiente a este caso. Utilice las opciones del menu ; para verlo, modificarlo, o salvarlo.			
Consulta al S.E. para obtener los tratamientos			
Pulse cualquier tecla para continuar ..			

7.20 irudia.- Abisua SA kontsultatzen don bitartean.

- Behin prozesua bukatuz gero, adituak erakutsi ahal izango du:
 - * Eboluzio Orria (7.21 irudia),

- * Aginduen Orria (7.22 irudia), eta
- * Azken gomendioak (7.23 irudia).

Honez gain, imprimagailuaren bidez eduki ditzake. Lehenengo biak ospitalean erabiltzen diran paper aurreimprimatuan imprimatuko dira.

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolucion	Recomendaciones	Hoja Ordenes	Salir
H. EVOLUCION PROBLEMA	Historia: 237022 VALOR	Tiempo : 0 VALOR DE CERTEZA	27/4/1992 11:30 EVOLUCION	
HIPERGLUCEMIA	Glucosa p.: 632	SEGURO		
DESHIDRATACION	Osmo. pla.: 328	SEGURO		
EAB	Ph : 7.36	SEGURO		
HIPONATREMIA	Sodio pla.: 131	SEGURO		
HIPOFOSFATEMIA	Fosfato p.: 3.2	SEGURO		
OSMOLARIDAD	Osmo. pla.: 328	SEGURO		
COMA	Esc. coma : 10	SEGURO		
Dosis insulina ordinaria : 2.3 U.I./Kg/hora.				
Av.Pag.- Adelante Re.Pag.- Atrás Teclas cursor ESC - Salir				

7.21 Irudia.- Eboluzio Orriaren erakuspena.

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolucion	Recomendaciones	Hoja Ordenes	Salir
H. ORDENES	Historia: 237022	Tiempo : 0	27/4/1992 11:32	
Dosis insulina ordinaria : 0.1 U.I./Kg.				
Bomba de insulina : 11.5 cc./h.				
Fisiológico 460 cc., durante 2 h. a un ritmo de 230 cc/h.				
Control de glucemia cada hora y constantes habituales cada hora				
Control de diuresis y labstix en cada micción.				
Av.Pag.- Adelante Re.Pag.- Atrás Teclas cursor ESC - Salir				

7.22 Irudia.- Aginduen Orriaren erakuspena.

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolución	Recomendaciones	Hoja Ordenes	Salir
RECOMENDACIONES		Historia: 237022	Tiempo : 0 27/4/1992 11:34	
-> VALORES DE FOSFORO EN PLASMA ALTERADOS SE RECOMIENDA NUEVO CONTROL EN LAS SIGUIENTES 24 HORAS				
Av.Pag.- Adelanto Re.Pag.- Atrás Teclas cursor ESC - Salir				

7.23 Irudia.- Azken gomendioen erakuspena.

SISTEMA EXPERTO CETODIA				
Ayuda	Hoja Evolución	Recomendaciones	Hoja Ordenes	Salir
Explicación objetivo: (terapeuticas-act fisiologico)				
datos-act	tiempo	(0)	seguro	conocido
datos-act	tem	(111.6)	seguro	conocido
shock	llenado-capilar	(2)	seguro	preguntado
shock	frialdad-extremidades	(s1)	seguro	preguntado
shock	cianosis	(no)	seguro	preguntado
datos-act	fc	(100)	seguro	conocido
datos-act	diuresis	(0)	seguro	conocido
shock	posibilidad-shock	(4)	seguro	dsh6
problemas-act	shock	(no)	seguro	dsh11
datos-act	ph	(7.36)	seguro	conocido
hidratacion	cant-fisiologico-ing	(460)	seguro	método
terapeuticas-act	fisiologico	(460)	seguro	th118
Av.Pag.- Adelanto Re.Pag.- Atrás Teclas cursor ESC - Salir				

7.24 Irudia.- Lortutako helburuaren (terapeuticas-act fisiologico) azalpena.

8. KAPITULUA: ONDORIOAK ETA ETORKIZUNERAKO JARRAIPIDEAK

8.1.- ONDORIOAK.

Azken bost kapituluetan MAMUR-en ezaugarriak esplotatu ditugu eta tresna honen aplikazioa CETODIA bezalako SA-en garapenerako, zeinek soluzio informatikoa ematen bait dio arazo mediku delikatu bati. Nahiz eta garatutako aplikazioak arazo errealak sortzen dituzten arazo teknikoak kontutan izanik ere, ez ditu MAMUREn ahalbide guztiak uztiatu; alderdi original batzuen erabilgarritasuna erakutsi du sailkapen arazoen ebazpean orientatutako arkitektura multimailaren kontestuan eta jarraian nabarmentzen ditugunak:

JB-aren eraikuntza eta mantenimendua errazten duen arkitektura modularra. Edozein motatako jakintza era sinple eta iniforme baten errepresentatzea lortzen da. Erregela guztiak, zein modulo edo jakintza motakoa den kontutan hartu gabe, formato berbera daukate eta interpretatzaile berberak interpretatuko ditu. Honek jakintza ingeneriaren lana errazten du, jakintza adituen pentsatzeko era berdinean errepresentatu ahal duelako, JB-a adituentzat ulergarria izan dedin egiten duelarik. Jakintza mota desberdinak formato berdin berdineko erregelen bidez errepresentatu daitezkeela eta era deklaragarrian iragarriko dela modulo bakoitza erabiliko den eran oinarritzen da. Barnetik erregela bakoitzak zeregin desberdina jokatu du zein modulok dituen arabera eta zein jakintza mota osatzen duten arabera, eragile gisa, mugapen gisa, edo dominioaren jakintzaren erregela sinpleen

gisa jokatzuz. Mugapenen Betetzean oinarritutako arrazonamendua erabiltzeak modulo bakoitzaren exekuzio osoarekin haien jakintza osoa erabiltzen guztietan, kontrol zikloa sinpletu egiten du. Planifikazio jerarkikoak JB-aren beheranzko diseinua errazten du. OOP-k ere JB-aren argitasuna eta egituraketari laguntzen dio.

Arkitektura multimailak dominioaren elementuak kontrolare-nengandik bereiztea haizutzen du. Lengoai deklaragarria erabiltzen da portaera deskribatzeko, hau arkitektura mota honetako ideia zentrala da. Mugapen maila desberdinak ere espezifika daitezke jakintza hipotetikodun moduloetan, eta honek bilaketa espazioa gehiago mugatzea haizutuko du une bakoitzean beharrezko jakintza hobeto fokatu ahal izateko. Jakintza teoriko eta enpirikoen errepresentazioa haizutzen du, espresibitate eta konputagarritasunaren arteko balatze onargarri batekin, jakintza mota bietarako lengoai bera erabiliz eta era deklarati boan. Moduloen jerarkian modulo baten kokapenak daukan jakintza mota zehazten du. Modularizazioa ezinbestekoa da aplikazio handien garapenerako.

RETE algoritmoaren inplementazio hobetu bat egin da. Iragazketak errendimenduan duen garrantzia dela eta algoritmo hau inplementatu dugu sistemaren beste ezaugarriari mezedo egin diezaion eran. Mugapenen Betetzeak konputazionalki eraginkorrak diren iragazketa teknikak behar ditu (2.4.3) RETE algoritmoa bezalakoak.

Erregelen erabilerak mugapen gisa RETE zuhaitza mugapen sare bihurtzen du. Seigarren kapituluari ikusi den bezala zuhaitza itxurako mugapen sare batek jarraitu behar den estrategiaren zehastapena **arazo erraz** (Easy Problem) bihurtzen du, atzera jotzerik gabe egingo bait da. Honela soluziora aldagaien kopuruari dagokionez denbora linealean iritsiko da, eraginkortasun konputazionala inkrementatzen delarik. Gainera, arrazonamendu monotonikoa erabiliz funtsezkotasun lokala funtsezkotasun orokorrarekin bat etorriko da, zein kasu honetan bermatuta geratuko den algoritmo beraren funtzionamenduarengatik, egintza

berri bakoitzeko zuhaitza eguneratu egiten duena. Azken batean eraginkortasuna hobetu egiten da jakintza deklaragarrien bidez, era ahalik eta sinpleenean.

JB-aren egitura modularra zuzenean errepresentatuta dago RETE zuhaitzan. Azpizuhaitz bat dago modulo bakoitzeko zuhaitz orokorraren barruan beharrianik gabe inolako nodorik errepikatzeko ez delarik. Azpizuhaitz bakoitzak aukera emango du modulo baten erregeletara bakarrik sartzeko baina RETE zuhaitz osoa eguneratuz egintza berri bakoitzeko, zuhaitz orokorraren bidez. Informazioaren eskutatzerik ez daukaneko eragozpena du, modulo batean ateratzen den guztia edozein bestetik erabilgarria da.

Zihurgabetasuna adierazten duten termino linguistikoak adituak aukera ditzake. Bai kopurua bai terminoak eurak. Aplika diezaizkioke hala premisei eta ondorioei nola erregelei euri.

8.2.- ETORKIZUNERAKO BIDEAK.

Aurretik esandako guztiaz eta CETODIA-ren garapenean hartutako esperientzia guztiaz honako alderdi hauetan sakontzeko beharrezkoa ikusten da:

JB-aren balioespenerako tekniken eransketa. Funtzionamendu zuzena lortzeko helburuarekin edozein sistemetan balioespena beharrezkoa da. JOS-en kasuan egitura egokia bada eta portaera aproposa bada egiaztatu beharrezkoa izango da, are gehiago JB-aren tamainua zenbat eta handiago egiten denean. MAMUR bezalako sistema batek V. Canivell-ek [CANI91] garatutako bezalako balioespen teknikek edukitzea oso interesgarria litzateke.

Kasuetan oinarritutako ikaskuntza teknikak. Mediku historial kantitate estimagarria eduki zitekeenez, zeinetatik atera

zitekeen ez bakarrik dominioaren jakintza, baizik eta kontrolari buruzkoa ere bai; jakintza berriak gehitu litzaizkioke JB-ari sistemaren erabilkeran zehar aditu desberdinengandik etor litzakeena. Gehiketa honetan erabil behar den zihurtasunak prozesu honen are gehiago eragotziko du.

Denboran oinarritutako arrazonamendu teknikak. Tratamendurako sistema baten, ohizkoa da, denborak eragindako prozesu batzuk jarraitu behar izatea, CETODIA-ren kasuan ihardunpide sinple eta partikular bat erabili da kontestu edo munduen (viewpoints) sorkuntzaren bitartez hobatu zitakeena ART, KEE, KNOWLEDGE CRAFT, etabarren egiten den bezala; MAMUR-en ez dagoen ahalbidea. Etengabeko operazioa ere haizutzen duen soluzio ideial baterako tratamendurako eredu bat definitu behar litzateke eta 2.5.4 atalean seinalatutako ezaugarridun tresna bat eduki Denbora Errealeko JOS bat eraikitzeke. Arlo honetan oso lan gutxi egin da.

BIBLIOGRAFIA.

- [AIEL75] Aiello, M. and Weyrauch, R.W.
Checking proofs in the metamathematics of first order logic.
Proc. of the 4th IJCAI, pp. 1-8, 1975
- [ALEG89] Alegria, I.; Artola, X.; Legorburu, P.; Sarasola, K.; Barrenetxea, T. eta Sagarna, A.
Informatika oinarriko hiztegia.
Elhuyar Kultur Taldea, PK 361, 20080 - Donostia, 1989.
- [ALLE83a] Allen, James F.
Maintaining Knowledge about Temporal Intervals.
Communications of the ACM 26(11), 1983, pp. 832-843, David Waltz (Ed.).
- [ALLE83b] Allen, J. and Koomen, J.
Planning using a temporal world model.
Proc. of IJCAI-83, Karlsruhe, pp. 741-747.
- [BEY782] Beyth-Marom, R.
How probable is probable? A numerical taxonomy translation of verbal probability expressions.
Journal of Forecasting, no.1, pp. 257-269, 1982.
- [BITN75] Bitner, J.R. and Reingold, E.M.
Backtrack Programming Techniques.
Communications of the ACM, Vol.18, no.11, pp. 651-656, 1975.
- [BOBR77] Bobrow D., Winograd T.
An Overview of KRL, a Knowledge Representation Language.
Readings in Knowledge Representation, pag. 263. Pub: Morgan Kaufmann, Eds: Brachman & Levesque.
- [BONI79] Bonissone P.P.
The problem of linguistic approximation in system analysis.
University Microfilms International Publications 80-14,618, 1979.
- [BRAC83] Brachman Ronald J., Fikes Richard E., Levesque Hector J.
KRYPTON: A Functional Approach to Knowledge Representation.
Readings in Knowledge Representation, pag. 411, Brachman & Levesque (Eds), Morgan Kaufmann Pub.
- [BRAC85] Brachman R., Levesque H. (Eds)
Readings in Knowledge Representation.
Morgan Kaufmann Pub. 1985.
- [BRAT86] Bratko Ivan
PROLOG Programming for Artificial Intelligence.
International Computer Science Series Addison-Wesley Pub., 1986.
- [BREU85] Breuker, J. and Wielinga, B.
KADS: Structured knowledge acquisition for expert systems.
Proc. V International Workshop on Expert Systems and Their Applications, Agence de L'Informatique, May 1985, Avignon, France, pp. 887-900.
- [BROW85] Browston L., Farrell R., Kant E., Martin N.
Programming Expert Systems in OPS5: An Introduction to Rule-Based Programming.
Addison-Wesley, 1985.

- [BROW89] Brown, David C. and Chandrasekaran, B.
Design Problem Solving, Knowledge Structures and Control Strategies.
Research Notes in AI, Pitman, London, Morgan Kaufmann Pub., California.
- [BUCH84] Buchanan, Bruce G. Shortliffe, E.H. (Eds.)
Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project.
Addison - Wesley Pub. Comp. , 1984.
- [BUCH88] Buchanan Bruce G., Smith Reid G.
Fundamentals of Expert Systems.
Annual Reviews Computer Science, 3, pp. 23-58, 1988.
- [BYLA91] Bylander Tom.
A Simple Model of Knowledge Compilation.
IEEE Expert, 6, (2), April 1991, pp. 73-74.
- [CAN191] Canivell, Verónica.
Validación de Bases de Conocimiento en SBC para diagnóstico.
Tesis Doctoral, Universidad de Dousto, 1991.
- [CARB83] Carbonell, J. G.
Learning by analogy: Formulation and generating plan from past experience.
R. S. Michalski, J. G. Carbonell, T. M. Mitchell (Eds.), Machine Learning (Toga, Palo Alto, CA, 1983).
- [CLAN85] Clancey W. J.
Heuristic Classification.
Artificial Intelligence, 27, pp. 289-350, 1985.
- [COX 83] Cox Brad J.
Object-Oriented Programming on C.
Unix Review, October/November 1983.
- [COX 84] Cox Brad J.
Message/Object Programming: An Evolutionary Change in Programming Technology.
IEEE Software, 1, (1), 1984.
- [COX 87] Cox Brad J.
Object-Oriented Programming: An Evolutionary Approach.
Addison-Wesley Pub., 1987
- [CHAN91] Chandrasekaran B.
Models versus Rules, Deep versus Compiled, Content versus Form. Some Distinctions in Knowledge Systems Research.
IEEE Expert 6, (2), April 1991, pp. 75-79.
- [DAV176] Davis, Randall
Use of Meta-Level Knowledge in the Construction and Maintenance of Large Knowledge Bases.
Dissertation Abstracts International, vol.37, Nov. 1976, Order: 76-25,988, p. 266.
- [DAV180] Davis, R.
Meta-rules: reasoning about control.
Artif. Intell. (Netherlands), vol.15, no.3, Dec. 1980, p.179-222.
- [DAV182] Davis Randall
TEIRESIAS: Applications of Meta-level Knowledge.
Knowledge-based systems in Artificial Intelligence, Davis R., Lenat D. B. (Eds), McGraw-Hill, 1982.

- [DAVI84a] Davis, R.
Diagnostic Reasoning Based on Structure and Behavior.
Artificial Intelligence, 24, pp. 347-410, 1984.
- [DAVI84b] Davis R., Buchanan B. G.
Meta-level Knowledge.
Rule-based expert systems: The MYCIN experiments of the Stanford Heuristic Programming Project.
Buchanan and Shortliffe (eds). Addison-Wesley, Reading.
- [DAVI87] Davis, Ernest
Constraint Propagation with Interval Labels.
Artificial Intelligence, 32, pp. 281-331, 1987.
- [DAVI89] Davis R.
Form and Content in Model-Based Reasoning.
Proc. 1989 Workshop on Model-Based Reasoning, tech. report, Boeing Computer Services, Bellevue Wash. 1989. pp. 11-27.
- [DAVI90] Davis, J. S.
Effect of modularity on maintainability of rule-based systems.
International Journal Man-Machine Studies, 32, (4), 1990, pp. 439-447.
- [DECH85] Dechter, R. and Pearl, J.
The anatomy of easy problems: A constraint-satisfaction formulation.
Proc. of IJCAI-85, pp. 1066-1072, Los Angeles, CA, 1985.
- [DECH88] Dechter, Rina and Pearl, Judea
Network-Based Heuristics for Constraint-Satisfaction Problems.
Artificial Intelligence, 34, pp. 1-38, 1988.
- [DECH89] Dechter, Rina and Meiri, Itay
Experimental evaluation of preprocessing techniques in Constraint Satisfaction Problems.
Proc. of the IJCAI-89, Detroit, Michigan, 20 - 25 August 1989, pp. 271 - 277.
- [DIET91] Dietterich, Thomas G.
Bridging the Gap between Specification and Implementation.
IEEE Expert 6, (2), April 1991, pp. 80-82.
- [DOYL79] Doyle, J.
A Truth Maintenance System.
Artificial Intelligence 12, no.3, pp. 231-272, 1979.
- [DRUM89] Drummond, M. and Tate, A.
AI Planning: A Tutorial and Review.
Artificial Intelligence Applications Institute, University of Edinburgh, AIAI-TR-30, January 1989.
- [DUBO85a] Dubois, D. Prade, H.
Théorie des possibilités. Applications à la représentation des connaissances en informatique.
Masson, Paris, 1985.
- [FIKE70] Fikes, R.E.
REF-ARF: A System for Solving Problems Stated as Procedures.
Artificial Intelligence, 1, 1970, pp. 27-120.
- [FIKE72a] Fikes, R.E.; Hart, P.E. and Nilsson, N.J.
Learning and Executing Generalized Robot plans.
Artificial Intelligence, 3, 1972.

- [FIKE72b] Fikes, R.E.; Hart, P.E. and Nilsson, N.J.
Some New Directions in Robot Problem Solving.
Machine Intelligence, 7, Meltzer, B. and Michie, D. (Eds.), Edinburgh University Press, 1972.
- [FIKE85] Fikes, R. Kohler, T.
The role of Frame-Based Representation in Reasoning.
Special Section on Architectures for Knowledge-Based Systems, Communications of the ACM, Vol. 28, no. 9, pp. 904-920, 1985.
- [FORG82] Forgy C. L.
RETE: A fast algorithm for the many pattern/many object pattern match problem.
Artificial Intelligence, 19, (1), pp. 17-37, 1982.
- [FOX 87] Fox, Mark S.
Constraint-Directed Search: A case Study of Job-Shop Scheduling.
Research Notes in AI, Pitman, London, Morgan Kaufmann Pub., California.
- [FREU78] Freuder, E.C.
Synthesizing Constraint Expressions.
Communications of the ACM, 21, 1978, pp. 958-966.
- [FREU82] Freuder, E.C.
A sufficient condition of backtrack-free search.
Journal ACM, Vol.29, no.1, pp. 24-32, 1982.
- [FREU89] Freuder, E. C.
Partial Constraint Satisfaction.
Proc. of IJCAI-89, Detroit, Michigan, 20 - 25 August 1989, pp. 278 - 283.
- [FURB84] Furbach U., Dirlich G., Freksa C.
Towards a Theory of Knowledge Representation Systems.
AI Methodology, Systems, and Applications. Bibel W., Potkoff B. (Eds.), Nort-Holland, 1984.
- [GASC79] Gaschnig, J.
A Problem Similarity Approach to Devising Heuristics: First Results.
Proc. IJCAI-79, Tokyo, Japan, August 1979, pp. 301-307.
- [GENE83] Genesereth, M.R.
An overview of meta-level architecture.
Proc of the AAAI Conf., pp. 119-124, 1983.
- [GODO89] Godo L., López de Mantaras R., Sierra G., Vordaguor A.
MILORD: The Architecture and the Management of Linguistically Exposed Uncertainty.
International Journal of Intelligent Systems, 4, pp. 471-501, 1989, John Wiley & Sons, Inc.
- [GOEL91] Goel Ashok K.
Knowledge Compilation. A symposium.
IEEE Expert, 6, (2), April 1991, pp. 71-93.
- [GUPT89] Gupta A., Forgy C., Newell A.
High-Speed Implementations of Rule-Based Systems.
ACM Transactions on Computer Systems, 7, (2), pag. 119, 1989.
- [GUSG87] Gusgen, H. W.; Junker, V. and Voss, A.
Constraints in a Hybrid Knowledge Representation System.
Proc. of the IJCAI-87, Milan, 1987, pp. 30-33.
- [GUSG88] Gusgen, Hans-Werner and Hertzberg, Joachim
Some Fundamental Properties of Local Constraint Propagation.
Artificial Intelligence, 36, pp. 237-247, 1988.

- [GUSG89a] Gusgen, Hans Werner
A Universal Constraint Programming Language.
Proc. 11th IJCAI-89, pp. 60-65, Morgan Kaufmann, San Mateo, California, 1989.
- [GUSG89b] Gusgen, Hans Werner
CONSAT: A System for Constraint Satisfaction.
Research Notes in AI, Pitman, Morgan Kaufmann Pub., 1989.
- [HARA80] Haralick, R.M. and Elliott, G.L.
Increasing Tree Search Efficiency for Constraint Satisfaction Problems.
Artificial Intelligence 14, no.3, pp. 263-313, 1980.
- [HAYE79a] Hayes Patrick J.
The Logic of Frames.
Readings in Knowledge Representation, pag. 287, Brachman and Levesque (Eds.), Morgan Kaufmann Pub., 1979.
- [HAYE83] Hayes-Roth, F.; Waterman, D. A. and Lenat, D. B.
Building Expert Systems.
Addison-Wesley, 1983.
- [HAYE85b] Hayes-Roth Barbara.
A Blackboard Architecture for Control.
Artificial Intelligence, 26, pp. 251-321, 1985.
- [KELL89] Keller, R.M.
In Defense of Compilation: A Response to Davis' 'Form and Content in Model-Based Reasoning'.
Proc. AAAI-90, Workshop on Model-Based Reasoning, Tech. Report, Boeing Computer Services, Bellevue Wash., 1989, pp. 22-31.
- [KELL91] Keller, Richard M.
Applying Knowledge Compilation Techniques to Model-Based Reasoning.
IEEE Expert, 6, (2), April 1991, pp. 82-87.
- [KENG89] Kong, Naiping and Yun, David Y.Y.
A Planning/Scheduling Methodology for the Constrained Resource Problem.
Proc. 11th IJCAI-89, pp. 998-1003, Morgan Kaufmann, San Mateo, California, 1989.
- [KLEE84] de Kleer, J. and Brown, J.S.
A qualitative physics based on confluences.
Artificial Intelligence, Vol.24, no.1, pp. 7-83, 1984.
- [KONO88] Konolige Kurt
Reasoning by Introspection.
Meta-Level Architectures and Reflection, Maes P., Nardi D. (Eds.), North-Holland, 1988.
- [KORF85] Korf, R.E.
Macro-Operators: A Weak Method for Learning.
Artificial Intelligence, no.26, 1985, pp. 35-77.
- [KOWA79] Kowalski R.
Logic for Problem Solving.
North-Holland, 1979.
- [LAIR87] Lair, John E. , Rosenbloom, Paul S. and Newell, Allen
SOAR: An Architecture for General Intelligence.
Artificial Intelligence. Journal 1987, no.33, pp. 1-64.
- [LENAT82] Lenat, D.B.
AM: Discovery in Mathematics as Heuristic Search.
Knowledge-based systems in AI. R. Davis-D.B. Lenat, McGraw-Hill I.B.C., 1982.

- [LEVE86] Levesque, H.
Making Beliefs out of Computers.
Artificial Intelligence, 30, pp. 81-108, 1986.
- [LOPE90] López de Mántaras, R.
Approximated reasoning models.
Ellis Horwood, Chichester, 1990.
- [MACK77] Mackworth, A.K.
Consistency in networks of relations.
Artificial Intelligence, Vol.8, no.1, pp. 99-118, 1977.
- [MACK85] Mackworth, A.K. and Freuder, E.C.
The complexity of some polynomial network consistency algorithms for constraint satisfaction problems.
Artificial Intelligence, Vol.25, no.1, pp. 65-74, 1985.
- [MAES88] Maes P.
Issues in Computational Reflection.
Meta-Level Architectures and Reflection, Maes P., Nardi D. (Eds.), North-Holland, 1988.
- [MCCA69] McCarthy, J. and Hayes, P. J.
Some philosophical problems from the Standpoint of Artificial Intelligence.
Machine Intelligence, 4, pp. 463-502, 1969.
- [MCDE82] McDermott, J.
RI: A Rule-Based Configurer of Computer Systems.
Artificial Intelligence, 19, (1), 1982.
- [MENG91] Meng, Chao-Chiang and Sullivan, Michael
Logos: A Constraint-Directed Reasoning Shell for Operations Management.
IEEE Expert 6, no.1, pp. 20-28, February 1991.
- [MILL88] Millán, José del R. y Cortés, Ulises
El Arte de la Representación.
Universitat Politècnica de Catalunya, 1988.
- [MINS63] Minsky, Marvin
Steps towards Artificial Intelligence.
E. A. Feigenbaum and J. A. Feldman (Eds.), Computer and Thought, McGraw-Hill, New York, 1963.
- [MINS81] Minsky, Marvin
A Framework for Representing Knowledge.
Mind Design, pp. 95-128, edited by J. Haugeland, Cambridge, MA: The MIT Press, 1981.
- [MINT87] Minton, Steven and Carbonell, Jaime G.
Strategies for Learning Search Control Rules: An Explanation-based Approach.
IJCAI-87, pp. 228-235, 1987.
- [MIRA90] Miranker, Daniel P.
TREAT: A New and Efficient Match Algorithm for AI Production Systems.
Research Notes in Artificial Intelligence, Pitman Publishing, London, 1990.
- [MONT74] Montanari, U.
Networks of constraints: Fundamental properties and applications to picture processing.
Inf. Sci, Vol.7, pp. 95-132, 1974.
- [MOOR82] Moore, R.
The Role of Logic in Knowledge Representation and Commonsense Reasoning.
Proceedings AAAI-82, pp. 428-433, 1982.

- [MORA87] Del Moral, A.
Lenguajes de Primer Orden y Representación del Conocimiento.
Tesis Doctoral, Universidad de Deusto, 1987.
- [MORA92] Del Moral, Anselmo; Pikaza, Juan M.; Aldamiz-Etxebarria, Luis and Diaz, David
CETODIA: Expert system for the treatment of Diabetic Ketoacidosis using the tool "MAMUR".
Sixteenth Annual Symposium on Computer Applications in Medical Care (SCAMC), A Conference of
the AMIA, November 8-11, 1992, Baltimore, Maryland. (Bidalita)
- [MYLO80] Mylopoulos, J.
An Overview of Knowledge Representation.
Proceedings of the Workshop on Data Abstraction, Databases, and Conceptual Modelling.
Communications of the ACM. pp. 5-12, 1980.
- [PASC86] Pascoe, Geoffrey A.
Elements of Object-Oriented Programming.
BYTE Magazine, August 1986, pp. 15-20.
- [PEAR83] Pearl, J.
On the Discovery and Generation of Certain Heuristics.
AI Magazine Winter/Spring, 1983, pp. 22-23.
- [PIKA92a] Pikaza, Juan M.; Del Moral, Anselmo and Aldamiz-Etxebarria, Luis.
MAMUR: A multilevel modular architecture incorporating declarative control based on Constraint
Satisfaction.
Fifth International Symposium on Knowledge Engineering, Sevilla, October 1992. (Bidalita)
- [PIKA92b] Pikaza, Juan M. y Arrieta, Inmaculada
Adimen Artifizialeko hitzen hiztegia.
(Prestakuntza).
- [PIKA92c] Pikaza, Juan M.; Del Moral, Anselmo y Aldamiz-Etxebarria, Luis.
MAMUR: Un caso de representación declarativa del conocimiento sobre el control basada en la
Satisfacción de Restricciones.
Primer Congreso Nacional de Programación Declarativa. PRODE-92. Madrid, 28 - 30 de Septiembre
de 1992. (Bidalita)
- [PLA287] Plaza, E.
EAR: Sistema d'Ajut a l'Adquisició i Estructuració de Coneixements.
Tesis Doctoral, Facultad d'Informàtica, Barcelona, 1987.
- [PLA289a] Plaza, Enric
KAIRDS: A framework for specifying control and meta-control in deterministic declarative
systems.
Report de Recerca, Grial 89/2, CEAB, 1989.
- [PURD83] Purdom, P.W., Jr.
Search Rearrangement Backtracking and Polynomial Average Time.
Artificial Intelligence 21, pp. 117-133, 1983.
- [QUIN85] Quinlan, James
A Comparative Analysis of Computer Architectures for Production System Machines.
Hawaii International Conference on System Sciences, pp. 194-200, Honolulu, January 1986.
- [RICH91] Rich, E.
Artificial Intelligence.
Mc Graw-Hill, 1991.
- [RITC83] Ritchie, G. D. and Hanna, F. K.
Semantic Networks: A general definition and survey.
Information Technology Research and Development, October 1983, pp. 169 - 186.

- [ROBI65] Robinson, A. J.
A machine-oriented logic based on the resolution principle.
Journal of the ACM, 12, 1965, pp. 23 - 41.
- [ROBS81] Robson, David
Object-Oriented Software Systems.
BYTE, Vol.6, no.8, pp. 74-86, August 1981.
- [ROSE88] Rosenbloom, Paul , Laird, John and Newell, Allen
Meta-Levels in SOAR.
Meta-Level Architectures and Reflection, P. Maes and D. Nardi (Eds.), North Holland, 1988.
- [SACE73] Sacardoti, E.D.
Planning in a Hierarchy of Abstraction Spaces.
Proc. of IJCAI-73, Stanford, CA, 1973, pp. 412 - 422.
- [SACE74] Sacardoti, E.D.
Planning in a hierarchy of abstraction spaces.
Artificial Intelligence, Vol.5, no.2, pp. 115-135, 1974.
- [SACE75] Sacardoti, E.D.
The non-linear nature of plans.
Proc. of IJCAI-75, Tbilisi, Georgia, 1975, pp. 206 - 214.
- [SCH75] Schank, Roger C. and Abelson, Robert P.
Scripts, Plans and Knowledge.
Proc. IJCAI-75, Tbilisi, Georgia, 1975, pp. 151 - 157.
- [SHAR88] Sharma, D. D. and Sridharan, N. S.
Knowledge-Based Real-Time Control: A Parallel Processing Perspective.
Proc. AAAI-88, Saint Paul, Minnesota, 1988, pp. 665 -670.
- [SIER89] Sierra, C.A.
MILORD: Arquitectura Multi-Nivell per a Sistemes Experts en Classificació.
Tesis Doctoral, UPC, 1989.
- [SLOM71] Sloman, A.
Interactions between Philosophy and AI: the role of Intuition and non-Logical Reasoning in Intelligence.
Artificial Intelligence, 2, pp. 209 - 225, 1971.
- [SMIT82] Smith, Brian C.
Prologue to "Reflection and Semantics in Procedural Language".
Ph.D. dissertation and Tech. Report MIT/LCS/TR-272, M.I.T., Cambridge, MA, 1982.
- [STAL77] Stallman, R.M. and Sussman, G.J.
Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis.
Artificial Intelligence, 9, 1977, pp. 135-196.
- [STEE80] Steele, G.L.
The definition and Implementation of a Computer Programming Language Based on Constraints.
Tech. Report AI-TR-595, MIT, Cambridge, MA, 1980.
- [STEE85] Steele, L.
Constraints as Consultants.
Progress in Artificial Intelligence, Chichester: Ellis Horwood, New York: John Wiley & Sons, 1985, pp. 146-165.

- [STEE87] Steels, Luc
The Deepening of Expert Systems.
AICOM, vol.0, no.1, pp. 9-16, Aug. 1987.
- [STEF81a] Stefik, Mark
Planning with Constraints (MOLGEN: Part 1).
Artificial Intelligence, no.16, 1981, pp. 111-140.
- [STEF81b] Stefik, Mark
Planning and Meta-Planning (MOLGEN: Part 2).
Artificial Intelligence, no.16, 1981, pp. 141-170.
- [STIC85] Sticklen, Jon , Chandrasekaran, B. , Josephson, John R.
Control Issues in Classificatory Diagnosis.
Proceedings of IJCAI-85, pp. 300-306, 1985.
- [STIC87] Sticklen, John , Smith, Jack W. , Chandrasekaran, B. and Josephson, John R.
Modularity of Domain Knowledge.
International Journal of Expert Systems, Vol.1, no.1, pp. 1-15, 1987.
- [TONG91] Tong, Chris
The Nature and Significance of Knowledge Compilation.
IEEE Expert 6 , (2). April 1991.pp. 88-91.
- [VOYE87] Voyer, Robert
Moteurs de Systemes Experts.
Editions EYROLLES, Paris, 1987.
- [WALT72] Waltz, D.L.
Generating Semantic Descriptions from Drawings of Scenes with Shadows.
Technical Report AI-TR-271, MIT, Cambridge, MA, 1972.
- [WEYH80] Weyhrauch, R.
Prolegomena to a Theory Mechanized formal Reasoning.
Artificial Intelligence, Vol.13, no.1, 2, Amsterdam, North Holland.
- [WILK83] Wilkins, D.E.
Representation in a domain independent planner.
Proc. of IJCAI-83, pp. 733-740, 1983.
- [WILK84] Wilkins, D. E.
Domain independent planning: representation and plan generation.
Artificial Intelligence, 22, 1984, pp. 269-301.
- [WINO75] Winograd, Terry
Frame Representation and the Declarative/Procedural Controversy.
Representation and Understanding: Studies in Cognitive Science, pp. 185-210, Bobrow and Collins (Eds.), Academic Press, 1975.
- [WINO80] Winograd, T.
Extended Inference Modes in Reasoning by Computer Systems.
Artificial Intelligence, 13 , pp. 5-26, 1980.
- [WOOD83] Woods, William A.
What's Important about Knowledge Representation?
Special Issue on Knowledge Representation. IEEE Computer 16, no.10, pp. 22-27, October 1983.
- [ZADE75] Zadeh, L.A.
Fuzzy logic and approximate reasoning.
Synthese 30 , 1975 , pp. 407-428.

-
- [ZADE78] Zadeh, L.
PRUF: a Meaning Representation Language for Natural Languages.
International Journal of Man-Machine Studies, 10, pp. 395-460, 1978.
- [ZWE889] Zweben, Monte and Eskoy, Megan.
Constraint Satisfaction with Delayed Evaluation.
Proc. 11th IJCAI-89, pp. 875-880, Morgan Kaufmann, San Mateo, California, 1989.