



HAL
open science

Système expert d'aide au diagnostic et à la maintenance de grands entraînements électriques

Sigurd Ahrens

► **To cite this version:**

Sigurd Ahrens. Système expert d'aide au diagnostic et à la maintenance de grands entraînements électriques. Energie électrique. Institut National Polytechnique de Grenoble (INPG), 1993. Français. NNT: . tel-02543113

HAL Id: tel-02543113

<https://hal.science/tel-02543113>

Submitted on 15 Apr 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

T 986 prêt-à-lire



THESE

présentée par

Sigurd AHRENS

Dipl.-Ing. de la Technische Universität Braunschweig, R.F.A

pour obtenir le grade de DOCTEUR

de l'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

(Arrêté ministériel du 30 mars 1992)

(Spécialité génie électrique)

=====

**SYSTEME EXPERT D'AIDE
AU DIAGNOSTIC ET A LA MAINTENANCE DE GRANDS
ENTRAINEMENTS ELECTRIQUES**

=====

Date de soutenance : 19 Novembre 1993

Composition du Jury :

Monsieur	J.-C. SABONNADIERE	Président
Messieurs	G. CHAMPENOIS	Rapporteur
	A. NICOLAS	Rapporteur
	D. POUPART	
	J. BIGEON	

Thèse préparée au sein du Laboratoire d'Electrotechnique de Grenoble

THESE

présentée par

Sigurd AHRENS

Dipl.-Ing. de la Technische Universität Braunschweig, R.F.A

pour obtenir le grade de DOCTEUR

de l'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

(Arrêté ministériel du 30 mars 1992)

(Spécialité génie électrique)

=====

**SYSTEME EXPERT D'AIDE
AU DIAGNOSTIC ET A LA MAINTENANCE DE GRANDS
ENTRAINEMENTS ELECTRIQUES**

=====

Date de soutenance : 19 Novembre 1993

Composition du Jury :

Monsieur	J.-C. SABONNADIÈRE	Président
Messieurs	G. CHAMPENOIS	Rapporteur
	A. NICOLAS	Rapporteur
	D. POUPART	
	J. BIGEON	

Thèse préparée au sein du Laboratoire d'Electrotechnique de Grenoble

JAULENT	Patrick	ENSGI
JAUSSAUD	Pierre	ENSIEG
JOST	Rémy	ENSPG
JOUBERT	Jean-Claude	ENSPG
JOURDAIN	Geneviève	ENSIEG
KUENY	Jean-Louis	ENSHMG
LACHENAL	Dominique	UFR PGP
LACOUME	Jean-Louis	ENSIEG
LADET	Pierre	ENSIEG
LE NEST	Jean-François	UFR/PGP
LESIEUR	Marcel	ENSHMG
LESPINARD	Georges	ENSHMG
LIENARD	Joël	ENSIEG
LONGEQUEUE	Jean-Pierre	ENSPG
LORET	Benjamin	ENSHMG
LOUCHET	François	ENSEEG
LUCAZEAU	Guy	ENSEEG
LUX	Augustin	ENSIMAG
MASSE	Philippe	ENSPG
MASSELOT	Christian	ENSIEG
MAZARE	Guy	ENSIMAG
MICHEL	Gérard	ENSIMAG
MOHR	Roger	ENSIMAG
MOREAU	René	ENSIMAG
MORET	Roger	ENSIEG
MOSSIÈRE	Jacques	ENSIMAG
OBLED	Charles	ENSHMG
OZIL	Patrick	ENSEEG
PANANAKAKIS	Georges	ENSERG
PAULEAU	Yves	ENSEEG
PERRET	Robert	ENSIEG
PERRIER	Pascal	ENSERG
PIAU	Jean-Michel	ENSHMG
PIC	Etienne	ENSERG
PLATEAU	Brigitte	ENSIMAG
POUPOT	Christian	ENSERG
RAMEAU	Jean-Jacques	ENSEEG
REINISCH	Raymond	ENSPG
RENAUD	Maurice	UFR/PGP
RIMET	Roger	ENSERG
ROBERT	François	ENSIMAG
ROGNON	Jean-Pierre	ENSIEG
ROSSIGNOL	Michel	ENSPG
ROYE	Daniel	ENSIEG
SABONNADIÈRE	Jean-Claude	ENSIEG
SAGUET	Pierre	ENSERG
SAUCIER	Gabrièle	ENSIMAG
SCHLENKER	Claire	ENSPG
SCHLENKER	Michel	ENSPG
SILVY	Jacques	UFR/PGP
SOHM	Jean-Claude	ENSEEG
SOLER	Jean-Louis	ENSIMAG
SOUQUET	Jean-Louis	ENSEEG
TICKIEWITCH	Serge	ENSHMG
TROMPETTE	Philippe	ENSHMG
TRYSTRAM	Denis	ENSGI
VEILLON	Gérard	ENSIMAG
VERJUS	Jean-Pierre	ENSIMAG
VINCENT	Henri	ENSPG

PRESIDENT DE L'INSTITUT
Monsieur Maurice RENAUD



Année 1993

PROFESSEURS DES UNIVERSITES

BARIBAUD	Michel	ENSERG
BARRAUD	Alain	ENSIEG
BARTHELEMY	Alain	ENSHMG
BAUDELET	Bernard	ENSPG
BAUDIN	Gérard	UFR PGP
BEAUFILS	Jean-Pierre	ENSIEG/ILL
BOIS	Philippe	ENSHMG
BOUVIER	Gérard	ENSERG
BRINI	Jean	ENSERG
BRUNET	Yves	CUEFA
CAVAIGNAC	Jean-François	ENSPG
CHARTIER	Germain	ENSPG
CHENEVIER	Pierre	ENSERG
CHERUY	Arlette	ENSIEG
CHOVET	Alain	ENSERG
COGNET	Gérard	ENSGI
COLINET	Catherine	ENSEEG
COMMAULT	Christian	ENSIEG
CORNUT	Bruno	ENSIEG
COULOMB	Jean-Louis	ENSIEG
COUTRIS	Nicole	ENSPG
CROWLEY	James	ENSIMAG
DALARD	Francis	ENSEEG
DARVE	Félix	ENSHMG
DELLA DORA	Jean	ENSIMAG
DEPEY	Maurice	ENSERG
DEPORTES	Jacques	ENSPG
DEROO	Daniel	ENSEEG
DESRE	Pierre	ENSEEG
DIARD	Jean-Paul	ENSEEG
DOLMAZON	Jean-Marc	ENSERG
DURAND	Francis	ENSEEG
DURAND	Jean-Louis	ENSPG
FAUTRELLE	Yves	ENSHMG
FOGGIA	Albert	ENSIEG
FORAY	Pierre	ENSHMG
FOULARD	Claude	ENSIEG
GALERIE	Alain	ENSEEG
GANDINI	Alessandro	UFR/PGP
GAUBERT	Claude	ENSPG
GENTIL	Pierre	ENSERG
GENTIL	Sylviane	ENSIEG
GUERIN	Bernard	ENSERG
GUYOT	Pierre	ENSEEG
IVANES	Marcel	ENSIEG
JACQUET	Paul	ENSIMAG
JALLUT	Christian	ENSEEG
JANOT	Marie-Thérese	ENSERG



DIRECTEURS DE RECHERCHE CNRS

ABELLO	Louis
ALDEBERT	Pierre
ALEMANY	Antoine
ALLIBERT	Colette
ALLIBERT	Michel
ANSARA	Ibrahim
ARMAND	Michel
AUDIER	Marc
AUGOYARD	Jean-François
AVIGNON	Michel
BERNARD	Claude
BINDER	Gilbert
BLAISING	Jean-Jacques
BONNET	Roland
BORNARD	Guy
BOUCHERLE	Jean-Xavier
CAILLET	Marcel
CARRE	René
CHASSERY	Jean-Marc
CHATILLON	Christian
CIBERT	Joël
CLERMONT	Jean-Robert
COURTOIS	Bernard
CRIQUI	Patrick
CRISTOLOVEANU	Sorin
DAVID	René
DION	Jean-Michel
DOUSSIÈRE	Jacques
DRIOLE	Jean
DUCHET	Pierre
DUGARD	Luc
DURAND	Robert
ESCUDIER	Pierre
EUSTATHOPOULOS	Nicolas
FINON	Dominique
FRUCHARD	Robert
GARNIER	Marcel
GIROD	Jacques
GLANGEAUD	François
GUELIN	Pierre
HOPFINGER	Emil
JORRAND	Philippe
JOD	Jean-Charles
KAMARINOS	Georges
KLEITZ	Michel
KOFMAN	Walter
LACROIX	Claudine
LANDAU	Ioan
LAULHERE	Jean-Pierre
LEGRAND	Michel
LEJEUNE	Gérard
LEPROVOST	Christian
MADAR	Roland
MARTIN	Jean-Marie
MERMET	Jean

SITUATION PARTICULIERE
PROFESSEURS D'UNIVERSITE

DETACHEMENT

BLOCH Daniel
BONNET Guy
BRECHET Yves
CAILLERIE Denis
GREVEN H el ene
LATOMBE Jean-Claude
PIERRARD Jean-Marie

ENSPG
ENSPG
ENSEEG
ENSHMG
CUEFA
ENSIMAG
ENSHMG

PERSONNES AYANT OBTENU LE DIPLOME
D'HABILITATION A DIRIGER DES RECHERCHES

BALESTRA	Francis
BALME	Louis
BECKER	Monique
BIGEON	Jean
BINDER	Zdeneck
BOE	Louis-Jean
BRECHET	Yves
CADOZ	Claude
CANUDAS DE WIT	Carlos
CHAMPENOIS	Gérard
CHOLLET	Jean-Pierre
COEY	Jean-Pierre
CORNUEJOLS	Gerard
COURNIL	Michel
CRASTES DE PAULET	Michel
DALLERY	Yves
DESCOTES-GENON	Bernard
DUGARD	Luc
DURAND	Madeleine
FERRIEUX	Jean-Paul
FEUILLET	René
FORAY	Pierre
FREIN	Yannick
GAUTHIER	Jean-Paul
GHIBAUDO	Gérard
GUILLEMOT	Nadine
GUYOT	Alain
HAMAR	Sylviane
HAMAR	Roger
HORAUD	Patrice
JACQUET	Paul
LATOMBE	Claudine
LE HUY	Hoang
LE GORREC	Bernard
LOZANO-LEAL	Rogelio
MACOVSKI	Mihaïl
MAHEY	Philippe
METAIS	Olivier
MONMUSSON-PICQ	Georgette
MORY	Mathieu
MULLER	Jean
MULLER	Jean-Michel
NGUYEN TRONG	Bernadette
NIEZ	Jean-Jacques
PERRIER	Pascal
PLA	Fernand
RECHENMANN	François
ROGNON	Jean-Pierre
ROUGER	Jean
ROUX	Jean-Claude
SKOTNICKI	Tomasz
TCHUENT	Maurice
THOMAS	Olivier
VAHLAS	Constantin

MEUNIER	Gérard
MICHEL	Jean-Marie
NAYROLLES	Bernard
PASTUREL	Alain
PEUZIN	Jean-Claude
PHAM	Antoine
PIAU	Monique
PIQUE	Jean-Paul
POINSIGNON	Christiane
PREJEAN	Jean-Jacques
RENOUARD	Dominique
SENATEUR	Jean-Pierre
SIFAKIS	Joseph
SIMON	Jean-Paul
SUERY	Michel
TEODOSIU	Christian
VACHAUD	Georges
VAUCLIN	Michel
WACK	Bernard
YAVARI	Ali-Reza
YONNET	Jean-Paul

CREUTIN	Jean-Dominique
DAO	Trongtich
DARONDEAU	Philippe
DAVID	Bertrand
DE LA SEN	Manuel
DELACHAUME	Jean-Claude
DENAT	André
DESCHIZEAUX née CHERUY	Marie-Noëlle
DIJON	Jean
DOREMUS	Pierre
DUPEUX	Michel
EL ADHAM	Karim
EL OMAR	Fovaz
EL-HENNAWY	Adel
ETAY	Jacqueline
FABRE	Suzanne
FAURE-BONTE	Mireille
FAVIER	Denis
FAVIER	Jean-Jacques
FELIACHI	Movloud
FERYAL	Haj Hassan
FLANDRIN	Patrick
FOREST	Bernard
FORESTIER	Michel
FOSTER	Panayolis
FRANC	Jean-Pierre
GADELLE	Patrice
GARDAN	Yvon
GENIN	Jacques
GERVASON	Georges
GILORMINI	Pierre
GINOUX	Jean-Louis
GOUMIRI	Louis
GROC	Bernard
GROSJEAN	André
GUEDON	Jean-Yves
GUERIN	Jean-Claude
GUESSOUS	Anas
GUIBOUD-RIBAUD	Serge
HALBWACHS	Nicolas
HAMMOURI	Hassan
HEDEIROS SILIVEIR	Hamilton
HERAULT	Jeanny
HONER	Claude
HUECKEL	Tomasz
IGNAT	Michel
ILIADIS	Athanasios
JANIN	Gérard
JERRAYA	Ahmed Amine
JUTTEN	Christian
KAHIL	Hassan
KHUONGQUANG	Dong
KILLIS	Andreas
KONE	Ali
LABEAU	Michel
LACAZE	Alain
LACROIX	Jean-Claude
LANG	Jean-Claude
LATHUILLERE	Chantal
LATY	Pierre

PERSONNES AYANT OBTENU LE DIPLOME

DE DOCTEUR D'ETAT INPG

ABDEL-RAZEK	Adel
AKSAS	Haris
ALLA	Hassane
AMER	Ahmed
ANCELLE	Bernard
ANGENIEUX	Gilbert
ATMANI	Hamid
AYEDI	Hassine Feri
A.BADR	Osman
BACHIR	Aziz
BALANZAT	Emmanuel
BALTER	Roland
BARDEL	Robert
BARRAL	Gérard
BAUDON	Yves
BAUSSAND	Patrick
BEAUX	Jacques
BEGUINOT	Jean
BELLISSANT née FUNEZ	Marie-Claire
BELLON	Catherine
BEN RAIS	Abdejettah
BERGER-SABBATEL	Gilles
BERNACHE-ASSOLANT	Didier
BEROVAL	Abderrahmane
BERTHOD	Jacques
BILLARD	Dominique
BLANC épouse FOULETIER	Mireille
BOCHU	Bernard
BOJO	Gilles
BOKSENBAUM	Claude
BOLOPION	Alain
BONNARD	Bernard
BORRIONE	Dominique
BOUCHACOURT	Michel
BRION	Bernard
CAIRE	Jean-Pierre
CAMEL	Denis
CAPERAN	Philippe
CAPLAIN	Michel
CAPOLINO	Gérard
CASPI	Paul
CHAN-TUNG	Nam
CHASSANDE	Jean-Pierre
CHATAIN	Dominique
CHEHIKIAN	Alain
CHIRAMELLA	Yves
CHILO	Jean
CHUPIN	Jean-Claude
COLONNA	Jean-François
COMITI	Jacques
CORDET	Christian
COUDURIER	Lucien
COUTAZ	Jean-Louis

SAUBAT épouse MARCUS	Bernadette
SCHMITT	Jean-Hubert
SCHOELLKOPF	Jean-Pierre
SCHOLL	Michel
SCHOLL	Pierre-Claude
SCHOULER	Edmond
SCHWARTZ	Jean-Luc
SEGUIN	Jean
SIWY	Jacques
SKALLI	Abdellatif
SKALLI HOUSSEYNI	Abdelali
SOUCHON	Alain
SUETRY	Jean
TALLAJ	Nizar
TEDJAR	Farouk
TEDJINI	Smail
TEYSSANDIER	Francis
THEVENODFOSSE	Pascale
TMAR	Mohamed
TRIOILLIER	Michel
TUFFELIT	Denis
TZIRITAS	Georges
VALLIN	Didier
VELAZCO	Raoul
VERDILLON	André
VERMANDE	Alain
VIKTOROVITCH	Pierre
VITRANT	Guy
WEISS	François
YAZAMI	Rachid

LAUGIER	Christian
LE CADRE	Jean-Pierre
LE GARDEVR	René
LE THIESSE	Jean-Claude
LEMAIGNAN	Clement
LEMUET	Daniel
LEVEQUE	Jean-Luc
LONDICHE	Henry
L'HERITIER	Philippe
MAGNIN	Thierry
MAISON	François
MAMWI	Abdullah
MANTEL épouse SIEBERT	Elisabeth
MARCON	Guy
MARTINEZ	Francis
MARTIN-GARIN	Lionel
MASSE	Dominique
MAZER	Emmanuel
MERCKEL	Gérard
MEUNIER	Jean
MILI	Ali
MOALLA	Mohamed
MODE	Jean-Michel
MONLLOR	Christian
MONTELLA	Claude
MORET	Frédéric
MRAYATI	Mohammed
M'SAAD	Mohammed
M'SIRDI	Kouider Nace
NEPOMIASTCHY	Pierre
NGUYEN	Trong Khoi
NGUYEN-XUAN-DANG	Michel
ORANIER	Bernard
ORTEGA MARTINEZ	Roméo
PAIDASSI	Serge
PASSERONE	Alberto
PEGON	Pierre
PIJOLAT	Christophe
POGGI	Yves
POIGNET	Jean-Claude
PONS	Michel
POU	Tong Eck
RAFINEJAD	Paiviz
RAGAIE	Harie Fikri
RAHAL	Salah
RAMA SEABRA SANTOS	Fernando
RAVAINE	Denis
RAZBAN-HAGHIGHI	Tchanguiz
RAZZOUK	Micham
REGAZZONI	Gilles
RIQUET	Jean-Pierre
ROBACH	Chantal
ROBERT	Yves
ROGEZ	Jacques
ROHMER	Jean
ROUSSEL	Claude
SAAD	Abdallah
SAAD	Youcef
SABRY	Mohamed Nabi
SALON	Marie-Christine

HERAULT Jeanny
HICTER Pierre
JANIN Bernard
JOLY Jean René
JOSELEAU Jean Paul
KAHANE André
KAHANE Josette
KRAKOWIAK Sacha
LAJZEROWICZ Jeanine
LAJZEROWICZ Joseph
LAURENT Pierre Jean
LEBRETON Alain
DE LEIRIS Joël
LHOMME Jean
LOISEAUX Jean Marie
LONGEQUEUE Nicole
LUNA Domingo
MACHE Régis
MASCLE Georges
MAYNARD Roger
NEMOZ Alain
OMONT Alain
PELMONT Jean
PERRIER Guy
PIERRE Jean Louis
RENARD Michel
RICHARD Jean Marc
RIEDTMANN Christine
RINAUDO Marguerite
ROBERT Jean Bernard
ROSSI André
SAXOD Raymond
SENGEL Philippe
SERGERAERT Francis
SOUCHIER Bernard
STUTZ Pierre
TRILLING Laurent
VALLADE Marcel
VAN CUTSEM Bernard
VIALON Pierre
VIDAL Micheal

Physique
Chimie
Géographie
Mathématiques Pures
Biochimie
Physique
Physique
Mathématiques Appliquées
Physique
Physique
Mathématiques Appliquées
Mathématiques Appliquées
Biologie
Chimie
Sciences Nucléaires I.S.N.
Physique
Mathématiques Pures
Physiologie Végétale
Géologie
Physique du Solide
Physique
Astrophysique
Biochimie
Géophysique
Chimie Organique
Thermodynamique

Mathématiques
Chimie C.E.R.M.A.V.

Biologie
Biologie Animale
Biologie Animale
Mathématiques Pures
Biologie
Mécanique
Mathématiques Appliquées
Physique
Mathématiques Appliquées
Géologie



Président de l'Université :

M. NEMOZ Alain

ANNEE UNIVERSITAIRE 1990 - 1991

MEMBRES DU CORPS ENSEIGNANT DE SCIENCES ET DE GEOGRAPHIE

PROFESSEURS DE 1ERE CLASSE

ADIBA Michel	Informatique
ANTOINE Pierre	Géologie I.R.I.G.M.
ARVIEU Robert	Physique Nucléaire I.S.N.
AURIAULT Jean Louis	Mécanique
BARJA Jean René	Statistiques - Mathématiques Appliquées
BECKER Pierre	Physique
BEGUIN Claude	Chimie Organique
BELORISKY Elie	Physique
BENZAKEN Claude	Mathématiques Pures
BERARD Pierre	Mathématiques Pures
BERNARD Alain	Mathématiques Pures
BERTRANDIAS Françoise	Mathématiques Pures
BERTRANDIAS Jean Paul	Mathématiques Pures
BILLET Jean	Géographie
BLANCHI Jean Pierre	A.P.S.
BOEHLER Jean Paul	Mécanique
BOITET Christian	Informatique et Mathématiques Appliquées
BORNAREL Jean	
BRUANDET Jean François	Physique
CARLIER Georges	Biologie Végétale
CASTAING Bernard	Physique
CHARDON Michel	Géographie
CHIBON Pierre	Biologie Animale
COHEN ADDAD Jean Pierre	Physique
COLIN DE VERDIERE Yves	Mathématiques Pures
CYROT Michel	Physique du Solide
DEBELMAS Jacques	Géologie Générale
DEMAILLY Jean Pierre	Mathématiques Pures
DENEUVILLE Alain	Physique
DEPORTES Charles	Chimie Minérale
DOLIQUE Jean Michel	Physique des Plasmas
DOUCE Roland	Physiologie Végétale
DUCROS Pierre	Cristallographie
FINKE Gerde	Informatique
GAUTRON René	Chimie
GENIES Eugène	Chimie
GERMAIN Jean Pierre	Mécanique
GIDON Maurice	Géologie
GIGNOUX Claude	Sciences nucléaires
GILLARD Roland	Mathématiques
GUITTON Jacques	Chimie

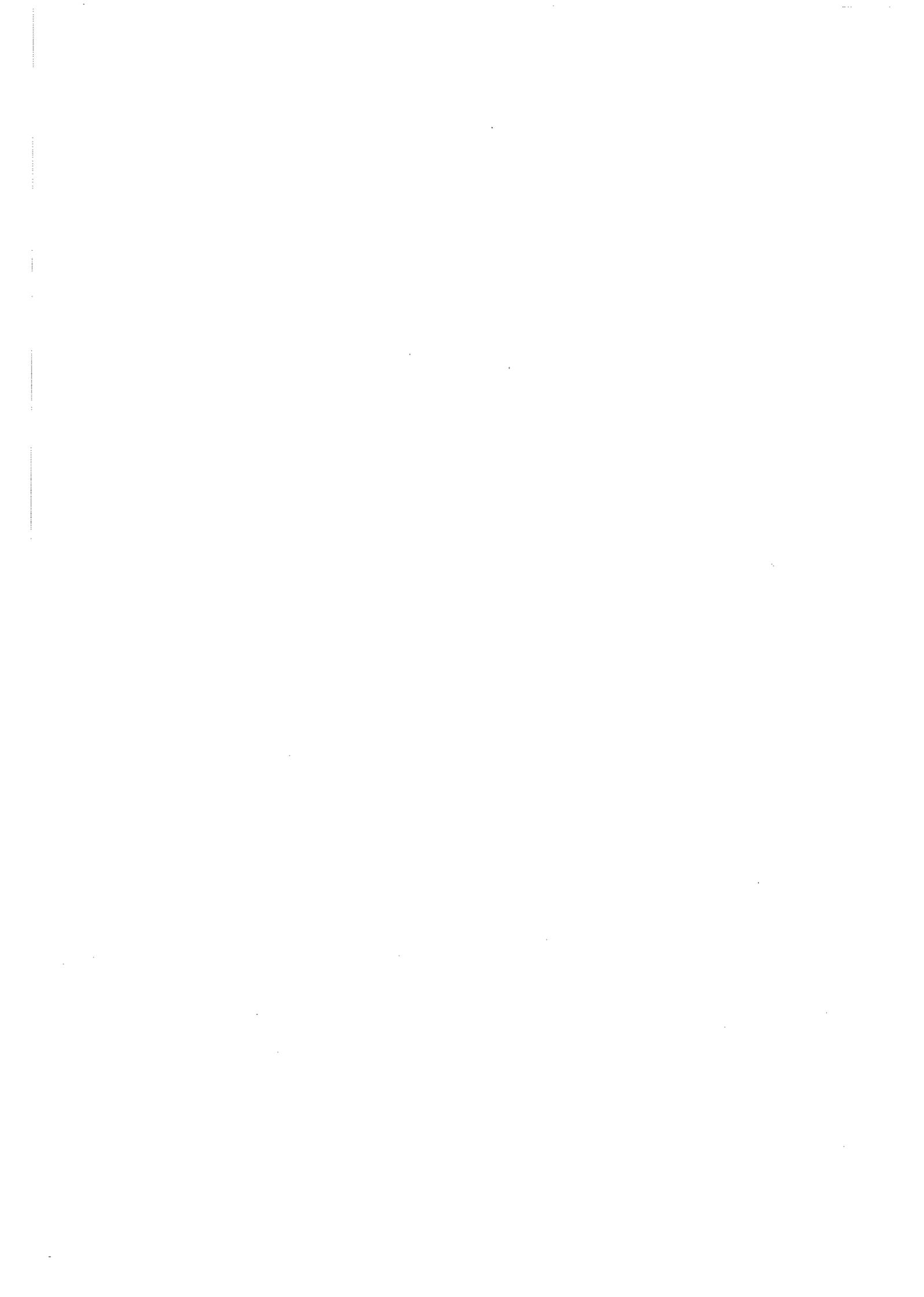
.. / ' ...

PAUTOU Guy
PECHER Arnaud
PELLETIER Guy
PERRIN Claude
PFISTER Claude
PIBOULE Michel
PORTESEIL Jean Louis
PUECH Laurent
RAYNAUD Hervé
REGNARD Jean René
ROBERT Claudine
ROBERT Danielle
ROBERT Gilles
SAJOT Gérard
SARROT REYNAULD Jean
SAYETAT Françoise
SERVE Denis
STOECKEL Frédéric
SCHOLL Pierre Claude
SUBRA Robert
TEMPERVILLE André
TISSUT Michel
TOURNIER Evelyne
VALLADE Marcel
VALLON Michel
VICAT Jean
VINCENS Maurice
VINCENT Gilbert
VIVIAN Robert
VOTTERO Philippe
WITOMSKI Patrick

Biologie
Géologie
Astrophysique
Sciences Nucléaires I.S.N.
Biologie
Géologie
Physique
Physique
Mathématiques Appliquées
Physique
Didactique des disciplines scientifiques
Chimie
Mathématiques Pures
Physique
Géologie
Physique
Chimie
Physique
Mathématiques Appliquées
Chimie
Mécanique
Biologie
Informatique et Mathématiques appliquées
Physique
Glaciologie
Physique
Chimie
Physique
Géographie
Chimie

PROFESSEURS DE 2EME CLASSE

APPARU Marcel	Chimie
ARMAND Gilbert	Géographie
ARNAUD Hubert	Géologie
ARTRU Marie Christine	Physique
ATTANE Pierre	Mécanique
BARATE Robert	Sciences Nucléaires
BARET Paul	Chimie
BARGE Jean	Mathématiques
BARLET Roger	Chimie
BERTIN José	Mathématiques
BLOCK Marc	Biologie
BLUM Jacques	Mathématiques Appliquées
BOITET Christian	Mathématiques Appliquées
BORRIONE Dominique	Automatique informatique
BOULON Marc	Mécanique
BOUTRON Claude	Glaciologie
BOUVET Jean	Biologie
BROSSARD Jean	Mathématiques
BRUGAL Gérard	Biologie
CAMPILLO Michel	Géophysique
CAVILLE Jean Yves	Chimie
CERFF Rudiger	Biologie
CHIARAMELLA Yves	Mathématiques Appliquées
CHOLLET Jean Pierre	Mécanique
COLOMBEAU Jean François	Mathématiques (ENSL)
COTTET Georges-Henri	Modélisation, calcul scientifique, statis.
COURT Jean	Chimie
CUNIN Pierre Yves	Informatique
DAVID Jean	Géographie
DEROUARD Jacques	Physique
DHOUILLY Danielle	Biologie
DUFRESNOY Alain	Mathématiques Pures
DUPUY Claude	Chimie
DURAND Mireille	Sciences Nucléaires
FONTECAVE Marc	Chimie
FOURNIER Jean Marc	Physique
GASPARD François	Physique
GIDON Maurice	Géologie
GIORNI Alain	Sciences Nucléaires
GONZALEZ SPRINBERG Gérardo	Mathématiques Pures
GOURC Jean Pierre	Mécanique
GUIGO Maryse	Géographie
GUMUCHIAN Hervé	Géographie
HACQUES Gérard	Mathématiques Appliquées
HAMMOU Abdelkader	Chimie
HERBIN Jacky	Géographie
HERINO Roland	Physique
HERZOG Michel	Biologie
JARDON Pierre	Chimie
JUTTEN Christian	Physique
KERCKHOVE Claude	Géologie
KOSAREW Siegmund	Math. fondamentales et appliquées
KLINGER Jurgen	Glaciologie
LAURENT Christine	Mathématiques
MANDARON Paul	Biologie
MARTINEZ Francis	Mathématiques Appliquées
MERCHEZ Fernand	Physique
MILAS Michel	Chimie
MOREL Alain	Géographie
MORIN Pierre	Physique
NGUYEN HUY Xuong	Informatique
OUDET Bruno	Mathématiques Appliquées



A mes parents
Annelore et Helmut

- * Madame Josiane Even pour avoir assuré avec beaucoup de patience la frappe de ce mémoire.
- * Mes amis Carsten-Sünke, Demba, Roland, Stephane, Volker, Wolfgang et tous les autres avec qui j'ai passé des moments agréables pendant ces trois années de recherche.
- * Tous les autres camarades de l'équipe IA et du laboratoire y compris les secrétaires et les techniciens de la plate-forme, notamment Marie-Thérèse Loubinoux et Bruno Mallet, qui ont tous contribué à cette ambiance magnifique et inoubliable qui règne au LEG.

Enfin, je remercie la Commission des Communautés Européennes qui a favorisé mon séjour en France en m'accordant une bourse dans le cadre du programme BRITE/EURAM.

Remerciements

Je tiens à exprimer mes remerciements à :

- * Monsieur Jean-Claude Sabonnadière, Professeur à l'INPG et Directeur du LEG, pour m'avoir encouragé à prendre ce sujet et pour l'honneur qu'il m'a fait en acceptant la présidence du jury.
- * Monsieur Alain Nicolas, Professeur à l'ECL et Directeur du CEGELY, pour l'intérêt qu'il a manifesté en m'honorant de sa présence dans le jury et en acceptant d'être l'un des rapporteurs.
- * Monsieur Gérard Champenois, Professeur à l'IUT de Poitiers, pour s'être intéressé à mon travail et avoir accepté d'être rapporteur de cette thèse.
- * Monsieur Daniel Poupart, Ingénieur de la société Jeumont-Industrie, qui m'a proposé ce travail, pour les discussions fructueuses que nous avons eues et pour l'honneur qu'il m'a fait d'être l'un des membres de mon jury de thèse.
- * Monsieur Jean Bigeon, Chargé de recherche du CNRS au LEG, pour les nombreux conseils et encouragements qu'il m'a prodigués au cours de ce travail, y compris dans les moments difficiles. Qu'il trouve ici l'expression de ma profonde gratitude.
- * Monsieur Robert Perret, Professeur à l'INPG et un des directeurs du LEG, pour l'accueil chaleureux au sein du LEG.
- * Tous les ingénieurs et techniciens de la société Jeumont-Industrie qui m'ont fait part de leur expérience et leur connaissance en diagnostic d'entraînements, notamment Messieurs Guilloux et Schurra. Sans leur support, ce travail n'aurait pas pu aboutir. Qu'ils soient remerciés chaleureusement de leur collaboration.
- * Florence François et Laurent Gerbaud pour leur précieux commentaires lors de la relecture des manuscrits de ce mémoire et pour leur amitié.



Sommaire

Chapitre II : Modélisation des connaissances de diagnostic dans le domaine des grands entraînements électriques	27
1 Introduction.....	27
2 L'expertise.....	27
2.1 Eléments de connaissance de diagnostic	28
2.2 La démarche du diagnostic	29
2.3 Analyse de la connaissance.....	31
3 Modélisation de la connaissance factuelle.....	32
3.1 Structuration.....	33
3.1.1 Décomposition fonctionnelle.....	33
3.1.2 Décomposition structurelle.....	34
3.1.3 Décomposition hybride.....	34
3.2 Définition des entités d'entraînements.....	36
3.2.1 Les composants	36
3.2.2 Les modules structurels	37
3.2.3 Les modules fonctionnels	39
3.2.4 Propriétés	39
3.3 Le modèle objet.....	41
3.4 Décomposition Objet de l'entraînement	43
4 Modélisation des connaissances déductives	45
4.1 Types de raisonnement.....	45
4.1.1 Raisonnement "par défaut".....	45
4.1.2 Raisonnement causal.....	45
4.2 Intégration des types de raisonnement	46
4.3 Conseils méthodologiques.....	47
4.3.1 Création d'hypothèses sur les causes.....	47
4.3.2 Création d'observations hypothétiques.....	49
4.3.3 Discussion	50
4.4 Structuration des connaissances déductives	51
4.4.1 Structuration selon le type de connaissance.....	51
4.4.2 Structuration selon le type de l'entité	52
5 La connaissance stratégique	53
5.1 Philosophie.....	54
5.2 Choix des règles à appliquer	54
5.3 Tri des hypothèses établies par les règles.....	56
6 Conclusion.....	57

Sommaire

Introduction.....	1
Chapitre I : Présentation des problèmes	3
1 Présentation de l'objet de l'étude.....	3
1.1 Les entraînements synchrones de grande puissance	3
1.2 Composants d'un entraînement synchrone	3
1.3 Le système de contrôle et de surveillance	4
1.4 Problème de diagnostic de grands entraînement électriques	5
2 Survol sur quelques méthodes de diagnostic	7
2.1 Diagnostic par redondance matérielle	7
2.1.1 Présentation	7
2.1.2 Redondances dans les entraînement électriques de grande puissance.....	8
2.1.3 Conclusion	9
2.2 Les méthodes de maintenance	10
2.2.1 Analyse des modes de défaillance, de leurs effets et de leur criticité (AMDEC)	10
2.2.2 La méthode des arbres de défaillances	12
2.2.3 Conclusion	14
2.3 Diagnostic par redondance analytique	15
2.3.1 Présentation	15
2.3.2 Discussion.....	16
2.3.3 Conclusion	17
2.4 Diagnostic par réseaux de neurones artificiels.....	17
2.4.1 Principe.....	17
2.4.2 Discussion.....	18
2.5 Les systèmes experts	18
2.5.1 Introduction	19
2.5.2 Structure d'un système expert	19
2.5.3 Les systèmes de maintien de vérité.....	23
2.5.4 Conclusion	24
2.6 Conclusion.....	25

Annexes

Annexe I :Diagnostic par l'approche système expert en génie électrique	1
1 Approches basées sur des règles de production.....	1
1.1 Approche avec stratégie guidée par les buts	1
1.2 Démarche par inventaire des défauts	4
1.3 Approche avec stratégie guidée par les données	7
2 Approches basées sur des représentations multiples de connaissances.....	11
2.1 Approche avec classification de scénarios de défauts	11
2.2 Approche avec représentation explicite de l'installation.....	17
2.3 Approches combinant des représentations de l'installation et de scénarios de défauts.....	21
3 Approches basées sur des modèles profonds.....	25
3.1 Principe de la méthode	26
3.2 Exemple d'illustration.....	27
3.3 Discussion.....	29
3.4 Commentaires.....	30
3.5 Conclusion	30
4 Conclusion.....	31
Annexe II: Outils d'enregistrement et de compression de données pour synthétisation d'une trace d'exploitation	33
1 Objectif.....	33
2 Démarche	33
3 Principe	34
4 Mise en œuvre.....	35
Annexe III : Détails sur la mise en œuvre de l'interface graphique.....	39
1 Types d'images	39
1.1 Les images synthétiques	39
1.2 Les images détaillées	39
2 Couplage entre images et objets d'une installation.....	40
2.1 Les activités	40
2.2 Un exemple.....	41
Annexe IV : Index des figures	43
Annexe V : Références bibliographiques.....	45
Annexe V : Références bibliographiques.....	53

Chapitre III : Implémentation informatique du prototype	59
1 L'architecture blackboard	59
1.1 Présentation.....	60
1.2 Discussion	61
2 L'outil choisi.....	61
3 Modélisation des données	
Représentation objet.....	62
4 Modélisation des sources de connaissance.....	68
5 Modélisation du contrôle du raisonnement.....	70
5.1 Le choix des règles à appliquer.....	70
5.1.1 Recensement des évènements.....	71
5.1.2 Sélection et application des sources de connaissances à appliquer.....	71
5.1.3 Discussion.....	74
5.2 Tri des hypothèses.....	74
5.3 Déroulement	76
6 L'interface graphique	78
6.1 Objectifs.....	78
6.2 Génération automatique de l'interface	79
6.3 Présentation.....	79
6.3.1 Les synoptiques hiérarchisées	79
6.3.2 Visualisation de l'état de l'installation.....	80
6.4 Mise en œuvre.....	81
7 Conclusion.....	81
 Chapitre IV : Exemple d'une session avec le prototype du système expert	 83
1 Une session utilisateur.....	84
1.1 Chargement des connaissances.....	84
1.2 Lancement du raisonnement	87
1.3 Simulation de défauts.....	88
2 Une session concepteur	92
2.1 Adaptation du système expert à un équipement particulier.....	93
2.2 Création d'un nouveau composant et rajout de nouvelles règles de diagnostic	93
3 Conclusion.....	94
 Conclusion générale	 95

Introduction

Enfin, nous illustrons le fonctionnement du prototype de système expert obtenu, par un exemple de session utilisateur.

Introduction

L'importance du diagnostic et de la maintenance en vue d'augmenter la sécurité, la fiabilité et la productivité, non seulement dans le domaine de la conversion d'énergie électrique mais aussi dans tous les domaines techniques, se manifeste par un nombre croissant de travaux. L'éventail des sujets comprend des capteurs intelligents et de plus en plus des méthodes de détection de défauts faisant appel à des ordinateurs puissants : méthodes d'estimation, filtres, réseaux de neurones artificiels et systèmes experts.

Le problème du diagnostic d'un système complexe repose sur la multitude de défauts possibles et sur leurs différents symptômes. Ceci nécessite la collaboration de nombreuses méthodes de détection et de surveillance pour arriver à une analyse globale sur l'état du système. C'est cette vision globale qui manque aux systèmes de surveillance classiques tels que les relais de protection.

Pour cette raison, nous avons été amenés à choisir une approche de type système expert pour pallier les problèmes du diagnostic et de la maintenance. Le but de ce travail est de développer une méthodologie et de trouver les structures informatiques permettant de prendre en compte les différents aspects du problème de diagnostic et de maintenance dans un environnement industriel. Nous avons choisi, comme base de notre étude le diagnostic d'entraînements synchrones de grande puissance, d'une part en raison du rôle important et souvent stratégique que jouent ces entraînements dans la motorisation de grands procédés industriels et d'autre part pour la diversité du matériel mis en jeu. La structure et les composants de ces entraînements s'adaptant à des contraintes spécifiques de chaque installation, il est indispensable qu'un système de diagnostic soit assez souple et évolutif pour pouvoir prendre en compte les caractéristiques de toute nouvelle installation.

Dans le chapitre I, nous présentons d'abord le cadre de notre étude. Puis, nous faisons une analyse non exhaustive des différentes méthodes utilisées en diagnostic dans le domaine du génie électrique.

Suite à cela, nous avons opté pour une approche système expert, dont nous présentons les concepts et principes dans le chapitre II.

Puis, nous avons pu implanter informatiquement cette approche, comme le décrit le chapitre III.

Chapitre I

Présentation des Problèmes

- le transformateur de puissance, qui, selon la configuration peut comporter un ou deux secondaires,
- le convertisseur statique qui comporte un redresseur et un commutateur de courant, liés entre eux par une self de lissage. Il peut être de forme hexaphasée, dodécaphasée ou mixte.
- la machine synchrone qui a une ou deux étoiles (stator) selon les contraintes imposées par le procédé,
- la partie commande et surveillance qui est réalisée dans des installations modernes en numérique pour des raisons de fiabilité et de souplesse,
- la partie excitation. Dans le cas d'une machine sans bague ni balais (brushless), il s'agit d'un convertisseur statique alimentant une génératrice asynchrone qui est située sur l'arbre de la machine synchrone. Le redressement se fait par un pont de diodes tournantes. Pour un entraînement nécessitant plus de dynamique, l'excitation est faite par un pont de Graëtz, mais cette fois-ci, la machine est dotée de bagues et balais.

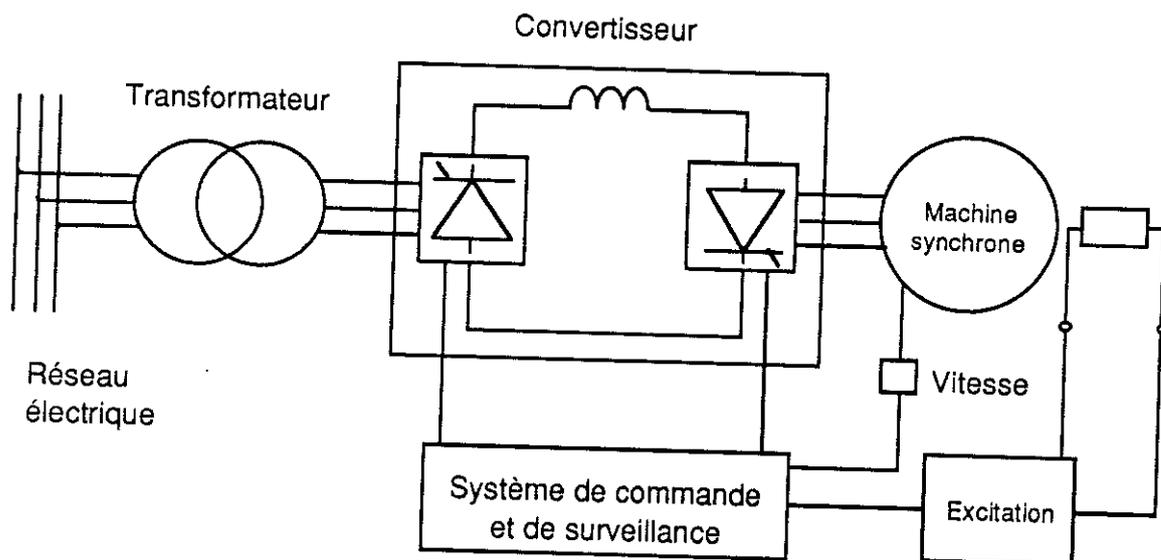


Figure I.1 Blocs dans un entraînement synchrone

1.3 Le système de contrôle et de surveillance

Nous partons de l'hypothèse qu'un système expert pour le diagnostic de pannes doit avoir accès aux mêmes informations sur l'équipement à diagnostiquer qu'un expert humain. Ceci implique que tous les paramètres du système de commande ainsi que les informations de la surveillance lui sont disponibles (à tout instant ou éventuellement sur requêtes).

Chapitre I

Présentation des problèmes

Dans ce chapitre, après avoir présenté l'objet de notre étude, le diagnostic de pannes pour de grands entraînements électriques, nous présentons différentes méthodes pouvant fournir des éléments d'aide au diagnostic.

1 Présentation de l'objet de l'étude

1.1 Les entraînements synchrones de grande puissance

Les grands entraînements synchrones dont la puissance peut atteindre jusqu'à 50 MW sont en train de prendre le relais des turbines à gaz ou à vapeur dans la motorisation de grands procédés industriels tels que les pompes et les compresseurs dans la pétrochimie ou les laminoirs dans la métallurgie. Ils représentent un meilleur choix économique par rapport aux turbines en raison de leurs nombreux avantages : ils sont moins encombrants que celles-ci tout en étant plus fiables et plus facilement maintenables. De plus, leur contrôle de vitesse est plus simple et ils offrent un meilleur rendement.

Les avantages au niveau de la maintenance et de la fiabilité sont d'autant plus décisifs que l'entraînement joue un rôle important dans le procédé industriel. L'enjeu économique du procédé demande souvent une extrême fiabilité ce qui nécessite d'avoir recours à des structures redondantes et une conception qui rend sa maintenance aisée.

Sachant que, par exemple, dans la pétrochimie le coût des pertes de production causées par un arrêt imprévu peut dépasser le coût de la motorisation, on comprend la nécessité d'une maintenance préventive et de méthodes de diagnostic appropriées pour réduire le plus possible le temps d'arrêt et pour programmer les arrêts indispensables.

1.2 Composants d'un entraînement synchrone

Dans le cas d'un entraînement électromécanique tel que celui que nous avons choisi comme base pour développer notre système expert de diagnostic, l'ensemble à prendre en compte pour la surveillance, le diagnostic et la maintenance comporte les éléments (blocs) suivants (figure I.1) :

- le réseau d'alimentation,

Le diagnostic des grands entraînements pose plusieurs problèmes inhérents au domaine:

- La diversité du matériel et de technologies rend ces installations complexes: l'interaction entre des systèmes électriques (réseau, transformateur, convertisseur), électromécaniques (machine, auxiliaires) et informatiques (commande par ordinateurs et logiciels) rend la diagnostic difficile en raison de la multiplicité et de la diversité des causes de pannes possibles.
- Les entraînements de grande puissance étant conçus sur mesure, leurs structures varient selon les applications.
- Ce sont des installations à priori très fiables. Il y a alors peu de pannes et par conséquent peu de retour d'expérience, ce qui rend l'acquisition d'expertise de diagnostic difficile.
- Le matériel est soumis à l'évolution technologique (capteurs, protections intelligentes, etc..)

Aussi, une solution pour le diagnostic des grands entraînements doit s'adapter aux caractéristiques de ce problème ouvert. Ainsi, un outil qui propose d'apporter une aide au diagnostic doit avoir les caractéristiques suivantes:

- une architecture ouverte pour s'adapter aux particularités des différentes installations et pour pouvoir intégrer des évolutions futures,
- une structure claire, modélisée dans un formalisme appropriée qui facilite l'intégration de nouvelles connaissances (expertise),
- il doit être facile à utiliser pour un utilisateur non-informaticien (par exemple, pour lui donner une aide dans des rares cas de pannes),
- il doit faciliter son adaptation aux installations variées pour lesquelles il doit donner une aide au diagnostic,
- il doit intégrer des outils d'analyse pour fournir des informations synthétiques sur l'exploitation de l'installation.

L'entraînement que nous avons choisi comme base de notre étude comporte un système de contrôle et de surveillance numérique qui facilitera les échanges entre le procédé et le système expert.

Les tâches de la commande sont réparties entre quatre processeurs numériques qui sont chacun responsables du matériel auquel ils sont associés. L'architecture du système de contrôle est proche de celle du convertisseur (figure I.2).

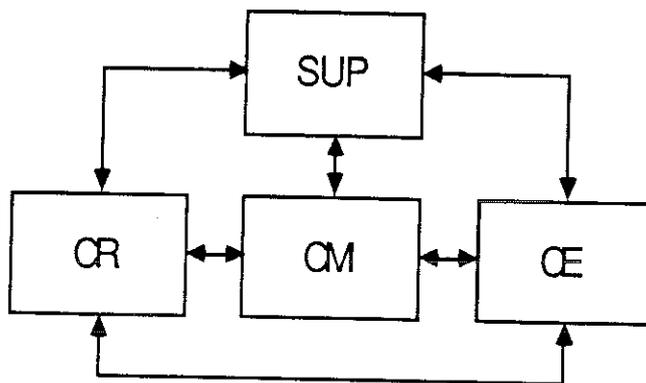


Figure I.2 Architecture du système de contrôle

A chaque pont du convertisseur est associé un processeur qui effectue les tâches de contrôle et de régulation ainsi que l'acquisition des données dont il a besoin. Il y a quatre processeurs (figure I.2) :

- CR, processeur pour le convertisseur côté réseau.
- CM, processeur pour le convertisseur côté machine.
- CE, processeur pour le convertisseur d'excitation.
- SUP, processeur de supervision.

1.4 Problème de diagnostic de grands entraînement électriques

Les entraînements électriques jouent souvent un rôle stratégique dans la motorisation de procédés industriels. Le coût d'une immobilisation suite à un arrêt intempestif pouvant excéder le coût de la motorisation, on cherche à créer des outils qui permettent d'augmenter la fiabilité et la disponibilité de ces installations.

L'approche classique qui consiste à avoir recours aux redondances pour les parties les plus critiques (cf. § I.2.1) augmente la complexité et le coût de l'équipement et ne résout pas le problème du diagnostic. Une aide au diagnostic peut viser plusieurs objectifs: la détection précoce d'anomalies avant arrêt intempestif et l'aide à l'analyse de la cause après incident pour actions curatives ou correctives. Ces objectifs nécessitent des connaissances différentes sur l'installation et l'utilisation d'approches différentes.

Si l'on prend l'exemple de deux capteurs, si leurs mesures $m1$ et $m2$ varient d'une manière significative, on peut dire qu'un des deux capteurs est tombé en panne. Ceci se fait par simple comparaison du résidu r (différence des valeurs filtrées $m1f$ et $m2f$) à un seuil d'incertitude qui est fonction des tolérances et du bruit des capteurs. Cependant, on ne peut pas dire lequel continue à fonctionner normalement alors que cela serait important pour la poursuite du fonctionnement. Pour un diagnostic plus précis avec une localisation du défaut, il faut instrumenter la mesure avec au moins trois capteurs (figure 1.4). Ainsi il est possible de se fier aux signaux identiques de deux capteurs pour décider de la panne du troisième. Ceci nécessite un dispositif de vote pour identifier celui qui est en panne. Ce dispositif s'appuie sur les trois résidus $r1$, $r2$, $r3$, calculés par le détecteur (figure 1.4):

$$\begin{aligned} r1 &= m1f - m2f \\ r2 &= m1f - m3f \\ r3 &= m2f - m3f \end{aligned}$$

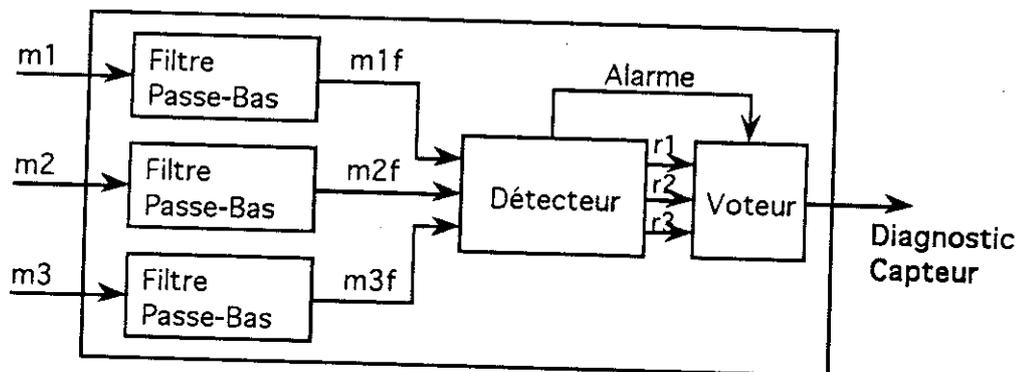


Figure I.4 Redondance matérielle triple avec localisation du défaut [BRU 90]

2.1.2 Redondances dans les entraînements électriques de grande puissance

Dans le domaine des entraînements de grande puissance la fiabilité est souvent augmentée par une redondance simple du système de commande entier. Actuellement, on arrive à un temps de fonctionnement sans arrêt théorique de 5 ans.

Cette redondance est justifiée pour des installations à enjeu économique important comme, par exemple, les entraînements dans des processus pétrochimiques.

D'une part, le système de capteurs et l'électronique de commande sont les plus sensibles aux défauts. D'autre part, le coût du système de commande est relativement faible par rapport au coût de l'ensemble (machine, transformateur, convertisseur). Puis, il y a parfois des redondances au niveau des thyristors dans les bras des convertisseurs (possibilité de

2 Survol sur quelques méthodes de diagnostic

Maintenant, nous allons présenter les principales méthodes applicables en diagnostic d'installations industrielles. Après un bref survol des méthodes existantes, nous développerons les différentes approches de diagnostic par système expert. En raison du nombre important de méthodes, ce survol ne peut pas être exhaustif. Notre but est uniquement de vérifier l'adaptation des différentes approches retenues en vue d'une utilisation dans les entraînements électromécaniques de grande puissance.

2.1 Diagnostic par redondance matérielle

2.1.1 Présentation

La méthode consiste en la multiplication des composants nécessaires au système pour qu'il remplisse ses fonctions. Elle permet aussi de s'affranchir d'une ou plusieurs défaillances [VIL 88] [PAG 80]. Ainsi, la disponibilité et la fiabilité du système sont augmentées car le fonctionnement est assuré même en présence des pannes tolérées. La redondance matérielle s'applique normalement pour des composants qui sont importants pour la sécurité (systèmes de refroidissement dans les centrales nucléaires) ou dont la probabilité intrinsèque de panne est élevée (capteurs, chaînes de commande dans les avions).

En fonction de l'application et du but visé pour la redondance le diagnostic est plus ou moins précis. Pour pouvoir détecter un défaut d'un capteur, il suffit de le dupliquer ce qui donne une redondance double (figure 1.3).

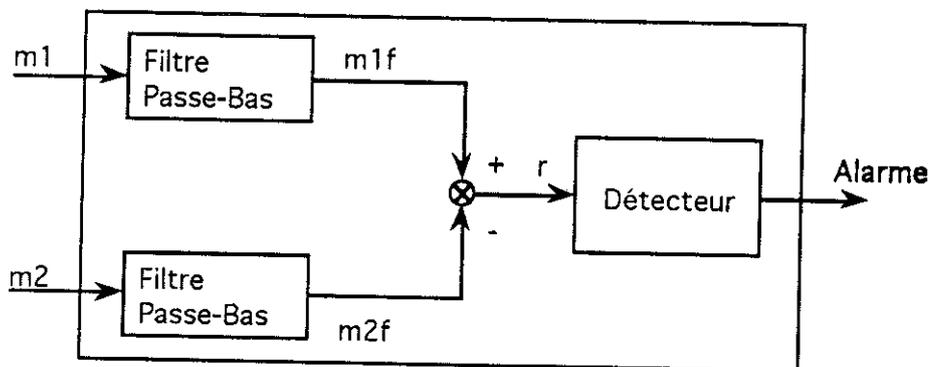


Figure I.3 Redondance matérielle double [BRU 90]

simple et permet une tolérance envers les pannes prévues et parfois une réparation en cours de fonctionnement.

Cependant, elle augmente le prix et la complexité de l'équipement (nécessité d'installer des systèmes de détection) et peut poser des problèmes d'emplacement des capteurs supplémentaires. Enfin, il y a le problème de l'interdépendance des pannes : des systèmes identiques de même conception et même fabrication fonctionnant dans le même environnement ne vont-ils pas tomber simultanément en panne ?

Dans le domaine des entraînements, des redondances ont fait leur preuve pour les cas d'applications cités mais elles ne peuvent pas résoudre tous les problèmes liés au diagnostic.

2.2 Les méthodes de maintenance

Le but de la maintenance est de réduire les anomalies de fonctionnement des systèmes. Parmi les méthodes et techniques de maintenance, nous allons en présenter deux qui interviennent dans la phase de conception des systèmes :

- * l'analyse des modes de défaillances, de leurs effets et de leur criticité (AMDEC),
- * les arbres de défaillance.

Ces méthodes consistent à faire un diagnostic préalable du système afin d'en améliorer sa conception. L'idée consiste à répertorier tous les défauts possibles avec toutes les causes possibles, les moyens éventuels de les détecter, etc...

2.2.1 Analyse des modes de défaillance, de leurs effets et de leur criticité (AMDEC)

a) Présentation

Cette analyse, connue aussi sous le nom de FMECA (Failure Modes, Effects and Criticality Analysis), est une méthode inductive pour l'étude systématique des causes et des effets des défaillances qui peuvent affecter un composant.

Elle s'appuie sur les points suivants [VIL 88]:

- La décomposition du système en blocs fonctionnels.
 Cette étape permet de décrire le système sous forme d'un diagramme composé de blocs correspondants aux fonctions nécessaires à la bonne marche du système. Par exemple, pour un entraînement, cela donnera (figure I.5):

fonctionner avec un thyristor en court-circuit permanent), des auxiliaires (alimentations ou systèmes de refroidissement, etc...).

Le principe de la redondance électronique consiste à avoir deux systèmes de commande fonctionnant en même temps. Pendant qu'une électronique A est active, c'est à dire elle commande le convertisseur, l'autre (B) est en veille, produisant les mêmes signaux de commande que A. Une électronique de surveillance C compare les entrées et sorties des deux. Le rôle de l'électronique C est de basculer du système actif sur le système en veille lors de l'apparition de certains problèmes.

Le raisonnement choisi est le suivant :

- Si un défaut est détecté par le système actif et non par le système en veille (ex. apparition d'une surintensité, surcharge, manque tension ...), on suppose qu'il s'agit d'un défaut dans le système actif (défaut capteur ou défaut carte électronique de commande) et on bascule sur le système en veille car on considère que celui-ci marche correctement.
- Si un défaut est détecté par l'électronique en veille et non sur le système actif, on suppose que la panne est au niveau du système en veille. L'entraînement continue à fonctionner normalement mais dorénavant sans redondance de commande.
- Si un défaut est détecté par les deux électroniques, ses causes sont probablement hors électronique. Dans ce cas, il peut y avoir différentes stratégies de procéder, par exemple, arrêter l'entraînement (la stratégie la plus prudente), arrêter et redémarrer (suite à la signalisation de certains défauts) ou forcer la poursuite de la marche au risque de casser du matériel.

Cependant, il peut y avoir des problèmes avec la redondance électronique, par exemple :

- des basculements intempestifs d'une commande sur l'autre dont les causes sont difficilement analysables,
- une perturbation de l'électronique par des parasites engendrés par des effets électromagnétiques, par exemple, peut se traduire par l'apparition d'impulsions de commande de thyristors intempestives ce qui peut entraîner une surintensité et ainsi un basculement de la commande sans qu'il y ait une cause matérielle dans l'électronique, les capteurs et la puissance.

2.1.3 Conclusion

En résumé, la redondance matérielle s'applique dans les domaines où la sécurité a une très haute priorité ou est fondamentale (par exemple, aéronautique, centrales nucléaires), et où la fiabilité et la disponibilité d'un système peuvent être augmentées à un coût raisonnable comme dans le cas des grands entraînements. Le concept de la redondance matérielle est

Cependant, elle est lourde à mettre en œuvre car elle nécessite l'analyse de tous les composants. De plus, décrire tous les modes de défaillance possible d'un composant et connaître les probabilités de panne n'est pas réalisable, même en se limitant aux défaillances uniques. En réalité, il faut prendre en compte, d'une part l'interaction entre les composants, et d'autre part les défaut dépendants qui sont difficilement envisageables. Pour ces raisons, cette méthode est actuellement utilisée dans les domaines où la sécurité est primordiale (aéronautique, nucléaire, vols spatiaux) ou dans les domaines où on dispose d'un grand nombre de données statistiques [BRU 90], par exemple la production automobile [SUH 92][JAU 92].

A notre connaissance, actuellement cette méthode est peu utilisée dans le domaine des grands entraînements électriques.

2.2.2 La méthode des arbres de défaillances

a) Présentation

C'est une méthode déductive qui vise à identifier pour chaque événement indésirable (interruption du fonctionnement, apparition d'un événement catastrophique, etc...) l'ensemble des événements élémentaires qui peuvent contribuer à son apparition. La méthode des arbres de défaillance ou arbres des causes est très employée dans les domaines de l'automobile, du nucléaire, de la chimie, de l'aéronautique, etc... pour évaluer la fiabilité d'un système [VIL 88][LIM 91]. Elle consiste en une recherche exhaustive des combinaisons possibles d'événements qui conduisent à un défaut de fonctionnement ou une panne. La représentation la plus utilisée est celle qui associe à chaque événement indésirable la liste de tous les enchaînements logiques des événements qui le provoqueraient. L'analyse est menée sur les causes de tous les événements jusqu'à arriver à des événements de base qui ne nécessitent pas de développements plus poussés. Un événement de base est, par exemple, défaillances premières (panne cataleptique intempestive, sans cause extérieure), les défaillances secondes (conséquences d'autres pannes) et les défaillances de commande (erreurs humaines, etc...).

Quand l'arbre est établi et les probabilités des événements élémentaires connues, on peut déterminer la probabilité de l'événement indésirable et les combinaisons les plus probables qui conduisent à son apparition.

L'exemple simpliste de la figure I.6 du circuit électrique montre l'arbre de défaillances pour l'événement "La lampe ne s'allume pas".

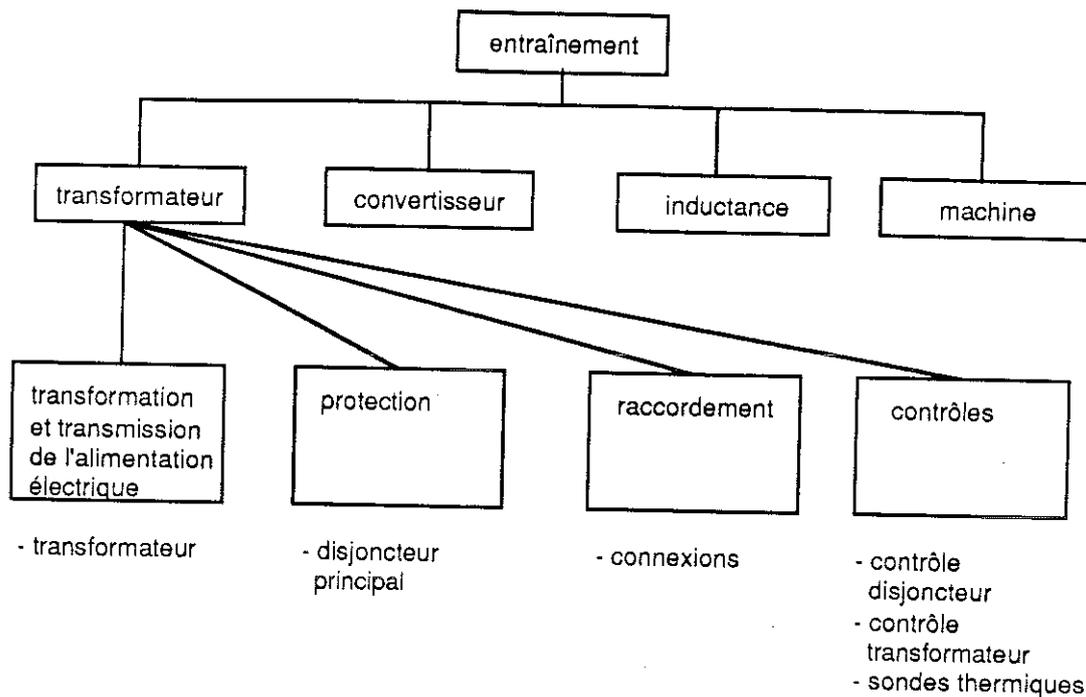


Figure I.5 Exemple de décomposition d'un entraînement en blocs fonctionnels

- Une liste la plus complète possible de tous les défauts des composants avec leurs causes les plus probables.
- L'étude des effets des modes de défaillance des composants.
Cette étude systématique décrit les conséquences d'un défaut sur un composant en supposant que ce défaut est unique.
- La classification des modes de défaillance par ordre de criticité établie à partir de la gravité du mode et de la probabilité de son apparition.
- La définition des moyens et interventions de détection, localisation, prévention pour chaque mode de défaillance.

b) Discussion

Cette méthode présente l'avantage d'envisager tous les défauts et les moyens de détection (capteurs). La prise en compte des probabilités de pannes sur les composants permet d'identifier les points faibles d'une installation et d'y pallier en prévoyant des redondances ou des actions de prévention. La méthode peut être un outil de dialogue entre les différentes équipes de maintenance, de conception et le client.

Cependant, cette méthode présente plusieurs inconvénients:

- L'inconvénient principal vient de l'exhaustivité nécessaire pour établir le diagnostic : uniquement les causes prévues peuvent être déduites. La difficulté réside en la lourdeur de la mise en œuvre de la méthode pour un système complexe. Pour cette raison, elle s'appuie souvent sur une analyse AMDEC effectuée préalablement.
- Il n'est pas toujours évident de savoir jusqu'à quel niveau de décomposition on doit mener l'analyse.
- La méthode est difficile à utiliser pour des systèmes complexes fortement interconnectés. Une étude exhaustive de tous les défauts et de leurs combinaisons s'avère rédhibitoire pour un entraînement. Ainsi il n'est pas envisageable de construire, par exemple, un arbre de défaillances pour l'événement indésirable "arrêt intempestif de l'entraînement" car les causes possibles sont multiples et non exclusives. Ceci peut faire intervenir des facteurs supplémentaires comme le mode d'exploitation, les conditions dans l'environnement, l'historique de l'entraînement, etc ...

Dans le domaine des entraînements, des arbres de défaillances ont été utilisés pour analyser les pannes de cartes analogiques de commande. L'idée était d'analyser les causes de quelques combinaisons de signaux logiques issus de protections pour donner à l'utilisateur une indication sur la carte à changer après un défaut. L'expérience a montré que ces arbres étaient lourds à mettre en œuvre et difficiles à modifier (pour prendre en compte une nouvelle combinaison de causes d'un défaut) car implémentés par une logique câblée. De plus comme ils étaient basés sur un nombre restreint de signaux logiques les indications données étaient très succinctes.

2.2.3 Conclusion

En conclusion, à priori, les méthodes de maintenance présentées ici peuvent apporter une aide au diagnostic en tant qu'outil d'analyse. Cependant, la nature systématique, rigoureuse et exhaustive de ces approches entraîne une lourdeur de mise en œuvre et d'utilisation telles qu'elles sont difficilement utilisables dans le domaine des grands entraînements.

En effet, alors qu'une analyse AMDEC, même incomplète, peut apporter des éléments de diagnostic et une structuration de l'entraînement, la construction d'un arbre de défaillances couvrant tous les cas de défauts imaginables n'est pas envisageable pour l'ensemble d'un entraînement.

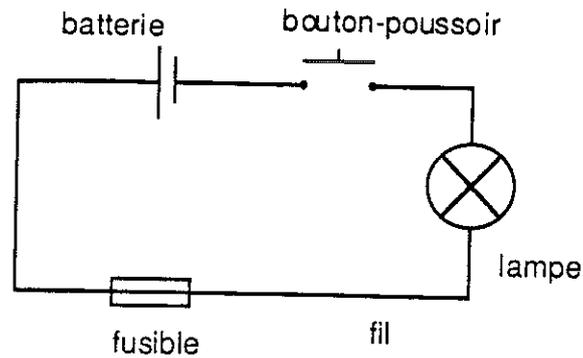


Figure I.6 Exemple pour établir un arbre de défaillances [VIL 88]

Par exemple, si la lampe dans le circuit de la figure I.6 ne s'allume pas lorsqu'on appuie sur le bouton-poussoir, les causes évidentes sont les suivantes :

1. le bouton-poussoir est bloqué
2. la batterie est déchargée
3. le fusible est ouvert
4. un fil est coupé
5. la lampe est grillée

Ceci est représenté par l'arbre suivant (figure I.7) :

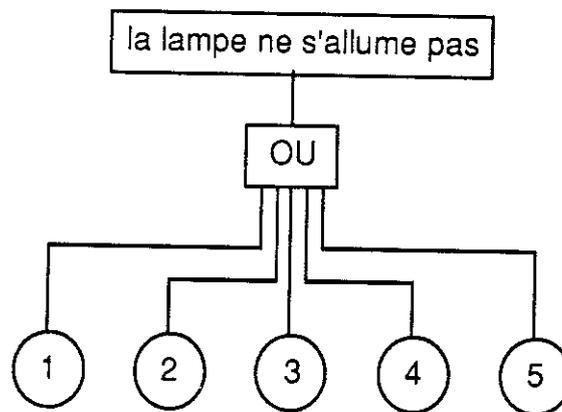


Figure I.7 Arbre de défaillances pour l'exemple de la figure I.6 [VIL 88]

b) Discussion

Cette méthode présente différents avantages:

- Elle permet d'analyser toutes les causes possibles d'un défaut et donc d'améliorer la conception du système.
- La décomposition arborescente permet un diagnostic relativement rapide des défauts.

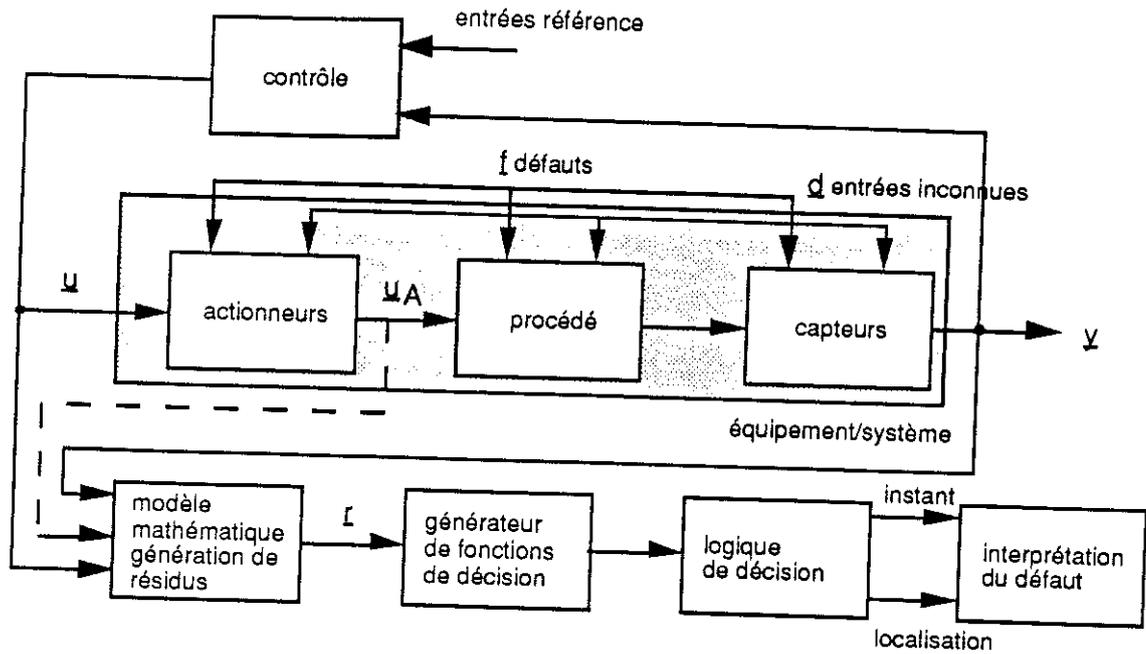


Figure I.8 Détection de défauts par redondance analytique [FRA 90]

- \underline{u} : vecteur d'entrée
- \underline{u}_A : vecteur de sortie des actionneurs
- \underline{f} : vecteur des défauts
- \underline{y} : vecteur des sorties mesurées
- \underline{r} : vecteur des résidus

2.3.2 Discussion

Quels que soient le modèle et la réalisation choisis le problème principal est l'exhaustivité des défauts possibles dans un système complexe comme un entraînement de grande puissance. On ne peut pas prévoir tous les défauts ni connaître toutes leurs signatures. Celles-ci risquent de varier avec le mode de fonctionnement dans lequel un défaut intervient. Quand plusieurs défauts ont des signatures similaires il faut prendre en compte des informations supplémentaires pour les distinguer. Toutes les connaissances nécessaires pour le diagnostic ne sont pas exprimables sous forme analytique et nécessitent un traitement en amont de la phase de détection. Il peut s'avérer nécessaire d'incorporer de la connaissance heuristique sur le procédé dans le système de détection et localisation pour interpréter les résidus et identifier la cause d'un défaut [PAT 89].

La création d'un modèle analytique d'un système complexe et son adaptation au procédé sont des tâches difficiles. Il ne semble pas envisageable de créer un tel modèle qui couvre l'ensemble convertisseurs, commande et machine d'un entraînement complexe. Cependant il serait intéressant d'utiliser des modèles analytiques pour des problèmes partiels, par

2.3 Diagnostic par redondance analytique

2.3.1 Présentation

Le but de cette approche est la détection et la localisation d'anomalies dans un système dynamique. Les anomalies qu'on veut pouvoir identifier portent sur les capteurs, les actionneurs et sur les composants du procédé. Elles peuvent être de nature abrupte (défauts cataleptiques) ou lente (usure, vieillissement). Selon l'origine et la nature des défauts, il existe différentes approches de la redondance analytique [BRU 90] [PAT 89]: l'estimation d'état par observateurs ou dans l'espace de parité [FRA 90] [BER 93a] [CAT 92] et l'identification de paramètres du procédé [ISE 84] [BUC 91]. Ces approches sont toutes fondées sur l'existence d'un modèle mathématique du procédé.

L'idée principale de cette approche est d'utiliser le modèle mathématique en parallèle avec le procédé pour vérifier la cohérence entre ses comportements attendu et observé. En pratique, ceci peut consister à simuler le système en parallèle avec le procédé et à comparer des grandeurs significatives simulées avec les grandeurs mesurées du procédé.

L'apparition d'écarts significatifs entre le comportement attendu et celui observé se traduit par la modification de résidus qui sont exploités par la suite pour détecter et localiser un défaut éventuel. La figure I.8 montre la structure du procédé de détection par redondance analytique [FRA 90].

Le modèle mathématique prend comme entrées les sorties mesurées \underline{y} et les entrées \underline{u} ou \underline{u}_A du système. Les signaux résultants \underline{r} sont ensuite utilisés pour générer des fonctions de décision. La base pour la décision sur l'apparition d'un défaut est sa signature, qui traduit les manifestations caractéristiques des effets du défaut. La détermination de l'instant et de la localisation du défaut nécessite alors la connaissance a priori des signatures de tous les défauts qu'on veut détecter. Une des méthodes de localisation consiste à utiliser des modèles du système avec défaut [CAT 92].

Le suivi du système nécessite l'exploitation du modèle mathématique en temps réel ce qui impose des contraintes pour la réalisation du modèle. Quelques exemples de mise en œuvre du modèle sont l'utilisation d'un modèle simplifié [CAT 92] [NOR 91], l'implantation du modèle sur un calculateur analogique [SCH 93], ou l'utilisation d'un modèle qualitatif au lieu du modèle quantitatif du système [MON 92]. Un problème du modèle mathématique du procédé utilisé est qu'il doit être à la fois suffisamment précis pour détecter les défauts du procédé et robuste par rapport aux erreurs de modélisation et aux entrées inconnues du procédé.

2.4.2 Discussion

Conceptuellement, un réseau de neurones peut être considéré comme une boîte noire, il n'est pas nécessaire de construire un modèle du système à diagnostiquer, contrairement aux méthodes analytiques (cf. § I.2.3). De ce fait, on cherche à utiliser cette méthode pour interpréter des signaux complexes tels que les spectres de vibrations [WAT 92] ou de décharges partielles [OGI 91] pour lesquels des méthodes classiques de traitement de signal [CAP 92] [DRO 92] sont difficiles à établir.

Cependant, il n'existe pas encore une méthodologie pour maîtriser les problèmes inhérents qui sont principalement le choix de la structure, de la taille du réseau et des algorithmes d'apprentissage pour un problème précis. La construction d'un réseau est, à l'heure actuelle, un processus itératif, la solution n'étant pas assurée à priori.

Des recherches récentes sur l'application de réseau de neurones dans le cadre du diagnostic industriel donnent lieu à penser que cette approche sera intégrée dans des systèmes de diagnostic futurs.

A notre connaissance, ce type d'approche n'est pas encore utilisée dans le domaine des grands entraînements. En raison de la complexité de ces derniers, si on souhaite utiliser cette méthode, ce ne pourra être que ponctuellement pour des aspects partiels (par exemple: vibrations dans la machine, décharges dans le transformateur, etc...).

2.5 Les systèmes experts

Les problèmes auxquels on se heurte en faisant du diagnostic de systèmes industriels sont principalement dûs à la complexité des systèmes réels par opposition aux modèles nécessairement simplifiés que l'on utilise.

Les approches présentées ci-dessus ont pour caractéristique d'être systématiques, parfois exhaustives ainsi qu'algorithmiques.

Or, le diagnostic d'un système complexe est un problème de nature ouverte :

- les données nécessaires pour le diagnostic ne sont pas toujours accessibles, ce qui conduit à l'utilisation de raisonnements basés sur des données incomplètes. Or, les modèles algorithmiques peuvent difficilement prendre en compte des informations imprécises ou incomplètes.

exemple, pour surveiller des grandeurs non mesurables directement comme les paramètres statoriques et rotoriques de la machine qui peuvent donner des indications sur des défauts naissants.

2.3.3 Conclusion

L'approche analytique est un élément indispensable pour le diagnostic d'un système complexe tel qu'un grand entraînement. Il semble intéressant de pouvoir l'utiliser dans un système de diagnostic intégré mais elle ne peut pas être la seule représentation possible dans un système de diagnostic idéal.

2.4 Diagnostic par réseaux de neurones artificiels

Ces dernières années, une nouvelle approche pour le diagnostic commence à émerger: les réseaux de neurones artificiels. Issus de la recherche en neurophysiologie et en informatique, cette approche fait objet de recherches dans beaucoup de domaines comme, par exemple, la compréhension et la synthétisation de la langue naturelle, la reconnaissance de formes, la classification, le contrôle [QUE 91] et le diagnostic [FAR 91] [WOO 91] [OGI 91] [WAT 92]. Cette technologie comporte des caractéristiques très intéressantes comme la faculté d'apprentissage et d'auto-organisation, d'abstraction et la perspective de performances de calcul en parallèle inégalées.

2.4.1 Principe

Le principe de diagnostic par réseaux de neurones est basé sur la reconnaissance de formes. Pour identifier des défauts dans un système, il faut disposer d'un nombre suffisant d'exemples de bon fonctionnement et de défauts que l'on fait apprendre au réseau. Pendant la phase d'apprentissage, les exemples sont présentés au réseau en entrée avec les diagnostics correspondants à la sortie. Le réseau s'auto-organise, apprenant à relier les exemples montrés aux diagnostics. Après l'apprentissage, le réseau ne reconnaît pas seulement les exemples appris mais également des paradigmes leur ressemblant ce qui correspond à une certaine robustesse par rapport aux déformations de signaux par bruit.

Pour un traitement par réseau de neurones, les exemples doivent être mis en forme pour être entrés dans le réseau. Ce sont des grandeurs caractéristiques des défauts à diagnostiquer, filtrées et prétraitées par exemples par une discrétisation [CHO 91] ou une analyse de Fourier (FFT) [WAT 92].

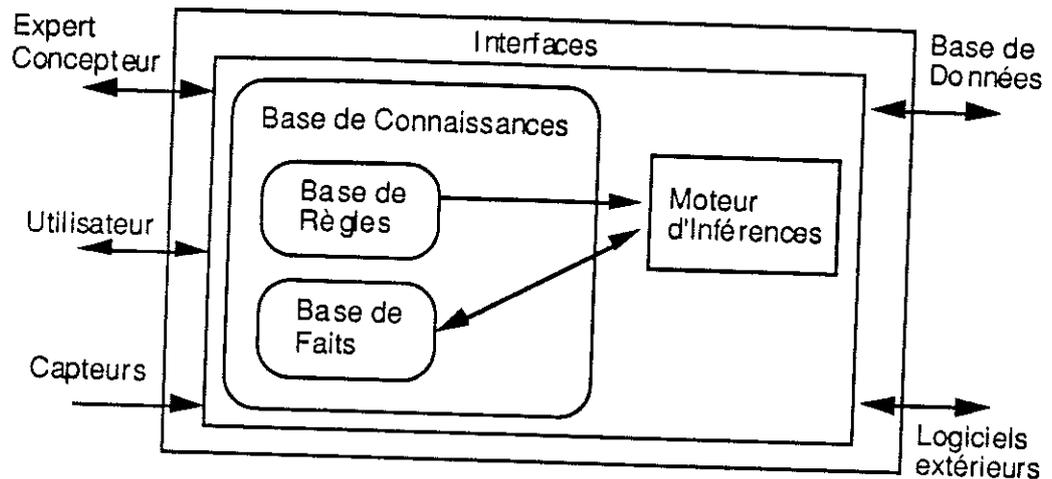


Figure I.9 Architecture d'un système expert [HAT 91]

- La **base de connaissances** contient différentes formes du savoir :
 - la **base de faits** : c'est la mémoire de travail du système qui contient les **données** initiales décrivant le problème à traiter, puis les hypothèses émises et les faits apparus pendant le raisonnement (c'est-à-dire les objets du monde ou les faits concernant les objets ou classifications ou événements).
 - La **base de règles** : c'est la connaissance permanente, les règles de diagnostic et les règles influant sur la stratégie.
- Le **moteur d'inférences** : c'est le mécanisme de **raisonnement** exécutant les règles modifiant les faits selon une stratégie soit intégrée dans le moteur, soit définie par le concepteur.
- Les **interfaces** assurant la communication du système avec l'extérieur et son intégration dans un système général d'information.

Les systèmes experts se distinguent en principe par la manière dont sont représentés les faits et les règles et par le mécanisme de raisonnement. Par la suite, nous résumerons les principales caractéristiques en vue d'une application aux problèmes du diagnostic.

a) Les faits (données)

Les faits sont les données manipulées par les règles. Ils peuvent être des expressions simples, par exemple "*alarme-surintensité-premier-niveau*" ou "*le moteur chauffe*". Des approches plus souples permettent des faits composés, c'est-à-dire des faits pouvant en contenir d'autres par exemple (*alarme (surintensité 1)*).

- Souvent, les modèles nécessaires n'existent pas. Quand ils existent, comme par exemple les simulateurs dans le domaine des réseaux électriques, le diagnostic est un problème trop large et divers pour être résolu d'une manière systématique et exhaustive dans un temps raisonnable.

Dans ces cas, l'homme fait normalement appel à des hypothèses pour pallier les manques de ses connaissances sur le problème. Ceci lui permet de réduire un vaste domaine d'investigation en n'exploitant que les possibilités les plus pertinentes et d'arriver à la solution dans un temps raisonnable. Un moyen important pour faire ceci est son expérience, chose qui échappe à la modélisation algorithmique.

2.5.1 Introduction

Les systèmes experts sont des outils conçus pour pouvoir modéliser l'approche des experts dans un domaine spécifique. Ils permettent de formaliser leurs connaissances et fournissent un mécanisme pour les exploiter.

La séparation entre les connaissances nécessaires pour résoudre un problème et les mécanismes de raisonnement exploitant ces connaissances est un concept de base des systèmes experts [HAT 91] [FAR 85] [ERM 89] [WAT 86]. [HAY 83]. Cela les distingue de l'algorithmique classique qui mélange étroitement les connaissances et leur exploitation. D'une manière générale, les connaissances prennent différentes formes :

- objets du monde (par exemple: une machine, des ponts convertisseur, des thyristors, etc...),
- faits concernant ces objets (par exemple: "le courant nominal du moteur est 1000 A"),
- événements (exemple: "le transformateur chauffe"),
- règles heuristiques de savoir faire (par exemple: "si le moteur chauffe, le ventilateur est peut être hors fonction"),
- règles stratégiques sur l'organisation du raisonnement (par exemple: "avant de choisir une action il faut envisager toutes les causes possibles"), etc...

2.5.2 Structure d'un système expert

Les systèmes experts mettent en œuvre des outils pour modéliser les différents aspects du savoir. La structure classique d'un système expert reflète la séparation explicite des différents aspects, comme le montre la figure I.9. Elle contient plusieurs parties: un moteur d'inférence, une base de connaissance et des interfaces.

Les premiers référencent les faits directement dans les règles, par exemple

Si "alarme-surintensité"
 et "alarme-déséquilibre courant"
 Alors "panne-puissance-convertisseur côté réseau".

La conclusion consiste à ajouter un nouveau fait à la base de faits. Les moteurs d'ordre zéro sont essentiellement utilisés pour effectuer des diagnostics ou des classifications. Dans ce cas, la base de règles est équivalente à un arbre de décision, comme cela est illustré dans l'annexe A-I.1.1 et A-I.1.2.

Les moteurs d'ordre 1 se distinguent des moteurs d'ordre zéro par la possibilité d'utiliser des variables dans les règles. Ainsi une règle peut s'appliquer sur plusieurs faits du même type, par exemple :

Si X est un moteur
 et F est le fusible de X
 et état de F est fondu
 Alors état de X est arrêté

Dans cet exemple, la variable X peut être remplacée par un fait de type moteur quelconque de la base de données, puis la variable F prendra le fusible qui protège le moteur en question. Cela suppose que des relations existent dans la base de données entre les moteurs et les fusibles qui les protègent. Les expressions *fondu* et *arrêté* sont des faits qui représentent des propriétés des objets.

La puissance d'expression des règles avec variables est plus grande que celle des moteurs d'ordre zéro. Une seule règle permet ainsi de résumer un ensemble de règles d'un tel moteur. Cela est un avantage car les règles sont génériques et indépendantes d'une application particulière. Cependant, le fonctionnement d'un moteur d'inférences d'ordre 1 est informatiquement plus complexe et moins rapide qu'un moteur d'ordre zéro. Ceci peut être contraignant pour des applications en temps réel qui travaillent sur un grand nombre de données. Une solution à ce problème peut consister en l'utilisation d'un système d'ordre 1 pour la description des règles génériques et une traduction ultérieure de celles-ci en règles d'ordre zéro par compilation pour une base de faits figée. Une telle approche a été utilisée pour la création d'un système expert de surveillance de centrales nucléaires [CHE 90].

c.2) L'exploitation des règles

Un moteur d'inférences peut fonctionner selon deux mécanismes de base :

- un raisonnement en **chaînage avant**, c'est-à-dire guidé par les données, au cours duquel les règles sont utilisées dans le sens conditions vers conclusions. Un tel raisonnement est approprié en diagnostic lorsqu'il s'agit de réagir pour une situation

Les faits peuvent également être représentés par des structures complexes s'appuyant sur le formalisme d'objet ou de "frames" [MEY 88] [HAT 91] [WED 90]. C'est un formalisme très puissant qui permet de représenter des objets du monde réel sur le plan structurel et fonctionnel. Il est particulièrement adapté pour représenter le grand nombre de données diverses, nécessaires pour le diagnostic d'installations complexes. L'approche objet en particulier permet de bien structurer les composants multiples d'une installation et de modéliser les caractéristiques de chacun d'eux. Un autre avantage en est la souplesse d'utilisation et l'évolutivité des produits conçus avec cette approche.

La notion de méthode inhérente au concept objet facilite la création des interfaces entre l'utilisateur et le procédé. On trouvera des exemples de représentation de faits par objets dans les chapitres suivants.

b) Les règles

Les règles expriment un petit élément de connaissance, déclaratif et autonome. Le format le plus utilisé est celui des règles de production de la forme.

Si conditions alors conclusions (α).

Ce formalisme est bien adapté pour représenter le raisonnement humain : une règle est déclenchée, c'est-à-dire ses conclusions sont effectuées si toutes les conditions dans la partie *Si* d'une règle sont remplies. La connaissance traduite par une règle peut être entachée d'incertitude. Ceci peut être réalisé en rajoutant un coefficient numérique (α) à la conclusion de la règle. Ce coefficient sera en quelque sorte un facteur de confiance relatif à cette connaissance. Ainsi il est possible de faire un raisonnement approximatif qui peut prendre en compte des faits et des règles incertains et imprécis. L'exemple dans l'annexe A-I.1.3 illustre un tel raisonnement.

Le formalisme utilisé dans les règles dépend du mécanisme de raisonnement et de sa façon de fonctionner. Le rôle du mécanisme de raisonnement consiste à sélectionner les règles et de les appliquer l'une après l'autre pour former le raisonnement. Pour ceci, ce mécanisme s'appuie sur les faits dans la base de faits manipulées dans les règles.

c) Caractéristiques des moteurs d'inférences

c.1) Fonctionnement avec ou sans variables

Suivant la nature des faits manipulés dans les règles, on distingue les systèmes à base de logique des propositions (moteurs d'ordre zéro) et ceux à base de logique des prédicats (moteurs d'ordre 1).

qui s'avère défaillant dans une phase ultérieure du raisonnement. Il se peut aussi que suite à une observation anormale, plusieurs hypothèses soient créées. Après confirmation d'une de ces hypothèses on admet que la cause a été identifiée. Si on écarte la possibilité de causes multiples il faut enlever les autres hypothèses et tout ce qui s'en suit car elles ne sont plus pertinentes.

Ainsi un système de maintien de vérité peut être une aide précieuse pour gérer la base de faits. Par contre, son fonctionnement ne doit pas être obligatoire et systématique, il doit être pilotable par la stratégie du diagnostic, car il existe aussi des cas où les conclusions sur un fait doivent rester après sa disparition. Ceci est le cas pour les alarmes transitoires, qui disparaissent après quelque temps ou qui sont effacées volontairement par l'utilisateur. Enlever les conclusions dans un tel cas correspondrait à oublier un événement important ce qui serait problématique dans un système de diagnostic.

Un exemple d'application des systèmes de maintien de vérité est donné dans le chapitre 2.3 qui traite du diagnostic dans le domaine des circuits électroniques. Dans cet exemple, l'approche est utilisée pour gérer des valeurs hypothétiques propagées à travers un modèle du circuit à diagnostiquer, le diagnostic étant fondé sur les contradictions entre valeurs hypothétiques et mesurées.

2.5.4 Conclusion

Les concepts présentés sont très généraux, si bien que lorsque l'on souhaite traiter une application réelle, on se trouve confronté à de nombreux problèmes. En annexe A-I, nous présentons quelques approches par système expert de diagnostic d'équipements techniques pour illustrer les concepts mentionnés ci-dessus. On s'apercevra que la notion de système expert est très générale et que la mise en œuvre pratique d'un tel système est fortement liée à l'application étudiée. On constate qu'il n'y a pas de modèle unique qui s'applique d'une manière universelle à tous les problèmes et toutes les applications.

Parmi les nombreuses approches qui existent, nous ne présentons que celles qui nous semblent les plus intéressantes pour le diagnostic de grands entraînements électriques. Nous en développons les caractéristiques les plus intéressantes au travers d'exemples sur différentes applications.

particulière. Cela permet par exemple de créer des hypothèses suite à des messages d'alarmes venant du procédé. Dans ce cas, le système évalue les conditions de toutes ses règles et ne retient que celles qui réagissent sur l'alarme présente. Celles-ci sont ensuite appliquées de sorte que leurs conclusions puissent ajouter de nouveaux faits à la base. Le système présenté dans le chapitre 2.1.3 utilise ce type de raisonnement.

- Un raisonnement en **chaînage arrière**, c'est-à-dire guidé par les buts, où les règles sont utilisées dans le sens conclusions vers conditions. Ce type de raisonnement intervient aussi dans le diagnostic quand il s'agit de vérifier une hypothèse. L'hypothèse devient l'objectif dont le système cherche à démontrer la validité. Un exemple d'un système de diagnostic utilisant ce type de raisonnement se trouve dans le chapitre 2.1.1. Le langage PROLOG est un exemple typique mettant en œuvre ce formalisme [FAR 85] [DEL 88].

Il existe des moteurs qui permettent de combiner les deux modes de fonctionnement. Dans tous les cas, le moteur effectue un cycle dans lequel il sélectionne la règle à appliquer en premier, parmi les règles applicables dans sa base de règles.

La stratégie de raisonnement à appliquer, c'est-à-dire le type de chaînage et la méthode de résolution de conflit quand plusieurs règles sont applicables, doivent être spécifiées par le concepteur.

2.5.3 Les systèmes de maintien de vérité

Les systèmes de maintien de vérité (TMS = Truth Maintenance System ou ATMS = Assumption-based TMS) [DEK 87], [DER 91] [HAT 91] sont des concepts informatiques qui peuvent faire partie du mécanisme de raisonnement d'un système expert comme c'est le cas dans ART [TRI 91] ou qui sont couplés avec un mécanisme de création d'hypothèses [GEN 84] [REI 87] [SPU 90] (cf. annexe A-I.3).

Ces systèmes ont pour objectif d'assurer la cohérence d'une base de connaissances en révisant continuellement les déductions sensibles à l'évolution de la base de faits. Ils permettent par exemple de retrouver toutes les déductions qui ont été effectuées à partir d'un fait. Lorsque ce fait n'est plus valable, on peut considérer que tout ce qui a été déduit à partir de celui-ci ne l'est plus et ainsi enlever toutes ses conclusions de la base de connaissances grâce au système de maintien de vérité.

Cette propriété est très intéressante dans le domaine de diagnostic. En effet, on peut, par exemple, être amené à enlever toutes les conclusions effectuées suite au signal d'un capteur

- * s'appuyer sur les données des capteurs de l'entraînement et donc être couplé avec un système d'acquisition de signaux,
- * permettre une bonne évolutivité du système expert pour faciliter d'enrichir sa connaissance au fur et à mesure,
- * permettre la portabilité du système expert pour qu'il puisse facilement être adapté aux diverses applications possibles d'une famille d'entraînements électriques,
- * pouvoir intégrer d'autres méthodes de détection, par exemple à base de redondance analytique (cf. § I.1.3) ou de réseaux de neurones artificiels (cf. § I.1.4)...

Dans le chapitre suivant, nous présentons les différents concepts qui nous permettrons d'atteindre ces objectifs.

2.6 Conclusion

Nous avons présenté dans ce chapitre et en annexe A-I les principales démarches de diagnostic existantes, appliquées au domaine de l'électrotechnique. Ce tour d'horizon ne peut pas être complet, surtout en ce qui concerne les spécificités des démarches à base de connaissances. Nous avons montré les difficultés et limites des démarches classiques. Ces dernières constituent des éléments nécessaires mais pas satisfaisants si elles sont appliquées indépendamment l'une de l'autre. Actuellement, aucune des démarches classiques utilisée seule, ne peut résoudre le problème du diagnostic pour une application électrotechnique industrielle.

Parmi les méthodes présentées dans ce chapitre, nous avons opté pour une approche de type système expert. En effet, c'est l'approche la plus générale qui permet d'intégrer les autres démarches et donc d'utiliser au mieux les spécificités de chacune. De plus, elle semble appropriée pour enrichir la démarche de diagnostic à partir des expériences acquises sur le traitement de problèmes réels. Dans l'annexe A-I, nous présentons quelques approches aptes à modéliser des connaissances de diagnostic. Ce qu'il faut retenir, c'est que ces approches se distinguent essentiellement par la manière dont les connaissances sont structurées et par la complexité de la stratégie qu'elles utilisent.

Pour le diagnostic d'un système complexe, il s'est avéré qu'une bonne structuration des connaissances manipulées est indispensable pour la maintenabilité et donc la faisabilité d'un système expert de diagnostic. Ceci est surtout vrai pour la connaissance statique (les données du problème, situations typiques et/ou l'installation en question) et la connaissance dynamique (le savoir-faire, les heuristiques et la stratégie du diagnostic). Chaque type de connaissance nécessite un modèle approprié pour prendre en compte les nécessités du domaine d'application. Le modèle choisi est souvent un compromis entre plusieurs méthodes de représentation. Ce compromis dépend fortement des particularités du domaine d'application et de la nature de la connaissance disponible. Le problème à résoudre est donc le choix de ce compromis pour notre application. Nous avons essayé dans notre système de dégager quelques règles méthodologiques et des règles de conseil qui nous ont conduit à une implantation informatique.

Ainsi, pour notre système expert d'aide au diagnostic, nous nous sommes défini les objectifs suivants :

- * fournir une aide au diagnostic pour l'analyse et la compréhension de défauts. En effet, le diagnostic ne peut pas être entièrement automatisé en raison de l'étendu des problèmes et la connaissance incomplète en diagnostic,

Chapitre II

Modélisation des connaissances de diagnostic
dans le domaine des grands entraînements électriques

maintenance destinés à l'utilisateur, des analyses AMDEC partielles et quelques comptes-rendus d'interventions suite à de rares pannes. Nous avons utilisé ces documents, et surtout les comptes-rendus d'interventions, pour discuter avec plusieurs experts différents (problème de disponibilité) de leur démarche face à différents problèmes.

Remarque: La connaissance ainsi recueillie n'est que partielle, en raison des difficultés auxquelles nous avons été confrontés pour l'extraire. De plus, les exemples des pannes analysés avec les experts restent en général très spécifiques. Cependant, nous avons pu mettre en œuvre un formalisme de représentation pour ces connaissances. Pour cela, nous avons commencé par identifier certains éléments utilisés pour le diagnostic.

2.1 Éléments de connaissance de diagnostic

A partir des schémas électriques et de la documentation disponible de quelques installations indicatives, nous avons répertoriés les éléments suivants:

- la structure de l'entraînement (en fait, la première action consiste à se procurer un schéma à jour de l'installation en panne): configuration des ponts du convertisseurs (hexaphasée, dodécaphasée ...) et des auxiliaires (alimentations, refroidissements, ...)
- l'existence de redondances pour le système de contrôle commande, thyristors, refroidissement ..
- le type de matériel utilisé : thyristors, contacteurs, protections, capteurs, machine, paliers, etc...
- les signaux du système de surveillance (les alarmes signalées et celles provoquant un déclenchement, les grandeurs mesurées, les consignes, les limitations des régulateurs, les seuils de protection, etc ...)

D'autres éléments reflètent une expérience et une compréhension approfondie du fonctionnement des équipements:

- les problèmes possibles des composants de l'installation, leurs modes de défaillance possibles et les symptômes qui permettent de les identifier, par exemple: contacteur défaillant, bloqué ouvert ou fermé, transformateur de mesure en court-circuit, thyristor claqué, auto-amorçant, etc ...,
- le contexte du fonctionnement dans lequel un défaut s'est produit. Ceci s'exprime par les manifestations extérieures du défaut, c'est à dire les signaux fondamentaux et les messages du système de contrôle et des protections ainsi que des observations de l'opérateur,
- le fonctionnement correct de l'installation ainsi que la connaissance sur l'historique (incidents antérieurs et caractéristiques spécifiques à une installation). Cette connaissance est souvent issue de l'expérience acquise pendant la mise en route de l'installation, et se

Chapitre II

Modélisation des connaissances de diagnostic dans le domaine des grands entraînements électriques

1 Introduction

Dans ce chapitre nous allons décrire la démarche suivie pour mettre en œuvre un système expert de diagnostic dans le domaine des grands entraînements électriques. La démarche des experts est analysée et la connaissance en jeu identifiée. Ensuite, nous allons montrer comment les différents types de connaissances ont été structurés et formalisés en vue de leur implantation dans un outil informatique. L'exploitation de ces connaissances nécessite une stratégie qui simule le raisonnement d'un expert du domaine. Nous proposons un modèle pour la stratégie de diagnostic qui prend en compte des considérations pratiques sur l'utilisation de l'outil informatique. La vocation de cet outil à créer étant de fournir une aide pour l'utilisateur, il a fallu prendre en compte, à tous les niveaux de modélisation de la connaissance, les aspects pratiques de son utilisation comme la communication homme-machine, l'explication du conseil donné et son intégration dans l'environnement industriel de production.

Un aspect important de notre démarche est la généralité du système expert à créer. En fait, notre objectif n'était pas de créer le système expert pour une installation particulière mais de créer l'outil qui permet de générer des systèmes experts pour des installations d'entraînements diverses avec un minimum d'effort d'adaptation.

Ceci nécessite un modèle faisant coexister des connaissances génériques (générales pour tous les entraînements) et spécifiques (propres à une structure particulière).

2 L'expertise

Le premier pas vers la représentation des connaissances de diagnostic consiste en une étape d'acquisition de la connaissance des experts. Le savoir-faire en diagnostic n'existant pas sous une forme écrite et bien organisée, nous avons interrogé les experts et exploité les documents disponibles sur quelques installations indicatives. Les documents existants étaient des schémas électriques d'entraînements, des notices sur le principe de fonctionnement des entraînements synchrones, et sur leur mise en route, des manuels de

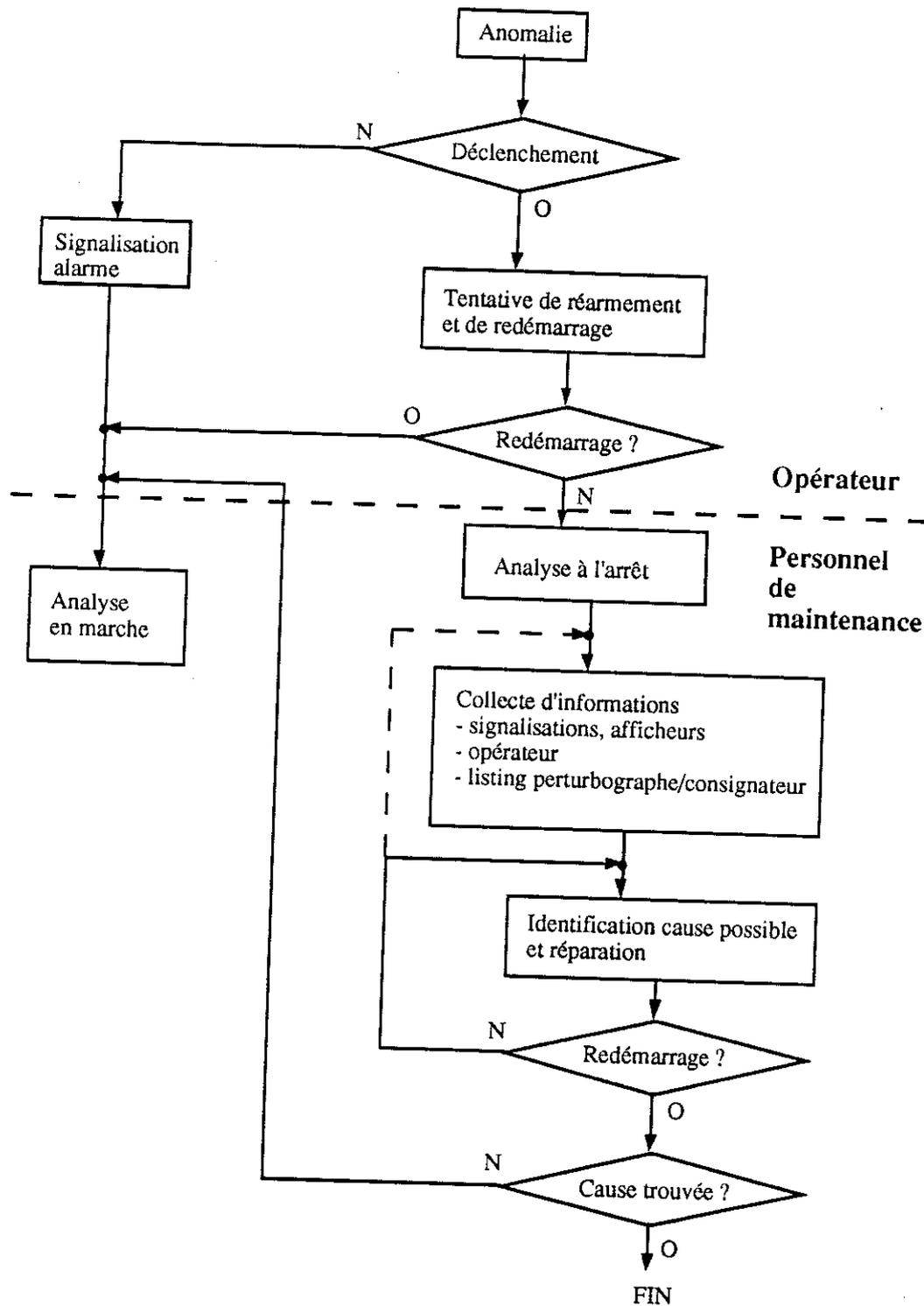


Figure II.1 Organigramme de la démarche de diagnostic d'une anomalie

Celle-ci consiste à instrumenter et à analyser les mesures accessibles sous tension (sans consignation), mais des actions correctives ne sont normalement pas possibles. Dans ce cas, il faut souvent attendre l'apparition prochaine de l'incident pour avoir des enregistrements exploitables.

traduit en une série de points de référence pour les grandeurs principales dans les différents points de fonctionnement,

- des méthodes de test pour vérifier le bon fonctionnement du matériel ou pour détecter les causes d'un défaut.

A ces éléments s'ajoute la connaissance de bon sens électrotechnique, facteur qui croît avec l'expérience de l'expert. C'est là le problème majeur des entretiens : les différents types de connaissances manipulées n'étant pas explicitement séparées, l'expert a du mal à exprimer "les règles du diagnostic" utilisables telles quelles dans un système expert. Un travail très important consiste donc à faire une analyse critique de la connaissance obtenue et à essayer d'en dégager un formalisme capable de structurer les différents aspects de ce savoir.

2.2 La démarche du diagnostic

La démarche des experts a dû être mise en évidence à travers des exemples discutés avec eux. En fait, la démarche en face d'un problème n'est pas toujours identique. Elle est difficilement généralisable, souvent itérative et dépend de plusieurs caractéristiques comme la gravité potentielle du problème, les conditions d'exploitation et les actions envisageables en conséquence. La figure II.1 montre un organigramme simplifié de la démarche des experts en fonction des circonstances d'une situation donnée : une anomalie est détectée par les capteurs du système de surveillance (par exemple: apparition d'une alarme) ou par une observation anormale de la part de l'opérateur chargé du fonctionnement du procédé.

Si l'entraînement est arrêté par la surveillance (déclenchement d'une protection et signalisation d'une alarme) une action habituelle de l'opérateur consiste, si possible, à supprimer les alarmes et à tenter de redémarrer. Cet essai de redémarrage nécessite un effacement des alarmes enregistrées car le système de contrôle interdit normalement le démarrage en présence d'alarmes. Si l'anomalie était transitoire l'entraînement recommence à tourner normalement. Cependant, tant que l'incident n'est pas analysé, il y a le risque qu'il se reproduise (problème réseau, défaut intempestif suite à un défaut naissant mais pas franc, etc ...).

C'est maintenant qu'interviennent les experts. Leur approche consiste à collecter d'abord un maximum d'informations sur l'incident. Pour ceci, ils interrogent l'utilisateur et analysent les enregistrements de la surveillance (liste d'alarmes sur une imprimante). Si l'entraînement tourne sans possibilité de l'arrêter pour des raisons de production, seulement une analyse en marche est envisageable.

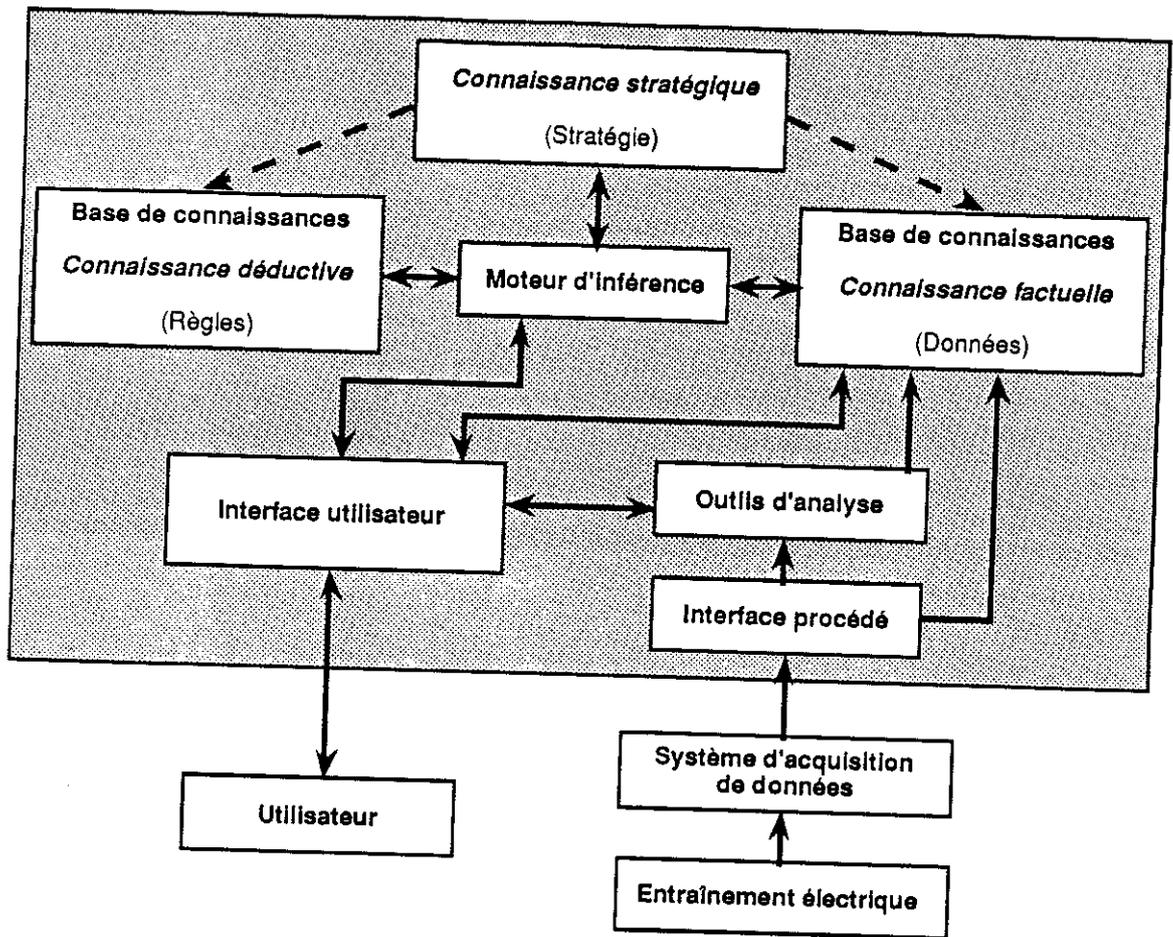


Figure II.2 Architecture du système expert d'aide au diagnostic

3 Modélisation de la connaissance factuelle

La connaissance factuelle regroupe tous les aspects statiques, c'est-à-dire les données disponibles sur l'entraînement à diagnostiquer. Il s'agit de représenter tous les composants avec leurs caractéristiques importantes pour le diagnostic. Par exemple, pour un ventilateur de réfrigération, il est important de connaître sa puissance et son courant nominal pour pouvoir juger d'une éventuelle surcharge et de connaître ses modes de défaillance caractéristiques. Ensuite il faut pouvoir le situer dans l'installation, connaître ses circuits de commande et de protection. En fait, tous ces détails dépendent de la qualité du support que l'on attend du système expert pour le diagnostic.

De plus, il est important de savoir interpréter les signaux émis par la surveillance, les grandeurs manipulées et de connaître le fonctionnement des protections. Ainsi, il y a un grand nombre d'informations supplémentaires et de la documentation à intégrer dans cette

Si l'entraînement est arrêté, l'analyse est facilitée par la possibilité d'accéder aux composants pour vérification et de faire des tests sur le fonctionnement des sous-ensembles. Il est possible de faire plusieurs essais de démarrage, de tester systématiquement les éléments des convertisseurs, etc ...

Lorsque la cause est identifiée, on passe à la réparation puis l'exploitation de l'installation peut continuer. Il se peut que la cause trouvée n'était qu'une conséquence d'une cause ultérieure n'ayant pas encore été identifiée. Dans ce cas les problèmes persistent ou réapparaissent après un certain temps de fonctionnement et l'analyse doit recommencer.

2.3 Analyse de la connaissance

L'analyse critique de la démarche et de l'expertise obtenue nous a conduit à définir trois catégories de connaissance [GEN 91]:

- la **connaissance factuelle** qui englobe les données physiques manipulées. Ce sont, par exemple, le matériel à diagnostiquer et les signaux issus du système de contrôle,
- la **connaissance déductive** qui regroupe les règles de diagnostic. Celles-ci permettent par exemple d'interpréter le résultat d'un test, de soupçonner ou innocenter un composant et d'identifier une cause probable d'un défaut,
- la **connaissance stratégique** qui traduit la façon d'enchaîner les règles sur les faits. Il s'agit ici de trouver une représentation du processus de diagnostic.

Remarque: Le **raisonnement** appliqué en cours de diagnostic se construit en appliquant les règles de diagnostic sur les données selon la stratégie qui est implantée dans le système expert. Ceci nous amène à l'architecture du système expert illustrée par la figure II.2 [GER 93].

Pour obtenir un système évolutif et souple, il est important de séparer explicitement ces trois catégories. Ceci n'est pas chose aisée car l'aspect stratégique (surtout) est souvent sous-jacent à la démarche personnelle de chaque expert et implicite aux règles qu'il utilise. Pourtant, cette séparation est indispensable si l'on veut réutiliser une partie de la connaissance dans un autre contexte, par exemple pour une installation différente.

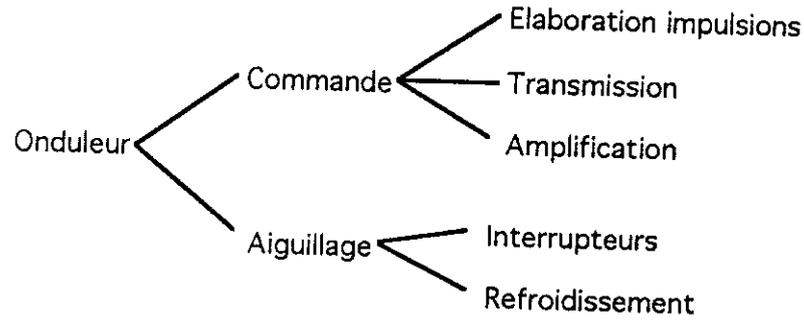


Figure II.3 Exemple de décomposition fonctionnelle

3.1.2 Décomposition structurelle

On découpe l'objet à diagnostiquer en sous-objets de plus en plus élémentaires (figure II.4). C'est une décomposition naturelle pour un expert car elle met en évidence des structures physiques réelles. Le convertisseur, par exemple, peut être considéré comme l'association de six bras qui sont chacun composés de thyristors, de dispositifs d'allumage, de radiateurs, etc... Une telle décomposition est souvent utilisée pendant la recherche d'un composant défaillant, par exemple, lorsqu'on cherche un contacteur défaillant dans un circuit électrique.

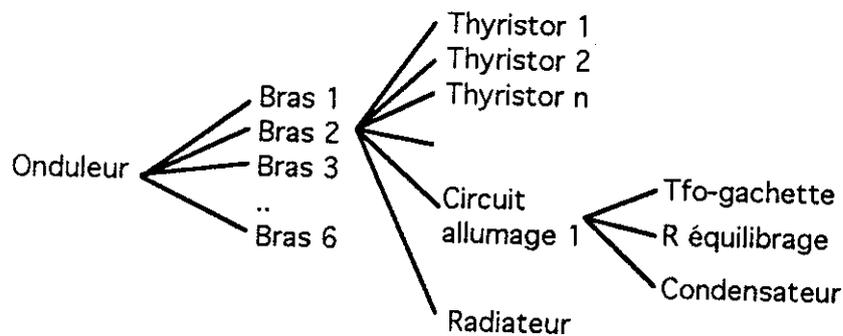


Figure II.4 Exemple de décomposition structurelle

3.1.3 Décomposition hybride

C'est un mélange des deux décompositions précédentes (figure II.5). On met ainsi en évidence l'association entre des fonctionnalités et des structures physiques. Cette décomposition est difficile à mettre en œuvre car il n'y a pas de façon unique pour l'établir. Cependant, par rapport à une décomposition fonctionnelle, elle est plus naturelle car elle aboutit forcément aux entités structurelles à un niveau de décomposition. Ainsi, comme elle mélange fonction et structure, elle nous semble la plus appropriée pour structurer les données utilisées dans la démarche des experts. En effet, ces derniers ont tendance à tester

l'état des fonctions (par exemple, la commande ne marche pas) puis des sous-fonctions (par exemple, défaut de transmission d'impulsions) et ensuite à basculer sur la représentation structurelle pour identifier le composant en panne (par exemple, un câble entre l'électronique et un relais déconnecté ou mal isolé).

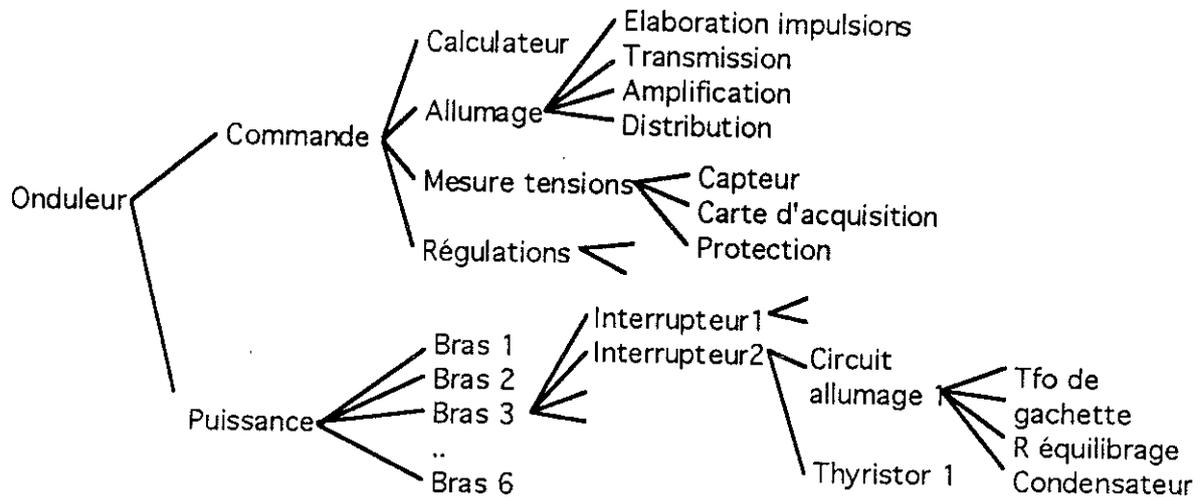


Figure II.5 Exemple d'une décomposition hybride

En résumé, les experts raisonnent de plusieurs façons, c'est à dire structurellement et fonctionnellement. Leur raisonnement favorise parfois une représentation plus fonctionnelle et parfois plus structurelle [GEN 91].

Ainsi, nous avons retenu une décomposition hybride qui commence par une décomposition du système en modules fonctionnels qui sont eux-mêmes décomposés en d'autres modules fonctionnels ou en composants mettant en œuvre les fonctions correspondantes. Cette décomposition nous semble au mieux correspondre à l'image qu'ont les experts des entraînements. En effet, ils retrouvent les fonctions qu'ils ont l'habitude de mettre en cause et de vérifier (l'allumage de l'onduleur, la régulation du courant, etc...) ainsi que les composants qui servent à les mettre en œuvre. Les composants représentent le niveau le plus bas de la décomposition et sont considérés comme des éléments interchangeables. De ce fait, ils déterminent le niveau de détail du diagnostic, par exemple, si celui-ci s'arrête pour un interrupteur au niveau du circuit d'allumage ou au niveau du transformateur de gachette (figure II.5).

La mise en œuvre de la décomposition de l'entraînement électrique s'est avérée délicate car il n'y a pas une façon unique à la réaliser. Souvent on est tenté de mélanger fonction et structure, d'où la difficulté pour modéliser leurs interactions. En fait, il peut y avoir des

connaissance statique; comme par exemple, les valeurs et les allures caractéristiques des principales grandeurs manipulées en plusieurs points de référence du fonctionnement.

Toutes ces connaissances factuelles concernent plutôt l'installation particulière à diagnostiquer. Par contre, d'une certaine manière, elles sont indépendantes du savoir-faire du diagnostic et de la stratégie de procéder dans l'analyse. Donc, c'est essentiellement la connaissance factuelle qui varie selon les particularités propres de chaque installation. Pour ces raisons, elle doit être séparée de la connaissance de savoir-faire qui est plus générique et donc valable pour un certain ensemble d'installations.

3.1 Structuration

Comme les composants et les signaux peuvent être nombreux, il faut bien les structurer pour pouvoir y accéder aisément. L'intérêt de cette structuration est d'identifier des structures généralisables sur lesquels portent des règles génériques. Ceci permet de configurer le système expert de façon modulaire pour pouvoir l'adapter facilement selon l'installation qui doit être diagnostiquée.

L'analyse de la connaissance experte montre une certaine genericité des règles utilisées. Par exemple, une règle mettant en cause le refroidissement du convertisseur est générique par rapport au système de refroidissement, car elle ne présuppose rien sur sa nature (ventilateurs, circuit refroidissement à eau ou hydro-ventilé, etc ...). Par contre, une règle sur ce dispositif de refroidissement n'est générique que pour tout dispositif équivalent.

Cette structuration doit être la plus "naturelle" possible, c'est-à-dire elle doit inclure les notions de sous-systèmes que les experts ont l'habitude de manipuler (par exemple les sous-systèmes allumage, excitation, convertisseurs côté réseau et machine, ...). Dans le souci de trouver la structuration la mieux adaptée à l'approche des experts, nous avons envisagé les décompositions suivantes.

3.1.1 Décomposition fonctionnelle

Chaque sous-objet devient une des fonctionnalités que doit remplir l'objet (figure II.2). Par exemple, pour le convertisseur, on trouve la fonction commande, la fonction puissance, etc... Il existe des procédures de tests qui permettent de vérifier le bon état de certaines fonctions. Cette décomposition correspond à l'approche des experts qui ont tendance à déduire le bon état des sous-fonctions et du matériel les mettant en œuvre si une fonction marche bien. Si par exemple (figure II.3), la commande a été testée avec succès, les sous-fonctions transmission et amplification des impulsions ainsi que les câbles et les transformateurs d'allumage fonctionnent bien.

fonctions qui sont mises en œuvre par plusieurs composants, mais également des composants participant à plusieurs fonctions, de sorte qu'on met l'accent soit sur l'aspect fonctionnel, soit sur l'aspect structurel, soit en superposant ces deux aspects.

Le but de la décomposition est de fournir un support pour le raisonnement par une bonne structuration des données. Ainsi, elle peut faciliter la formulation de règles de diagnostic, mais seule, elle n'est pas suffisante pour faire du diagnostic (cf. exemple annexe I.2.2).

La réalisation de la décomposition des entraînements que nous avons étudiés, s'est faite par la définition de structures de base pour représenter les entités fonctionnelles et structurelles que nous avons jugées importantes pour les tâches du diagnostic. Ces structures servent à contenir de la connaissance sur les éléments qu'ils représentent ainsi que des informations sur les relations entre les uns et les autres.

Dans la suite, nous allons présenter comment nous avons trouvé et modélisé les entités permettant de mettre en œuvre une décomposition des entraînements.

3.2 Définition des entités d'entraînements

La définition de modules fonctionnels et de composants pour la structuration des entraînements a nécessité l'analyse d'exemples de schémas d'installations et la connaissance des concepts manipulés dans le raisonnement des experts.

Ainsi, dans un premier temps, nous avons identifié des structures et des types de composants apparaissant souvent en analysant des schémas d'installations différentes. Ensuite, ils ont été formalisés, pour permettre de représenter les éléments fonctionnels et structurels de l'entraînement.

Les différents éléments identifiés ont été classés d'une manière hiérarchique dans une taxinomie [VOG 88], c'est à dire une classification. Cela permet de bien gérer les propriétés de chaque type d'élément tout en profitant des définitions de propriétés communes entre eux (figure II.6).

3.2.1 Les composants

Les protections sont un bon exemple pour l'identification de types de composants. Ici, la fonction remplie consiste à couper un circuit suite à un courant trop important. Le courant de coupure est alors une propriété commune de toutes les protections ainsi que l'état qu'elles prennent à un instant donné. Puis, il existe plusieurs types de protections qui varient selon

leur technologie, rapidité, fonctions supplémentaires (par exemple, un sectionneur-fusible), représentation dans le schéma et la manière dont elles risquent de tomber en panne.

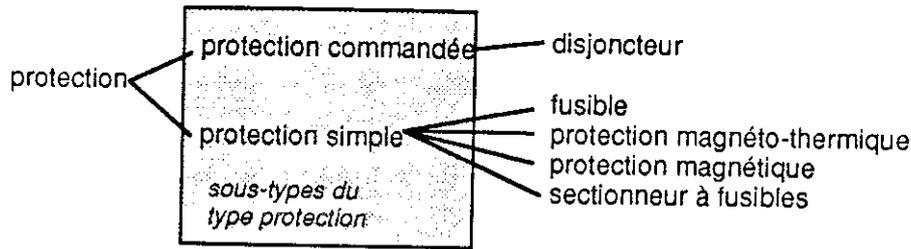


Figure II.6 Exemple de différents types de composants

Remarque: La création de types d'éléments abstraits *protection* et de types intermédiaires *protection-commandée* et *protection-simple* (figure II.6) n'a pas forcément un sens physique mais permet d'éclaircir les critères de classification et facilite la création ultérieure de nouveaux types.

Bien qu'il soit possible de créer des éléments de type *protection* ou des sous-types (figure II.6), nous recommandons d'utiliser les types les plus précis pour décrire l'installation. Ceci permet d'améliorer la précision du diagnostic.

L'intérêt des types abstraits de la classification consiste en leur utilisation dans le raisonnement. Ainsi, il est possible d'écrire des règles génériques qui portent sur le diagnostic d'une entité dans son environnement.

Exemple: *Si le ventilateur ne tourne pas, alors vérifier l'état de sa protection*

Dans cette règle il n'est pas nécessaire de prendre en compte tous les détails de la protection. Ceux-ci seront nécessaires par contre, lorsqu'il s'agit d'approfondir le diagnostic sur les causes d'un éventuel déclenchement dans une phase de raisonnement ultérieure.

3.2.2 Les modules structurels

Un exemple d'un **module** structurel est le module auxiliaire dans la figure II.7. Cette structure sert à représenter l'environnement dans lequel se trouvent les composants.

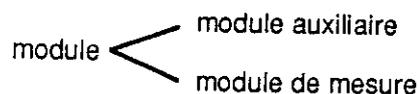


Figure II.7 Exemple de différents types de modules structurels

Son rôle consiste à regrouper ces composants dans un moule commun. Les exemples que nous allons présenter par la suite servent à décrire l'environnement dans lequel se trouvent les composants d'une installation. Cet environnement peut être caractérisé par la structure dans laquelle les composants sont intégrés et par la fonction à laquelle ils participent. La distinction n'étant pas toujours facile à faire, nous avons procédé comme nous l'avons jugé utile pour le diagnostic. L'intérêt de ces concepts est de définir et de figer la sémantique pour l'écriture des règles du diagnostic et de fournir des moules pour la création des objets qui représenteront l'installation par la suite.

Comme on le verra par la suite, les éléments seront reliés entre eux par des références, formant ainsi un espèce de réseau sémantique. Par exemple, le *module auxiliaire* est la traduction d'une partie typique du schéma électrique de connexion d'un actionneur (figure II.8).

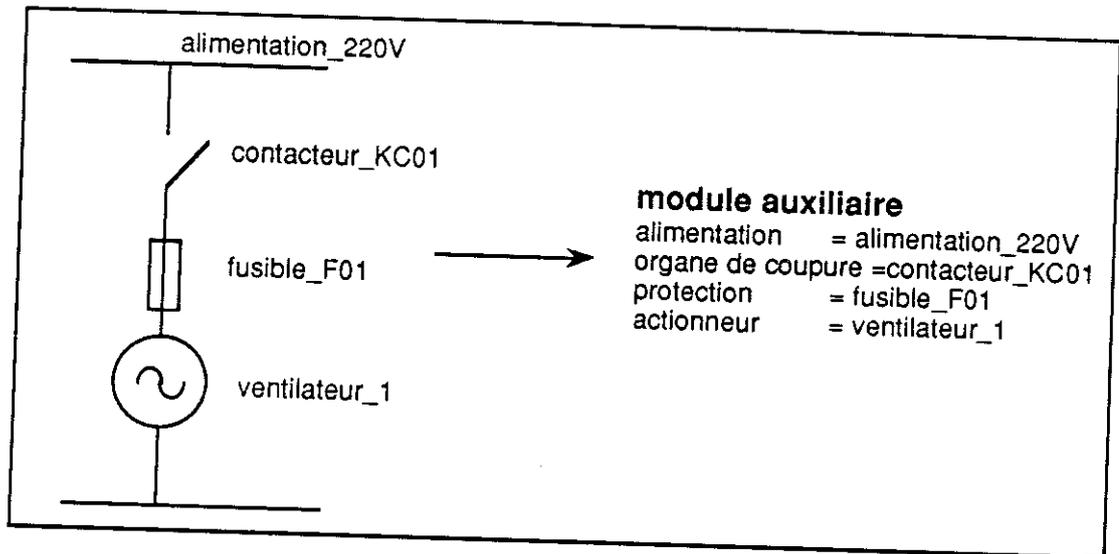


Figure II.8 Exemple de propriétés d'un module structurel

Les propriétés du module (*alimentation, protection, etc...*) sont des références à d'autres composants ou modules; par exemple, le module auxiliaire a une alimentation (qui constitue un autre module structurel), un contacteur (qui est un composant de type organe de coupure), une protection et un actionneur (figure II.8).

La définition des propriétés des modules permet de préciser la nature exacte de chacune d'elles. Par exemple, le courant de coupure d'une protection est un nombre entier et se mesure en Ampère, de même l'alimentation d'un module auxiliaire est une référence sur un module de type alimentation synthétisant ainsi une petite partie d'une installation. Les connexions, câbles, fils de commande, etc ne sont pas représentés explicitement dans les

modules, car leur modélisation explicite alourdirait inutilement la mise en œuvre du système expert. Pour ces détails importants pendant le diagnostic, les schémas électriques sont disponibles sur papier. Ils pourraient aussi être intégrés dans une base de données d'images si besoin est [WOD 90].

3.2.3 Les modules fonctionnels

Un autre type de structures sont les **modules fonctionnels** qui représentent les **sous-systèmes** d'un équipement. Ils ne reflètent pas forcément une partie du schéma électrique comme les modules structurels mais regroupent les entités participant à la fonction du module. Par exemple, le système d'excitation de la figure II.9 regroupe plusieurs modules dont l'alimentation, le convertisseur alimentant l'excitatrice, l'excitatrice, le convertisseur alternatif-continu pour fournir le courant rotorique, le rotor. De même, le module excitation contient une référence sur le courant mesuré en entrée du convertisseur.

excitation
 alimentation =
 commande du gradateur =
 pont gradateur =
 mesure de courant =
 courant du gradateur =
 excitatrice =
 pont à diodes tournantes =
 bobines rotoriques =

Figure II.9 Exemple de propriétés d'un module fonctionnel

3.2.4 Propriétés

Les propriétés servent à représenter les caractéristiques des différents éléments dont est composé un entraînement électrique. Il y a deux sortes de propriétés :

- les propriétés communes à un type d'élément avec une **valeur liée au type** (figure II.10), par exemple, tous les contacteurs auront les mêmes modes de défaillance.
- les propriétés communes à un type d'élément avec une **valeur propre pour chaque élément** (figure II.11), par exemple, l'état signalé actuel varie d'un contacteur à l'autre.

Nous avons rattaché toutes les connaissances relatives au diagnostic pour un type de composant aux propriétés avec valeur liée au type, par exemple, les observations et les modes de défaillance possibles et des connaissances opératoires pour le diagnostic et pour la réparation (figure II.10)

propriété	explication
modes de défaillance	Une liste des modes de défaillances caractéristiques d'un composant ou d'un système. Un contacteur, par exemple, peut avoir les modes suivants: (bloqué ouvert, bloqué fermé, bobine grillée, défectueux, ..)
procédure de vérification	documentation d'une procédure de test pour vérifier l'état d'un contacteur
source de connaissance hypothèses-causes source de connaissance hypothèses-effets source de connaissance évaluation-hypothèses source de connaissances de transfert-signalisations source de connaissances de maintenance	références sur des connaissances opératoires utilisées pendant le raisonnement

Figure II.10 Exemple de propriétés communes à un type d'élément

Le but des définitions ci-dessus est de structurer les données représentant l'installation à diagnostiquer. Ceci est nécessaire pour deux raisons: d'une part, ces définitions fournissent les concepts sur lesquels s'appuie l'écriture des règles de diagnostic. D'autre part, elles serviront de moule pour la création des éléments décrivant les installations. Comme on verra dans le chapitre suivant, des objets informatiques sont créés selon les structures définies ici. La mise en œuvre des définitions des différentes structures ainsi que celle des éléments représentant les installations, se fait à travers le modèle Objet.

propriété	explication
état-signalé	propriétés caractérisant l'état. Présence d'une alarme sur un contacteur
état-fonctionnel	(ouvert/fermé), pour affichage et interfaçage avec le système d'acquisition
état-conclu	pour la communication avec l'utilisateur
modes de défaillance observés modes de défaillances suspectés observations hypothétiques	des propriétés utilisées pour le raisonnement: observations et hypothèses pour la communication avec l'utilisateur conséquences hypothétiques d'observations
localisation	une référence sur la localisation où se trouve un contacteur (armoire-puissance 1, tableau T2)
actions préventives	une liste de références sur des actions préventives
accessibilité	une note qui peut être exploitée par la stratégie
surveillance-état	une référence sur un capteur indiquant sa position actuelle (fin de course, etc)
systèmes	une référence sur le module auquel un contacteur est associé
alimentation	une référence sur une alimentation ou signal qui contrôle un contacteur, propriété d'un <i>organe de coupure électrique</i>
composants-aval	référence sur les composants et systèmes alimentés, propriété d'une <i>alimentation</i>
etc ...	

Figure II.11 Exemples de propriétés particulières aux différents éléments

3.3 Le modèle objet

Pour mettre en œuvre notre décomposition, nous nous sommes appuyés sur le modèle objet [MEY 88]. Ce formalisme se prête très bien à la représentation des connaissances factuelles. Celui-ci est particulièrement intéressant quand il s'agit de développer des logiciels réutilisables et évolutifs. Il est, par exemple, possible d'ajouter de nouvelles propriétés ou caractéristiques ou de nouvelles fonctionnalités aux objets sans grande modification dans un programme existant. La structure modulaire des objets facilite leur utilisation dans un contexte différent ce qui capitalise l'effort investi dans leur développement. Par la suite, nous présentons brièvement quelques concepts inhérents au modèle objet utilisé.

Lorsqu'on parle de décomposition objet, on fait intervenir la notion de classe. Une **classe** est un concept générique qui permet de modéliser un ensemble d'éléments ayant la même structure, les mêmes propriétés et caractéristiques. Ces dernières constituent les **champs** d'une classe. Un élément réel d'une telle famille est représenté par un **objet**, appelé aussi une **instance** de la classe correspondante (figure II.12). L'instance a les mêmes champs que sa classe, remplis de valeurs propres à l'objet réel. Par exemple, l'élément *FD035_2* (figure II.12) est un objet de la classe des protections. La valeur de ces champs permettent de le caractériser (par exemple, son état actuel, son courant de coupure et le système dans lequel il se trouve).

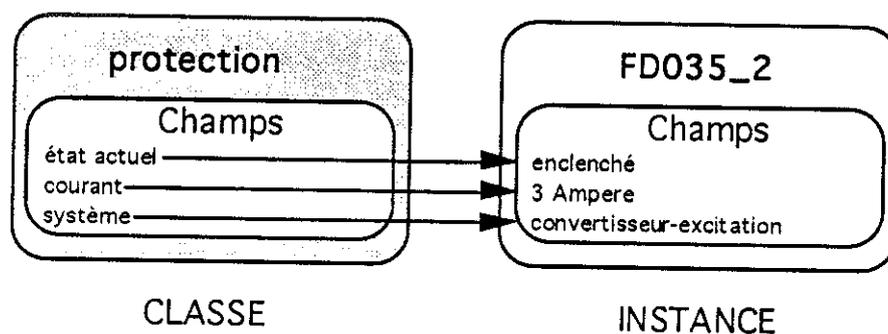


Figure II.12 Le concept de classe

Les objets sont aussi capables de répondre à certains **messages** exprimant leur comportement vis-à-vis d'une sollicitation extérieure. Par exemple, on peut demander à une protection de mettre à jour son image dans l'interface graphique en lui envoyant le message *mettre à jour-graphique* (figure II.13). Les messages servent à invoquer les fonctions qui sont définies pour les classes, appelées "**méthodes**" dans la terminologie informatique.

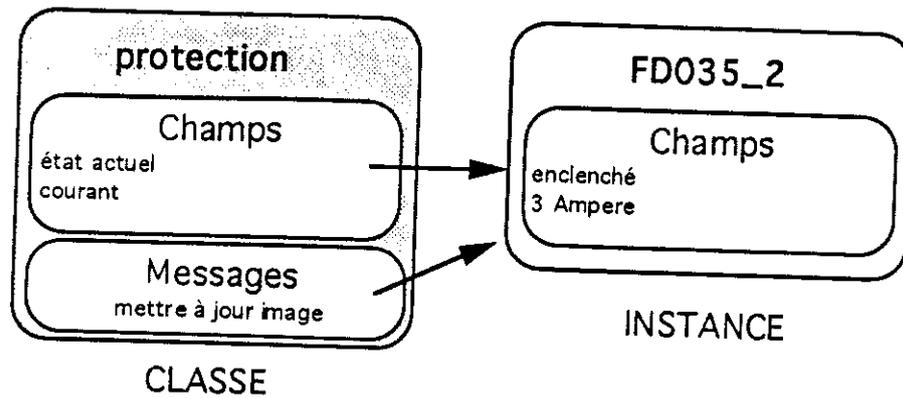


Figure II.13 Le concept de messages

Les méthodes sont associées aux classes et auront la même définition pour toutes les instances d'une classe, mais fourniront des réponses propres à chacune de ces instances.

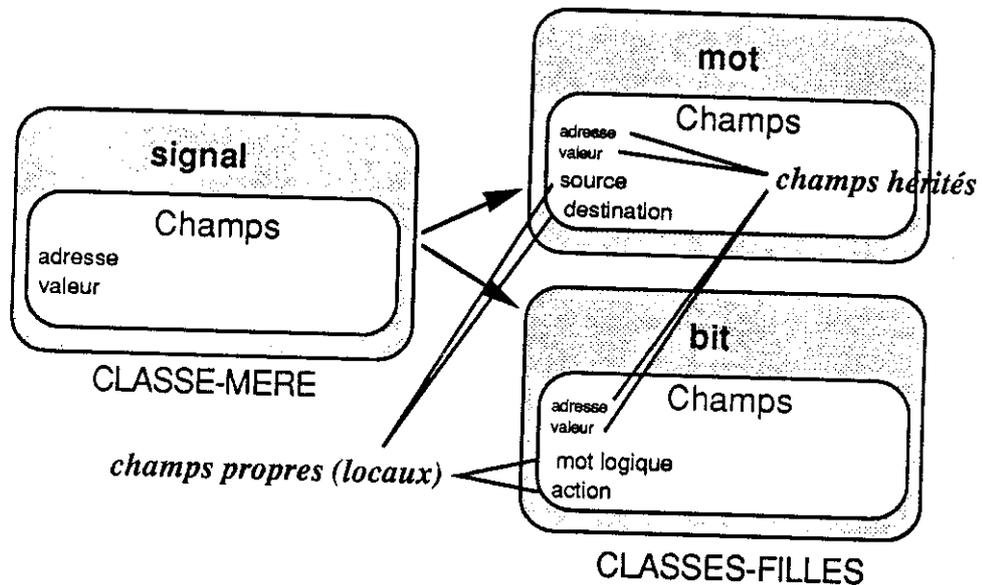


Figure II.14 Le concept d'héritage

Les différentes classes peuvent être structurées d'une manière hiérarchique ce qui fait partager aux classes-filles les propriétés (champs et messages) des classes-mères (concept d'héritage, figure II.14). Par exemple, les classes bit et mot sont des sous-classes de la classe signal. De ce fait elles disposent des mêmes champs que la classe signal et elles savent répondre aux mêmes messages que celle-ci. Cependant, il est possible de définir différentes méthodes pour les classes-filles afin de leur associer un comportement spécifique. Ainsi, on peut avoir un comportement différent en envoyant le message *afficher_valeur*, par exemple, à des instances des classes signal, mot ou bit. Dans ce cas, les méthodes différentes peuvent consister à imprimer la valeur du signal dans le terminal, à afficher la valeur d'un mot à l'aide d'une jauge et à afficher une lampe allumée ou éteinte en fonction de l'état d'un bit.

3.4 Décomposition Objet de l'entraînement

Le modèle objet permet de mettre en œuvre les structures décrites ci-dessus sous forme de classes. Les éléments même des installations sont représentés par les objets (instances des classes). Le concept d'héritage permet de regrouper les propriétés communes à plusieurs types d'éléments dans une classe générale que nous appelons *objet-installation*. Ainsi, les classifications de composants et de modules (cf. chapitre II.3.2) se traduisent naturellement en classes et sous-classes, connectées par des liens d'héritage, c'est-à-dire des relations mère-fille. Ainsi, les lignes dans les figures II.6 et II.7 du chapitre II.3.2, par exemple, sont des liens d'héritage.

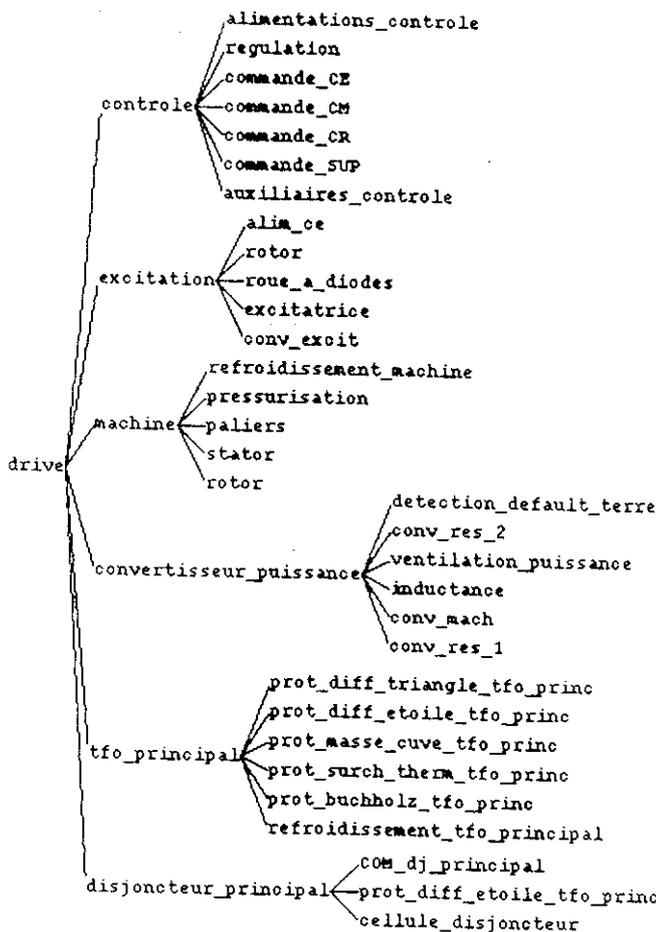


Figure II.15 Exemple de décomposition d'un entraînement

La relation entre les objets d'une installation se fait par des liens de décomposition qui traduisent une référence sur d'autres objets. Par exemple, les champs *système* dans

l'exemple de la figure II.12 et *mot logique* et *source* (figure II.14) servent à mettre en œuvre ce type de lien. Ils prennent comme valeur une référence sur l'objet auquel un objet est relié.

Ainsi, une installation d'entraînement peut se décomposer comme le montre la figure II.15.

La figure II.15 ne montre que les premiers niveaux de la décomposition de l'entraînement étudié. Chaque sous-système continue à se décomposer en sous-ensembles jusqu'à ce qu'on arrive aux composants et aux signaux associés aux capteurs et signalisations, comme l'illustre la figure II.16 pour la décomposition de la machine.



Figure II.16 Exemple de décomposition d'une machine

4 Modélisation des connaissances déductives

Les connaissances déductives expriment le savoir-faire des experts. Elles englobent des relations entre les symptômes observés et leurs causes possibles, et inversement, les conséquences possibles de certains défauts. Le raisonnement peut changer entre des phases déductives (des symptômes vers les causes) et abductives (des causes vers des symptômes) pour atteindre un aperçu global d'une situation de défaut.

4.1 Types de raisonnement

Nous avons distingué les deux types principaux de raisonnement pour permettre l'exploitation des connaissances. Ainsi, il devient possible d'intégrer des connaissances partielles (cf. § II.2.1) sur le diagnostic dans le système expert, en les découplant de la connaissance stratégique permettant d'exécuter les règles.

4.1.1 Raisonnement "par défaut"

Un raisonnement par défaut, qui consiste à relier directement les symptômes observés à un certain nombre de causes possibles. Par exemple, l'apparition d'une surintensité dans le convertisseur peut avoir beaucoup de causes différentes:

- problème d'allumage
- problème de réseau de puissance
- problème d'un thyristor
- problème de commande (manque de mesure du retour du courant)
- etc..

4.1.2 Raisonnement causal

Le raisonnement causal est basé sur une compréhension profonde du fonctionnement de l'installation et l'interaction entre ses composants. Lorsque plusieurs hypothèses de diagnostic existent en même temps, ce raisonnement doit permettre de diriger la recherche vers l'hypothèse la plus pertinente en s'appuyant sur les autres observations disponibles. Un autre but de ce raisonnement est de fournir des éléments pour expliquer le défaut a posteriori. Nous allons l'illustrer autour de l'exemple suivant:

Si les fusibles du transfo-synchro sont fondus (signalisation par un capteur)
 alors manque tensions machine
 alors manque retour fréquence
 alors commande allumage incohérente possible du convertisseur côté machine
 alors allumage incohérent possible sur convertisseur côté machine

alors allumage diamétral du convertisseur côté machine possible
alors détection d'une surintensité

Ainsi, il est possible qu'un défaut intrinsèque d'un transformateur de mesure provoquant la fusion d'un fusible, se manifeste par l'apparition d'une surintensité.

Malheureusement, ce raisonnement n'est pas toujours valable: il peut bien y avoir allumage incohérent sur le convertisseur côté machine sans qu'il y ait apparition d'une surintensité. Toutefois, ce raisonnement est très intéressant parce qu'il permet de ne pas poursuivre dans un premier temps les hypothèses a priori sur la surintensité, mais de focaliser directement sur la cause à l'origine. Il permet aussi de donner une explication globale et cohérente de l'incident, ce qui est un aspect important pour aider l'utilisateur du système expert dans sa compréhension.

4.2 Intégration des types de raisonnement

Nous avons fait collaborer ces deux types de raisonnement dans notre système expert. En fait, dans le domaine des entraînements électriques, il n'est pas possible d'intégrer a priori tous les raisonnements causaux dans la base de connaissances. Cependant, dans les cas où le raisonnement causal n'aboutit pas à une conclusion (à cause de l'incomplétude de la base de connaissances), il est tout de même intéressant de pouvoir proposer un certain nombre d'hypothèses pour aider l'utilisateur.

Les deux types de raisonnement sont mis en œuvre par le formalisme des règles de production du premier ordre de la forme

```

Si <condition>
    et <condition>
    et ...
alors <action>
    et <action>
  
```

Ce formalisme présente l'avantage d'être bien compris par les experts et surtout de permettre l'expression aisée des règles utilisées dans le diagnostic. Cependant, ce formalisme est trop général pour permettre un contrôle du raisonnement. Ainsi, il faut éviter de mélanger les deux types de raisonnement pour éviter des rebouclages imprévisibles et ainsi des résultats inexploitable.

4.3 Conseils méthodologiques

Pour cette raison, nous proposons une méthodologie qui doit guider le concepteur du système dans la formulation des règles de diagnostic. Le but de ces conseils méthodologiques est d'assurer la compatibilité des règles entre elles et de permettre l'implantation explicite d'une stratégie de diagnostic. Ces conseils sont les suivants:

1. On raisonne explicitement sur des observations existantes pour les différents entités de l'entraînement électrique
2. Les hypothèses sont, comme les observations, modélisées d'une manière explicite par des objets.
3. Pour distinguer le raisonnement par défaut du raisonnement causal il existe deux types d'hypothèses différents :
 - a) les *hypothèses sur les causes* d'une *observation* qui sont utilisées dans le raisonnement par défaut
 - b) les *hypothèses sur les effets* d'une *observation* qui sont utilisées dans le raisonnement causal.
4. Le raisonnement par défaut permet de créer des hypothèses sur les causes suite à des observations établies.
5. Le raisonnement causal permet de créer des hypothèses sur les effets. Une hypothèse sur un effet exprime le fait qu'on s'attend à une observation en conséquence d'une autre observation.

4.3.1 Création d'hypothèses sur les causes

Il est possible de créer plusieurs hypothèses suite à une observation en appliquant plusieurs règles indépendantes (figure II.17), comme le montre l'exemple suivant:

- R1: SI observation vibrations sur paliers
alors créer hypothèse-cause desalignements de l'arbre
- R2: SI observation vibrations sur paliers
alors créer hypothèse-cause balourd-thermique sur rotor

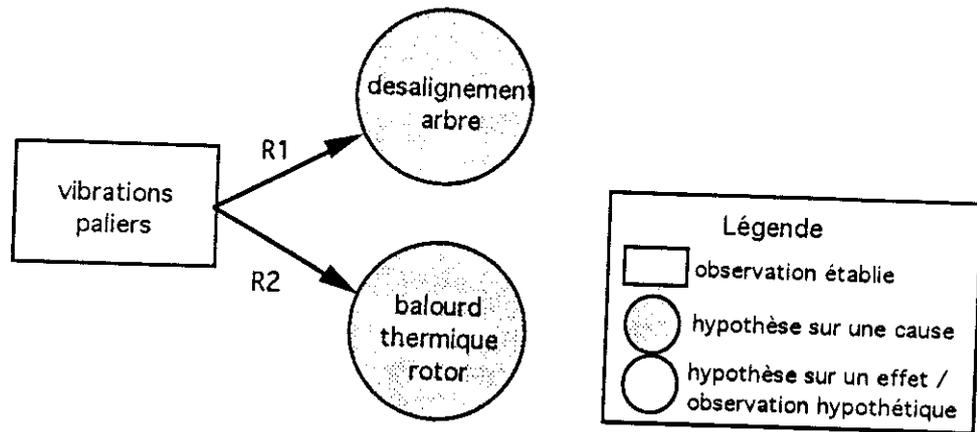


Figure II.17 Principe de création d'hypothèses sur les causes d'une observation anormale

De plus, nous déconseillons de créer des hypothèses sur les causes à partir d'autres hypothèses sur les causes (figure II.18) car cela peut entraîner la création d'un nombre d'hypothèses important et difficilement contrôlable. Comme les hypothèses sont communiquées à l'utilisateur celui-ci risque d'être mal guidé par un choix trop grand d'hypothèses à vérifier. Outre cette considération pratique, ce conseil correspond bien à la démarche des experts qui consiste à se convaincre d'abord de la pertinence d'une hypothèse avant d'essayer de la affiner par des sous-hypothèses.

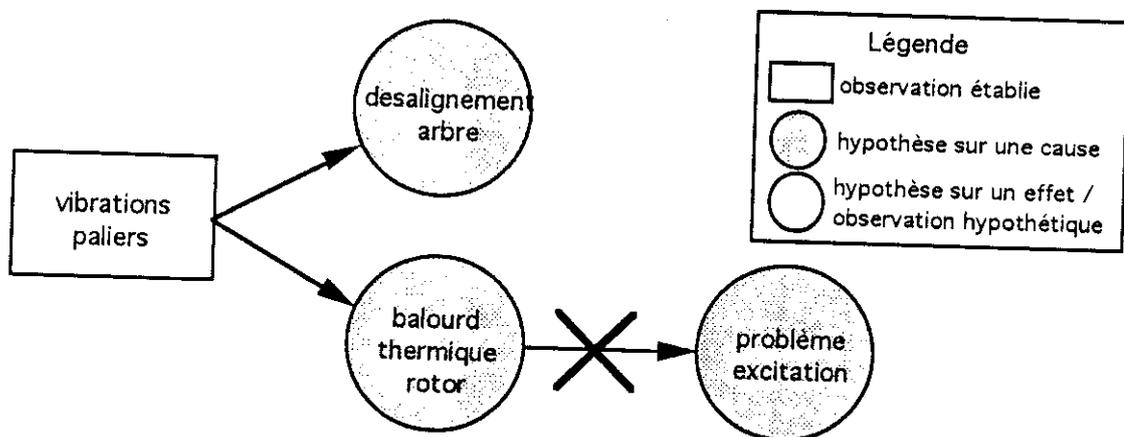


Figure II.18 Conseil pour la création des hypothèses sur les causes

Continuons à discuter notre exemple. Une fois l'hypothèse-cause *balourd-thermique* sur le rotor confirmée, elle va se transformer en observation. Par la suite l'émission de l'hypothèse-cause sur l'excitation peut avoir lieu (figure II.19):

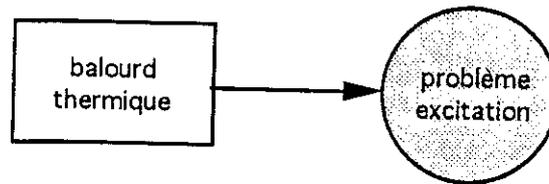


Figure II.19 Création d'observations

En effet, il est possible de créer des observations suite à d'autres observations. Ceci permet de propager une observation sur un composant dans l'objet qui modélise l'environnement auquel il appartient et d'exécuter éventuellement d'autres règles de diagnostic.

4.3.2 Création d'observations hypothétiques

Le raisonnement causal permet de créer des observations hypothétiques, c'est à dire des hypothèses sur les effets suite à des observations établies ou suite à d'autres hypothèses sur les effets.

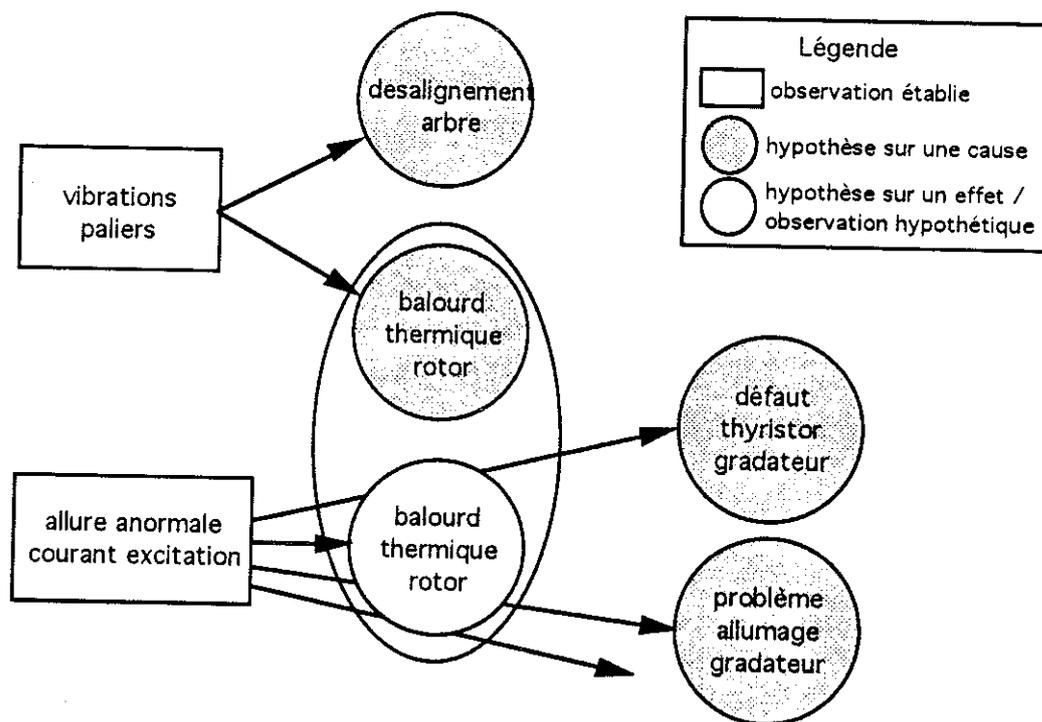


Figure II.20 principe de utilisation d'observations hypothétiques

L'idée des hypothèses sur les effets consiste à exploiter leur présence pour la stratégie du raisonnement. La présence simultanée, par exemple, d'une observation ou d'une hypothèse-cause *balourd thermique* et d'une hypothèse-effet *balourd thermique* sur rotor (figure II.20), peut, dans certaines conditions, permettre l'interprétation que l'observation

vibrations paliers est une conséquence de l'observation *allure anormale* sur *courant excitation* qui est à la base de l'hypothèse-cause *balourd thermique* sur *rotor*. Dans ce cas, il conviendrait de tester d'abord les hypothèses-causes liées à l'observation initiale *allure anormale* avant de s'occuper des effets secondaires...

En raison de l'interprétation des hypothèses-effet par la stratégie choisie, nous déconseillons de créer des hypothèses-effets suite à des hypothèses sur les causes. En effet, bien qu'il soit conceptuellement intéressant de faire un raisonnement hypothétique, ceci poserait des problèmes de gestion d'un grand nombre d'hypothèses-effets et d'interprétation différente selon l'origine des hypothèses-effet.

4.3.3 Discussion

Ce traitement peut paraître lourd mais il représente une solution élégante pour décider de la causalité d'événements rapides tels qu'ils arrivent après un incident dans un convertisseur. En fait, il n'est pas possible d'enregistrer la séquence temporelle correcte des alarmes déclenchées pour les exploiter dans le bon ordre.

Une autre solution consisterait à créer des règles traitant toutes les combinaisons des événements signalés possibles pour déterminer quelles hypothèses créer pour chaque cas. Cependant, ces règles sont difficiles à trouver et trop limitatives puisqu'il n'est pas possible de prévoir tout ce qui peut se passer. Aussi, il n'y aura pas d'explication cohérente de l'incident.

Remarque: L'hypothèse sous-jacente de notre démarche est que la plupart des défauts sont des défauts simples. Les défauts multiples indépendants sont assez peu probables. Par contre, il peut y avoir des défauts en cascade, par exemple, des défauts naissants non encore détectés qui, lorsqu'ils s'aggravent, provoquent un deuxième défaut, et ce n'est que celui-ci qui se manifeste clairement par le déclenchement d'une protection.

Pour pouvoir détecter des défauts naissants (vieillessement, usure, conditions d'exploitation malsaines, etc ...) il faut intégrer dans le système expert des méthodes de surveillance plus fines que les seuils grossiers des systèmes de protection utilisées à l'heure actuelle. L'objectif de ces méthodes est de créer des observations nécessaires pour que ces phénomènes puissent être interprétés par les règles. Par exemple, il serait possible d'analyser cycliquement l'allure d'un courant en la comparant à une allure de référence ou de prendre en compte le résultat d'une méthode de redondance analytique [BER 93b] pour détecter des anomalies avant que l'incident se soit suffisamment aggravé pour faire déclencher une protection.

4.4 Structuration des connaissances déductives

Le raisonnement décrit ci-dessus est mis en œuvre grâce au formalisme des règles de production. En effet, le nombre de règles utilisées pouvant devenir important, il est nécessaire de les structurer comme cela a été fait pour les composants des entraînements.

Une bonne structuration des règles est importante pour deux raisons. Premièrement, elle permet de mieux retrouver les règles recherchées et elle facilite l'ajout de nouvelles règles. Deuxièmement, elle permet de mettre en place une stratégie pour appliquer les règles. Le but de cette stratégie est de gérer intelligemment l'application des règles de diagnostic. Elle décide, quelles règles il faut appliquer à un moment donné tout en évitant d'en exécuter inutilement. Pour ces raisons, nous avons regroupé les règles en paquets pour former des **sources de connaissance** [GEN 91].

La formation de ces sources de connaissance s'appuie sur deux critères: le type de connaissances et le type d'objet concerné par la source de connaissance à créer.

4.4.1 Structuration selon le type de connaissance

L'apport d'une source de connaissance pour l'avancement du diagnostic est désigné par le type des règles qu'elle contient. Ainsi, nous distinguons six types de règles selon la connaissance qu'elles expriment :

a) Règles de transfert de signalisations

Ces règles constituent l'interface entre le système d'acquisition de signaux (capteurs, protections, signalisations d'état, etc...) et le raisonnement. Elles traduisent les signaux bruts des signalisations associées aux composants en observations explicites pour le raisonnement.

Remarque: Ceci permet de dissocier le raisonnement du système d'acquisition de données d'un entraînement particulier. Les observations du procédé peuvent ainsi être fournies par l'interprétation de signaux (par les règles), mais aussi "à la main" par l'utilisateur du système expert.

b) Règles de création d'hypothèses sur les causes

Ces règles permettent de mettre en œuvre le raisonnement "par défaut", décrit ci-dessus (cf. § II.4.1.1). Leur rôle consiste à créer des hypothèses sur les causes possibles des observations anormales perçues.

c) Règles de création d'hypothèses sur les effets

Ces règles mettent en œuvre le raisonnement causal dans le but de construire des relations de cause à effet entre les différentes observations anormales apparues. Elles créent des observations hypothétiques, conséquences possibles des observations établies. La vérification de concordances entre observations, observations hypothétiques et hypothèses est pris en charge par la stratégie décrite dans le système expert de diagnostic.

d) Règles d'évaluation d'hypothèses

Ces règles ont pour mission d'évaluer des hypothèses pour les confirmer ou les infirmer. Ainsi, elles permettent de décharger l'utilisateur de ces vérifications. Mais ces règles sont rares dans notre système expert car la plupart des hypothèses nécessitent une action de vérification par l'utilisateur qui est difficilement automatisable.

Cependant, il est parfois possible de déduire la validité d'une hypothèses à partir des informations disponibles pour notre système expert.

e) Règles de maintenance

Ces règles interprètent les causes identifiées et proposent des actions de maintenance en fonction du mode de défaillance constaté.

f) Règles stratégiques

Les règles stratégiques ont un rôle particulier dans le raisonnement. Elles décident quelles sources de connaissance sont à appliquer à un instant donné mettant en œuvre ainsi la stratégie du raisonnement.

4.4.2 Structuration selon le type de l'entité

Les connaissances qui concernent un type de matériel sont regroupés en sources de connaissance qui lui sont spécialisées. Ceci a permis d'établir une stratégie pouvant mieux filtrer les règles qu'elle cherche à appliquer en évitant des essais inutiles d'application de règles aux éléments non concernées.

Les différentes sources de connaissances sont reliées aux définitions des classes comme le montre l'exemple de la figure II.21. L'accès aux sources de connaissances se trouve ainsi facilité pour tous les éléments d'une installation. Il suffit, par exemple, de consulter le champ *SdC_hypothèses_causes* de n'importe quel objet protection pour connaître la source de connaissance spécialisée pour les protections, c'est-à-dire l'objet *CHC_protect* dans ce cas.

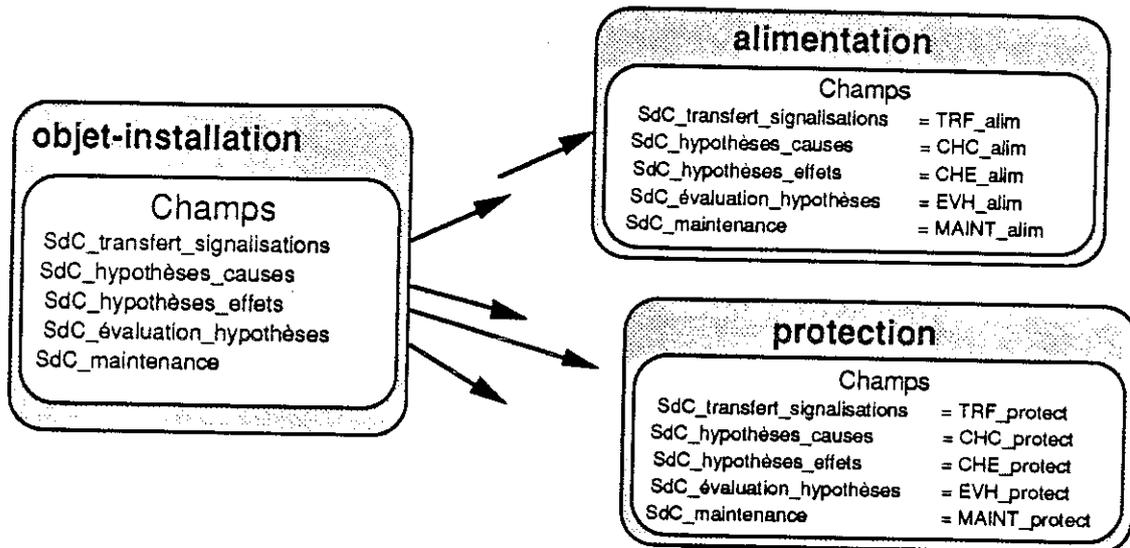


Figure II.21 Exemple d'association de sources de connaissance (SdC..) aux classes

Les sources de connaissances sont donc créées suivant les deux critères précédents (type de connaissances et les classes d'objets sur lesquelles elles portent), aussi, leur nombre est important. Cependant, elles sont hautement spécialisés selon la tâche à effectuer et l'objet à traiter. Cette structuration permet donc de tenir compte de la grande variété de composants qui peut exister et des connaissances locales associées. En effet, il y a un spécialiste pour chaque action et chaque type de composant dans notre système expert.

Maintenant, nous allons présenter la connaissance stratégique qui est chargée de faire collaborer les différents spécialistes entre eux.

5 La connaissance stratégique

Après avoir modélisée l'installation à diagnostiquer et structuré les règles de diagnostic, il est nécessaire de déterminer quand et comment vont s'appliquer les règles. Il s'agit alors de simuler le raisonnement de l'expert et de faire part à l'utilisateur des conclusions obtenues.

Le but de la stratégie est de créer des hypothèses sur les causes des problèmes observés et de choisir l'hypothèse à évaluer parmi toutes les hypothèses existantes à un instant donné. C'est ainsi que la démarche du diagnostic se réalise progressivement, guidée par les observations existantes et les règles du diagnostic, contrainte par le coût de vérification des composants. La création d'une hypothèse sur la cause d'un problème est suivie par une action de vérification, c'est à dire un test qui permet de la confirmer ou de l'infirmer. Plusieurs

hypothèses peuvent coexister en même temps et donc demander des actions de vérification ou de réparation différentes.

5.1 Philosophie

Le choix de l'action à réaliser de préférence est normalement un compromis entre la pertinence de l'hypothèse à tester et le coût de vérification de celle-ci. L'objectif du raisonnement est de proposer à l'utilisateur les actions à entreprendre pour l'assister dans la recherche de la cause d'un problème. L'assistance dont peut avoir besoin l'utilisateur est difficilement prévisible. Elle dépend de plusieurs facteurs, par exemple, la connaissance que l'utilisateur a sur l'installation, la situation spécifique où intervient un défaut, la disponibilité d'outils d'analyse, etc... L'assistance et l'explication pour l'utilisateur est un problème non-trivial qui a fait l'objet de différents travaux de recherches [BAU 91] [YOO 88a] [YOO 88b], et que nous n'avons pas approfondi. Ainsi, nous proposons à la fois toutes les actions de test pertinentes, ainsi que des critères qui permettent de les trier. C'est lui qui connaît le mieux les contraintes de la situation et ce doit être lui qui décide quelle hypothèse à poursuivre et quel test à effectuer. Alors, au lieu d'insister sur la réponse d'une question, le système expert propose à son utilisateur une liste de toutes les questions pertinentes (= des hypothèses à vérifier par une action de vérification) triée selon plusieurs critères possibles, au choix de l'utilisateur. La stratégie a donc les deux aspects suivants à traiter.

5.2 Choix des règles à appliquer

Le choix des règles à appliquer s'inscrit dans le cycle de fonctionnement du système expert. Prévu pour fonctionner sur site en parallèle avec l'installation, le système effectue un cycle de tâches correspondant à différentes phases. Celles-ci sont pilotées d'une manière procédurale, c'est à dire par application séquentielle d'actions à entreprendre. Au cours de ce cycle d'actions, les règles stratégiques vont décider s'il faut appliquer les sources de connaissance. Lorsque c'est le cas, la règle stratégique utilisée fait appliquer la source de connaissance sélectionnée sur l'objet de l'installation pour lequel la condition d'application a été remplie. Voici les tâches principales du cycle de notre système expert:

- *charger-signaux*

Un nouveau jeu de données est demandé au système de surveillance. Cette tâche se charge de la communication avec le système de surveillance et de la mémorisation et de l'affichage des données récupérées.

- *déterminer-conditions-exploitation*

Cette tâche met à jour un objet synthétisant les conditions de l'exploitation actuelles à partir de certains signaux enregistrés par le système d'acquisition. L'utilisateur peut, lui

aussi modifier cet objet pour entrer ses observations sur le contexte associé au défaut (par exemple, défaut intervenant au démarrage ou par grande chaleur ...)

- *transférer observations*

Il s'agit de traduire les enregistrements de signaux bruts des capteurs en observations explicites associées aux composants de l'installation. Ceci permet de dissocier le raisonnement de l'existence de tel ou tel capteur. Les observations qui sont à la base des règles du diagnostic peuvent ainsi être rentrées aussi "à la main" par l'utilisateur. Cette tâche est remplie par une règle stratégique qui applique les sources de connaissance de transfert d'observations par les signalisations dont le signal a été activé.

- *créer-hypothèses-causes*

Pour chaque entité de l'installation pour laquelle une observation anormale existe, on applique sa source de connaissance créant les hypothèses sur les causes de l'observation possible.

- *créer-hypothèses-effets*

Pour chaque entité pour laquelle existe une observation ou une observation hypothétique, on applique sa source de connaissance créant des hypothèses sur les effets de l'observation possible.

- *évaluer-hypothèses*

Pour chaque objet pour laquelle existe au moins une hypothèse sur une cause non encore évaluée, on applique sa source de connaissance contenant des règles pour confirmer ou infirmer les hypothèses. Ceci peut éviter à l'utilisateur d'apporter lui-même la réponse à un test.

- *évaluer-résultats-vérifications*

Si, suite à une hypothèse sur un module complexe (par exemple, test d'allumage), des tests sur les composants de ce module ont été proposés, alors les règles de cette tâche vérifient si tous les tests ont été bien effectués et synthétisent le résultat pour la première hypothèse.

- *proposer-remèdes*

Pour chaque entité identifiée défaillante, on applique sa source de connaissance qui propose des actions de maintenance curatives ou correctives adaptées au mode de défaillance identifié.

- *gestion-système*

C'est une phase nécessaire pour des raisons techniques, par exemple pour libérer la mémoire de données inutiles ou pour effectuer des actions cycliques, comme celles que des rappels systématiques d'intervalles de maintenance préventive qui peuvent être définis pour les éléments des entraînements.

L'application des sources de connaissance de diagnostic aboutira à la création d'hypothèses et à la proposition d'actions de vérification et de maintenance par l'utilisateur. Ensuite, notre système expert doit trier ces actions pour lui donner une aide dans sa décision de choix.

5.3 Tri des hypothèses établies par les règles

Le deuxième aspect de la stratégie consiste à établir un ordre d'exécution dans les actions proposées suite aux hypothèses présentes à un instant donné. Ces hypothèses sont communiquées à l'utilisateur via les tests qu'elles requièrent. Ensuite, l'utilisateur peut choisir parmi différents critères de tri :

- *priorité* : critère principal qui tient compte de la pertinence de l'hypothèse. C'est un coefficient fixé arbitrairement positif ou négatif. Son élaboration suit quelques règles générales :
 - * une hypothèse évoquée par plusieurs observations et plusieurs règles est plus pertinente que celle supporté par moins d'observations.
 - * Une hypothèse créée suite à une observation secondaire est moins pertinente que celle ayant une observation primaire à son origine.
 - * Une hypothèse sur la cause qui est une conséquence possible d'une observation autre que celle à partir de laquelle elle a été émise, est moins prioritaire que les autres.

Par exemple, si il y a une hypothèse qui met en cause un élément et, en même temps, il y a une observation hypothétique sur cet élément, on peut dire qu'on s'attendait à ce que l'élément ne fonctionne pas correctement. Il faut alors s'occuper d'abord des hypothèses sur l'observation à l'origine du mauvais fonctionnement. Ceci se fait en réduisant la priorité de l'hypothèse mettant en cause le module dont sont fonctionnement anormal est une conséquence possible d'une autre observation.

- *coût/accessibilité* : critère tenant compte du coût de vérification du composant à tester, c'est-à-dire de son accessibilité. Un thyristor situé dans une armoire de haute tension, par exemple, est moins accessible que la carte élaborant les impulsions d'allumage en raison de la nécessité de consigner l'équipement avant d'accéder à la partie sous tension.
- *date de création* : critère servant pour des fins de mise au point du raisonnement. En affectant la date aux objets-raisonnement, c'est à dire les observations et hypothèses, il est possible de détruire ceux qui sont obsolètes car vieilliss.

- *date de réponse* : critère permettant de garder la trace des tests (hypothèses vérifiées) effectués.

La gestion des critères pour le tri des hypothèses se fait par les méthodes associées aux objets spécifiques au raisonnement (observations, hypothèses-causes, hypothèses-effets, tests) (cf. § II.3.3). Ceci permet de réagir instantanément pour adapter la priorité des hypothèses lorsque les données changent.

L'état d'avancement du diagnostic est documenté en permanence par les observations, les différentes hypothèses et les liaisons que la stratégie va établir entre ces dernières, lorsque les règles du diagnostic sont appliquées. Cette documentation peut, par la suite, être utilisée pour donner des explications à l'utilisateur en cours de sessions du système expert.

6 Conclusion

Nous proposons donc une structuration de la connaissance en trois catégories :

- la connaissance factuelle qui contient les données manipulées par les experts: données relatives à l'installation, ses composants, signaux, fonctions, pannes, etc... et données relatives au raisonnement, observations, hypothèses, tests.
Nous avons utilisé une décomposition structurelle et fonctionnelle hiérarchique pour les entraînements. Les installations seront modélisées à l'aide de la représentation objet: les composants sont représentés par des objets de modèles génériques (les classes).
- La connaissance déductive qui contient les règles de diagnostic et de maintenance. Ces règles sont regroupées en paquets pour former des sources de connaissances, spécialistes pour une action d'inférence et un type de composant bien défini. La structuration proposée permet d'introduire des connaissances évolutives de façon incrémentale, découplée de la manière dont elles vont être exploitées.
- La connaissance stratégique qui modélise le processus de diagnostic. Elle regroupe deux aspects différents: l'organisation de l'application des sources de connaissances et le tri des actions proposées à l'utilisateur selon plusieurs critères à son choix.

Nous avons donc dû trouver une architecture informatique, capable de prendre en compte tous ces différents aspects et de respecter particulièrement l'indépendance des connaissances utilisées dans chacune de ces trois catégories, ce que nous présentons dans le chapitre suivant.

Chapitre III

Implémentation informatique du prototype

Chapitre III

Implémentation informatique du prototype

Avant de présenter l'implémentation informatique du prototype de système expert, il est important de bien définir ses différents niveaux d'utilisation. Dans la suite nous distinguons :

- * le **concepteur** du système expert, c'est-à-dire celui, qui met en place les sources de connaissance et les définitions des classes d'éléments et qui introduit les règles de diagnostic.
- * le **configureur**, dont le rôle consiste à adapter le système expert à une installation particulière. Cela consiste à définir la base de données de l'installation en question et ne demande pas de connaissances particulières en diagnostic.
- * l'**utilisateur** final qui consulte le système expert pour obtenir des informations ou l'aide au diagnostic. Celui-ci ne doit avoir aucune notion de l'organisation des connaissances à l'intérieur du système expert. Toute la communication avec l'utilisateur final se passe par l'intermédiaire d'interfaces graphiques de sorte qu'un minimum de connaissances en informatique lui soit requis.

1 L'architecture blackboard

Comme nous l'avons vu dans le chapitre précédent, le diagnostic nécessite la manipulation de données et de règles heuristiques selon une stratégie bien adaptée (figure II.2). Nous avons choisi une architecture de type blackboard parce que celle-ci permet de séparer explicitement les données, les règles et la stratégie, assurant ainsi une bonne évolutivité du système expert. Cette architecture permet, en particulier, de guider le raisonnement en fonction des informations disponibles à un moment donné ce qui correspond bien au processus de diagnostic des experts. De plus, il permet de faire collaborer entre eux les différents spécialistes (bases de règles) en assurant une bonne cohérence entre eux et les données qu'ils modifient.

Malheureusement, le concept de blackboard n'est pas un formalisme unique utilisable tel quel pour tous les problèmes. Il ne remplace en aucun cas des modèles de représentation de données ou de règles, mais fournit une sur-couche pour gérer la collaboration de ces éléments.

Dans la suite nous allons montrer le fonctionnement de notre blackboard et comment les différents éléments ont été mis en œuvre.

1.1 Présentation

L'idée de base d'une architecture blackboard (tableau noir) est celle-ci : plusieurs experts collaborent pour résoudre un problème, chacun apportant les compétences de son propre domaine et effectuant une tâche particulière. Les experts ne communiquent pas directement entre eux ; ils voient l'information sur un tableau noir et la modifient selon leurs compétences individuelles (figure III.1). Des données d'entrée sont mises sur le tableau et l'action des experts consiste à ajouter leurs conclusions pour faire évoluer la solution. Le tableau noir correspond à l'ensemble de données présentes (la base de faits) et représente l'état du problème à un instant donné; il contient à la fois la description de l'installation, l'ensemble de ses signaux, ainsi que les observations et hypothèses créées par les différents experts. Les différents experts correspondent aux sources de connaissances que nous avons constituées (cf. § II.4.4), chacune apportant un élément précis à la solution du problème pour faire progresser le raisonnement (c'est-à-dire le diagnostic).

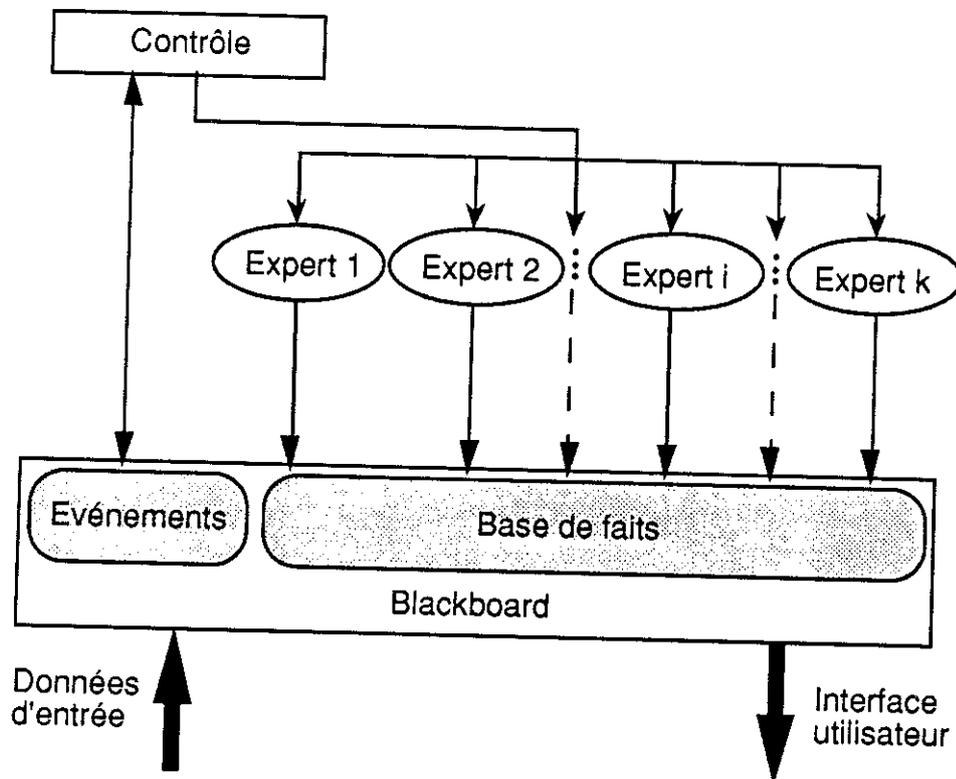


Figure III.1 L'architecture blackboard

L'ordre dans lequel vont agir les experts est fixé par le contrôle. En fait, comme les experts ne se connaissent pas, ils sont incapables de décider qui doit agir prioritairement. Le contrôle

est le seul à posséder une vue d'ensemble du problème et à connaître tous les experts. Il est alors capable de choisir quel expert doit agir à un instant donné, en fonction de l'état de l'avancement du diagnostic.

1.2 Discussion

L'architecture blackboard est un concept assez général permettant d'utiliser des types de contrôle les plus divers. Le type de contrôle utilisé dépend des problèmes spécifiques des applications [LAA 89]. Développé initialement pour la reconnaissance et compréhension de la parole [ERM 80], ce concept a été utilisé pour des problèmes d'analyse de situations, de planification d'actions [HAR 88], de diagnostic [TAL 86], [BUT 89], [CHA 91], de conception [GEN 91], et de problèmes complexes dans des domaines divers [ENG 88].

Nous ne développerons pas ici les différents types de contrôle existants et les spécificités des applications pour lesquels ceux-ci s'appliquent. Une comparaison de différentes approches peut être trouvée dans [LAA 89].

Dans la suite nous allons montrer l'implantation de notre blackboard, qui recouvre l'implantation des données, des sources de connaissance et du contrôle mettant en œuvre la stratégie.

2 L'outil choisi

Pour implanter notre prototype, nous avons utilisé un générateur de système expert: SMECI (Système Multi-Expert de Conception et d'Ingénierie) [SME 91]. Nous avons choisi cette boîte à outils commercialisée car elle contient les éléments dont nous avons besoin pour mettre en œuvre notre approche :

- un formalisme objet permettant de représenter la connaissance factuelle de notre application,
- un formalisme des règles de production d'ordre 1, compatibles avec le formalisme objet, pour représenter la connaissance déductive,
- un moteur d'inférences paramétrable permettant de mettre en œuvre des stratégies diverses,
- un environnement graphique avec une bibliothèque de fonctionnalités qui facilite la création de l'interface utilisateur
- un mécanisme de trace qui permet d'avoir à tout moment des explications sur les règles appliquées et les objets modifiés,
- utilisation du langage Lelisp pour définir les méthodes des objets et des procédures dans les règles.

SMECI ne contient pas un blackboard mais dispose de tous les éléments nécessaires pour sa programmation.

3 Modélisation des données: Représentation objet

Toutes les données présentes dans le système sont représentées en utilisant la représentation objet [SME 91] [GEN 91]. Ceci est vrai pour les informations relatives au domaine d'application, mais aussi pour toutes les structures relatives au raisonnement.

En ce qui concerne le domaine d'application, l'utilisateur de SMECI a la possibilité de créer interactivement les classes (appelées catégories dans SMECI) d'objet dont il a besoin pour la modélisation de son problème. Pour cela, il lui suffit de décrire les champs et les méthodes se rapportant à cette classe. Nous avons ainsi pu représenter fidèlement la structuration de la connaissance factuelle présentée au chapitre précédent.

Tout objet est une instance d'une seule classe.

Remarque: Ceci est une restriction par rapport à d'autres langages à objets permettant la multi-instanciation, c'est-à-dire qu'un objet peut être une instance de plusieurs classes à la fois [MEY88] [BIG 93], ce qui offre plus de souplesse dans la structuration des objets. Cependant, cette restriction permet d'éviter les problèmes d'ambiguïtés possibles dans ces dernières sans pour autant imposer des difficultés insurmontables pour notre application.

Les champs qui décrivent les propriétés des objets ont leur propre propriétés qui les caractérisent : les **facettes**. Le rôle des facettes est de spécifier la nature du champ, les types de valeurs autorisés, des actions à effectuer lorsqu'une valeur est affectée au champ etc.... La figure III.2 en montre quelques exemples. Les facettes simplifient considérablement la mise en œuvre de la décomposition de l'installation en objets et, par la suite, l'adaptation du système expert à des entraînements variés.

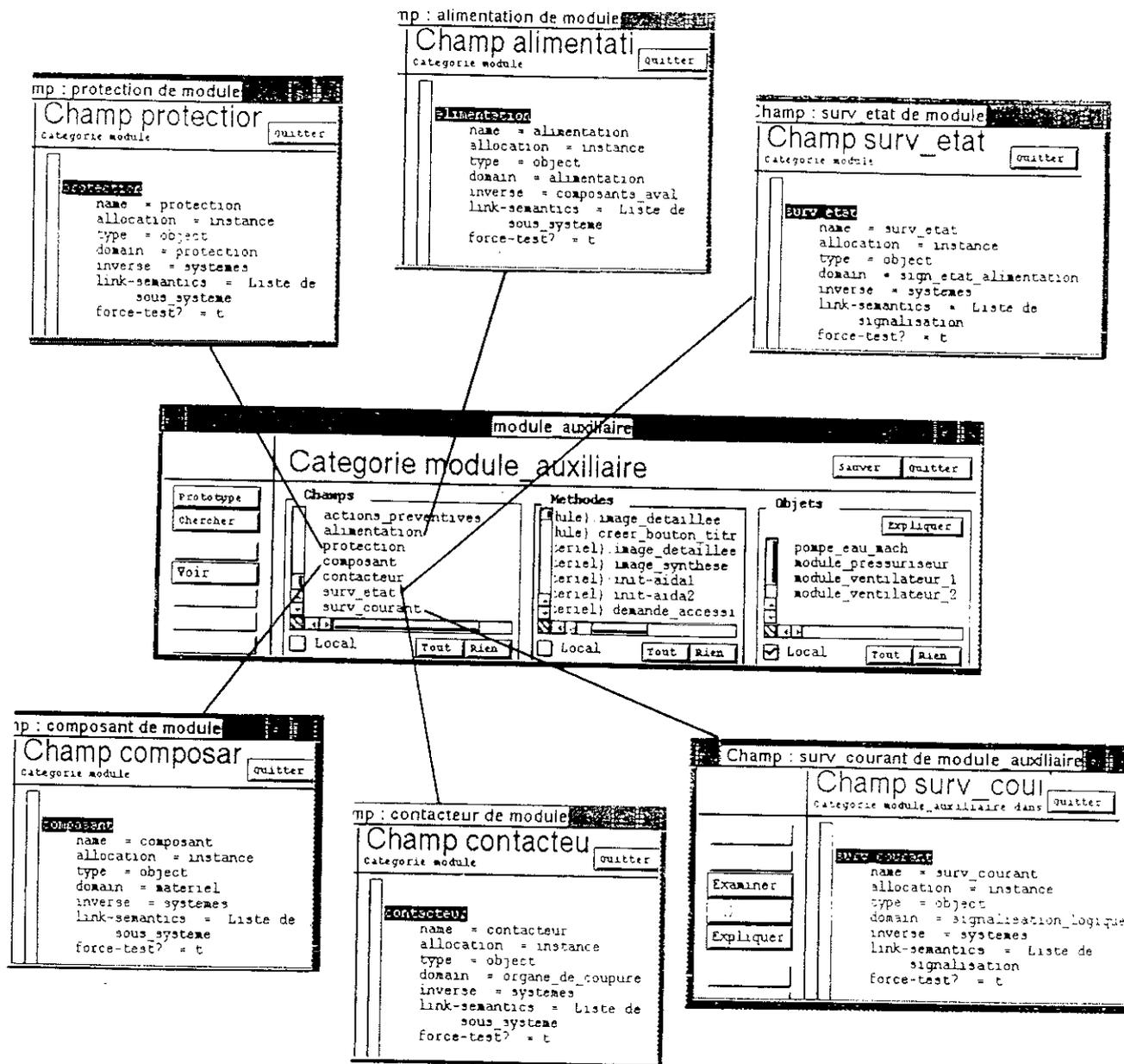


Figure III.2 Exemple de définition de champs par facettes

La création d'un objet se fait par instanciation d'une classe. Ensuite, chaque objet peut remplir ses champs avec les valeurs qui lui sont propres.

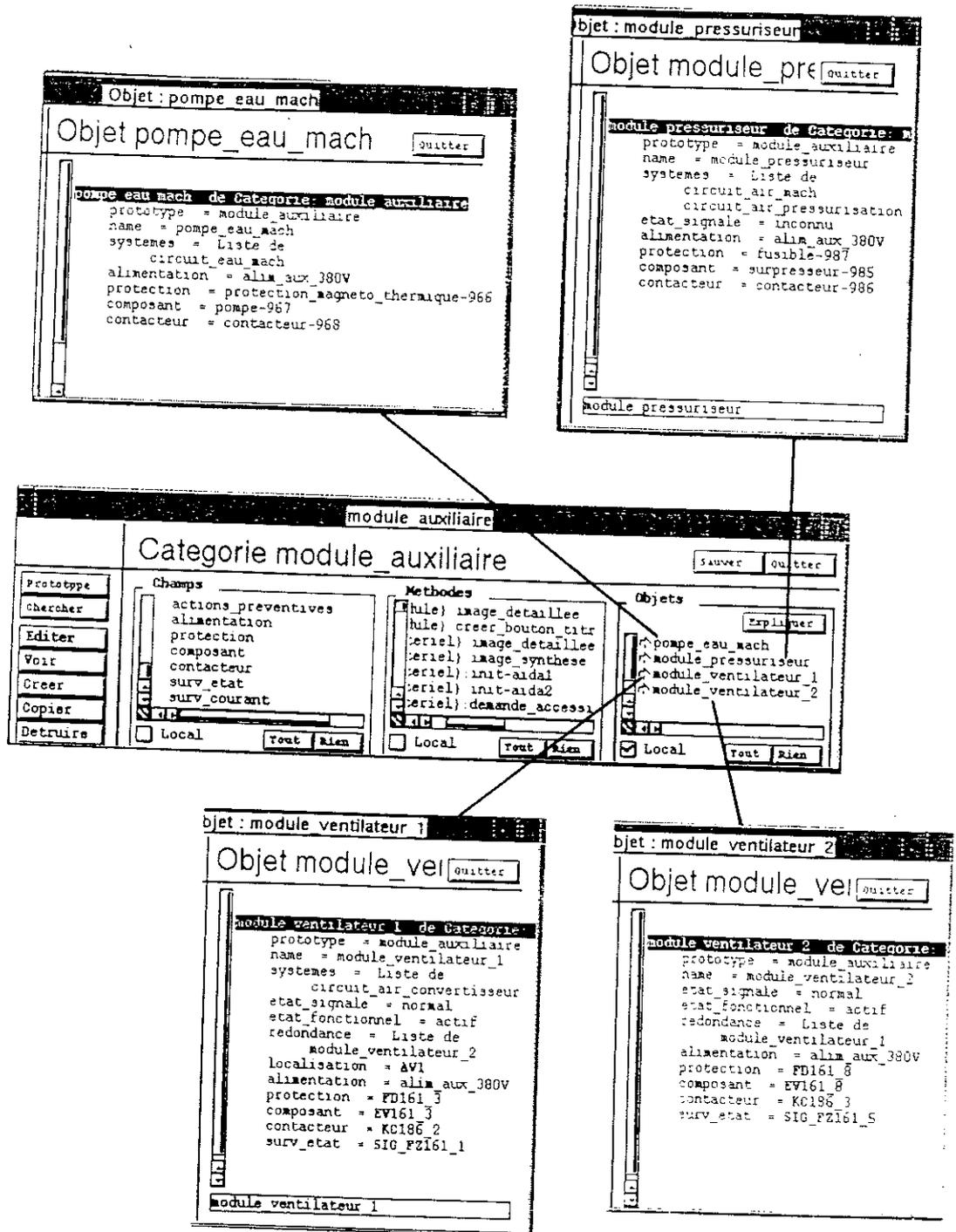


Figure III.3 Exemple de création d'instances d'une classe (catégorie)
(Dans la figure n'apparaissent que les champs ayant une valeur)

Le fonctionnement de notre blackboard impose quelques champs communs à tous les objets. Ainsi, la compatibilité et la communication des règles sont assurées dans les différentes sources de connaissances, comme on le verra par la suite. Les champs communs sont définis dans la classe *objet-installation* amont de toutes les autres classes définissant les objets d'une installation (cf. § II.3.2.4):

Les champs *mod_def_suspecté*, *mod_def_observé* et *observations_hypothétiques* mémorisent les observations et hypothèses existantes pour un composant (figure III.3).

Les champs *état_conclu*, *état_signalé* et *état_fonctionnel* rendent compte de l'état des composants (figure III.3)(cf § II.3.2.4.2). Leur contenu peut être modifié par les signalisations des capteurs ou des entrées de la part de l'utilisateur.

D'autres champs, également communs à toutes les classes, servent à fixer des valeurs communes pour tous les objets d'un même type (cf chapitre §II.3.2.4.1), par exemple, les champs référençant les sources de connaissances associées (*SdC-création-hypothèses-causes*, *SdC-création-hypothèses-effets*, etc ...) ou ceux qui définissent les valeurs possibles pour un champ d'un objet (*état_signalé*, *état_conclu*, *état_fonctionnel*, etc ..., figure III.4). Ceci se fait par l'intermédiaire d'un objet spécifique appelé **prototype** de la classe.

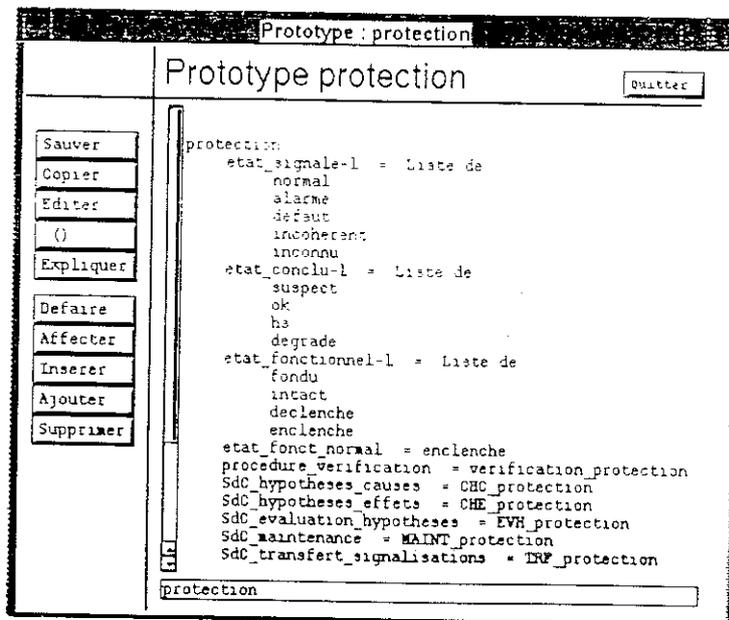


Figure III.4 Exemple d'implémentation de valeurs communes à tous les objets d'une classe.

Les classes représentent les concepts de référence de notre application. Celles-ci ne sont pas indépendantes les unes des autres mais s'insèrent dans une hiérarchie de classes, créant ainsi une taxinomie (classification) (cf. § II.3.2). La relation entre les niveaux de cette hiérarchie correspond à une spécialisation successive. Le concept d'héritage du modèle objet se réalise à travers la hiérarchie des classes (cf. § II.3.3) : les classes-filles héritent des propriétés (champs et méthodes) de leurs classes-mère. La figure III.5 montre une petite partie de la taxinomie *matériel* qui représente les éléments disponibles pour décrire les composants des entraînements. Sur la partie gauche, la figure III.5 montre les classes dans leur hiérarchie

avec quelques exemples d'objets sur la partie droite. Ces derniers sont reliés entre eux par des liens de décomposition qui sont réalisés à travers les différents champs qui les composent.

Outre la taxinomie *matériel*, dans notre prototype, il y a d'autres taxinomies, par exemple, pour les signaux, les grandeurs manipulées, les objets utilisés par la stratégie (observations, hypothèses ...), les outils de trace d'exploitation, etc ...

La décomposition des entraînements en modules et composants telle qu'elle a été présentée dans le chapitre II.3.2 a été mise en œuvre pour l'exemple d'une installation complète. Les figures II.15 et II.16 du chapitre II.3.4 montrent des exemples de cette décomposition pour l'entraînement complet et son sous-système machine. La décomposition est faite en termes de sous-systèmes, composants, protections, signalisations et signaux. Elle se précise au fur et à mesure qu'elle s'affine, jusqu'au niveau des capteurs, signalisations et des signaux. Ces derniers apparaissent aux feuilles de l'arborescence (extrémités de l'arbre de décomposition, figure II.16).

Remarque: Dans ces figures apparaissent uniquement les noms des objets représentant les entités correspondantes de l'installation, par opposition à la figure III.5 qui montre quelques détails des différents objets.

L'ensemble des objets représentant les entités d'un entraînement constituent la base de données de l'installation. Son utilisation à plusieurs objectifs. Elle sert de mémoire pour des conclusions intermédiaires issues de règles de diagnostic et d'outil de communication avec l'utilisateur pour la documentation de l'état actuel de l'installation.

Remarque: Pour la documentation, une interface graphique a été ajoutée à la base de données. Son rôle consiste à afficher toutes sortes d'informations incluses dans les objets telles que la structure de l'entraînement, l'état de ses composants, des paramètres électriques et physiques, des données historiques, etc ...

4 Modélisation des sources de connaissance

La représentation de la connaissance déductive se fait à l'aide de règles de production. La syntaxe utilisée pour les règles est proche de celle du langage naturel, ce qui est un point très important pour l'utilisation du système expert par des non-informaticiens (figure III.6).

```

defregle RC_palier_echauffement1
auteur ahrens
explication
  L'observation d'un echauffement sur le palier PAL de la machine
  MACH permet de creer l'hypothese d'un defaut bague sur PAL.

fin-explication
soit PAL un palier parmi $(FOCUS)
  PALIERS un paliers parmi les systemes de PAL
  MACH une machine parmi les systemes de PALIERS
  OBS une observation parmi les mod_def_observe de PAL

; SI l'echauffement du palier a ete observe
si mode^OBS = $echauffement

; ALORS creer une hypothese sur un defaut bague du palier
alors
action
  $(generation_hypothese PAL 'defaut_bague 'RC_palier_echauffement1
    (list PAL) OBS)
fin-regle

```

Figure III.6 Exemple d'une règle de diagnostic

On distingue trois parties principales d'une règle:

- une partie **explication** qui explique l'action de la règle sous texte libre. Il est possible d'utiliser des variables dans l'explication qui sont remplacées par les objets instanciés par la règle. Nous utilisons cette partie des règles pour donner à l'utilisateur une justification sur les hypothèses et les tests proposés en cours d'exécution du système expert. Cela traduit en quelques sortes une partie du raisonnement appliqué pour le diagnostic.
- une partie **condition** où on trouve les mots clés :
 - *SOIT* ... qui permet d'instancier des objets des classes parmi tous les objets présents. Pour pouvoir restreindre l'application de la règle sur un objet précis, nous avons ajouté une fonction de filtrage \$(FOCUS) au formalisme des règles de SMECI.
 - *SI* ... qui permet de déclarer des conditions d'application de la règle. Il est possible d'utiliser des instructions Lelisp comme condition.
- une partie **action** avec les mots clés :
 - *ALORS* ... qui permet d'exprimer en langage naturel les actions à réaliser comme l'affectation de valeurs ou la création d'objets.
 - *ACTION* ... qui permet d'utiliser des instructions Lelisp dans la conclusion de la

règle. Nous utilisons cette possibilité lors la création d'hypothèses pour gérer des actions liées à la stratégie, et qui ne doivent pas préoccuper le concepteur.

Comme nous l'avons vu dans le chapitre II.4.4, nous regroupons des règles en sources de connaissances. Ceci est réalisé par des **bases de règles**, qui sont utilisé dans SMECI pour contrôler le fonctionnement de son moteur d'inférence [SME 91], figure III.7.

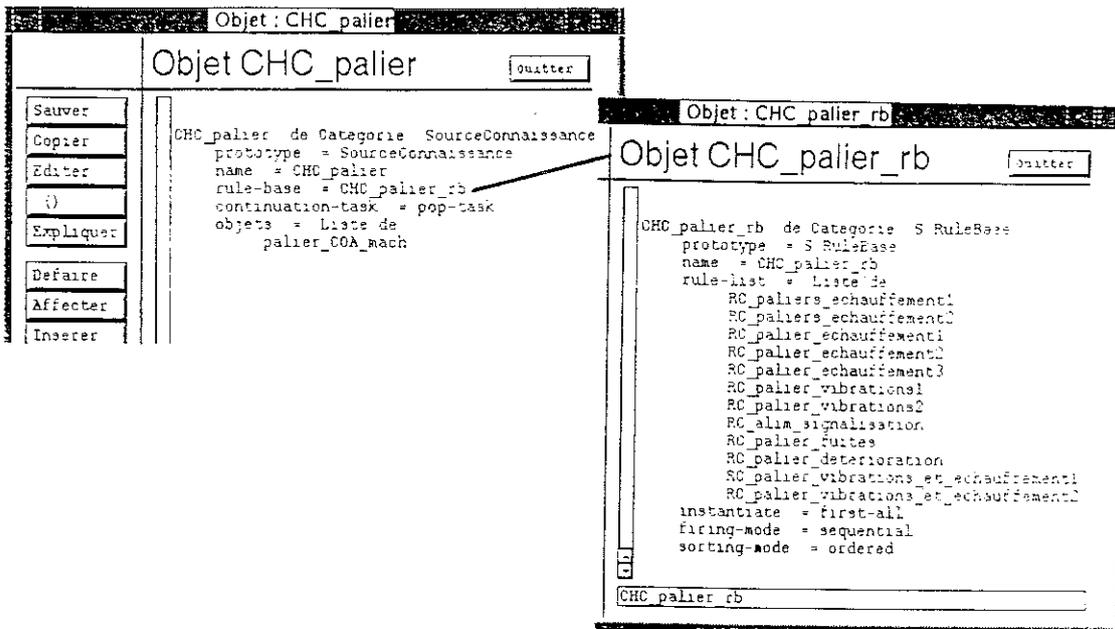


Figure III.7 Exemple d'une source de connaissance

Une base de règles constitue un objet d'une classe prédéfinie dans SMECI. Un tel objet contient différents champs: *instantiate*, *sorting-mode*, et *firing-mode* (figure III.7). Nous avons réglé ces champs de sorte que le moteur d'inférence utilise toutes les règles d'une source de connaissance.

C'est le contrôle du blackboard qui décide de l'application des différentes sources de connaissance en fonction des données sur le blackboard. Le champ *objets* dans les sources de connaissances permet de restreindre l'application de ses règles sur les objets sélectionnés par le contrôle. Ceci évite des tentatives d'applications inutiles de règles sur des objets autres que ceux sélectionnés par le contrôle lors de l'exécution du moteur d'inférences.

Le champ *continuation-task* des objets *sources de connaissance* sert à gérer la poursuite du fonctionnement du système expert après traitement d'une source de connaissance. Pour celui-ci, sa valeur est figée à *pop-task* pour rendre le contrôle au contrôle de blackboard [SME 91].

Comme le montre la figure III.4, les différentes sources de connaissances sont liées aux classes des objets sur lesquels elles portent par le prototype de la classe. Ceci comporte l'avantage d'un accès aisé aux différentes sources de connaissances. Il suffit d'envoyer un message, par exemple *appliquer-SdC-creation-hypotheses* à un objet pour activer la source de connaissance correspondante. Cela facilite considérablement l'accès au grand nombre de sources de connaissances pour le contrôle du raisonnement réalisé par le système expert.

5 Modélisation du contrôle du raisonnement

Comme on l'a vu dans le chapitre III.5, la stratégie du diagnostic comporte deux parties indépendantes: le choix des règles à appliquer, ainsi que le tri des hypothèses et de leurs tests proposés.

5.1 Le choix des règles à appliquer

Cela est fait par le contrôle du blackboard qui a pour mission de sélectionner et d'appliquer les sources de connaissances en fonction de l'état d'avancement du raisonnement et des données. Nous l'avons réalisé en créant la tâche *cycle-diagnostic* qui met en œuvre un fonctionnement cyclique du système expert (figure III.8).

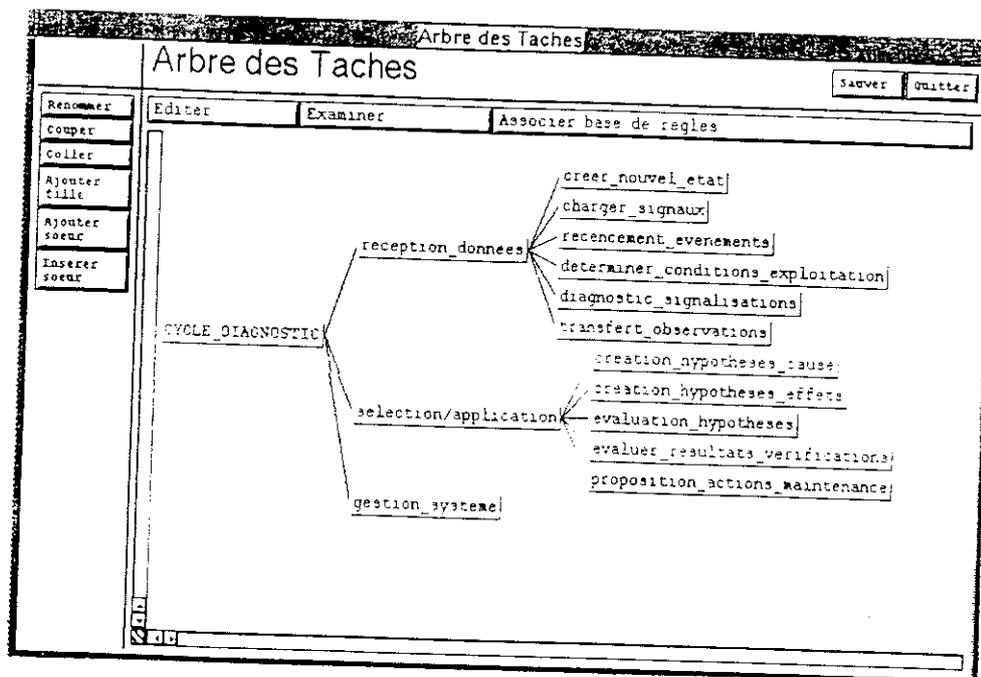


Figure III.8 Les tâches du contrôle

Celle-ci est divisée en plusieurs sous-tâches qui effectuent les différentes actions du cycle de contrôle présentées dans le chapitre II.5.2. Maintenant, nous allons expliquer en partie, comment le contrôle du blackboard réalise ces tâches, qui sont essentiellement *recensement-événements* et *sélection/application* (figure III.8).

5.1.1 Recensement des événements

Après chargement des données du système d'acquisition, la tâche *recensement-événements* de la tâche *réception_données* établit une liste de tous les nouveaux événements apparus depuis le dernier cycle. Un événement correspond à l'apparition d'une nouvelle information dans les données du blackboard.

Ces informations sont de nature diverse :

- apparition d'une alarme ou d'une nouvelle valeur d'une grandeur mesurée : le contenu du champ *valeur* d'un objet des classes *signal* ou *grandeur-mesurable* change,
- création ou suppression d'un *objet-raisonnement* (cf. § II.5.3): entrée ou retrait d'une *observation* par l'utilisateur, création d'*hypothèses* ou de propositions de *tests* par une règle,
- modification de l'état d'un composant, par exemple: déclenchement d'une protection. Ceci se traduit par une modification du contenu de son champ *état_fonctionnel* de *<enclenché>* à *<déclenché>* suite à une entrée par l'utilisateur ou à un signal d'un capteur. L'évaluation de l'état d'un composant suite à un test modifie le champ *état_conclu*, synthétisant le diagnostic pour un composant. Ceci constitue également un événement.

Le contrôle utilise un **agenda** pour gérer la liste de ces événements. Pour ceci, nous avons complété l'agenda utilisé normalement par SMECI pour la gestion des tâches du raisonnement (champ *task-stack*), par les champs *nouveaux-événements* et *événements-traités* (figure III.9) qui sont utilisés dans les étapes suivantes.

5.1.2 Sélection et application des sources de connaissances à appliquer

Rappelons ici que le contrôle est le seul à avoir une vision globale du problème, c'est-à-dire des différents sources de connaissance disponibles et de la stratégie de diagnostic.

Notre philosophie de sélection est la suivante : on sélectionne à un instant donné les sources de connaissance d'un type dans l'ordre que nous avons défini dans le chapitre II.5.2.

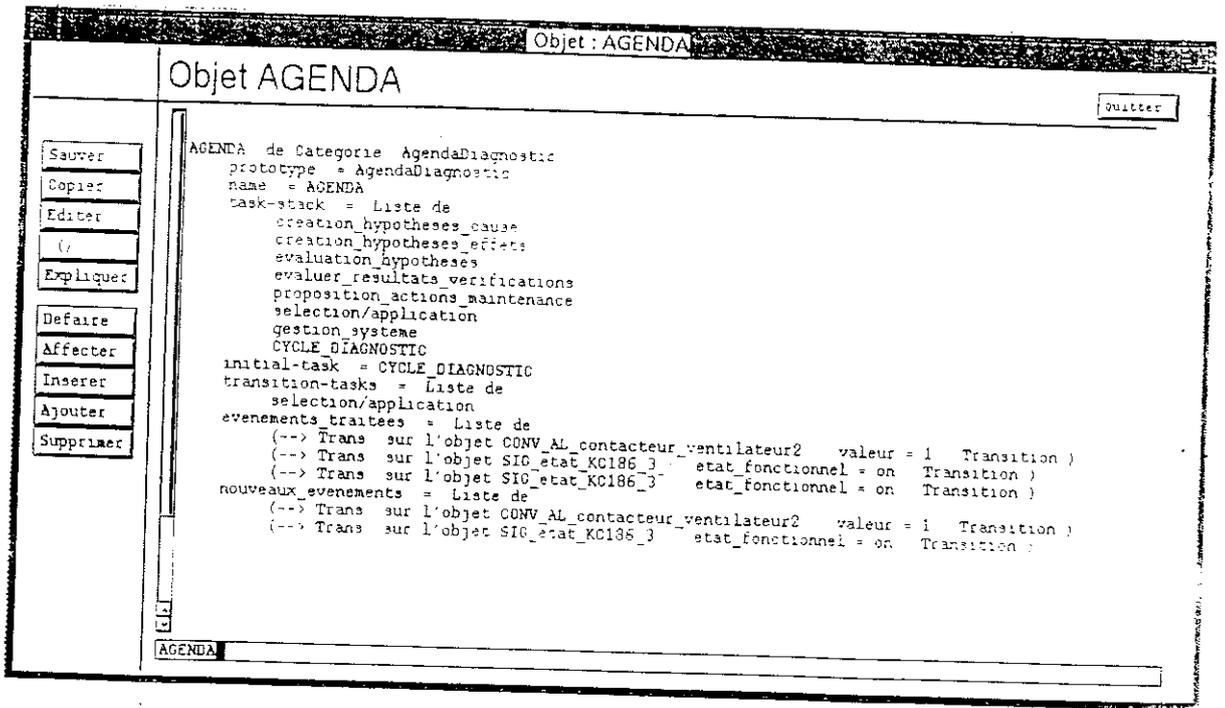


Figure III.9 L'agenda du contrôle

Ceci est effectué par l'application séquentielle des sous-tâches de la tâche de contrôle *cycle-diagnostic*. (figure III.8). Celles-ci contiennent des règles stratégiques (cf. § II.4.4.1) qui sélectionnent, en fonction des événements répertoriés dans l'agenda, une ou plusieurs sources de connaissances et préparent leur application en leur affectant les objets sur lesquels elles doivent travailler, et les empilent sur l'agenda des tâches (champ *task-stack*, figures III.9 et III.10).

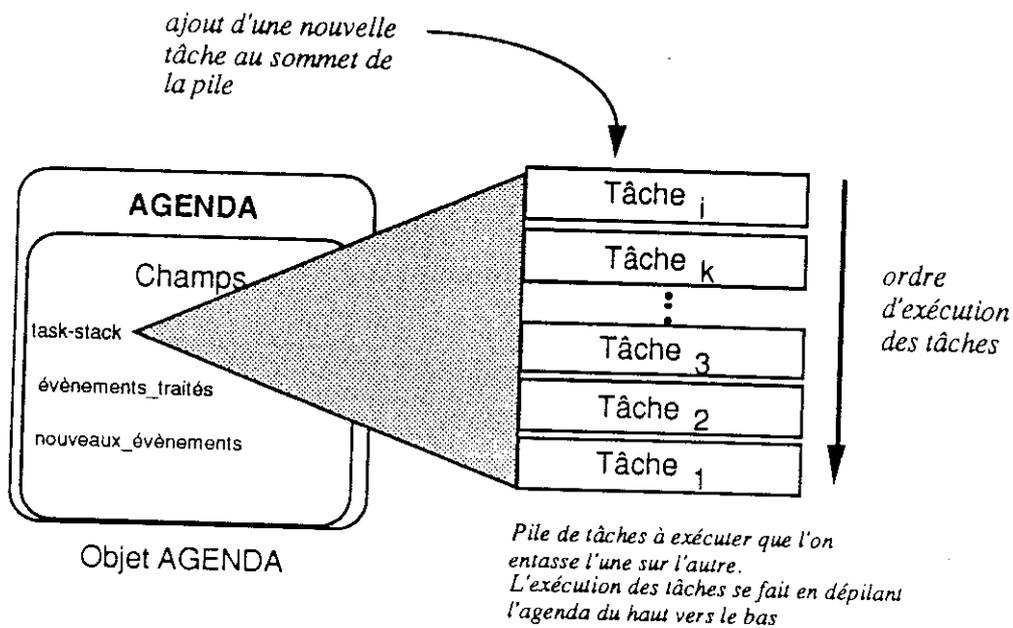


Figure III.10 Exemple d'utilisation de l'agenda

Lorsqu'une règle stratégique a terminé la sélection, le système continue avec l'application de la tâche au sommet de la pile, c'est-à-dire au début de la liste des tâches du champ *task-stack*, qui est la dernière source de connaissance sélectionnée.

Ainsi s'appliqueront toutes les sources de connaissances sélectionnées avant que les règles stratégiques de la prochaine tâche de contrôle ne sélectionnent d'autres sources de connaissances. La figure III.11 montre un exemple d'une règle stratégique, sélectionnant des sources de connaissance pour créer des hypothèses.

```

defregle RSTRAT_creation_hypotheses_causes
auteur ahrens
explication
    Créer des hypothèses sur les causes des observations sur OBJ
fin-explication
soit AGENDA un AgendaDiagnostic parmi $(S-get-agenda)
    EV un quelconque parmi les nouveaux événements de AGENDA
    TELT un S.TransitionElt parmi $(car EV)
    TRANS une S.Transition parmi $(cdr EV)

    OBS une observation parmi l'objet de TELT
    POS la position de TELT
    OBJ le cible de OBS

; Si une observation a été créée sur l'objet OBJ
si POS = 2
et new-value^TELT <> old-value^TELT

; ALORS appliquer sa source de connaissance pour créer des hypothèses
alors action
    $(send 'appliquer_SdC_hypotheses_causes OBJ)
    ; mémoriser la transition pour pouvoir l'enlever à la fin du cycle
    $(memo_pour_enlever TELT TRANS)
fin-regle

```

Figure III.11 Exemple d'une règle stratégique

A la fin du cycle, le contrôle supprime les événements qui ont été traités pour éviter que ceux-ci donnent lieu à d'autres applications de sources de connaissance dans le cycle suivant. Ainsi, le contrôle surveille qu'une source de connaissance ne soit appliquée qu'une seule fois suite à un événement. Ceci est important pour éviter des activations inutiles qui amèneraient des conclusions identiques multiples : une fois un remède à un défaut proposé, il ne faut pas répéter la même action car ceci risque de perturber l'utilisateur du système expert.

Le pilotage par les événements et la communication entre les règles des différentes sources de connaissance nécessitent une formalisation des données communes. Celles-ci sont notamment les *objets-raisonnement*, c'est-à-dire les observations, les différentes hypothèses et tests (cf. § II.5), et les caractéristiques stockées dans les champs communs (cf. § II.3.2.4) des objets d'une installation.

5.1.3 Discussion

Le contrôle que nous avons réalisé est de type procédural car les différents types de sources de connaissances sont utilisées dans un ordre figé. Toutes les connaissances stratégiques sont centralisées dans le contrôle ; les sources de connaissances n'ont aucune influence sur leur sélection. Cet aspect distingue le contrôle dans notre approche des contrôles plus sophistiqués que les systèmes à architecture blackboard permettent de réaliser. Un exemple en est le système C.O.C.A.S.E. [GEN 91] qui fait intervenir les sources de connaissances dans l'étape de sélection. Pour ceci, une partie des connaissances stratégiques est incluse dans les sources de connaissances, sous forme de conditions d'application pour chacune d'elles.

Dans C.O.C.A.S.E., chaque source de connaissance évalue d'abord elle-même si elle peut s'appliquer et passe une proposition d'action au contrôle. Ensuite, le contrôle sélectionne et applique une seule source de connaissance parmi celles qui sont applicables. Après chaque application, l'étape de réception recommence avec l'évaluation de toutes les sources de connaissance. Une telle approche est plus souple et permet de réaliser une stratégie plus évoluée. Cependant, la souplesse doit être payée par une certaine lourdeur informatique et par conséquent un fonctionnement lent.

Nous rappelons que le contrôle doit dépendre des spécificités du problème à résoudre. Dans notre cas, un contrôle relativement simple suffit pour mettre en œuvre le raisonnement de diagnostic. Il en découle les avantages de l'efficacité et d'une vitesse raisonnable du fonctionnement de notre système expert.

5.2 Tri des hypothèses

Les hypothèses créées par le raisonnement sont, comme les observations sur lesquelles elles s'appuient représentées explicitement comme des objets de la classe *objet-raisonnement*. La figure III.12 montre la taxinomie de cette classe avec ses sous-classes et leurs champs.

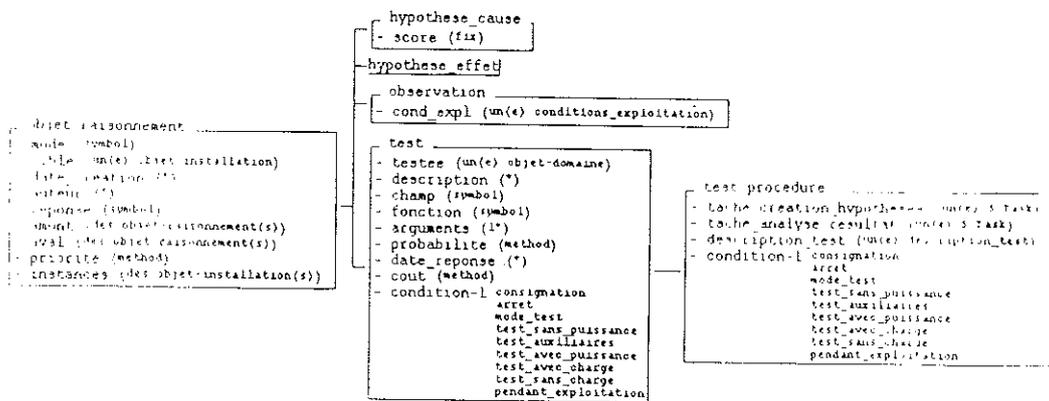


Figure III.12 Classification des classes pour le raisonnement

Tout *objet-raisonnement* est affecté dans les champs prévus à l'objet sur lequel il porte (cf. § II.3.2.4.2). Ceci permet un accès aisé à toutes les observations et hypothèses qui concernent l'objet analysé. Les champs suivants de la classe *objets-raisonnement* servent à mémoriser un certain nombre d'informations relatives aux observations et hypothèses (cf. § II.5.3):

- mode : mode de défaillance observé ou soupçonné sur l'objet-installation. Le système expert effectue des tests de cohérence vérifiant que seuls les modes de défaillances définis au niveau du prototype de l'objet en question sont permis,
- cible : objet-installation sur lequel porte l'objet-raisonnement,
- date-création : date à laquelle l'instance a été créée,
- auteur : responsable de la création. Ceci peut être une règle de diagnostic ou l'utilisateur du système expert,
- réponse : utilisé pour les hypothèses et les tests, résultat de l'évaluation d'une hypothèse ou d'un test : confirmation ou infirmation,
- amont : liste de références sur des objets-raisonnement en amont, par exemple, les hypothèses ayant donné naissance à un test,
- aval : liste de références sur des objets-raisonnement en aval, par exemple, les hypothèses créées suite à une observation,
- priorité : méthode qui renvoie la priorité de l'objet-raisonnement. Les méthodes sont différentes pour les tests, les hypothèse-causes et les observations,
- score : valeur fixe qui intervient dans la détermination de la priorité de l'objet. Cette valeur sert à renforcer ou à affaiblir la priorité d'un test, d'une observation ou d'une hypothèse

instances : liste de références sur des objets-installation qui ont contribué à la création de l'objet. Ce champ permet en particulier d'éviter d'appliquer plusieurs fois la même règle sur la même instantiation et ainsi de reboucler.

Pendant le raisonnement, les *objets-raisonnement* créés par les règles sont reliés les uns aux autres à l'aide des références dans les champs *amont* et *aval*. Ceci permet de fournir une explication cohérente du diagnostic ou des tests proposés, et en même temps, d'évaluer une priorité des tests proposés (par la méthode priorité). Les liens entre les objets-raisonnement sont créés automatiquement par les fonctions de création qui doivent être utilisées par les règles de diagnostic.

5.3 Déroutement

L'ordre des actions d'inférences du système expert est typiquement le suivant :

- 1) Apparition d'un signal correspondant à une alarme.
Ceci se fait soit par la lecture d'un fichier de données issues du procédé, soit par l'entrée interactive de celle-ci par l'utilisateur.
- 2) Affichage automatique de l'alarme dans l'interface graphique.
Toute action d'affichage liée à l'état des composants, comme par exemple la présence d'une alarme, est gérée par les méthodes des objets-installation et ne font intervenir aucun raisonnement par règles de diagnostic.
- 3) Création d'un objet observation par une règle de transfert.
Ainsi, la présence d'une alarme est traduite dans l'observation d'un mode de défaillance sur un objet ; une instance de l'observation est alors créée explicitement et est affecté au champ *mode-def-observé* de l'objet en question. La création de l'observation représente un événement suite auquel la stratégie va appliquer plusieurs sources de connaissances.
- 4) Déclenchement des règles de diagnostic.
Ceci se fait avec la création d'hypothèses sur les causes possibles de l'observation et sur des effets possibles. Les hypothèses sont également des instances dont la création est perçue via des nouveaux événements par le contrôle. Elles sont affectées aux champs *mod-def-suspectés* ou *observations-hypothétiques* de l'objet sur lequel elles s'appuient.

5) Création automatique d'un test pour chaque hypothèses de cause.

Un test est le moyen de communiquer l'hypothèse à l'utilisateur et de lui donner l'occasion de fournir une réponse, c'est-à-dire de confirmer ou d'infirmer l'hypothèse. Si plusieurs hypothèses existent pour un composant incriminant toutes le même mode de défaillance, elles sont toutes synthétisées dans un seul test.

L'ensemble de tous les tests pertinents à un moment donné sont affichés dans l'éditeur de tests et sont triés par la priorité ou tout autre critère sélectionné par l'utilisateur (figure III.13). Ainsi, l'utilisateur peut choisir à tout moment à quel test il veut répondre et se renseigner sur les tests déjà effectués. A chacun d'eux est associé une interface graphique donnant accès aux explications justifiant leur existence et gérant le dialogue avec l'utilisateur.

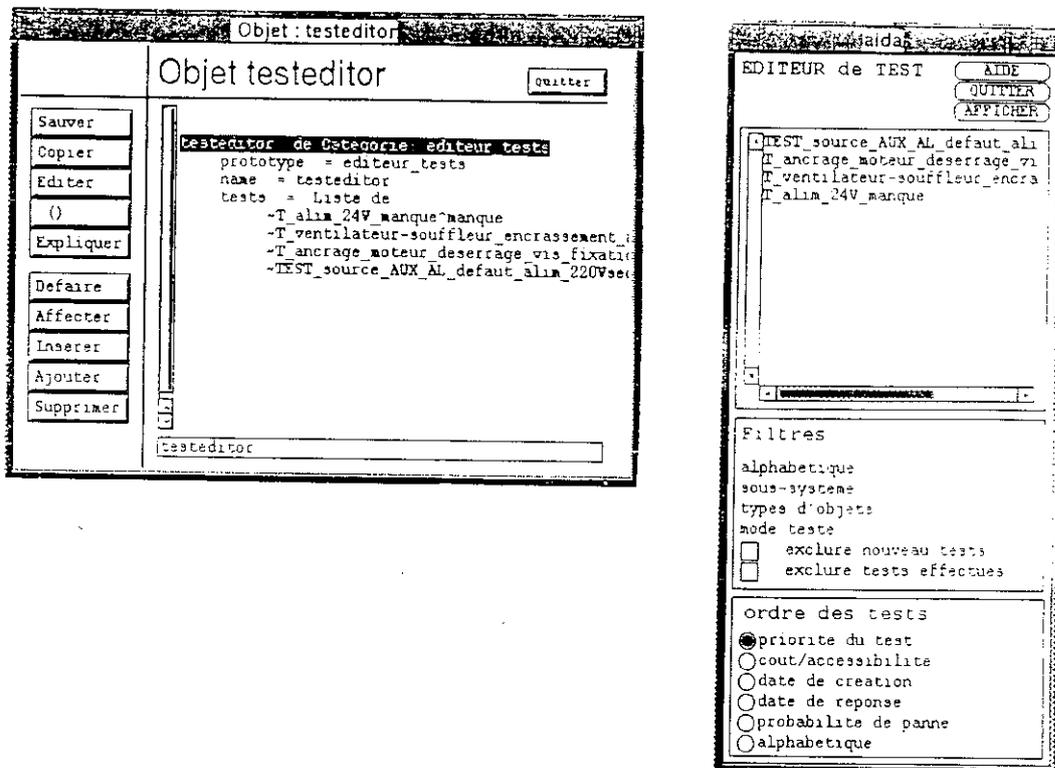


Figure III.13 Choix du test à effectuer dans l'éditeur de tests

6 L'interface graphique

6.1 Objectifs

L'objectif de l'interface graphique consiste à communiquer avec l'utilisateur en cachant tous les détails informatiques qui risquent de le perturber. L'utilisateur n'est pas censé être un spécialiste en informatique. Lors des rares pannes où il sera amené à consulter le système expert, sa première préoccupation sera de résoudre le problème de son procédé. Pour cette raison, l'utilisation du système expert doit être la plus facile possible et nécessiter un minimum d'apprentissage.

L'interface graphique que nous avons réalisée pour notre prototype de système expert porte sur les objets de l'installation et les conclusions de raisonnement, c'est-à-dire sur les tests proposés à l'utilisateur. En particulier, elle permet :

- de rendre transparente l'installation en facilitant l'accès à ses composants, aux signaux et aux grandeurs manipulées, ceci afin de visualiser leur état, d'entrer des observations et de consulter les hypothèses éventuelles les concernant,
- d'accéder aux informations associées aux composants (fichiers de documentation), aux signaux et aux différentes grandeurs (valeurs de références en plusieurs points de fonctionnement, enregistrement de données historiques, etc ...),
- d'informer l'utilisateur des actions à entreprendre et d'expliquer ses conclusions sans l'obliger de chercher ces informations dans le mécanisme de trace du système expert, ce qui demanderait une connaissance approfondie de son fonctionnement et de sa structure.
- d'être mis elle-même en œuvre automatiquement à partir de la base d'objets d'installation (cf. § III.6.2)

Pour un système expert de diagnostic dont le rôle est essentiellement d'expliquer à son utilisateur ce qui ne va pas, et au mieux, pourquoi (explication étroitement liée à la connaissance de l'utilisateur), l'interface joue un rôle primordial. L'idée qu'un bon dessin vaut plus qu'un long discours nous a amené à visualiser la structure de l'installation et les signaux à l'aide d'images. La décomposition de l'installation en sous-systèmes et composants a été, en majeures parties, inspirée par des réflexions sur l'interface graphique.

6.2 Génération automatique de l'interface

La mise en œuvre d'une interface graphique est normalement une tâche onéreuse. Pour cette raison, nous avons conçu un générateur d'interfaces graphiques qui supplée le concepteur du travail fastidieux de programmation manuelle de l'interface. Les seules données nécessaires à spécifier sont les objets de la base de données de l'installation. Celle-ci contient la structure, les composants et les signaux disponibles d'une installation. C'est la même base de données qui sert pour adapter le système expert à une installation spécifique.

6.3 Présentation

6.3.1 Les synoptiques hiérarchisées

La base de données d'une installation contient les sous-systèmes, les composants et les signaux de cette installation. Les différentes entités sont représentées par des objets qui sont liés entre eux. On peut donc, à partir des liens de décomposition, établir une **synoptique hiérarchisée** de l'installation (figure III.14), de ses sous-systèmes, de leurs sous-systèmes respectifs etc..., jusqu'au niveau des composants. Ceci amène à une structure de l'interface graphique qui reflète fidèlement la structure modulaire de la base de données de l'installation. Comme le nombre de composants est trop important pour être visualisé simultanément à l'écran, et comme l'affichage de trop d'informations disperserait l'attention de l'utilisateur, seul un niveau de décomposition de l'entraînement est visualisé. La figure III.14 montre des exemples où pour la répartition des entités présentées, nous nous sommes inspirés de celle des schémas électriques utilisés habituellement en électrotechnique. Cet affichage se fait dans une fenêtre de l'écran et montre le système sélectionné avec les entités qu'il contient. Il est possible de descendre dans la hiérarchie des sous-systèmes en sélectionnant les entités de l'image synoptique apparaissant à l'écran à l'aide de la souris. Cette action ajoute à l'écran une fenêtre qui détaille le sous-système sélectionné.

6.3.2 Visualisation de l'état de l'installation

Inspiré par les interfaces des systèmes de surveillance modernes [WEY 89] [ABB 89] [ALE 91], nous avons intégré de la couleur dans l'interface. Grâce à la couleur et également à l'allure des images qui peuvent changer dynamiquement, on peut attirer l'attention de l'utilisateur pour lui signaler les états des composants et la présence de conditions anormales par alarmes.

a) Les alarmes et défauts

L'affichage d'une entité en rouge signifie qu'un signal de défaut est présent dans cette entité ou une des entités qu'elle contient. Ceci signifie un incident grave qui a provoqué l'arrêt de l'installation. La couleur jaune correspond à la présence d'une alarme signifiant un dépassement d'un seuil critique par une grandeur physique ou électrique.

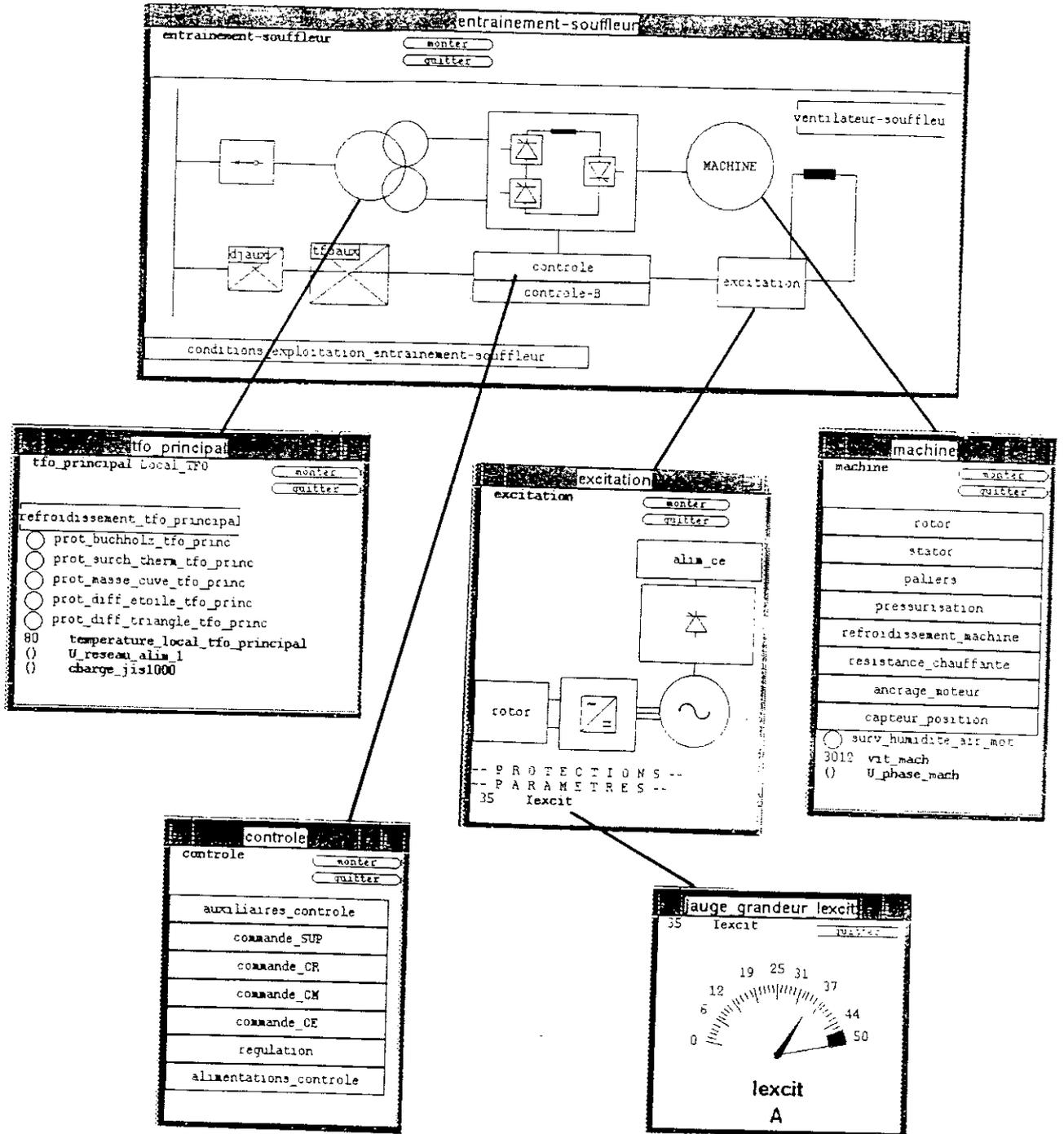


Figure III.14 Exemple de synoptiques hiérarchisées d'un entraînement

b) Les états des composants

Les composants qui peuvent prendre plusieurs états, comme, par exemple, une protection qui peut être déclenchée ou enclenchée, affichent leur état actuel en modifiant leurs images. L'information sur l'état d'une entité peut venir du système des capteurs ou d'une conclusion d'une règle de diagnostic.

c) La synthétisation de l'information

La visualisation des alarmes et défauts signalés par les capteurs ou le système de contrôle se répercute sur les synoptiques de niveau de décomposition supérieur (figure III.14). Ceci permet, depuis la synoptique du plus haut niveau, d'identifier rapidement l'origine de l'événement, simplement en descendant dans la hiérarchie. En même temps, ce mécanisme fait la synthèse des événements à n'importe quel niveau souhaité. Cette façon d'afficher une alarme est bien plus explicative que le simple affichage d'une étiquette. Toutefois, l'interface graphique ne remplace pas l'affichage classique. Elle est un plus optionnel pour les utilisateurs qui cherchent à mieux comprendre le contexte de l'événement.

6.4 Mise en œuvre

L'implantation des images qui visualisent les objets représentant l'installation, fait appel aux outils AIDA (langage à primitives graphiques basé sur Lelisp) et MASAI (générateur d'interfaces graphiques) de la société ILOG. Les images sont, comme les objets représentant l'installation, implémentées dans le formalisme objet. Elles sont plutôt des applications graphiques, car outre l'image affichable, elles peuvent avoir un comportement (par exemple, être sensibles aux actions de la souris comme c'est le cas des boutons). D'une manière générale, les images sont créées par des fonctions-constructrices qui font appel aux primitives de Lelisp et d'AIDA. Ces fonctions sont intégrées dans le cadre du système expert par l'utilisation dans les méthodes (cf. figures II.2 et II.3). Chaque classe dispose de ses propres méthodes de création d'interface qui lancent les fonctions-constructrices. Ce rattachement aux classes permet de retrouver facilement les fonctions graphiques responsables de l'allure des entités.

Des détails techniques sur la mise en œuvre de l'interface graphique et son association à la base de données d'objets représentant une installation se trouvent dans l'annexe III.

7 Conclusion

En conclusion, l'architecture utilisée pour notre prototype de système expert, est caractérisée par deux concepts fondamentaux [GEN 91].

- une séparation la plus grande possible entre la connaissance relative au domaine du diagnostic (les experts / sources de connaissance) et la manière à résoudre le problème (le contrôle),
- une unification de l'échange de données entre les différents experts. Ceux-ci ne communiquent que par l'intermédiaire d'une zone commune : le tableau.

Outre l'efficacité et une vitesse raisonnable du fonctionnement du système expert, effectuées par le contrôle de notre blackboard, il s'ajoute d'autres avantages inhérents aux architectures blackboard, qui sont évolutivité et uniformité de la connaissance représentée. L'évolutivité est assurée par le concept des sources de connaissances. Il est aisé de remplacer ou de rajouter une source de connaissance sans avoir une vision globale du problème. Ainsi, il est facile d'intégrer la connaissance sur un nouveau type de composant. L'uniformité est atteinte par la formalisation de la communication entre les sources de connaissance via le blackboard et le modèle des données commun qui le met en œuvre.

En particulier, notre prototype présente les caractéristiques suivantes:

- Le contrôle du type blackboard s'intègre parfaitement dans SMECI, ce qui facilite son adaptation à des versions futures de ce logiciel.
- Le raisonnement mis en œuvre à travers le blackboard est particulièrement efficace : aucune source de connaissance n'est appliquée quand il n'y a pas eu d'événements significatifs. L'évaluation des règles stratégiques dans ce cas est très rapide.
- L'architecture du système est ouverte : il est très facile d'introduire dans le cycle de contrôle de nouvelles tâches, par exemple pour effectuer des analyses systématiques sur les allures des grandeurs mesurées. Pour ceci on peut faire appel aux méthodes de traitement du signal ou de reconnaissances de formes.
- L'ajout de nouvelles sources de connaissance est très simple. Il suffit de définir leur nom au niveau des classes des composants concernés pour qu'elles soient prises en compte automatiquement par le système.
- Toutes les phases importantes du contrôle (gestion des événements, sélection et application des sources de connaissances) sont prises en charge par le système expert. Le concepteur des sources de connaissances n'a pas besoin d'intervenir dans ce fonctionnement. Il peut, par contre, suivre l'évolution des différents champs des objets nécessaires à cette gestion pour en comprendre le fonctionnement.
- Il est possible de simuler des défauts. Cet aspect est important, en particulier pour tester sans risque de nouvelles connaissances implantées dans le système expert. Dans le chapitre IV.1.3, nous présentons un exemple de simulation de défauts.

Chapitre IV

Exemple d'une session avec le prototype

du système expert

Chapitre IV

Exemple d'une session avec le prototype du système expert

Le prototype a été conçu pour fonctionner en parallèle avec l'installation à diagnostiquer. La plupart du temps, il n'y a pas de problèmes à diagnostiquer et son rôle se limite à enregistrer et à analyser cycliquement les données enregistrées par le système d'acquisition, et à documenter l'état de l'installation. Pour ceci, il dispose d'un certain nombre d'outils implémentés avec le formalisme Objet, comme la base de données de l'installation, pour synthétiser et visualiser la masse d'information mémorisées. Un exemple d'un tel outil est le suivi de l'historique sur les grandeurs fondamentales, pendant toute la durée d'exploitation de l'installation (courants, températures, etc ...). Le but de ces enregistrements est de fournir des données exploitables comme renseignements supplémentaires pour aider les experts dans leur analyse d'un défaut (cf. annexe II, historique)

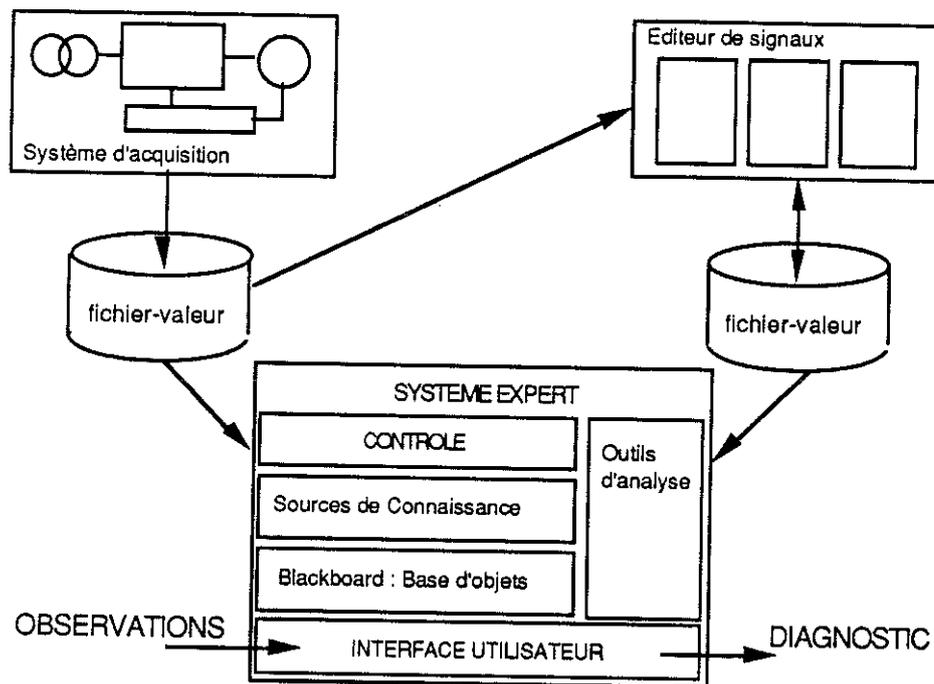


Figure IV.1 Interfaçage entre le système expert et l'équipement

En fait, le manque de données fiables au début des interventions des experts peut ainsi être pallié par une trace synthétique du fonctionnement, ce qui est plus facilement exploitable que les enregistrements classiques. En effet, ceux-ci s'avèrent difficiles à interpréter lorsque la masse de données ou la durée des enregistrements deviennent importantes.

Le transfert de données entre le système d'acquisition et le système expert est prévu, dans un premier temps, par fichiers textes (figure IV.1).

Tant que cet interfaçage n'existe pas, il est possible de simuler l'installation et son système d'acquisition en créant manuellement les fichiers-valeur à la main, à l'aide d'un éditeur de signaux qui fait partie des outils que nous avons créés. Il est aussi possible, avec cet outil, de fournir ces données interactivement pour tester le fonctionnement du prototype.

1 Une session utilisateur

1.1 Chargement des connaissances

Après chargement du système expert dans l'ordinateur, l'environnement SMECI est initialement vierge et ne possède aucune connaissance sur l'installation ou sur les règles de diagnostic (figure IV.2).

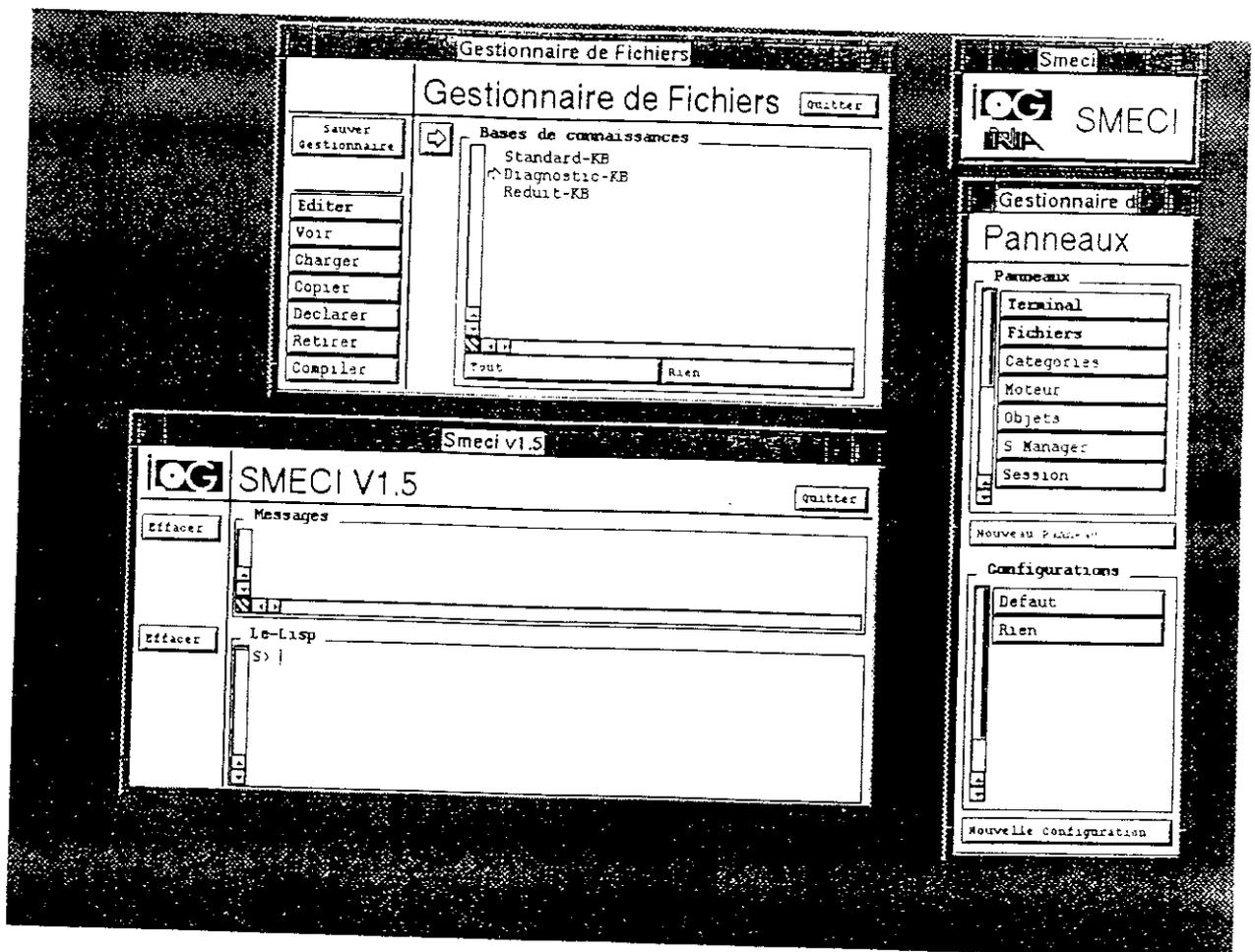


Figure IV.2 Le système expert initialement vide

Il est donc nécessaire, dans un premier temps, de charger dans SMECI les différents fichiers relatifs au fonctionnement de notre prototype. Ce chargement est très simplifié puisque tous ces fichiers sont regroupés dans une base de connaissance : *Diagnostic-KB*. Cette base contient à la fois les informations relatives à l'installation à diagnostiquer (la base de données de l'installation) et toutes les connaissances relatives au diagnostic. Le chargement de celle-ci se fait en cliquant sur le bouton *charger* du gestionnaire de fichiers après l'avoir sélectionnée. Le système indique alors par un message dans le terminal de SMECI (figure IV.3) qu'il est prêt à démarrer.

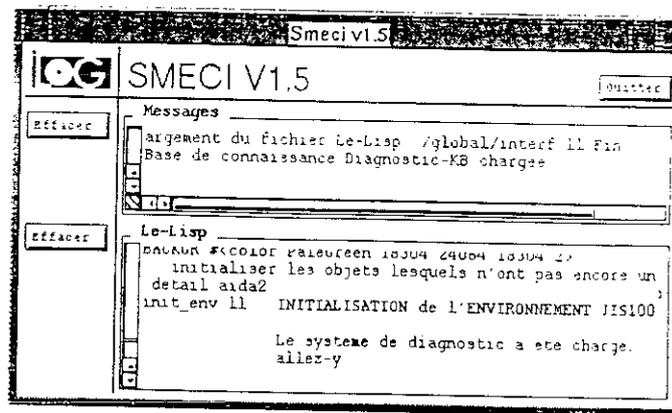


Figure IV.3 Le système expert après chargement de la base de connaissances

Ensuite, à l'écran apparaissent plusieurs fenêtres qui contiennent les fonctionnalités suivantes :

- l'interface de la base de données de l'installation qui montre la synoptique de l'installation (figure IV.4). L'accès à tous les composants et signaux de l'entraînement est possible en cliquant sur le dessin de l'objet à détailler,
- le gestionnaire de fonctionnalités, prévu pour remplacer le gestionnaire de l'environnement SMECI pour l'utilisateur final. Il regroupe toutes les fonctionnalités propres au système de diagnostic et permet de les activer à l'écran (figure IV.5, partie gauche),
- l'éditeur de test, encore vide. Il permet d'appliquer différents filtres (cf. § II.5.3) sur la liste des tests proposés et de les trier selon différents critères (figure IV.5, partie droite),
- le gestionnaire du raisonnement contenant un terminal et une série de boutons pour démarrer ou pour arrêter le raisonnement, et pour initialiser la base de faits (figure IV.6)

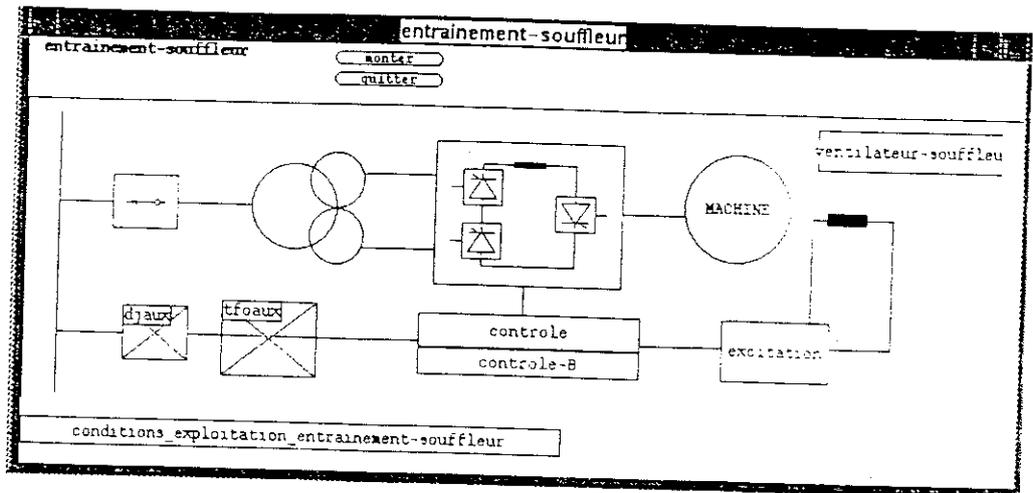


Figure IV.4 L'interface de la base de données de l'installation

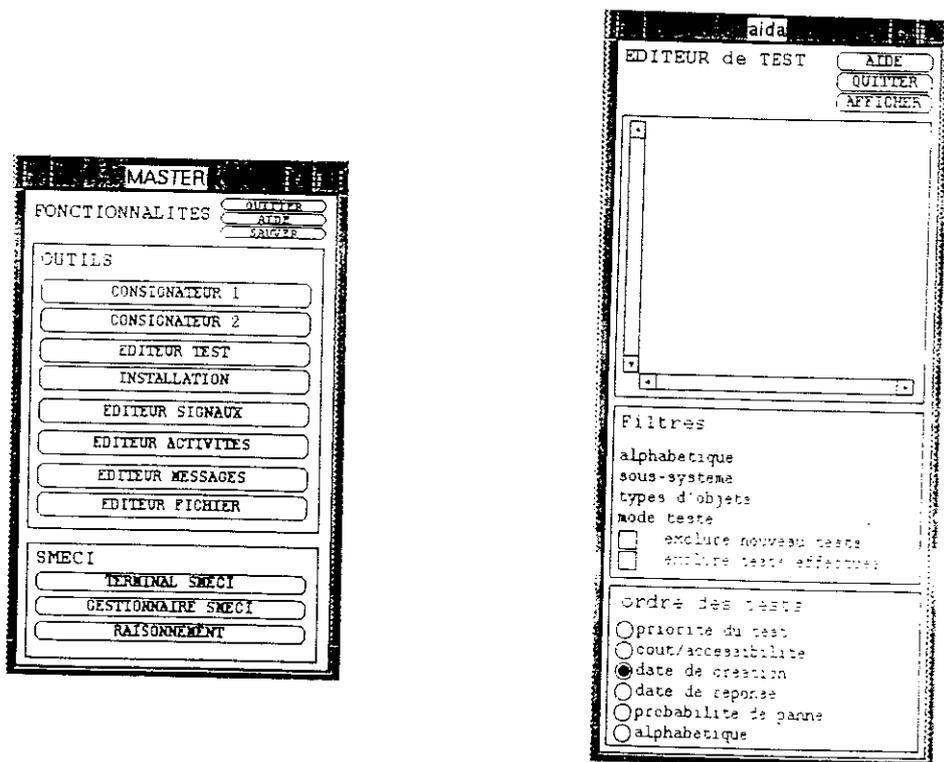


Figure IV.5 Le gestionnaire de fonctionnalités et l'éditeur de tests

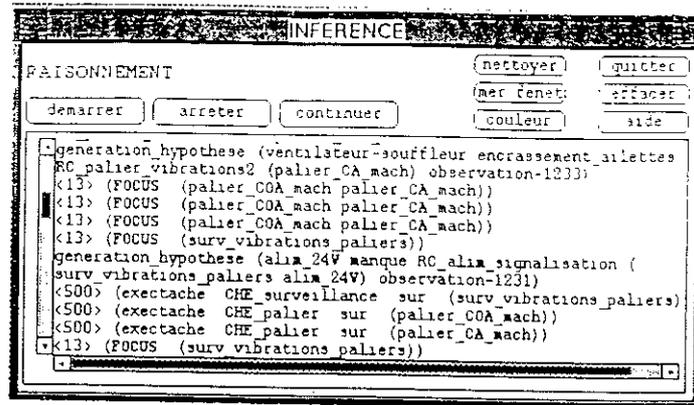


Figure IV.6 Le gestionnaire du raisonnement

1.2 Lancement du raisonnement

Au ce stade de l'utilisation du système expert, le raisonnement n'est pas encore activé. Cependant, il est déjà possible de consulter la base de données de l'installation via son interface (figure IV.4) et d'entrer des valeurs pour les signaux de l'installation via l'éditeur de signaux (figure IV.7)

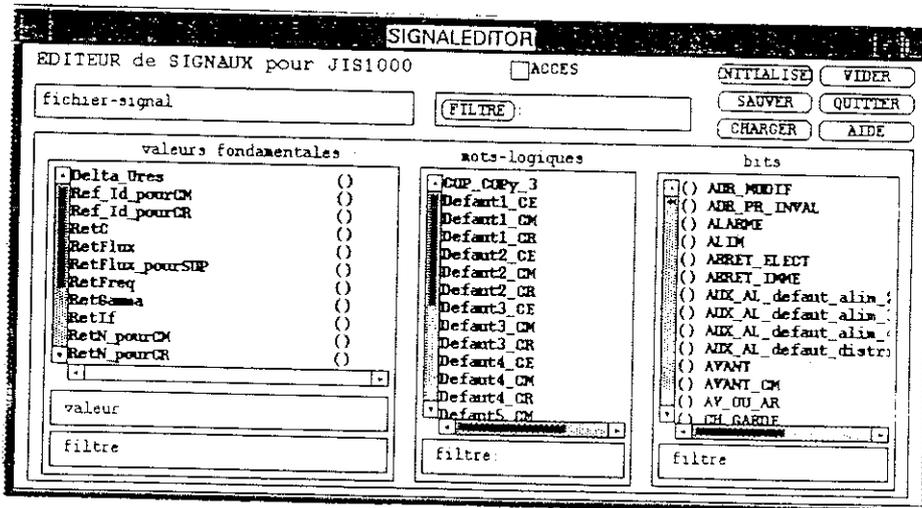


Figure IV.7 L'éditeur de signaux

La possibilité de consultation de la base de données est intéressante pour surveiller le fonctionnement de l'installation et comme aide de documentation en cas de problèmes. La figure 8 montre l'exemple d'une consultation de points de référence qui peuvent être enregistrés pendant la mise en route de l'installation.

regulation

regulation

References
 Ref_Id_pourCR

Grandeurs mesurees
 Ref_Id_pourCR
 519 RetC
 RetFlux
 RetFreq
 RetGamma
 RetIf
 RetN_pourCR
 157 Ret_Id
 Ures
 cosphi
 Udo
 surv_regulation

point de reference 1

GRANDEURS de REFERENCE du point de reference 1
 (point de reference 1 demarrage sans charge - charge desaccoupees essai avec puissance.

valeur	unite	objet-cible	nom de l'objet
()	()	temp_huile_palierGA_mach	ref1_temp_huile_palierGA_main
()	()	temp_palierGA_mach	ref1_temp_palierGA_mach
()	degres_Celsius	temp_air_cn	ref1_temp_air_cn
()	degres_Celsius	temperature_ambiante_convertisseur	ref1_temp_air_cn
300	A	Id_mesure	ref1_Id
()	degres_Celsius	temp_enroulement1_stator	ref1_temp_bob1_stator
()	()	temp_enroulement3_stator	ref1_temp_bob3_stator
()	degres_Celsius	temp_enroulement2_stator	ref1_temp_bob2_stator
()	degres_Celsius	temperature_local_tfo_principal	ref1_temp_amb_tfo
111	points	Ret_Id	mesure_reference-2

POINTS DE REFERENCE pour Ret Id

(Retour Id mesure par le CR filtre par un passe-bas du premier ordre a T = 4.44 ms

valeur	unite	point de reference	nom de l'objet
157	points	demarrage_50%In	mesure_reference-1050
20	points	demarrage_sans_charge	mesure_reference-1050
256	points	point_nominal	mesure_reference-1050
111	points	point de reference 1	mesure_reference-2

Figure IV.8 Exemple de consultation de points de référence enregistrés

Cependant, l'intérêt principal du système expert est d'assister l'utilisateur au diagnostic en cas de problèmes. Pour ceci, il faut démarrer le cycle de raisonnement en sélectionnant la fonctionnalité *raisonnement* dans le gestionnaire de fonctionnalités (figure IV.5). Ceci fait apparaître à l'écran le gestionnaire du raisonnement (figure IV.6), qui permet de démarrer le cycle de raisonnement décrit dans le chapitre III.5.3 en cliquant avec la souris sur le bouton *démarrer*. L'éditeur de signaux s'affiche pour recevoir des entrées, sinon seulement quelques commentaires défilant dans le terminal montrent que le raisonnement est activé. Maintenant, il est possible de simuler l'apparition d'une alarme ou de n'importe quel signal enregistré en chargeant un fichier de données préparé ou en entrant directement les valeurs dans l'éditeur.

1.3 Simulation de défauts

Supposons qu'on simule une alarme sur une alimentation en affectant la valeur "1" au signal *AUX_AL_défaut_alimentation_220_V_secouru* en cliquant dessus dans l'éditeur de signaux (figure IV.7). Suite à cela, plusieurs actions se passent dans notre système expert :

- la présence d'une alarme est immédiatement affichée dans l'interface de l'installation : l'objet représentant le contrôle change de couleur en devenant jaune. Il est possible de trouver l'origine de l'alarme en cliquant sur l'objet jaune, ce qui fait apparaître des détails

de l'objet, jusqu'à arriver au signal et au capteur qui l'a émis. Cet affichage a lieu indépendamment de l'activation du cycle de raisonnement ;

- un test va apparaître dans l'éditeur de test, créé par une des règles de la base de connaissance (figure IV.9, à gauche).

Lorsqu'une autre alarme est entrée, par exemple concernant les vibrations pour la machine (figure IV.10), d'autres tests sont proposés dans l'éditeur de test (figure IV.9, à droite).

Il est également possible de fournir des observations ou des hypothèses directement via l'interface de l'installation. Ainsi, il est possible de donner des informations qui ne sont pas instrumentées par des capteurs. L'utilisateur est guidé par un éditeur approprié limitant ses choix aux observations définies dans la base de connaissances (figure IV.11).

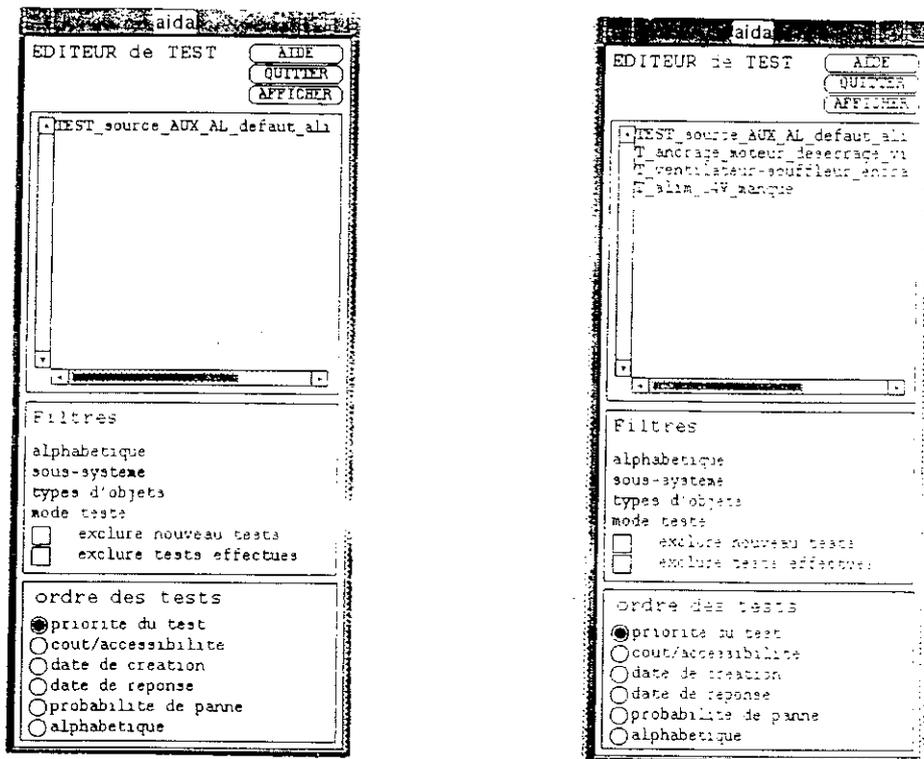


Figure IV.9 Exemple de tests proposés dans l'éditeur de tests

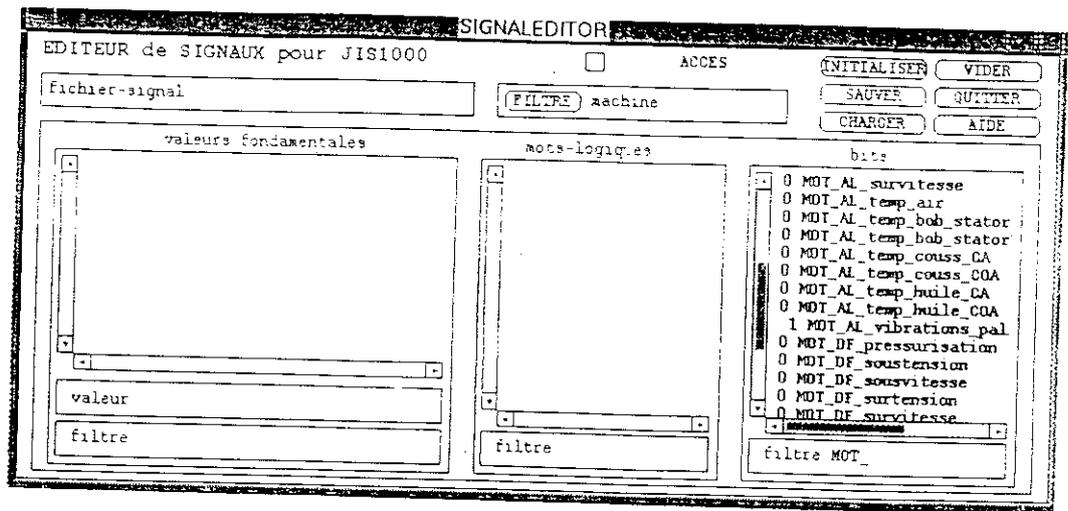


Figure IV.10 L'éditeur de signaux: entrée d'une alarme pour la machine

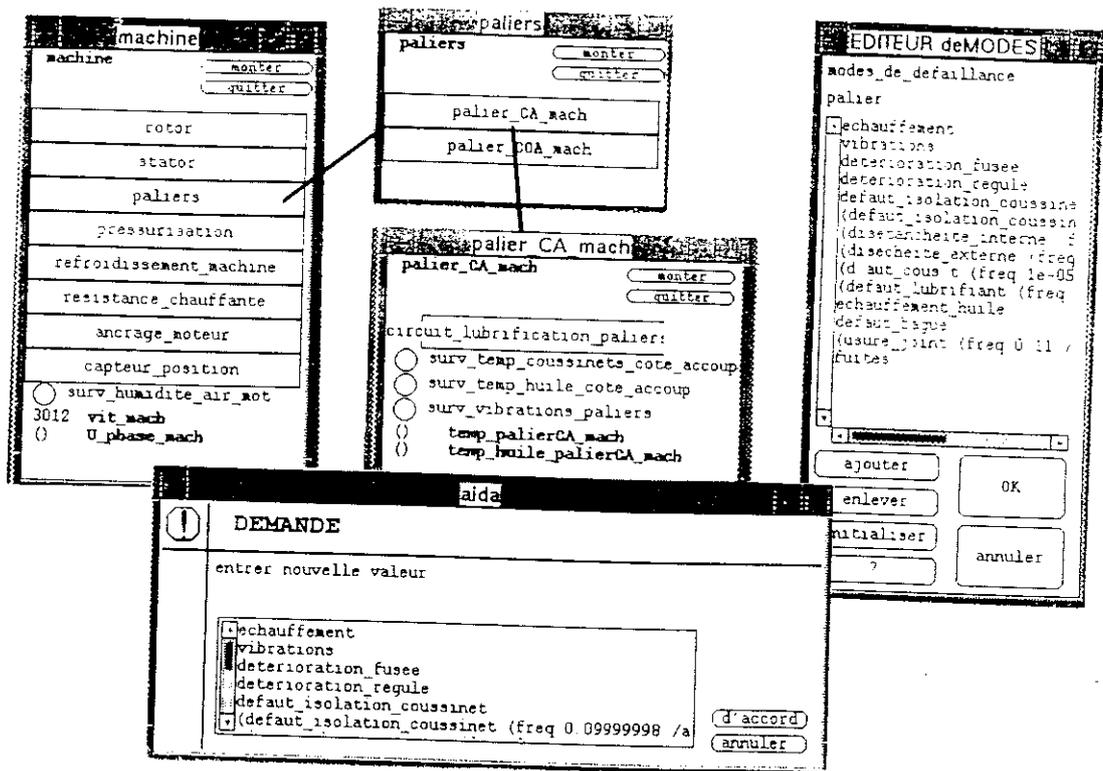


Figure IV.11 Exemple d'entrée d'observations

Maintenant, l'utilisateur peut sélectionner les tests dans l'éditeur de test pour afficher l'explication et ou pour donner des réponses (figure IV.12).

Comme le cycle de raisonnement continue pendant le dialogue, la réponse est prise immédiatement en compte par le raisonnement, ce qui peut résulter en le retrait de tests qui ne sont plus pertinents. Ainsi, la confirmation de l'hypothèse sur le *manque_alimentation_24_V* peut, par exemple, avoir comme conséquence le retrait des tests proposés suite à l'observation de vibrations. Ceci peut s'expliquer par le dysfonctionnement de la signalisation rattachée au capteur de vibrations, et la création d'une fausse alarme par la suite, de sorte que les hypothèses et les tests proposés suite à la fausse alarme sur les vibrations ne soient plus justifiées.

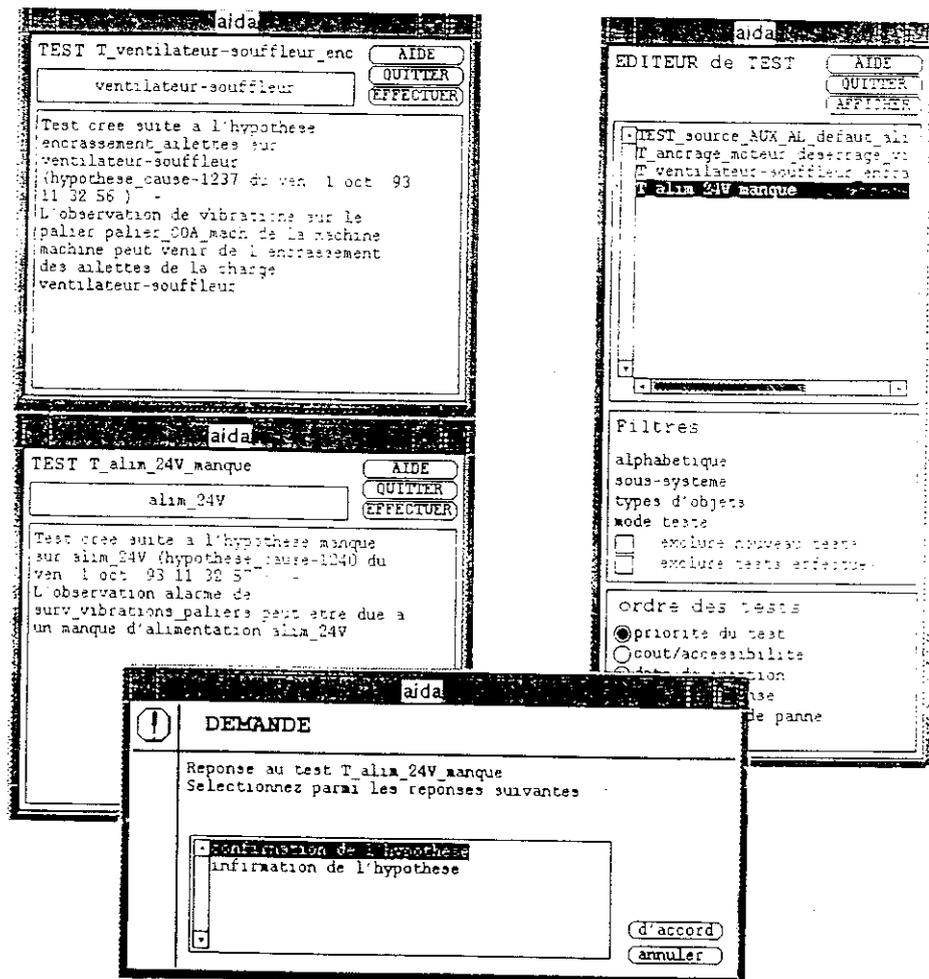


Figure IV.12 Exemple de réponse aux tests proposés

Ces fonctionnalités devraient suffire pour un utilisateur qui veut juste une aide pour son problème. En fait, il ne doit pas être confronté aux détails de fonctionnement du système tels que la gestion des observations, des hypothèses et du cycle de fonctionnement du contrôle.

Par contre, le concepteur du système expert peut, pour des raisons de mise au point, être amené à suivre de plus près le mécanisme d'inférence. Pour ceci il a à sa disposition tout l'environnement de développement de SMECI, dont des éditeurs pour les classes, les objets et des règles, ainsi qu'un mécanisme de trace sophistiqué pour le raisonnement. Ainsi il est facile d'accéder aux règles de diagnostic et de tester leur application (figure IV.13).

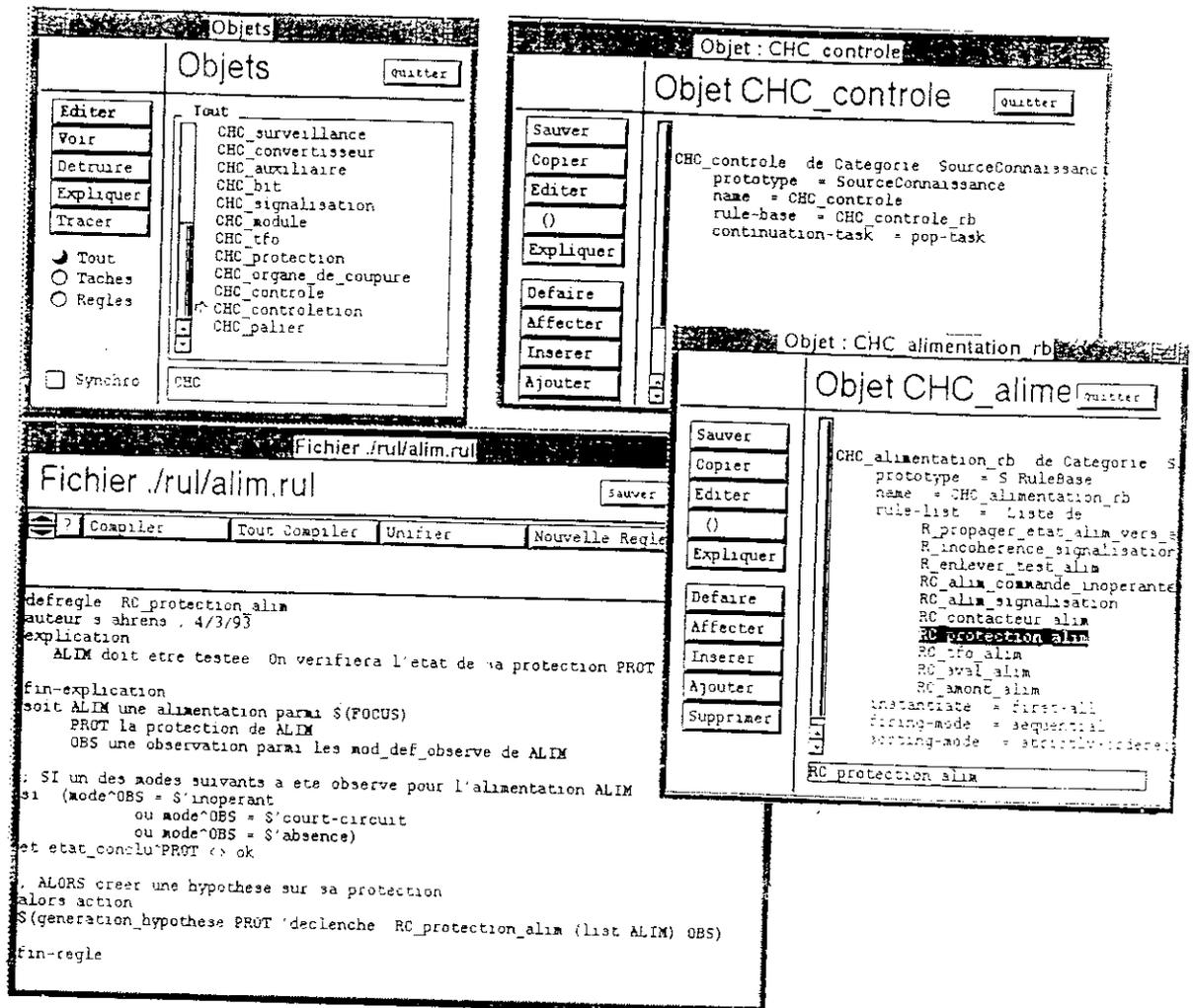


Figure IV.13 Exemple d'outils de trace du système expert

2 Une session concepteur

Outre l'utilisation du système expert par un utilisateur final pour consultation telle que cela a été illustré dans le paragraphe précédent, il peut y avoir **deux niveaux différents de concepteur** qui se distinguent par la complexité et les connaissances requises sur le système expert (cf. début § III).

2.1 Adaptation du système expert à un équipement particulier

Une fois que toutes les règles de diagnostic pour toutes les configurations possibles des équipements sont rentrées dans les différentes sources de connaissances, il suffit de mettre en place la base de données de l'installation. Ceci consiste à créer les instances représentant l'installation à partir des différentes classes existantes et de remplir leurs champs afin de décrire les relations entre les sous-systèmes et leurs composants. Cette tâche ne demande pas de connaissances sur le diagnostic ni sur les structures qui les mettent en œuvre (règles, bases de connaissances, stratégie).

2.2 Création d'un nouveau composant et rajout de nouvelles règles de diagnostic

Cette tâche demande au concepteur des notions sur le formalisme des règles utilisé, sur les sources de connaissances auxquelles les règles sont à relier et une compréhension générale de la stratégie implantée. Cependant, le concepteur ne doit pas intervenir dans les règles stratégiques ni dans le contrôle du raisonnement.

L'ajout d'un nouveau type de composant, par exemple, nécessite les actions suivantes:

1. Créer une nouvelle classe: celle-ci est à insérer dans la classification des classes existantes. On choisira comme classe-mère celle dont la nouvelle classe est une spécialisation afin de réutiliser au mieux le concept d'héritage du formalisme objet, c'est-à-dire des champs déjà définis. Puis, on créera les champs propres (locaux) à la nouvelle classe, par exemple pour référencer des signalisations ou des composants qui lui sont associés, en précisant dans les facettes de ces champs le type des objets référencés (cf. figure III.2, § III.3). Ceci facilitera ultérieurement la création d'objets, lors de la mise en place de la base de données des installations.
2. Définir des méthodes de création d'image pour l'interface graphique du nouveau composant.
3. Créer des sources de connaissance correspondant aux nouveaux composants et leur associer les règles le concernant. Cependant, ceci n'est pas toujours nécessaire. En effet, si les règles ne sont pas nombreuses et si une partie de règles existantes dans les sources de connaissance de la classe-mère sont aussi valables pour les nouveaux composants, on peut les intégrer dans les sources existantes en évitant ainsi d'en créer de nouvelles. Si on décide de créer de telles sources, on y intègre les règles valables existantes et on affectera leurs noms à la définition de la nouvelle classe pour que celles-ci soient connues par la stratégie.

4. Entrer les modes de défaillance et les observations possibles pour le nouveau type de composant et écrire les règles s'y référant.
5. Compléter la classe de l'élément auquel est relié le nouveau composant par un champ qui le référence. Pour pouvoir accéder à celui-ci par l'interface graphique, il faut également ajouter son image synthétique à l'image détaillée de cet élément (cf. annexe III). Ceci peut être facilité par l'utilisation d'un générateur d'interface graphique interactif, nécessitant l'écriture d'un minimum de code.
6. Finalement, créer les différentes règles de diagnostic et les ajouter aux sources de connaissance qui seront activées par une observation anormale.

3 Conclusion

Grâce à son interactivité le prototype reste simple à utiliser, en particulier pour un utilisateur final. Il se présente sous forme d'un système de surveillance permettant l'accès aisé à toutes sortes d'informations relatives au fonctionnement et au diagnostic sur l'équipement. Dans le cas de présence d'anomalies détectées par les capteurs ou par l'utilisateur, il propose des diagnostics ou des actions pour parvenir au diagnostic du défaut.

L'adaptation du système expert à une installation particulière correspond à la mise en place de la base de données pour l'équipement. Pour l'introduction de nouvelles connaissances dans le système expert, le formalisme des sources de connaissances, ainsi que les outils d'édition et de trace de règles facilitent l'ajout de nouvelles règles.

Il reste à relier le prototype du système expert avec un système d'acquisition de données et de compléter les sources de connaissances pour pouvoir le tester dans des conditions réelles.

Conclusion générale

Conclusion générale

Les travaux réalisés dans le cadre de cette thèse, nous ont permis d'aborder le problème du diagnostic dans le domaine des grands entraînements électriques. Pour résoudre celui-ci, parmi les différentes démarches possibles en génie électrique, nous avons choisi les systèmes experts. D'une part, c'est une approche globale qui offre la possibilité d'intégrer les autres démarches qui ne couvrent généralement qu'un aspect partiel du diagnostic. D'autre part, les systèmes experts permettent de bien formaliser la connaissance des experts sur un problème donné (par exemple : règles d'heuristique, savoir-faire) et de la rendre accessible à un non expert.

Ainsi, nous avons réalisé un prototype de système expert pour l'aide au diagnostic et à la maintenance de grands entraînement électriques. Il fournit une aide aux techniciens de maintenance pour détecter, analyser et comprendre des défauts.

Pour réaliser ce prototype de système expert, nous avons dû structurer la connaissance relative aux entraînements (leurs composants, leurs capteurs, etc...) et nous avons dû acquérir, analyser, reformuler et formaliser la connaissance des experts (règles d'heuristique et stratégie). Ce travail est une étape difficile, y compris pour les experts, si bien que nous avons pu acquérir et structurer qu'une partie faible, mais significative, de la connaissance des experts. Cette connaissance s'appuie sur les observations faites sur l'entraînement (signaux des capteurs, observations des techniciens de maintenance...). Ainsi, le système expert a été conçu pour être couplé avec un système d'acquisition de signaux et fonctionner en parallèle avec l'entraînement.

Pour faire collaborer ces différentes connaissances et systèmes, nous avons mis en œuvre le prototype de système expert autour d'une architecture blackboard. Cela facilite d'une part la structuration et exploitation des connaissances, et assure d'autre part une bonne évolutivité et maintenabilité du système expert. De plus, de par sa conception modulaire, celui-ci peut être facilement adapté à diverses applications d'entraînements de même type.

Actuellement, l'expertise implantée dans le système expert n'est que partielle. Cependant, grâce au formalisme des connaissances que nous avons défini, elle pourra être enrichie par des non-informaticiens. En effet, la traduction informatique des règles d'expertise est devenu relativement simple. Cependant, l'extraction et l'analyse de la connaissance reste un problème complexe, purement électrotechnique. Aussi, cette tâche nécessite-t-elle une forte participation des experts.

Maintenant, outre l'enrichissement des connaissances, le système expert doit être connecté au système d'acquisition pour être validé dans des conditions réalistes par des experts.

Ensuite, on pourra envisager d'intégrer dans le système expert d'autres méthodes de détection pour optimiser certains aspects du diagnostic, par exemple pour détecter des défauts locaux, difficilement accessibles dans la machine, ou pour faire du diagnostic prédictif. Ces méthodes pourront être basées, par exemple, sur la redondance analytique ou des réseaux de neurones artificiels.

Annexes

Annexe I

Diagnostic par l'approche système expert en génie électrique

Pour illustrer la diversité des approches de diagnostic par système expert qui existent, nous allons en présenter quelques unes qui nous semblent les plus intéressantes pour le diagnostic de grands entraînements électromécaniques. On distingue essentiellement deux types d'approches :

- * celles basées exclusivement sur les règles de production, c'est-à-dire, les systèmes expert "classiques"
- * celles, plus récentes, dite aussi de seconde génération, qui intègrent des modèles. De ce fait, elles s'appuient sur des méthodes de représentation de connaissance multiples.

1 Approches basées sur des règles de production

Nous allons tout d'abord étudier les approches classiques par système expert, qui s'appuient uniquement sur le formalisme de règles de production pour la représentation de connaissances.

1.1 Approche avec stratégie guidée par les buts

L'idée, dans cette approche, est de vérifier systématiquement tous les défauts connus par le système expert par l'évaluation successive de chaque défaut répertorié dans les règles. Chaque défaut devient un but dont le système essaie de prouver l'existence. La démarche ressemble à un parcours d'un arbre à conditions qui doivent toutes être vérifiées pour l'identification du défaut. Les conditions sont spécifiées par des règles de production qui s'enchaînent les unes après les autres, guidées par les sous-objectifs à vérifier dans leur partie condition. La stratégie du parcours de l'arbre est alors le chaînage arrière.

1.1.1 Exemple

Prenons l'exemple du système expert de Novak et al [NOV 89] qui identifie les défauts au niveau des composants électriques dans un véhicule de transport de minerais. Le système électrique d'un tel véhicule consiste en plusieurs moteurs à courant continu, un transformateur, un redresseur et des circuits de couplage contenant des contacteurs, des protections et une logique de contrôle.

Les règles qui permettent d'identifier les défauts ont été développées à l'aide d'une analyse du schéma électrique du véhicule. Ainsi pour chaque défaut envisagé une analyse qualitative du circuit a fourni un scénario, c'est-à-dire une liste d'observations résultantes qui a ensuite été traduite en règles de production. Une telle démarche a également été utilisée dans [MIT 89] pour le système hydraulique d'un tel véhicule. Voici un exemple de règles de production utilisées dans cette application (figure A-I.1).

OBJECTIF
problème

REGLE 1

SI tension entre points 13 et 14 = normale
ALORS transformateur du circuit de contrôle = ok

REGLE 2

SI transformateur du circuit de contrôle = ok
ET disjoncteur du circuit de contrôle est fermé
ET tension entre points 15 et 26 = 0
ALORS problème = disjoncteur du circuit de contrôle

Figure A-I.1 Exemple d'une règle [NOV 89]

Dans cet exemple le moteur d'inférence commence par évaluer l'objectif à prouver qui est "*problème*". Pour ceci il cherche dans les conclusions de toutes les règles jusqu'à ce qu'il en trouve une qui contient le but à atteindre dans sa partie conclusion. Ainsi le moteur trouve la règle 2 et cherche à montrer que le problème est un problème du disjoncteur de contrôle.

Pour atteindre ce but, le moteur d'inférence cherche à vérifier la première clause de la règle 2. Il cherche d'abord dans sa base de faits si l'attribut de l'objet "*transformateur du circuit contrôle*" est connu. Supposons que l'information n'existe pas dans la base de faits ; dans ce cas, le moteur d'inférence cherche dans la base de règles pour trouver une règle qui évalue l'attribut de "*transformateur du circuit contrôle*" dans sa conclusion, ce est qui est le cas dans la règle 1. Maintenant le moteur d'inférence cherche à prouver que "*la tension entre points 13 et 14 est normale*". A nouveau il cherche d'abord dans la base de faits si l'attribut de "*tension entre points 13 et 14*" est connu et puis, s'il ne trouve pas cette information il cherche une règle lui permettant de l'établir. Comme il n'y a pas d'autres règles pour continuer cette démarche de chaînage arrière, le moteur demande à l'utilisateur de fournir l'information. Si celui-ci répond que la "*tension entre points 13 et 14*" est "*normale*" la règle 1 s'applique et ajoute le fait "*transformateur du circuit de courant contrôle = OK*" à la base de faits.

Maintenant, le moteur d'inférence retourne à la deuxième clause de la règle 2 et cherche à vérifier si le fait simple "*disjoncteur du circuit de contrôle est fermé*" est vrai. Pour ceci le moteur d'inférence cherche ce fait dans la base de faits et, s'il ne le trouve pas va demander à l'utilisateur de vérifier si le fait "*disjoncteurs de circuit de contrôle est fermé*" est vrai. Si l'utilisateur confirme ce fait le raisonnement continue en vérifiant la troisième clause de la règle 2. Si cette clause s'avère aussi vraie, la règle 2 s'applique, le but est atteint et le diagnostic "*problème = disjoncteur du circuit de contrôle*" est établi.

1.1.2 Discussion

Les faits qui sont traités par les clauses dans les règles peuvent être des constantes simples vraies ou fausses ou des structures plus élaborées regroupant plusieurs faits sous forme de données composées (*transformateur de circuit = OK*). Les règles font appel aux faits directement : il n'est pas possible de traiter des variables dans les règles (moteur d'inférence d'ordre zéro). La stratégie du procédé de diagnostic est codée implicitement dans les règles. L'ordre des buts à tester et l'ordre des vérifications à effectuer est spécifié par l'ordre des clauses dans les prémisses des règles. Les faits qui n'existent pas dans la base de faits sont d'abord évalués par des règles, sinon demandés à l'utilisateur.

Ce fonctionnement est proche du formalisme de PROLOG [DEL 88] qui fait aussi un parcours de l'ordre de résolution par chaînage arrière, en profondeur d'abord, mais qui permet de créer des règles plus génériques permettant d'utiliser des variables dans les clauses. Un exemple d'utilisation de PROLOG est le système de diagnostic de locomotives électriques de la société ABB [NAN 91].

1.1.3 Conclusion

Ce formalisme est conceptuellement simple mais très rigide. De ce fait, il impose une grande rigueur dans la formulation des règles. La démarche suivie dans l'exemple présenté consiste à vérifier systématiquement les composants d'un système et ceci dans l'ordre où ils apparaissent dans le circuit électrique. Il n'est pas possible de tenir compte d'une situation spécifique pour établir une priorité des actions proposées ni de prendre en compte la probabilité des pannes.

L'utilisateur du système expert est étroitement guidé par les questions posées au moment de l'évaluation des prémisses des règles. Il est impossible de ne pas répondre à une question car le système se trouverait ainsi bloqué.

La base de faits n'étant que faiblement structurée, la connaissance est presque entièrement représentée par des règles. Celles-ci mélangent la structure du circuit électrique avec des heuristiques de pannes et la stratégie du diagnostic. Ceci pose le problème de la maintenance et de l'évolution de la base de connaissance. Le manque de généralité dans les règles fait que la base de connaissance d'un système à diagnostiquer n'est pas réutilisable par exemple pour des systèmes similaires.

1.2 Démarche par inventaire des défauts

Une démarche très semblable à l'approche ci-dessus a été proposée également pour le diagnostic du matériel électrique dans les postes de distribution d'énergie [KAM 91], de machines tournantes [PRO 90], ou pour le diagnostic d'entraînements alimentés par un convertisseur statique [DEB 91] [DEB 92]. Ici, la connaissance est exprimée également par des règles qui se succèdent, formant ainsi un inventaire structuré de défauts. L'ordre d'application des règles peut se résumer sous forme d'un arbre (figures A-I.2, A-I.3) et dépend des réponses successives de l'utilisateur sur les questions liées aux branchements. L'interrogation du système expert commence par le point le plus haut de l'arbre (c'est à dire la racine) correspondant au niveau d'abstraction le plus élevé, et avance successivement vers le sous-système en panne, puis vers la cause. Une fois le niveau le plus bas atteint dans l'arbre (c'est-à-dire les feuilles de la structure), à chaque cause identifiée est attribuée une proposition de "remède" (figure A-I.3)

1.2.1 Exemples

Les figures A-I.2 et A-I.3 montrent des exemples d'une telle arborescence qui peut se présenter sous forme d'une arborescence de défauts (figure A-I.2) ou d'un arbre de décision (figure A-I.3). Quel que soit le contenu de l'arbre, il peut être réalisé par règles de production parcourues en chaînage arrière comme cela a été présenté dans le chapitre ci-dessus.

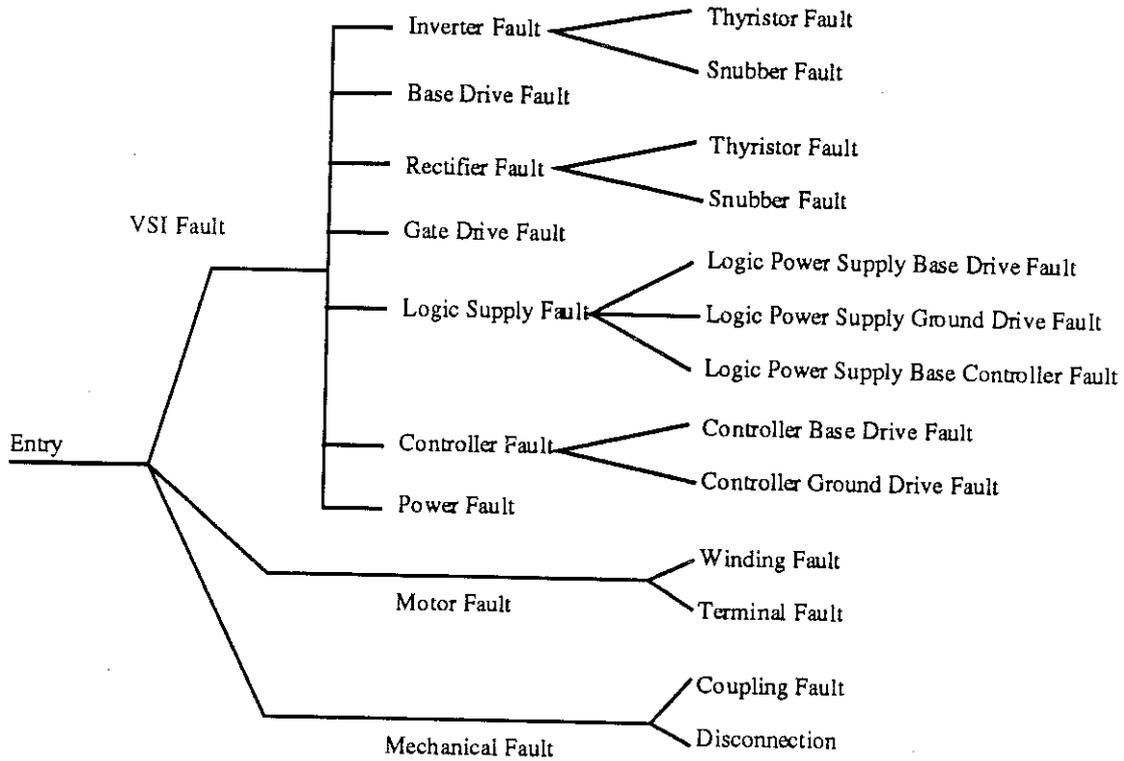


Figure A-I.2 Exemple d'arbre de défauts [DEB 92]

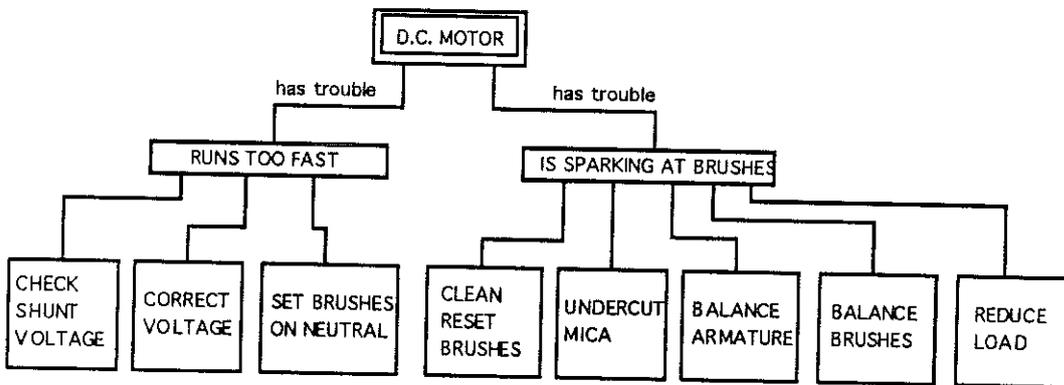


Figure A-I.3 Exemple d'un arbre de décision [PRO 90]

1.2.2 Discussion

L'approche est caractérisée par une vérification successive de toutes les branches correspondant aux défauts envisagés. L'ordre de vérification est codé une fois pour toutes dans les règles. Par conséquent, le déroulement de l'analyse est invariant quelles que soient les informations disponibles au début du diagnostic ce qui présente l'avantage d'avoir un déroulement entièrement prévisible ce qui peut s'avérer important pour des systèmes de diagnostic de temps réel [NAN 91].

Le concept de l'arborescence de défauts est trop rigide pour pouvoir réagir sur des combinaisons variées d'informations en entrée. Le déroulement du diagnostic est invariant car codé dans les règles une fois pour toute. De ce fait la recherche du diagnostic peut être très inefficace pour analyser des situations réelles. Si l'existence d'une information (= un fait) suffit pour explorer successivement les différentes branches, que fait-on si on a plusieurs faits à la fois ? Ceci revient à explorer indépendamment différentes branches. Si on veut tenir compte d'une interprétation spécifique à une combinaison spécifique de faits, il faut créer un sous-arbre avec la condition combinée à sa racine. Dans ce cas, que fait-on avec les autres branches à condition simple ? Faut-il les verrouiller, c'est-à-dire ajouter des conditions supplémentaires qui évitent de les exploiter dans le cas où seulement la combinaison doit être explorée ?

1.2.3 Conclusion

La démarche du type inventaire de défauts est une solution simpliste car elle ne prend pas en compte la complexité du diagnostic dans des situations réelles. Dans un entraînement électrique, un défaut peut se manifester de plusieurs manières et avoir des conséquences sur d'autres sous-systèmes que celui dont il est originaire. De ce fait, il n'est pas toujours possible de décider de son origine ni de l'identifier dans une approche quasi-linéaire.

La rigidité du formalisme pose rapidement des problèmes de mise à jour quand une nouvelle pièce de connaissance doit être rajoutée. L'ajout de nouvelles connaissances est pratiquement inévitable puisqu'il est illusoire de penser qu'on puisse prévoir toutes les combinaisons de symptômes possibles et anticiper toutes les analyses.

Par contre, pour modéliser des actions dont l'ordre est figé est invariant, comme, par exemple, des actions de vérification systématiques suite à certaines alarmes, cette démarche paraît adaptée pour une vérification des causes les plus probables modélisées par un arbre de décision.

1.3 Approche avec stratégie guidée par les données

L'approche décrite précédemment fixe une fois pour toutes la démarche à suivre dans les règles permettant de mettre en œuvre l'arbre de raisonnement. Le déroulement de l'analyse dans cette approche est invariant quelles que soient les informations disponibles au début du diagnostic. Or, l'analyse experte de situations réelles procède d'une manière plus efficace et plus souple, prenant en compte les informations disponibles des capteurs de l'installation pour orienter l'analyse.

Cette démarche peut être modélisée par une approche du type guidé par les données. L'exemple suivant met en œuvre une telle approche dans un système expert de diagnostic de groupes turbo-alternateurs, développée par la Société Westinghouse [CAM 87] [GON 86].

1.3.1 Exemple

Le but de ce système expert est d'améliorer la surveillance d'un ensemble de groupes turbo-alternateurs. Installé dans un centre de surveillance à Orlando (FLORIDA, USA), qui est connecté en ligne aux systèmes d'acquisition de données de 14 groupes turbo-alternateurs dans des centrales réparties dans tout le pays, il fournit à l'opérateur une information sur les conditions de fonctionnement de ses équipements et également des recommandations sur les actions à réaliser. Ce système analyse en ligne et en temps réel les informations de 225 détecteurs sur des conditions détaillées par groupe, collectées et filtrées par les systèmes de surveillances et transmises dans le centre de diagnostic. En cas d'une anomalie, le système centralisé sort un diagnostic en terme de conditions anormales possibles avec un facteur correspondant de confiance.

Le système comporte des connaissances heuristiques issues de séances de discussions d'un groupe d'experts, c'est-à-dire qu'il connaît un certain nombre de défauts et leurs symptômes. Il fait également appel à une base de données contenant les caractéristiques de tous les groupes connectés de même que l'historique de tous les problèmes rencontrés et des solutions correspondantes. Les connaissances sont représentées à l'aide de règles de production comportant des facteurs de confiance. La base de règles est structurée d'une manière hiérarchique (réseau sémantique, figure A-I.5) et exploitée par une démarche guidée par les données.

Le processus de raisonnement commence par une évaluation des données des capteurs ce qui permet de pondérer la contribution de chaque capteur. Les données des capteurs sont traduites en faits et affectées d'un facteur de confiance.

Ce diagnostic sur les capteurs se fait par des règles de production qui s'appliquent directement sur les signaux des capteurs. Ainsi, il est possible d'écarter les contributions des capteurs en panne ou de réduire le poids des capteurs suspects. Un exemple d'un tel fait est celui-ci "*hot coil indicated by normalized temperatures* [0.9]" créé par une règle qui synthétise les signaux de plusieurs sondes thermiques (figure A-I.4).

Aux faits sont associés des facteurs de confiance, pris dans l'intervalle [+1..-1]. Un degré de 1 signifie que le fait est considéré comme définitivement vrai, -1 traduit un fait définitivement faux. Un degré de 0 signifie l'absence d'information sur un fait.

Les règles travaillent directement sur les faits, c'est-à-dire elles contiennent des références directes sur les faits dans leurs conditions et conclusions. La figure A-I.4 montre quelques exemples de règles de production. Les règles disposent de deux coefficients de confiance pour pondérer leur conclusions. Chaque règle peut comporter deux coefficients [HAT 91]. Le premier coefficient peut être interprété comme degré d'encouragement pour la validité de la conséquence si la condition est remplie, c'est-à-dire si le facteur de confiance du fait de la condition est positif. Le deuxième coefficient correspond au degré de découragement pour la conclusion si la condition n'est pas remplie, c'est-à-dire si le fait de la condition a un facteur de confiance négatif.

R1:	IF	arcing exists
	THEN	cracked or broken strands resulting in hydrogen leak (2, 10^{-5})
R2:	IF	hot coil indicated by normalized temperatures
	THEN	cracked or broken strands resulting in hydrogen leak (5, 10^{-3})
R3:	IF	hydrogen leak exists
	THEN	cracked or broken strands resulting in hydrogen leak (300, 1)
R4:	IF	test procedure for detection shows stator winding leak
	THEN	cracked or broken strands resulting in hydrogen leak (... ..)

Figure A-I.4 Exemples de règles de diagnostic d'un turbo-alternateur [GON 86]

Le coefficient du fait conclu se calcule en fonction des coefficients des faits de la partie condition (partie IF) et des coefficients propres de la règle. L'exploitation des coefficients pour la stratégie de diagnostic est mise en œuvre par le moteur d'inférences.

Plusieurs règles peuvent contribuer indépendamment à une conclusion, la contribution de chacune consistant à la modification du facteur de confiance du fait conclu. On dit que les règles forment un réseau d'inférences [HAT 91] [WAT 86] car leur interaction peut être

représentée graphiquement par un réseau. La figure A-I.5 illustre le réseau d'inférences mis en œuvre par les règles de la figure A-I.4.

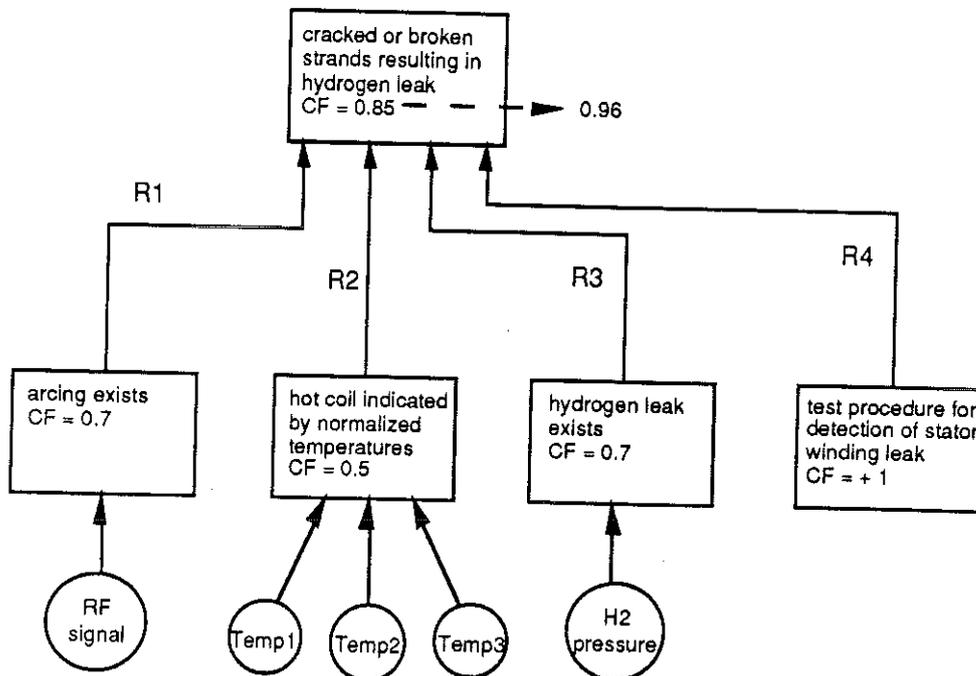


Figure A-I.5 Réseau d'inférences mis en œuvre par les règles de la figure A-I.4 [GON 86]

Le rôle du moteur d'inférences est de propager chaque modification d'un coefficient de confiance d'un fait à travers le réseau. Dès qu'un coefficient d'un fait change ou un nouveau fait est rentré dans la base, toutes les règles ayant ce fait dans leur partie condition sont (ré)évaluées automatiquement, et le poids des conclusions qui en dépendent se trouve ainsi modifié. Le système expert dispose donc à tout moment d'une liste d'hypothèses ordonnée par leur coefficient de confiance. L'hypothèse avec le score le plus élevé représente le diagnostic en cours. Il se peut que, par manque de faits, toutes les règles n'aient pas été prises en compte pour renforcer le diagnostic. Dans ce cas le moteur peut modifier sa stratégie et chercher les règles les plus intéressantes pour renforcer le diagnostic. Pour ceci il adopte un comportement guidé par les buts et remonte les conditions des règles pour trouver le fait non existant le plus pertinent pour augmenter la confiance du diagnostic.

Puis le système demande à l'utilisateur de fournir ce fait, ce qui correspond normalement à une mesure ou un test à effectuer. Dans l'exemple de la figure A-I.5 le fait "*detection of stator winding leak*" n'est pas inclue dans la base initiale de faits de sorte que la règle R4 ne s'applique pas au début. Ce n'est qu'après que l'utilisateur ait donné le résultat du test (en lui affectant, par exemple, un degré de confiance de 1) que la règle R4 est déclenchée. Sa contribution fait monter la confiance accordée au diagnostic "*cracked or broken strands resulting in hydrogen leak*" de 0,85 à 0,96.

1.3.2 Discussion

Ce fonctionnement est plus souple et moins contraignant par rapport aux systèmes décrits précédemment. Ceux-ci ne peuvent pas conclure sur une règle, si un fait de la condition indispensable pour la preuve d'une hypothèse n'existe pas dans la base de faits et ainsi l'hypothèse ne peut pas être établie (pas de diagnostic, au pire des cas).

Le raisonnement présenté ici, par contre, peut pallier au manque d'un fait : le résultat en est simplement que l'hypothèse n'atteint pas son meilleur score sans demande d'informations supplémentaires à l'utilisateur.

Ainsi les faits difficiles ou coûteux à apporter jouent le rôle de confirmer le diagnostic par un test ou une action de mesure supplémentaire.

Une fois que la meilleure hypothèse a atteint un coefficient suffisamment élevé, elle est présentée comme le diagnostic et les actions curatives associées à celles-ci sont proposées à l'utilisateur. Pour ce dernier (et aussi le système d'acquisition des données), cette approche offre l'avantage de pouvoir fournir à tout moment une information quelconque au système expert mais aussi la possibilité de ne pas répondre à une question posée.

1.3.3 Conclusion

Conceptuellement le fait que cette approche soit basée sur un raisonnement incertain est un avantage parce qu'elle reflète bien la réalité du diagnostic. Un expert lui aussi ne dispose souvent que de données incomplètes, incertaines et imprécises pour arriver à son diagnostic.

Cependant la mise en œuvre de ce raisonnement à travers les coefficients de confiance dans les règles n'est pas évidente à implanter pour le concepteur du système de diagnostic technique.

Issus à l'origine d'un système expert d'aide à la prospection de minerais PROSPECTOR [HAT 91], [WAT 86], les coefficients dans les règles avaient la signification de probabilités conditionnelles et de données statistiques existant dans ce domaine.

Faute de données correspondantes pour un système technique, il faut choisir des valeurs arbitraires pour initialiser les règles. Le choix de ces coefficients peut s'avérer une tâche fastidieuse nécessitant plusieurs itérations jusqu'à ce que le raisonnement donne les résultats escomptés.

2 Approches basées sur des représentations multiples de connaissances

Dans les approches ci-dessus, la connaissance du diagnostic est essentiellement modélisée par des règles, la stratégie de son exploitation étant implicitement fixée et non explicitée par une représentation appropriée, les faits servant d'entrée pour les règles et de conclusions intermédiaires n'étant que faiblement structurées.

L'inconvénient principal d'un manque de structuration explicite des connaissances et de la stratégie est la maintenabilité difficile du système expert. Ainsi la mise au point devient lourde quand la base de connaissances est enrichie par de nouvelles règles. Des exemples ont montré qu'il fallait abandonner un outil basé uniquement sur le formalisme de règles de production en raison de son manque de puissance de structuration [NOV 89] [MIT 89] [MIT 91].

Remarque: nous mettons en garde le lecteur sur le fait que pour présenter les méthodes qui suivent, nous avons dû le faire autour d'applications spécifiques (DIVA, STEAMER, SEPT) pour illustrer le problème de collaboration entre les différentes méthodes. En effet, les méthodes utilisées ainsi que les stratégies utilisées sont fortement liées à l'application traitée.

2.1 Approche avec classification de scénarios de défauts

Par la suite, nous présentons un exemple qui, entre autres, fait également appel aux règles de production, mais qui dispose de modèles pour structurer explicitement la connaissance du domaine et la stratégie appliquée. La proximité du modèle de représentation utilisé à l'approche des experts (à leur "modèle mental") a permis de résoudre le problème complexe du diagnostic vibratoire [DAV 90]. L'exemple montre que le raisonnement peut être très complexe et nécessiter plusieurs modèles de représentations de connaissances et différentes stratégies.

2.1.1 L'exemple du système expert de diagnostic vibratoire des arbres DIVA de EdF

Le système expert DIVA [GUI 90] [MOR 87] [DAV 87] a été développé par EdF en collaboration avec GEC-ALSTHOM et les Laboratoires de Marcoussis. Commencé en 1985, il a atteint le stade de prototype validé en 1989. Il est déjà opérationnel pour analyser des situations hors ligne, mais son intégration au poste de surveillance et d'aide au diagnostic dans les centrales nucléaires est prévue après que sa base de connaissances aura été complétée.

DIVA est basé sur un système de surveillance vibratoire qui a été mis en service à EdF en 1978 pour gagner une bonne image de l'état de santé d'une machine à partir de son comportement vibratoire. Le système d'acquisition enregistre et traite les principaux paramètres d'exploitation : les vibrations, la vitesse, la puissance, les températures, etc...

A partir de ces enregistrements et d'une connaissance du contexte d'exploitation, le diagnostic consiste à identifier la ou les causes responsables d'une évolution des vibrations et, en particulier, de déterminer les conséquences pour trouver les mesures à prendre. L'apport principal de DIVA consiste à interpréter les données restituées par le système d'acquisition qui sont peu parlantes pour un exploitant de centrale. L'objectif de cette interprétation est d'identifier des situations typiques dont les causes et l'évolution sont connues.

Pour ce faire, la base de connaissances contient des descriptions de situations typiques (appelées "prototypes"), structurées dans une hiérarchie (figure A-I.6). Un prototype comprend les manifestations des défauts dans un contexte d'exploitation et une relation à d'autres prototypes plus spécialisés. Le diagnostic commence au plus haut niveau (situations générales) et se dirige vers les prototypes spécialisés à des problèmes précis. La stratégie générale est du type "reconnaître/raffiner" et consiste en un parcours itératif de la hiérarchie des prototypes, essayant d'arriver au plus profond. Pour ce faire, il réitère sur chacun des prototypes envisagés le cycle suivant :

- acquisition d'information dans le contexte du prototype courant,
- évaluation de la correspondance entre la description typique et la situation réelle,
- choix du nouveau prototype courant.

Le diagnostic final est élaboré à partir des différents prototypes reconnus. La représentation des connaissances fait appel aux méthodes orientées objet. Chaque prototype comprend des descriptions et des méthodes pour identifier une situation. Ceci permet d'utiliser plusieurs méthodes de raisonnement et donc, pour chaque prototype, la méthode la mieux adaptée.

La hiérarchie de prototypes (figure A-I.6) représente le processus du diagnostic mais ce n'est ni un arbre de décision, ni une arborescence de défauts.

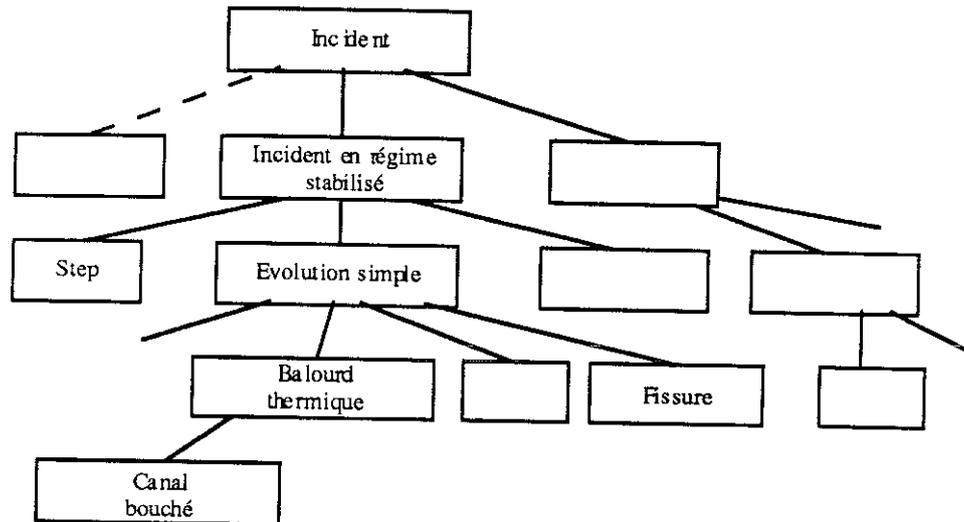


Figure A-I.6 Exemple d'une vue partielle de la hiérarchie des prototypes [MOR 87].

- Ce n'est pas un arbre de décision car chaque nœud dans l'arbre correspond à une situation qu'il décrit et qui peut être donnée comme diagnostic intermédiaire. Le parcours de cette hiérarchie est heuristique, et les situations sont toujours plus ou moins bien reconnues et plusieurs hypothèses peuvent être menées en parallèle.
- Ce n'est pas non plus une arborescence de défauts élémentaires (ex. frottement, perte d'ailette, etc...) car les manifestations peuvent varier considérablement suivant leur stade de développement ou les conditions d'exploitation.

A l'inverse, un prototype décrit un défaut en termes de ses manifestations attendues. Le même défaut élémentaire peut donc être à la racine de plusieurs prototypes, chacun décrivant une manifestation typique différente.

La description brève des étapes du cycle de fonctionnement mettra en évidence la représentation des connaissances et les stratégies de contrôle locales pendant chaque phase de raisonnement.

2.1.2 Stratégie de diagnostic

2.1.2.1 L'acquisition d'informations

L'acquisition d'informations pour le prototype correspond à remplir les composants du prototype avec les données du système de surveillance. Un composant est un objet (par exemple un paramètre d'exploitation comme la puissance active, températures de vapeur, d'huile, etc....) qui regroupe plusieurs attributs décrivant le composant. Pour chaque

attribut, il y a des informations sur la valeur attendue, les méthodes à utiliser pour acquérir l'information, etc... La figure A-I.7 montre un exemple d'un attribut.

Nom de l'attribut: *sensibilité H2*
 auteur :
 définition:
 propriété de: *évolution de vibration*
 prérequis: *toujours vérifié*
 genre: *monovalué*
 pondéré: *oui*
 comment: *système d'acquisition, utilisateur, historique*
 texte: *La vibration vous semble-t-elle sensible à la pression H2 ?*
 valeurs-possibles: *{oui, non}*
 pour-déduire:

Figure A-I.7 exemple d'un attribut pour l'acquisition de données [DAV 87]

L'ordre des composants à remplir peut être spécifié pour chaque prototype. La façon d'évaluer un attribut d'un prototype, c'est à dire comment remplir un champ, est spécifiée localement dans les informations qui caractérisent un attribut (appelé "slots" ou "facettes" dans la littérature), par exemple, question à l'utilisateur, consultation d'une base de données historiques ou déduction à partir d'autres faits. Dans ce dernier cas, les règles utilisables sont attachées à l'attribut dans le slot *<pour déduire>* (ce qui peut entraîner de nouvelles évaluations en chaînage arrière, cf. chapitre 2.1.1), une méthode particulière (analyse d'un spectre ou calcul d'amplitude d'un step, c'est à dire d'un changement brusque) ou requête au système d'acquisition. Dans l'exemple de la figure A-I.7, le slot *<comment>* spécifie que la valeur de l'attribut *<sensibilité Hz>* est d'abord demandé au système d'acquisition. Si elle n'est pas disponible, une question sera posée à l'utilisateur, et si celui-ci ne sait pas répondre, la valeur sera cherchée dans une base de données historique.

L'acquisition de données sur le prototype courant se termine lorsque tous les composants sont évalués ou lorsque la valeur obtenue pour un composant correspond à une valeur de rejet. Dans ce cas, le prototype est infirmé et il n'est pas nécessaire de poursuivre son évaluation.

2.1.2.2 L'évaluation du prototype

L'évaluation de la correspondance entre la description du prototype et les valeurs acquises peut se faire de plusieurs manières, spécifiées au niveau du prototype. Une méthode consiste à noter chaque composant en mesurant la distance entre sa valeur attendue et observée. Une combinaison de ces notes, pondérée par l'importance relative de chacun des composants donne la plausibilité du prototype.

Une autre méthode, plus générale, consiste à utiliser des règles pour juger globalement de l'adéquation de la situation réelle avec la description prototypique. Le fait de coupler au prototype même le soin de se reconnaître, permet alors d'utiliser des méthodes différentes et de mieux adaptées à une situation.

Trois conclusions de cette étape sont possibles :

- le prototype est confirmé car son coefficient de plausibilité indique que les valeurs observées correspondent dans une certaine mesure à ce qu'on attendait,
- le prototype est infirmé car la situation réelle ne correspond pas à ce qu'on attendait,
- le système juge qu'il n'en sait pas assez pour conclure car trop d'informations manquent.

2.1.2.3 Choix du nouveau prototype

Le choix du nouveau prototype à examiner permet de raffiner le diagnostic. Le système expert va alors sélectionner pour les prototypes reconnus celui qui est le plus plausible et puis va choisir un des fils (un prototype en aval dans la hiérarchie) de ce dernier comme nouveau prototype courant pour le prochain cycle. Les connaissances de contrôle permettant de sélectionner un des fils d'un prototype reconnu sont également attachées au prototype (slot *<pour raffiner>*). Ce sont des règles qui déterminent une priorité sur les prototypes-fils à envisager ou à ne pas envisager. La priorité n'est pas une plausibilité mais une connaissance de stratégie (un prototype peut être évoqué en priorité parce que facile à infirmer bien qu'a priori peu plausible). Lorsque tous les prototypes évoqués ont été traités, le système s'arrête. Le diagnostic final est constitué des différents prototypes reconnus.

2.1.3 Conclusion

Nous avons essayé de montrer à travers la description du fonctionnement de DIVA la forte structuration des connaissances mises en jeu par le diagnostic vibratoire. La connaissance s'articule autour de descriptions de situations typiques classifiées dans une hiérarchie.

Les prototypes sont des structures de données complexes auxquelles sont attachées des connaissances opératoires (des règles, des procédures, etc...) permettant de reconnaître les situations. On peut appréhender chaque prototype comme un petit système expert qui, quand il est invoqué par la stratégie globale, met en œuvre son propre raisonnement pour identifier la situation qu'il connaît.

Les concepts utilisés dans DIVA peuvent apparaître complexes mais ils représentent au mieux la démarche des experts du domaine. Ce type de collaboration de différentes méthodes s'avère difficilement généralisable d'une type d'application à l'autre.

Ainsi, ce modèle n'est pas transposable au diagnostic dans le domaine des grands entraînements. Dans ce dernier, la connaissance ne correspond pas à une classification de situations de défauts typiques. En fait, les défauts auxquels on est confronté dans les grands entraînements sont le plus souvent atypiques bien que les manifestations se ressemblent souvent. Il n'y a pas non plus des situations où les manifestations des défauts évoluent avec le temps ou le mode de fonctionnement, sauf peut être pour le transformateur de puissance, la machine ou la charge entraînée.

Par contre, l'instrumentation de ces composants se fait à l'heure actuelle par capteurs à seuils. Les informations "tout ou rien" que donnent ces capteurs (protections thermiques, sondes de vibrations avec seuil d'alarme, etc...) ne permettent pas une surveillance continue assez fine pour détecter des dégradations lentes causées par le vieillissement ou l'usure.

Une conséquence de cette situation est un manque de connaissances sur ces phénomènes lents dans le domaine des entraînements.

2.2 Approche avec représentation explicite de l'installation

Tandis que les connaissances dans l'approche précédente sont structurées autour d'une classification de situations typiques de défauts, d'autres systèmes experts utilisent un modèle structurel et fonctionnel du système à diagnostiquer sur lequel s'appuie un raisonnement à base de règles.

Ce type de modélisation est souvent utilisée pour des procédés complexes où sont présentés différentes technologies mécanique, électrique, électronique et informatique et où les pannes sont aussi de nature complexe (causes possibles multiples pour un symptôme donné). Une telle approche consiste à modéliser d'une part explicitement le système à diagnostic et d'autre part la stratégie du diagnostic. Dans la suite, nous allons présenter les méthodes utilisées en les illustrant à l'aide de l'exemple du système expert STEAMER. Ici, le modèle de l'équipement est utilisé pour mettre en œuvre la stratégie du diagnostic.

2.2.1 L'exemple du système expert STEAMER

STEAMER est un système expert d'aide à la maintenance d'un émetteur de télévision [WOD 90]. Une telle installation est très complexe et, elle fait intervenir des techniques variées comme l'électronique, la mécanique et l'hydraulique.

2.2.1.1 Représentation de l'installation

STEAMER utilise une décomposition structurelle en blocs de ce matériel. Un bloc représente un élément de l'équipement qui peut être défaillant et qui peut lui-même contenir d'autres blocs (figure A-I.8).

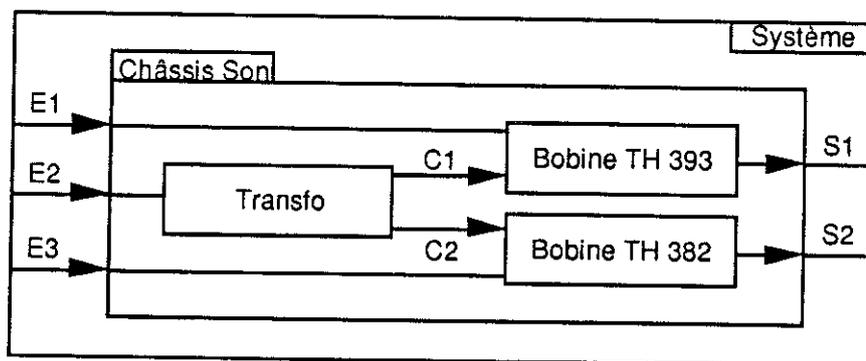


Figure A-I.8 Exemple d'une représentation structurelle dans STEAMER [WOD 90]

Les blocs sont reliés entre eux par des connexions qui véhiculent des signaux caractérisés par des grandeurs physiques mesurables.

On remarque que chaque **bloc** représente une fonction qui peut être défaillante et est successivement décomposable en un sous-ensemble de blocs et connexions. Cette hiérarchisation permet de structurer l'équipement afin de réduire la complexité à un niveau de décomposition de profondeur donné.

Dans STEAMER, un bloc est caractérisé par différents aspects par exemple :

- son état, qui est correct, incorrect, suspect ou inconnu,
- son type qui peut être un des éléments prédéfinis (par exemple relais, fusibles, ventilateurs, etc...), indiquant alors un comportement particulier, connu du système,
- une note d'accès, mesurant la facilité d'accéder au bloc pour effectuer des mesures,
- une note de sécurité, traduisant la présence de tensions ou températures élevées,
- une note de fiabilité, qui rend compte de la fiabilité intrinsèque du composant.

Chaque bloc contient des fonctions de transfert pour ses sorties. Une telle fonction de transfert est une relation qualitative entre une sortie d'un bloc et ses entrées:

$$FT(\text{Sortie, Module}) = \{\text{Entrées}\}.$$

La philosophie d'utilisation de ces fonctions de transfert est, comme on le verra par la suite, de pouvoir identifier les signaux d'entrée d'un bloc dont dépend un signal de sortie. Par exemple, dans la figure A, S1 est la sortie de la bobine TH393, sa fonction de transfert est donc $FT(S1, \text{Bobine TH393}) = \{E1, C1\}$.

Mais S1 peut également être considéré comme sortie du *châssis son* et dans ce cas, sa fonction de transfert est $T(S1, \text{châssis son}) = \{E1, E2\}$. Il existe alors plusieurs fonctions de transfert pour un signal, une pour chaque perspective c'est-à-dire niveau de décomposition considéré.

Une connexion est caractérisée par la nature du signal qu'elle porte (commande, vidéo bande de base, etc...), son état, sa fonction de transfert et par son état de référence. Il est important pour le fonctionnement de ce système expert que tous les signaux puissent être comparés à leur valeur de référence qui doit être constant dans le temps.

2.2.1.2 La stratégie

La stratégie du diagnostic est fondée sur l'idée que l'observation d'un signal incorrect en sortie d'un bloc peut être provoquée soit par l'état incorrect d'une des entrées du bloc, soit par la défaillance du bloc même. L'hypothèse sous-jacente est que l'utilisateur doit être capable de comparer l'état de chaque signal qu'il observe ou mesure avec un état de référence

constant dans le temps. Le raisonnement prend la forme du cycle de propagation de signaux anormaux, sélection et vérification d'hypothèses suivant :

a. Sélection de capteurs

La démarche consiste à utiliser d'abord des règles d'expertise qui sélectionnent un certain nombre de signaux de capteurs (alarmes) dont les états vont être propagés dans le réseau des blocs et des connexions. Un capteur est toujours lié à une connexion dont il mesure l'état du signal (booléen: correct ou non). Grâce aux fonctions de transfert, une propagation de cet état aux connexions et blocs amont est possible.

b. Propagation des états signalés

Lors de la propagation des signaux des capteurs sélectionnés, les états corrects permettent d'innocenter les blocs amont, tandis que les états incorrects vont suspecter ces blocs. La propagation porte seulement sur les blocs d'un seul niveau de décomposition considéré dans le cycle actuel de STEAMER

c. Sélection d'une hypothèse

L'opération précédente crée une liste de blocs suspects. Un critère faisant intervenir les notes affectées à ces blocs, la présence de témoins complémentaires et des considérations d'ordre heuristique, permet de choisir le bloc le moins coûteux (en termes de temps, de facilité d'accès, de sécurité, etc...) à vérifier.

d. Vérification de l'hypothèse

Vérifier un bloc consiste à vérifier ces sorties suspectes. Pour chacune de celles-ci, on vérifie les entrées correspondantes par la fonction de transfert. Si une des entrées au moins est dans un état incorrect, on propage alors cette information en amont et on continue la recherche. Si toutes les entrées sont correctes, le bloc vérifié est alors déclaré incorrect. Dans ce cas, le bloc défaillant devient le bloc système (on descend dans le niveau de décomposition) et on réitère le processus de recherche jusqu'à ce qu'on soit arrivé sur un bloc terminal, c'est-à-dire ne contenant pas d'autres blocs. Dans ce cas le diagnostic s'arrête et fait place à l'aide à la réparation.

2.2.2 Discussion

Autour de l'exemple du système expert STEAMER, nous montrons comment la structure de l'installation peut être utilisée par une stratégie générale de diagnostic. Dans ce sens, le système expert présenté peut être appréhendé comme un système à base de modèle et comparé à l'approche basée sur les modèles "profonds" comme nous l'exposons dans le chapitre 3 de cette annexe. Cependant, sa stratégie de diagnostic fait appel aux connaissances

heuristiques pour sélectionner l'hypothèse la plus pertinente à poursuivre. Son modèle de comportement n'étant pas détaillé, il ne permet pas une simulation fine du comportement des composants. Il se contente d'identifier des sorties des composants comme normal ou anormal, en se basant sur des états de référence connues et constants dans le temps qui est aussi de la connaissance heuristique.

La description des éléments de l'installation, les blocs et les connexions, a été réalisée par une approche orientée objet [MEY 88]. Ceci a permis d'ajouter facilement d'autres fonctionnalités comme un module graphique et un module d'aide à la réparation qui repose sur des photos des composants et des instructions détaillées de démontage et réparation. Ces fonctionnalités supplémentaires sont un atout important du système. Peut-être faut-il plutôt considérer le système de diagnostic comme un fonctionnalité importante d'un système d'aide à l'exploitation de l'installation ?

La décomposition en blocs et connexions d'une installation se fait selon un formalisme relativement rigide, ce qui ne facilite pas cette tâche. Comme cette décomposition en blocs et connexions est la base de la stratégie de recherche, il faut déjà tenir compte des nécessités du diagnostic lorsque cette décomposition est faite.

2.2.3 Conclusion

Dans le prototype que nous proposons à partir du chapitre III, nous avons fait une décomposition des entraînements en modules, semblable à la décomposition en blocs décrit précédemment. L'idée initiale de faire guider le diagnostic par le réseau des modules comme cela est fait dans l'exemple correspond bien à la démarche des experts dans certains cas, par exemple, lorsqu'il s'agit de remonter un signal douteux dans une chaîne de commande. Cependant, cette approche n'est pas suffisante pour la stratégie globale du diagnostic des entraînements.

En fait, l'idée est fondée sur l'hypothèse que les effets des pannes restent limités aux modules qu'elles affectent. Généralement, ceci n'est pas le cas dans un entraînement : un défaut électrique dans la machine, par exemple peut se manifester par une surcharge constatée au niveau du convertisseur.

Dans ce cas, une stratégie comme celle citée dans l'exemple dirigerait la recherche du défaut un module convertisseur au lieu du module machine. Ce petit exemple montre que la stratégie unique ne s'applique pas dans toutes les situations. De ce fait, il n'est pas possible, dans le domaine des entraînements, de faire guider le raisonnement uniquement par un

parcours systématique de sa structure, comme cela est montré dans l'exemple de STEAMER.

2.3 Approches combinant des représentations de l'installation et de scénarios de défauts

Comme précédemment, pour en expliquer les principes, nous nous appuyons sur un exemple d'une application précise. Avec SEPT, nous présentons un exemple dans un domaine de l'électrotechnique qui a connu de nombreux travaux de recherche sur l'application de systèmes experts pendant la dernière décennie: le diagnostic et la gestion de réseaux de transport d'énergie [WOL 86] [TAL 86] [FUJ 86] [LIU 86] [KAL 87] [CAR 88] [KER 89] [NEB 90] [BIJ 91] [MAR 91] [BES 93] etc...

Dans l'exemple choisi, on fait collaborer une modélisation explicite de l'équipement à diagnostiquer avec des modèles de fonctionnement correct de ses composants et des heuristiques sur leurs défauts caractéristiques.

Remarque: La stratégie présentée dans l'exemple suivant est particulière au domaine du diagnostic de réseaux, mais elle n'est pas non plus exclusive pour ce type de problèmes.

2.3.1 L'exemple du système expert SEPT

Le domaine d'application de SEPT est le diagnostic de matériel de protection dans un poste à très haute tension (Suivi d'Evènements dans les Postes de Transport) du réseau électrique de EdF [BRE 90] [FAU 91]. Les protections et automates situés dans les postes réagissent automatiquement et provoquent le déclenchement de disjoncteurs pour isoler la partie du réseau en défaut après un incident (par exemple coup de foudre sur une ligne ou court-circuit sur un transformateur). Le système de protections est très complexe en raison de la connexité et la redondance entre les différents types de protections. La méthode de surveillance du système des protections consiste à enregistrer, lors d'un incident sur le réseau, le compte-rendu des informations émises par les protections. L'analyse de ces enregistrements permet de déduire à posteriori l'incident qui s'est produit sur le réseau et sa localisation, et de détecter des pannes latentes dans les protections et les disjoncteurs.

L'analyse des enregistrements de défaut est un problème difficile en raison de la dualité réseau - système de protection. Faire un diagnostic sur l'état du réseau à partir des informations des protections peut être problématique en cas d'anomalies multiples du

système de protection. De même, porter le diagnostic sur le système de protection nécessite de connaître le défaut initial qui s'est produit sur le réseau.

Le système expert SEPT aide l'opérateur dans l'analyse des incidents. Il porte un diagnostic en ligne dès qu'un incident se produit et fournit des explications à l'opérateur.

2.3.1.1 La stratégie

La stratégie de SEPT consiste à établir le diagnostic en quatre étapes :

a. La vérification du fonctionnement des protections

Elle se base sur la comparaison entre le comportement observé et un modèle de fonctionnement prévu par les plans de protections. Ceci nécessite la représentation explicite de toutes les protections et de leur bon fonctionnement.

Les connaissances ont été structurées fonctionnellement selon un **scénario** qui exprime le séquençement dans le temps d'apparitions de signaux émis par les protections. Un scénario est attaché à chaque type de protection décrivant son fonctionnement en situation normale et dans les principales situations d'anomalies. Il est composé d'un ensemble de **règles** qui expriment des conditions logico-temporelles C_i décrivant l'apparition de signalisation à un instant t_i fonction des instants des conditions précédentes (figure A-I.9).

R-PXI-ZON2-ACC 1 :

SI

- C0 : PXi.ZON2(t0)
- C1 : PXi.MR.DEB(t1) . t0 - T.PXi.STAD2NAT < t1 < t0
- C2 : PXi.AVAL(t2) . t0 - T.PXi.STAD2NAT < t2 < t0
- C3 : PXiTAC.R(t3) . t3 ≈ t0
- C4 : DT.PXi(t4) . t4 ≈ t0
- C5 : [DT.PXi.0(t5) . t5 ≈ t0] ∨ [DT.PXi.4(t6) . t6 ≈ t0] ∨ [DT.PXi.8(t7) . t7 ≈ t0]

ALORS

le scénario correspondant est correct.

"Déclenchement de la protection PXi en deuxième stade accéléré à l'instant t0. Un défaut est vu par cette protection sur la ligne en deuxième zone de l'instant t1 à l'instant t0"

$C0 \wedge \neg C1$: " Anomalie protection PXi, apparition d'une information de deuxième zone à l'instant t0, absence d'apparition de la mise en route dans les T.PXi.STAD2NAT millisecondes qui précèdent."

$C0 \wedge \neg C2$: " Anomalie protection PXi, apparition d'une information de deuxième zone à l'instant t0, absence d'apparition du directionnel aval dans les T.PXi.STAD2NAT millisecondes qui précèdent."

$C0 \wedge \neg C4$: " Anomalie protection PXi, apparition d'une information de deuxième zone à l'instant t0, absence d'apparition simultanée de la signalisation de déclenchement DT.PXi."

Figure A-I.9 Exemple d'une règle de test pour un type de protection dans SEPT [BRE 90]

b. Découpage en incidents élémentaires

Cette étape découpe un enregistrement de signaux issus des protections en plages temporelles correspondant chacune à un incident élémentaire. Ceci est nécessaire parce que souvent, plusieurs incidents élémentaires figurent sur un même enregistrement, enchevêtrés dans le temps. Un court-circuit franc sur une ligne, par exemple, va créer deux incidents élémentaires : d'abord un déclenchement avec réenclenchement automatique après un certain temps (dû au fonctionnement des automates) et puis un deuxième déclenchement.

Ce déclenchement s'appuie sur des heuristiques de filtrage d'intervalles, par exemple, sur le fait que le fonctionnement des protections doit être quasi synchrone et qu'il doit être compris dans une certaine plage de temps.

c. Estimation de l'ouvrage en défaut

Pour chaque intervalle de temps, on estime le lieu du défaut en utilisant une technique de vote. A priori, on considère que chaque ouvrage du réseau (lignes, transformateurs, disjoncteurs), peut être le siège de défaut et on crée des hypothèses (des candidats).

Puis, on utilise le comportement des protections ayant été identifiées comme fonctionnant correctement dans la phase de vérification pour voter en faveur ou au détriment des candidats. Le diagnostic retenu est le candidat qui a obtenu le plus grand score. L'exemple dans la figure A-I.10 illustre la technique de vote.

Si, pour un départ ligne, un scénario de fonctionnement en zone 1 de protection de distance a été déclaré valide, alors,
la ligne "première partie" (0 à 70 % de sa longueur) est "accusée" avec un poids de + 1,
la ligne "partie extrémité" (70 à 100% de sa longueur), le poste voisin, les barres, sont "innocentés" avec un poids de -1.

Figure A-I.10 Exemple d'une règle de vote [FAU 91]

d. Synthèse

Cette partie fait la synthèse entre les résultats des étapes précédentes. Ainsi il est possible de vérifier la cohérence globale entre le fonctionnement du système de protection et l'estimation du lieu du défaut. Dans cette phase des règles de production sont utilisées pour reconnaître et synthétiser le type du défaut (par exemple permanent ou fugitif) et pour donner l'explication à l'utilisateur.

2.3.1.2 Représentation de l'installation et des faits pour le raisonnement

SEPT utilise le **formalisme objet** (cf. chapitre II.3.3) pour modéliser le réseau et le matériel des postes. Ceci se fait par des classes, appelées "schémas" dans cet exemple [BRE 90]:

```
{Nom_du_schéma
  slot1: valeurs
  slot2: valeurs
  ...
}
```

Les attributs d'un schéma (les slots) correspondent aux propriétés de l'objet ou aux relations avec d'autres objets. Les valeurs peuvent être un nombre, une chaîne de caractères, le nom d'un autre schéma ou le nom d'une fonction (**attachement procédural**). Cette fonction peut être déclenchée lorsque le schéma correspondant reçoit un message.

SEPT modélise le fonctionnement du système de protection (exemple de la figure A-I.9) à travers les attachements procéduraux, c'est à dire que les règles qui concernent les différents types de protections sont attachées aux définitions de leurs classes. Ceci facilite la vérification des différentes protections car il suffit d'envoyer un message *'vérifie-toi'* aux protections de la partie du réseau en question. Au cours de l'étape de vérification, SEPT construit dynamiquement des schémas correspondants aux scénarios vérifiés (exemple de la figure A-I.11). Ceux-ci participent au vote dans l'étape estimation et constituent en même temps un élément important pour l'explication de l'incident à l'opérateur [BAU 91].

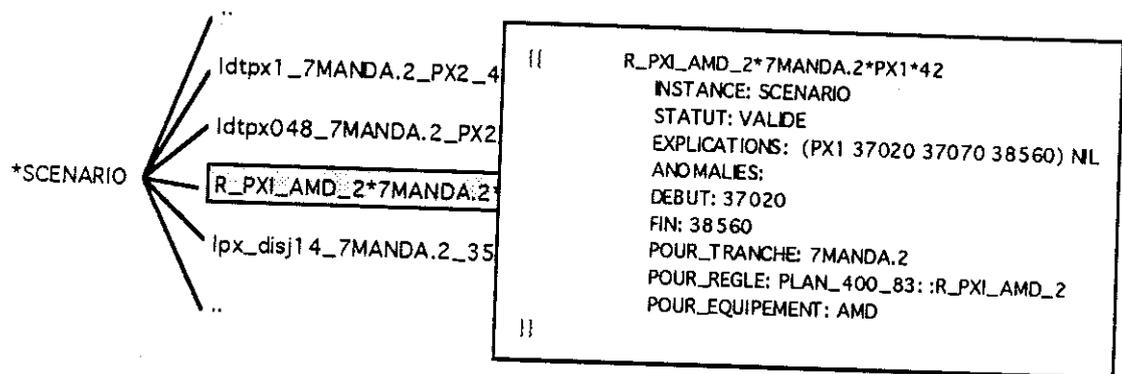


Figure A-I.11 Exemple de scénarios vérifiés [BRE 90]

La partie proprement déductive du diagnostic est réalisée par un système à base de règles de production du premier ordre qui travaille sur les schémas comme base de faits.

2.3.2 Discussion

Le système expert présenté ci-dessus permet un diagnostic sur les systèmes de protection en s'appuyant uniquement sur les enregistrements des signaux tracés. Ces enregistrements couplés avec la connaissance sur les séquences de signaux correctes des protections et sur leurs défauts caractéristiques, exprimées par des règles (exemple de la figure A-I.9), suffisent pour établir un diagnostic. Il n'y a pas besoin d'actions de vérification supplémentaires ni de participation active de l'opérateur, ce qui n'est pas le cas dans le domaine des entraînements. Ici, le diagnostic est plutôt un processus itératif; les enregistrements des signaux issus des protections sont très partiels et ne permettent pas une analyse complète du scénario du défaut. Le fonctionnement d'un entraînement est trop complexe pour pouvoir être décrit et vérifié par quelques règles tel que c'est possible dans l'exemple ci-dessus.

La représentation des faits s'appuie sur le formalisme objet pour modéliser l'installation (le réseau et les protections) et pour les conclusions intermédiaires (scénarios identifiés). Cette représentation est la plus utilisée ces dernières années à cause de sa puissance de structuration et sa souplesse, comme c'est illustré par les exemples précédents.

3 Approches basées sur des modèles profonds

Les approches présentées ci-dessus ont une caractéristique commune : elles font toutes appel aux connaissances heuristiques pour trouver l'hypothèse de défaut la plus pertinente pour expliquer les problèmes observés dans un système. Pour ceci, le raisonnement de diagnostic a besoin de connaissances sur les symptômes et les causes des défauts envisagés.

Cependant, les systèmes de diagnostic mettant en œuvre une telle approche sont limités car ils ne peuvent raisonner que sur les cas connus. Dès qu'un nouveau défaut apparaît dans une installation, il ne pourra pas être analysé par l'approche heuristique [MIL 87] [FRA 90]

L'idée de l'approche basée sur le modèle profond d'un procédé est de pallier à cet inconvénient en n'utilisant seulement les connaissances existant sur la structure du système à diagnostiquer et le comportement de ses composants pour élaborer le diagnostic [GEN 84].

La base de cette méthode est la détection d'incohérences entre le fonctionnement attendu et observé du système. Il s'agit de trouver le composant ou un ensemble de composants qui, quand ils sont en panne provoquent les observations incohérentes [DEK 87] [DAV 84] [DEV 91].

3.1 Principe de la méthode

Le principe de fonctionnement est la propagation de valeurs mesurées à travers le modèle du système et la conservation des hypothèses associées aux composants. La gestion des valeurs propagées et des hypothèses associées est normalement faite par un système de maintien de vérité (Truth Maintenance System TMS ou ATMS) [DEK 87] [DER 91].

Pour ceci l'approche se divise normalement en trois étapes :

1. Identification d'ensembles de conflits
2. Génération de candidats
3. Discrimination entre hypothèses

3.1.1 Identification d'ensembles de conflits

A chaque fois que le programme reçoit une nouvelle mesure, il applique tous les modèles (normalement des équations mathématiques décrivant le bon fonctionnement) dans lesquelles elle intervient et pour lesquels toutes les grandeurs sont connues, afin d'en calculer une nouvelle et de la propager de la même façon jusqu'à saturation. Au cours de cette propagation, à chaque fois qu'une valeur est calculée, le programme conserve les hypothèses associées, c'est-à-dire l'ensemble des composants supposés corrects pour aboutir à ce résultat.

Si le programme détecte une contradiction entre une prédiction et une mesure, ou entre deux prédictions, l'ensemble des hypothèses (dont une conduite à cette contradiction) constitue un ensemble de conflits (un "nogood"). Cela signifie qu'au moins une des hypothèses de bon fonctionnement d'un composant du nogood est fausse. Il y a alors au moins un des composants impliqués qui ne fonctionne pas correctement.

3.1.2 Génération de candidats

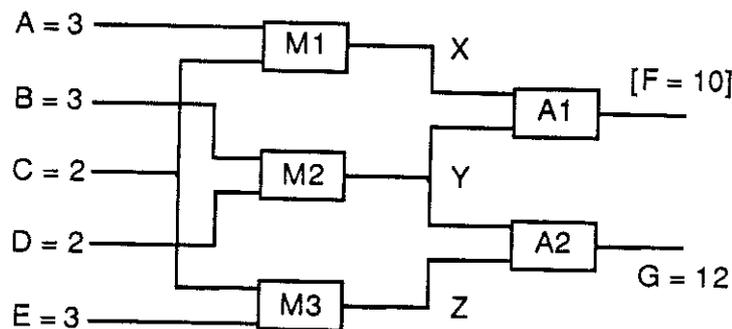
A partir des ensembles de conflits il est possible de déterminer les candidats. Un candidat est un ensemble de composants qui, s'ils ne fonctionnent pas correctement, permettent d'expliquer à eux seuls l'ensemble des contradictions. Ces candidats sont obtenus par croisement des ensembles de conflits : un candidat est formé en prenant un composant dans chacun des nogoods, et en ne considérant que les candidats à nombre de composants minimal.

3.1.3 Discrimination entre hypothèses

Le nombre de candidats générés pouvant être considérable, il faut le réduire. Le but est de choisir le candidat le plus plausible. Ceci peut se faire par propositions de mesures supplémentaires qui permettent d'éliminer des candidats. Il est également possible de faire appel à des connaissances supplémentaires, comme par exemple, la probabilité a priori de défaut des composants ou d'utiliser des modèles de composants avec défaut pour trier les candidats ou d'en éliminer une partie.

3.2 Exemple d'illustration

Nous allons illustrer le principe à l'aide d'un exemple simplifié souvent cité [DEK 87]. L'exemple montre un circuit électronique consistant en 3 multiplicateurs M1, M2, M3 et 2 additionneurs A1 et A2 (figure A-I.12). Le modèle structurel est défini par les relations des entrées et sorties entre composants, le modèle comportemental décrit les relations entre les entrées et les sorties d'un composant par des équations mathématiques.



Structure

multiplicateur (M1)	^	multiplicateur (M2)	^	multiplicateur (M3)	^
additionneur(A1)	^	additionneur(A2)	^	entrée1 (M1) = A	^
entrée2 (M1) = C	^	entrée1 (M2) = B	^	entrée2 (M2) = D	^
entrée1 (M3) = C	^	entrée2 (M3) = E	^	sortie (A1) = F	^
sortie (A2) = G	^				
sortie (M1)	=	entrée1 (A1)	=	X	^
sortie (M3)	=	entrée2 (A2)	=	Z	^
sortie (M2)	=	entrée2 (A1)	=	entrée1 (A2) = Y	

Comportement

multiplicateur (M-?) \Rightarrow	sortie (M-?) = entrée1 (M-?) * entrée2 (M-?)
additionneur (A-?) \Rightarrow	sortie (A-?) = entrée1 (A-?) + entrée2 (A-?)

Figure A-I.12 Exemple description structurelle et fonctionnelle d'un circuit électronique simple [DEK 87]

Supposons que les entrées valent $A=3$, $B=3$, $C=2$, $D=2$, $E=3$ et que les sorties mesurées valent $F=10$ (une contradiction) et $G=12$ (comme attendu).

3.2.1 Identification d'ensembles de conflit

Pas de raisonnement	Mesures	Prédiction	Hypothèses
0) début	$F=10, G=12$		$\{\}, \{\}$
1)	$A=3, C=2$	$X=6$	$\{M1\}$
2)	$B=3, D=2$	$Y=6$	$\{M2\}$
3)	$F=10, Y=6$	$X=4$	$\{A1, M2\}$

Ici on utilise comme entrées la valeur mesurée $F=10$ et la prédiction $Y=6$. L'ensemble des hypothèses doit alors inclure (MULT-2) pour la prédiction $Y=6$. On détecte une contradiction pour la prédiction $X=4$ qui est incohérente avec la prédiction $X=6$ (dans le pas 1). L'union des hypothèses associées aux étapes de raisonnement forment alors un premier ensemble de conflit $C1$: $C1 = (X=6 \{M1\}) \cup (X=4 \{A1, M2\}) = \langle M1 A1 M2 \rangle$

Pas de raisonnement	Mesures	Prédiction	Hypothèses
4)	$C=2, E=3$	$Z=6$	$\{M3\}$
5)	$X=6, Y=6$	$F=12$	$\{A1, M1, M2\}$

Ici apparaît une deuxième contradiction entre la mesure initiale $F=10$ liée à l'hypothèse $\{\}$ et la prédiction $F=12$ liée à l'hypothèse $\{A1, M1, M2\}$. On alors crée l'ensemble de conflits $C2$: $C2 = (F=10 \{\}) \cup (F=12\{A1 M1 M2\}) = \langle A1 M1 M2 \rangle$

Pas de raisonnement	Mesures	Prédiction	Hypothèses
6)	$Y=6, Z=6$	$G=12$	$\{A2, M2, M3\}$
7)	$F=10, Y=6$	$X=4$	$\{A1 M2\}$
8)	$F=10, X=6$	$Y=4$	$\{A1 M1\}$
9)	$Y=4, Z=6$	$G=10$	$\{A2, A1, M1, M3\}$

Troisième contradiction avec la valeur $G=12$ de l'étape 6)

$$C3 = (G=12\{\}) \cup (G=10\{A2, A1, M1, M3\}) = \langle A2 A1 M1 M3 \rangle$$

Les ensembles de conflits $C1$ et $C2$ étant identiques, il suffit d'en retenir un pour la génération de candidats.

3.2.2 Génération des candidats

Les candidats sont les composants qui doivent être en panne simultanément pour expliquer les contradictions apparues. La génération des candidats se fait par création des combinaisons des composants des ensembles de conflits en ne retenant que les combinaisons minimales, comme le montre la figure A-I.13.

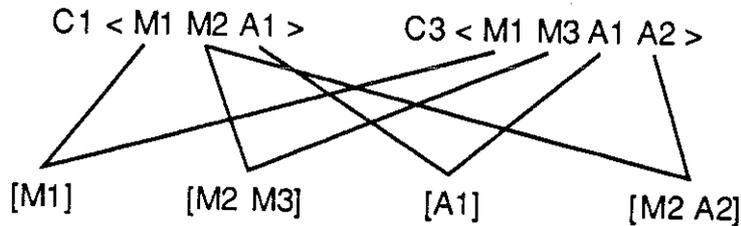


Figure A-I.13 Génération des candidats

3.2.3 Discrimination entre hypothèses

Si l'on considère que des défauts simples sont plus probables que les défauts multiples, on obtient les deux candidats A1 et M1. Le test le plus pertinent serait de mesurer la sortie de M1, la valeur de X puisque cette mesure permet de distinguer entre deux défauts uniques.

3.3 Discussion

Cette approche a été élaborée initialement par le diagnostic de circuits électroniques logiques [DAV 84] [DEK 87] mais s'applique également sur des cartes d'électronique analogiques [DEV 91] [DAG 90]. Une application sur le diagnostic des réseaux de distribution d'énergie avec l'interprétation d'alarmes a également été étudié [BES 93].

D'une manière générale, cette approche est adaptée aux systèmes qui ont les caractéristiques suivantes :

- topologie connue en détail,
- nombre de types de composants limité,
- existence de modèles du comportement assez détaillés pour chaque type de composant qui permettent de simuler leur comportement,
- un nombre important de mesures disponibles

Cette méthode présente les avantages suivants:

- la mise en œuvre rapide du modèle si on peut réutiliser les plans automatisés informatiquement de la phase de conception d'un système (circuits électriques ou électroniques),

- des connaissances heuristiques non requises. L'approche fonctionne sur des systèmes nouvellement conçus.

Elle présente cependant certains inconvénients:

- les modèles pour les composants n'existent pas toujours,
- il y a risque d'explosion combinatoire des hypothèses à gérer surtout quand le nombre d'observations (mesures disponibles) est insuffisant,
- les temps de calcul sont longs pour des systèmes réels contenant beaucoup de composants
- il faut introduire d'autres types de connaissances pour restreindre le nombre de candidats possibles.

3.4 Commentaires

Bien que cette approche soit souvent citée lorsqu'il s'agit du diagnostic de circuits électroniques ou de cartes analogiques, il existe aussi des approches basées sur l'heuristique dans ce domaine. Un exemple industriel est le diagnostic de cartes électroniques analogiques par arbres de tests [MAR 90] qui exploitent des connaissances heuristiques de techniciens, intégrant les probabilités de pannes et le coût des tests pour minimiser le nombre de tests à effectuer ainsi que le temps du diagnostic. La démarche d'un parcours heuristique d'un arbre de tests est comparable à celle décrite dans le chapitre 2.1.2.

Une autre démarche consiste à créer des hypothèses de défauts à partir de connaissances heuristiques et d'utiliser ensuite un simulateur contenant des modèles du système sans et avec défaut pour simuler les défauts soupçonnés [BIZ 90]. La décision sur l'hypothèse est prise par comparaison du comportement simulé et observé du circuit.

3.5 Conclusion

Une application de la démarche présentée dans l'exemple n'est pas envisageable pour les équipements d'entraînement électrique, d'une part car les modèles mathématiques des différents composants ne sont pas assez simples pour être utilisés ou car ils n'existent pas. Ainsi, pour un entraînement, les modèles de fonctionnement doivent être exprimés symboliquement, par exemple par des règles décrivant le comportement de ses composants [SPU 90]. D'autre part, le nombre de mesures disponibles sur les différents composants d'un tel équipement n'est pas suffisant de sorte que le nombre d'hypothèses à gérer soit trop important. De plus, les valeurs mesurées ne sont pas constantes dans le temps, comme ce le cas dans l'exemple présenté, ce qui augmente la complexité de la gestion d'hypothèses.

4 Conclusion

Dans cette annexe, nous avons présenté quelques approches de diagnostic en électrotechnique par système expert. Ces approches se distinguent essentiellement par la manière dont les connaissances sont structurées et par la complexité de la stratégie qu'elles utilisent. Les exemples présentés ci-dessus illustrent qu'il n'y a pas de modèle unique qui s'applique d'une manière universelle à tous les problèmes. Le modèle choisi est souvent un compromis entre plusieurs méthodes de représentation et dépend fortement des particularités du domaine d'application et de la nature de la connaissance disponible.



Annexe II

Outils d'enregistrement et de compression de données pour synthétisation d'une trace d'exploitation

1 Objectif

Nous avons du créer des outils d'aide à l'analyse de phénomènes lents (vieillessement, usure, encrassage, défauts naissants) dont on ne dispose pas d'expérience à l'heure actuelle. Ces outils sont un premier pas vers l'acquisition d'une connaissance pratique sur ces phénomènes dans des équipements d'entraînements de grande puissance. Une maîtrise de ces phénomènes pourra aboutir à une détection précoce d'anomalies avant que des seuils des protections utilisées à l'heure actuelle ne soient atteints.

2 Démarche

Enregistrement de grandeurs significatives de l'état des différents dispositifs d'une chaîne d'entraînement électrique: transformateur, convertisseurs, machine et charge éventuellement, par exemple:

- tensions et courants du réseau d'alimentation
- tensions et courants de la machine
- températures (ambiante, entrées et sorties des systèmes de refroidissement, des bobines statoriques de la machine, des paliers, d'huile du transformateur, etc..)
- vibrations sur la machine et la charge
- débits de fluide dans les circuits de refroidissement (pour une maintenance conditionnelle suite à l'encrassage)
- et pour corrélation avec le fonctionnement: vitesse et puissance fournies

A l'heure actuelle, la plupart de ces grandeurs sont surveillées par des capteurs à seuil, émettant un ou deux niveaux d'alarmes. Nous pensons qu'il est utile d'étudier ces valeurs de plus près car elles peuvent donner une image de l'état des différents composants d'une installation. Une telle surveillance se fait aujourd'hui pour le matériel des centrales nucléaires et les grands alternateurs [ALE 91][FOU 88] [MAR 90] et des grands moteurs stratégiques [SOT 89] [TAV 86] [TAV 89] [ZOC 90] [HAY 91] [KLE 91].

Pour l'analyse de certains problèmes du convertisseur statique, en particulier des problèmes de mise au point des réglages, il serait intéressant de connaître l'historique de l'exploitation, par exemple pour savoir si un point de fonctionnement a déjà été atteint depuis le réglage. Une trace pourra apporter des informations plus fiables qu'une enquête auprès des utilisateurs.

3 Principe

Le principe de l'enregistrement est très simple: on échantillonne les valeurs significatives sur un horizon glissant, par exemple une heure et on établie une analyse statistique sur cette fenêtre de temps. On retient:

- la valeur échantillonnée
- la valeur maximale $x_{\max} = \max(x_i)$
- la valeur minimale $x_{\min} = \min(x_i)$
- la valeur moyenne $x_{\text{moyenne}} = \frac{1}{n} \sum_{i=1}^n x_i$
- l'écart (la dispersion de la distribution) $\sigma^2 = \frac{1}{n} \sum_{i=1}^n (x_i - x_{\text{moyenne}})^2$

L'écart σ est une mesure de la distance moyenne des échantillons de la valeur moyenne. Pour une fonction de distribution gaussienne on a (figure A-II.1)

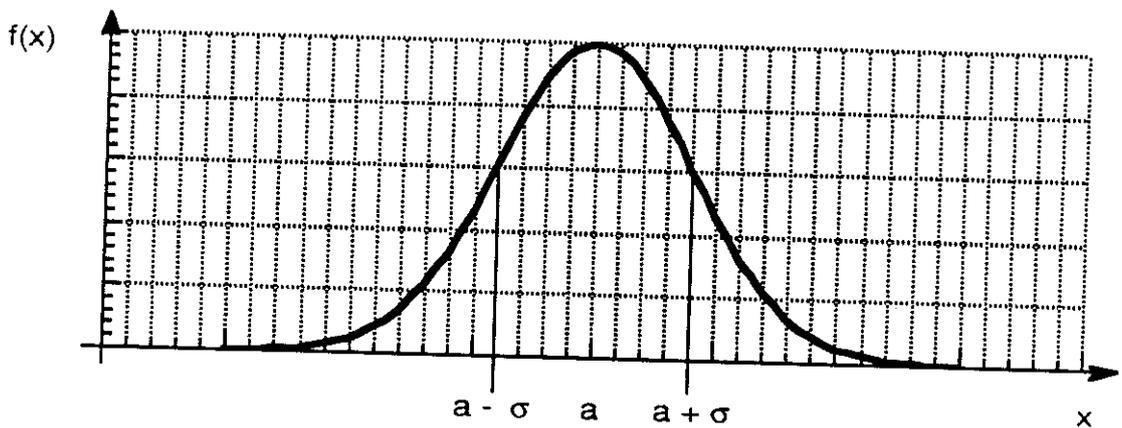


Figure A-II.1: Exemple d'une distribution gaussienne

68% des échantillons se trouvent dans l'intervalle:

$$x_{\text{moyenne}} - \sigma < x_{\text{moyenne}} < x_{\text{moyenne}} + \sigma$$

et moins de 1% en dehors de l'intervalle:

$$x_{\text{moyenne}} - 3\sigma < x_{\text{moyenne}} < x_{\text{moyenne}} + 3\sigma$$

Dès que le nombre d'échantillons prévu dans la statistique est atteint, les résultats sont enregistrés dans un traceur d'historique: on retient x_{moyenne} , σ^2 , x_{\max} , x_{\min} .

Celui-ci enregistre un nombre correspondant à sa profondeur de mémorisation. Une fois ce nombre atteint, une synthèse sur ses valeurs est mémorisée dans un autre traceur d'un niveau supérieur et ainsi de suite:

statistique heure -> historique jour

historique jour -> historique mois

historique mois -> historique an

historique an -> historique vie

4 Mise en œuvre

Nous avons implanté les outils proposés de manière modulaire et extensible grâce au formalisme objet (cf § II.3.3). Ainsi, les outils de trace d'historique s'intègrent naturellement dans la base de données de l'installation où sont inclus les signaux des grandeurs suivies. Les différents outils d'enregistrement sont tous des objets de la classe historique, fille de la classe *activité* (figure A-II.2). Pour faire tracer l'historique d'un signal, il suffit de créer une instance de *statistique*, de remplir son champ *profondeur* et de l'affecter au signal à tracer. Les outils réagissent automatiquement aux changements de valeur du signal. Ceci se fait par les méthodes dont disposent les objets des classes *statistique* et *historique* (mécanisme des valeurs actives, qui est un des concepts des langages orientés objet [MEY 88] [SME 91]). La synthétisation se fait également automatiquement lorsqu'on affecte un objet historique au champ *synthèse* d'un objet statistique ou historique. Le champ *profondeur* de celui-ci détermine, combien de valeurs sont stockées avant de passer à une synthèse au niveau suivant. La figure A-II.3 montre une chaîne d'enregistrement historique associée à une mesure. Les enregistrements peuvent ensuite être visualisés dans des éditeurs de courbes comme le montre la figure A-II.4.

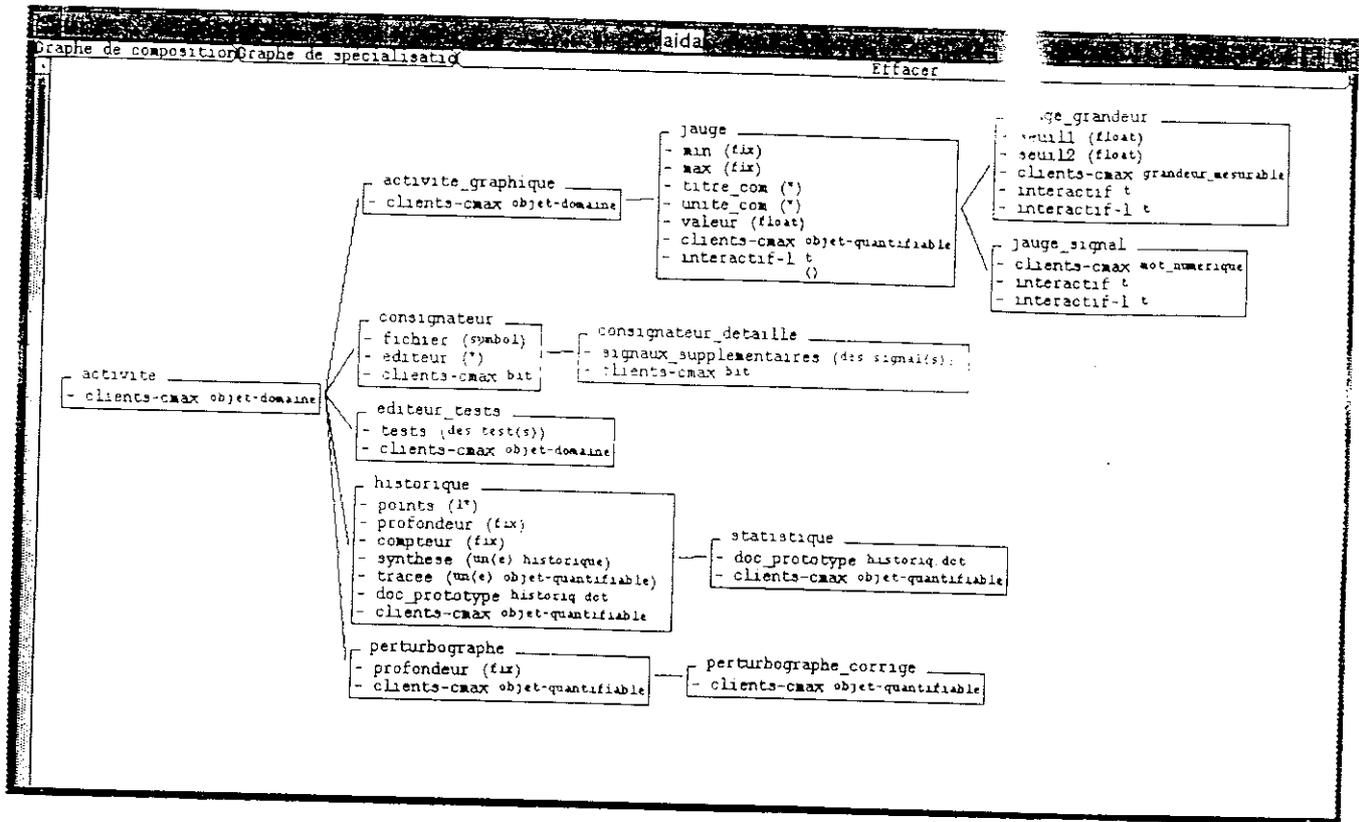


Figure A-II.2 Définition des classes pour les outils de trace

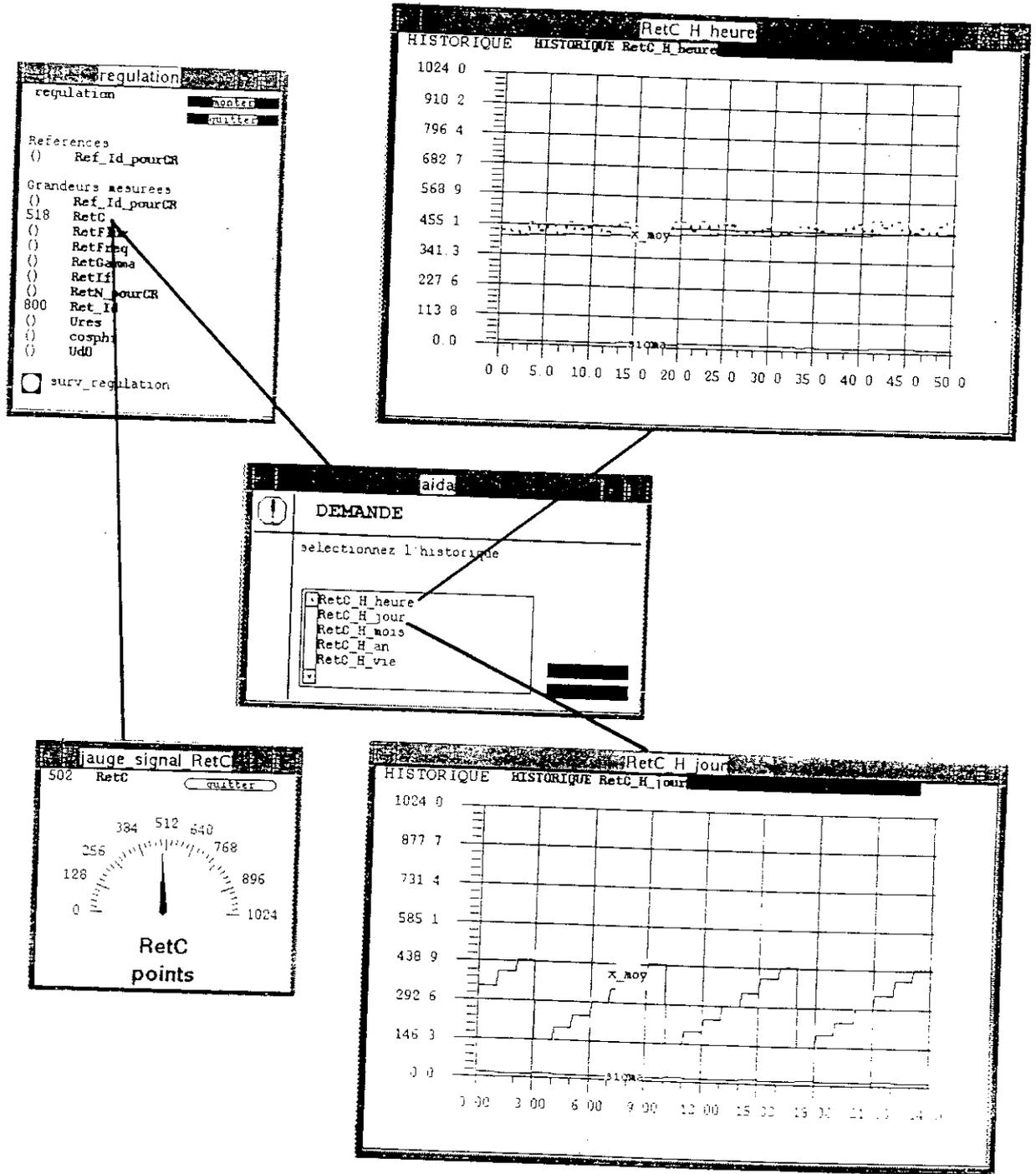


Figure A-II.4 Exemple de visualisation de l'historique enregistré

Annexe III

Détails sur la mise en œuvre de l'interface graphique

1 Types d'images

Pour une entité, il existe deux types de représentations graphiques: des images détaillées et des images synthétiques.

1.1 Les images synthétiques

Une image synthétique représente l'entité dans l'image du système dont elle fait partie (figure AIII.1). L'image synoptique peut changer son allure dynamiquement pour refléter l'état de l'entité. Elle a le comportement d'un bouton pour donner accès aux fonctionnalités disponibles pour elle (par exemple, zoom, éditeurs, documentation, etc...). Si l'entité se trouve dans plusieurs systèmes, il y aura multiplication de cette image pour qu'il y en ait autant de copies que de **systèmes-pères**. Ceci est nécessaire puisqu'une image, c'est-à-dire une structure informatique, ne peut se trouver qu'à un seul endroit à la fois.

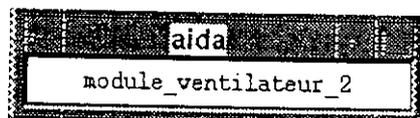


Figure AIII.1 Image synthétique du module_ventilateur_2

1.2 Les images détaillées

Une image détaillée représente le détail de l'entité avec sa structure et les sous-systèmes, composants et signaux qu'elle contient (figure AIII.2). Cette image est composée d'une image arrière-plan montrant la structure de l'entité et des images synoptiques des entités qu'elle contient. Les images détaillées n'existent que pour les systèmes; les composants finaux, par exemple, le fusible ou le ventilateur dans la figure AIII.2, ne peuvent pas être détaillés.

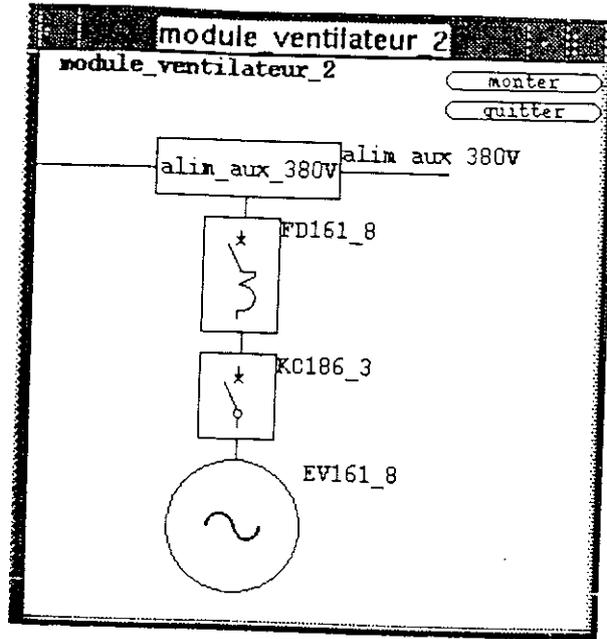


Figure AIII.2 Image détaillée du module_ventilateur_2

2 Couplage entre images et objets d'une installation

Le but de la représentation graphique des objets qui représentent une installation et de mieux documenter l'état des composants et sous-systèmes de l'installation. La **cohérence** entre l'affichage et la base de données nécessite la mise à jour de l'image d'un composant après une modification de son état. Cependant, cette mise à jour ne doit pas influencer la conception du système expert. Du point de vue fonctionnel, aucune différence ne doit exister pour le concepteur du système expert entre le fonctionnement avec ou sans interface graphique. Le concepteur qui adapte le système expert à une installation, doit se focaliser sur les aspects base de données de l'installation sans avoir à se préoccuper de détails d'implémentation informatique concernant l'interface graphique. Ceci est d'autant plus vrai pour un expert qui ajoute des règles de diagnostic. La parfaite modularité de l'interface est atteinte en faisant appel au concept des **activités**.

2.1 Les activités

Le couplage entre objets et ses images fait appel au concept des **démons** ou **valeurs actives** [SME 91]. L'idée consiste à déclencher un traitement automatiquement suite à la modification d'une variable, de la valeur d'un champ ou d'une caractéristique d'un objet, etc... Ce traitement peut être défini au niveau des classes (cas des démons), auquel cas le traitement est le même pour toutes les instances de la classe, ou au niveau des instances des classes (cas des activités). Dans ce dernier cas, le traitement est "personnalisé" pour chaque objet. La mise à jour des images dans l'interface graphique est géré par des activités.

Une activité est un objet qui sait répondre aux messages *do-before* et *do-after*. Un tel objet peut être associé à un objet, par exemple à un contacteur (figure AIII.3), pour effectuer des actions lorsque ce dernier subit une modification, c'est-à-dire un changement d'une valeur dans un de ses champs. Lorsqu'une telle modification intervient, ces messages sont envoyés automatiquement à l'objet défini comme activité et déclenchent les méthodes correspondantes [SME 91].

2.2 Un exemple

Nous avons prévu ces méthodes pour prendre en charge la mise à jour de l'image selon l'état de l'objet. Par exemple, lors d'une modification dans le champ *etat_fonctionnel* du contacteur *KC186_3*, le message *do-after* est envoyé à son activité *image_KC186_3*. Cet appel déclenche alors le traitement spécifié par la méthode *do-after* qui se charge de mettre à jour l'image. La figure AIII.3 donne un exemple du déroulement de ce mécanisme: suite à la modification de l'état du contacteur *KC186_3* qui devient *enclenché*, l'image change d'allure.

Notons aussi que d'autres activités peuvent être associées simultanément à un objet, par exemple, des éditeurs, d'autres images, des enregistreurs d'historique, etc...

Remarque:

La réalité dans notre implantation est un peu plus compliquée que l'exemple décrit ici. Ceci vient du fait que des références directes sur les applications graphiques de AIDA (et ceci est l'introduction d'un pointeur sur l'image dans la liste des activités) posent des problèmes de compatibilité lors des sauvegardes des objets. Toutefois, la solution retenue ne change rien dans l'idée du concept utilisé.

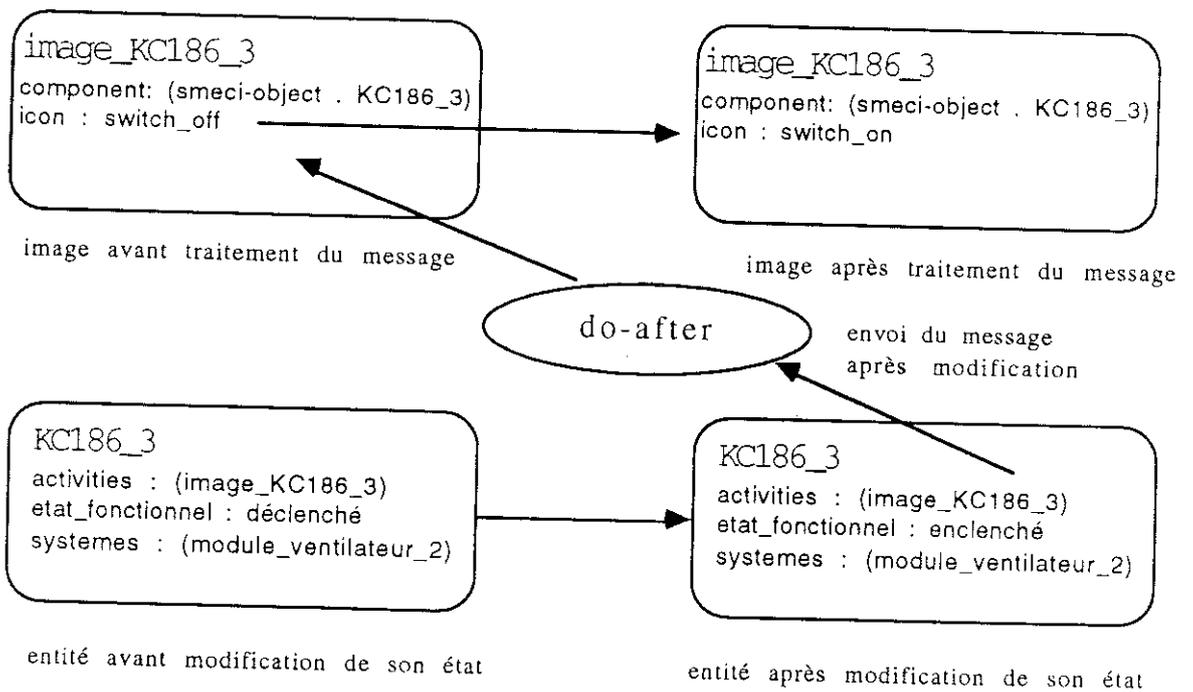
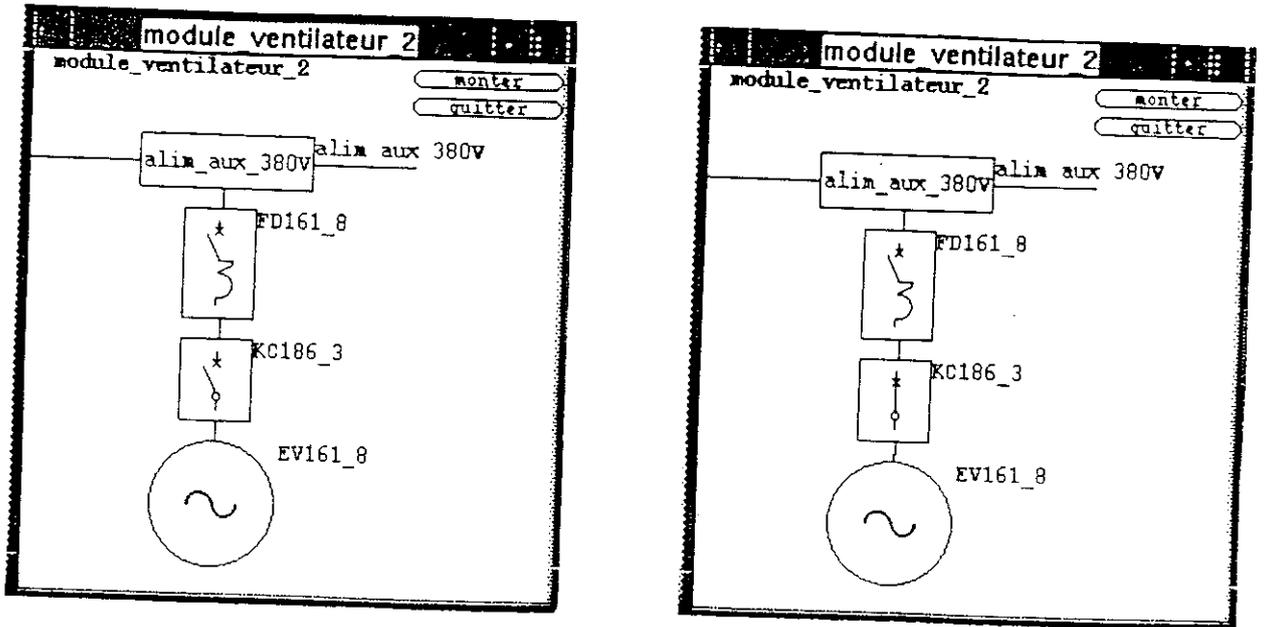


Figure AIII.3 Modification d'une image par activités

Index des figures

Figure I.1 Blocs dans un entraînement synchrone	4
Figure I.2 Architecture du système de contrôle.....	5
Figure I.3 Redondance matérielle double	7
Figure I.4 Redondance matérielle triple avec localisation du défaut	8
Figure I.5 Exemple de décomposition d'un entraînement en blocs fonctionnels	11
Figure I.6 Exemple pour établir un arbre de défaillances	13
Figure I.7 Arbre de défaillances pour l'exemple de la figure I.6	13
Figure I.8 Détection de défauts par redondance analytique	16
Figure I.9 Architecture d'un système expert	20
Figure II.1 Organigramme de la démarche de diagnostic d'une anomalie.....	30
Figure II.3 Exemple de décomposition fonctionnelle.....	34
Figure II.4 Exemple de décomposition structurelle	34
Figure II.5 Exemple d'une décomposition hybride	35
Figure II.6 Exemple de différents types de composants.....	37
Figure II.7 Exemple de différents types de modules structurels.....	37
Figure II.8 Exemple de propriétés d'un module structurel	38
Figure II.9 Exemple de propriétés d'un module fonctionnel.....	39
Figure II.10 Exemple de propriétés communes à un type d'élément.....	40
Figure II.11 Exemples de propriétés particulières aux différents éléments.....	40
Figure II.12 Le concept de classe	41
Figure II.13 Le concept de messages.....	42
Figure II.14 Le concept d'héritage.....	42
Figure II.15 Exemple de décomposition d'un entraînement	43
Figure II.16 Exemple de décomposition d'une machine	44
Figure II.17 Principe de création d'hypothèses sur les causes d'une observation anormale.....	48
Figure II.18 Conseil pour la création des hypothèses sur les causes	48
Figure II.19 Création d'observations.....	49
Figure II.20 principe de utilisation d'observations hypothétiques	49
Figure II.21 Exemple d'association de sources de connaissance (SdC..) aux classes	53
Figure III.1 L'architecture blackboard.....	60
Figure III.2 Exemple de définition de champs par facettes.....	63
Figure III.3 Exemple de création d'instances d'une classe (catégorie)	64
Figure III.4 Exemple d'implémentation de valeurs communes à tous les objets d'une classe.....	65
Figure III.5 Exemple de classes et d'instances	67
Figure III.6 Exemple d'une règle de diagnostic	68
Figure III.7 Exemple d'une source de connaissance.....	69
Figure III.8 Les tâches du contrôle	70
Figure III.9 L'agenda du contrôle.....	72
Figure III.10 Exemple d'utilisation de l'agenda.....	72
Figure III.11 Exemple d'une règle stratégique	73
Figure III.12 Classification des classes pour le raisonnement.....	75
Figure III.13 Choix du test à effectuer dans l'éditeur de tests	77
Figure III.14 Exemple de synoptiques hiérarchisées d'un entraînement	80
Figure IV.1 Interfacage entre le système expert et l'équipement	83
Figure IV.2 Le système expert initialement vide	84
Figure IV.3 Le système expert après chargement de la base de connaissances.....	85
Figure IV.4 L'interface de la base de données de l'installation.....	86
Figure IV.5 Le gestionnaire de fonctionnalités et l'éditeur de tests	86

Figure IV.6 Le gestionnaire du raisonnement.....	87
Figure IV.7 L'éditeur de signaux.....	87
Figure IV.8 Exemple de consultation de points de référence enregistrés	88
Figure IV.9 Exemple de tests proposés dans l'éditeur de tests.....	89
Figure IV.10 L'éditeur de signaux	90
Figure IV.11 Exemple d'entrée d'observations.....	90
Figure IV.12 Exemple Réponse aux tests proposés.....	91
Figure IV.13 Exemple d'outils de trace du système expert.....	92
Annexes	
Figure A-I.1 Exemple d'une règle	2
Figure A-I.2 Exemple d'arbre de défauts	5
Figure A-I.3 Exemple d'un arbre de décision	5
Figure A-I.4 Exemples de règles de diagnostic d'un turbo-alternateur	8
Figure A-I.5 Réseau d'inférences mis en œuvre par les règles	9
Figure A-I.7 exemple d'un attribut pour l'acquisition de données	14
Figure A-I.8 Exemple d'une représentation structurelle dans STEAMER	17
Figure A-I.9 Exemple d'une règle de test pour un type de protection.....	22
Figure A-I.10 Exemple d'une règle de vote	23
Figure A-I.11 Exemple de scénarios vérifiés	24
Figure A-I.12 Exemple description structurelle et fonctionnelle d'un circuit électronique simple	27
Figure A-I.13 Génération des candidats	29
Figure A-II.1 Exemple d'une distribution gaussienne	34
Figure A-II.2 Définition des classes pour les outils de trace	36
Figure A-II.3 Exemple d'une chaîne d'enregistrement historique	37
Figure A-II.4 Exemple de visualisation de l'historique enregistré.....	38
Figure AIII.1 Image synthétique du module_ventilateur_2.....	39
Figure AIII.2 Image détaillée du module_ventilateur_2.....	40
Figure AIII.3 Modification d'une image par activités.....	42

Bibliographie

Références bibliographiques

- [ABB 89] ABB
"Un système de communication homme-machine pour entraînement triphasé"
RGE, No.5, Mai 1989
- [ALE 91] W. Aleite, D. Wach
"Supervision and Fault Diagnosis for the Safe Operation of Nuclear Power Plants"
IFAC/IMACS - Symposium on Fault Detection, Supervision and Safety for
Technical Processes, Safeprocess '91, Preprints, Baden-Baden 1991, pp. 23-34
- [ANC 90] J. Ancelin
"KSE, L'intelligence artificielle surveille les centrales"
EDF La Recherche, sept 1990
- [BAU 91] Bau Dong-Yih
"Explication basée sur un modèle dans des systèmes d'aide au diagnostic :
Application au système à base de connaissances SEPT"
Thèse de l'Université de Paris-Sud Centre d'Orsay, 1991
- [BER 92] C.-S. Berendsen, G. Champenois, J. Davoine, G. Rostaing
"How to Detect and Localize a Fault in a DC/DC Converter ?", International
Conference on Industrial Electronics, Control, Instrumentation, and Automation,
IECON, San Diego 1992, pp. 536-541
- [BER 93a] C.-S. Berendsen, G. Rostaing, G. Champenois
"Distinguishing control loop failures from voltage fluctuations and load torque
impacts - an analytical approach", International Conference on Fault Diagnosis,
Tooldiag Toulouse 1993, pp. 582-589
- [BER 93b] C.-S. Berendsen
"Méthodes de diagnostic dans la partie commande des associations convertisseurs -
machines tournantes", Thèse de l'Institut National Polytechnique de Grenoble, 1993
- [BES 93] A. Beschta, O. Dressler, H. Freitag, M. Montag, P. Struß
"A model-based approach to fault localisation in power transmission networks"
Intelligent Systems Engineering, Vol. 2 No. 1, Spring 1993, pp. 3-14
- [BIG 93] J. Bigeon, F. Bounaas, D. Constant et al.
"Modélisation objet pour la C.F.A.O.",
Rapport Interne du Projet GSIP, CNRS GDR 895, janvier 1993
- [BIJ 91] R.W. Bijoch, S.H. Harris, T.L. Volkmann, J.J. Bann, B.F. Wollenberg
"Development and implementation of the NSP Intelligent Alarm Processor",
IEEE Transactions on Power Systems, Vol.6, No.2, Mai 1991
- [BIZ 90] L. Bizzari, F. Corazziari, S. Falciano, P.G. Gualaccini, L. Luminari, M. Savarese,
E. Trasatti, "Fault diagnosis of electronic circuits", 10ieme Journées Internationales
"Les Systèmes Experts et leurs Applications", Avignon 1990
- [BRE 90] Patrick Brézillon, Pascal Fauquembergue
"La Tâche de diagnostic dans le système expert SEPT"
10ieme Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1990, pp. 283-293
- [BRU 90] J. Brunet, D. Jaume, M. Labarrère, A. Rault, M. Vergé
"Détection et diagnostic de pannes - Approche par modélisation"
Edition Hermes, Paris 1990
- [BUC 91] G. Bucci, C. Cecati, D.Q. Zhang
"Real-time implementation of fault detection and control algorithms on electrical
drives", 4th European Conference on Power Electronics and Applications, EPE,
Florence 1991

- [BUT 89] K. Buttner, D. Sriram, M. Freiling
"An Object-Oriented Blackboard Architecture for Model-Based Diagnostic Reasoning"
Blackboard Architectures and Applications - Perspectives in Artificial Intelligence"
edited by V. Jagannathan, Academic Press Inc, London 1989
- [CAM 87] R.B. Cambien
"Diagnostic des groupes turbo-alternateurs"
SEE- Club 11, Diagnostic et Maintenance par Systèmes Experts- Journées d'études,
Paris, octobre 1987
- [CAP 92] C. Capdessus, Menad Sidmahme
"Analyse des vibrations d'un engrenage : cepstre, corrélation, spectre. Traitement du
signal en mécanique", Traitement du signal, Vol. 8, no 5, 1992, pp. 365-372
- [CAR 88] E. Cardozo, S.N. Talukdar
"A distributed expert system for fault diagnosis"
IEEE Transactions on Power Systems, Vol. PWRS-3, No.2, May 1988
- [CAT 92] Stéphane Catellani
"Etude sur la détection et la localisation de défauts en temps réel dans les éléments de
puissance d'une association convertisseur-machine",
Thèse de l'Institut National Polytechnique de Grenoble, 1992
- [CHA 90] Henri Chaffoin
"Nouvelles architectures de conduite et de supervision"
RGE, N° 2, Février 1990, pp. 28-31
- [CHA 91] J.-N. Chatain, B. Dury
"Apports du modèle du blackboard aux systèmes experts d'aide au diagnostic"
Actes de la Convention IA'91, Paris 1991
- [CHE 90] F. Chériaux
"3SE, Système expert temps réel de surveillance des sources électriques d'une
centrale nucléaire"
10ième Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1990, pp. 29-39
- [CHO 91] M.Y. Chow, S.O. Yee
"Methodology for on-line fault detection in single-phase squirrel-cage induction
motors using artificial neural networks"
IEEE Transactions on Energy Conversion, Vol. 6, No. 3, Sept. 1991, pp 536-545
- [DAG 90a] P. Dague, P. Deves, P. Luciani, P. Taillibert
"Diagnostic de systèmes analogiques"
10ième Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1990, pp. 125-136
- [DAG 90b] P. Dague, O. Jehl, P. Taillibert
"An interval propagation and conflict recognition engine for diagnosing continuous
dynamic systems",
G. Gottlob, W. Nejdl: Proceedings of the international Workshop Expert Systems in
Engineering - Principles and Applications, Vienna 1990, pp. 16-31
- [DAV 84] R. Davis
"Diagnostic Reasoning Based on Structure and Behaviour"
Artificial Intelligence, No. 24, 1984
- [DEB 91] K. Debebe, V. Rajagopalan, T.S. Sankar
"Expert Systems for Fault Diagnosis of VSI Fed AC-Drives"
IEEE Industry Applications Conference, 26th IAS Annual Meeting
Dearborn, Michigan 1991, pp. 368-373

- [DEB 92] K. Debebe, V. Rajagopalan, T.S. Sankar
"Diagnosis and Monitoring for AC Drives"
IEEE Industry Applications Conference, 27th IAS Annual Meeting
Houston, Texas 1992, pp. 370-377
- [DEK 87] J. de Kleer, B.C. Williams
"Diagnosing Multiple Faults"
Artificial Intelligence, No. 32, 1987
- [DEL 88] J.-P. Delahaye
"Cours de PROLOG avec TURBO PROLOG"
Editions Eyrolles, Paris 1988
- [DER 91] D. McDermott
"A general framework for reason maintenance"
Artificial Intelligence 50, August 1991
- [DEV 91] P. Devès, P. Taillibert, C. Fischer
"Diagnostic à base de modèles: une alternative aux systèmes experts"
11ème Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1991
- [DRO 92] K. Drouiche, M. Sidahmed, Y. Grenier
"Détection de défauts d'engrenages par analyse vibratoire. Traitement du signal en
mécanique", Traitement du signal, Vol. 8, no 5, 1992, pp. 331-343
- [ENG 88] Robert Engelmores, Tony Morgan (Editors)
"Blackboard Systems", Addison-Wesley 1988
- [ERM 89] Jean-Louis Ermine
"Systèmes Experts - Théorie et Pratique"
Lavoisier, Cachan 1989
- [ERM 91] J.-L. Ermine
"Un système à base de connaissances pour le diagnostic technique et la recherche
documentaire", Actes de la Convention IA'91, Paris 1991,
- [FAR 85] H. Farreny
"Les Systèmes Experts", Cepadues-Editions, Toulouse 1985
- [FAU 91] P. Fauquembergue, J.V. Esquevin
"Un système expert d'aide à la maintenance des systèmes de protection des réseaux de transport",
Note 92NIR0005, Service Etudes de Réseaux, Direction des Etudes et Recherches, EdF 1992
- [FRA 90] P.M. Frank
"Fault diagnosis in dynamic systems using analytical and knowledge-based
redundancy - A survey and some new results"
Automatica, Vol. 26, No. 3, 1990, pp. 459-474
- [FUJ 86] R. Fujiwara, T. Sakaguchi, Y. Kohno, H. Suzuki
"An intelligent load flow engine for power system planning"
IEEE Transactions on Power Systems, Vol 1. No.3, August 1986
- [GEN 84] M.R. Genesereth
"The Use of Design Descriptions in Automated Diagnosis"
Artificial Intelligence, No. 24, 1984
- [GEN 91] A. Gentilhomme
"C.O.C.A.S.E - Un système expert d'aide à la conception d'appareillages
électriques", Thèse, Institut National Polytechnique de Grenoble, 1991
- [GER 93] Laurent Gerbaud
"Aide à la conception des ensembles machine - convertisseur - commande : apport

- d'une démarche système expert", Thèse en génie électrique de l'Institut National Polytechnique de Grenoble, 1991
- [GON 86] A.J. Gonzales, R.L. Osborne, C.T. Kemper, S. Lowenfield
"On-line Diagnosis of Turbine-Generators using Artificial Intelligence"
IEEE Transactions on Energy Conversion, Vol.EC-1, No.2, June 1986, pp. 69-74
- [GUI 90] J. Guitton, J. Morel, R. Chevalier, B. Ricard
"DIVA, système expert d'aide au diagnostic vibratoire"
EDF La Recherche, Septembre 1990
- [HAR 88] C.J. Harris
"Application of Artificial Intelligence to Command and Control Systems"
Peter Peregrinus Ltd, London, 1988
- [HAT 91] Jean-Paul Haton, Nadjat Bouzid et al.
"Le Raisonnement en Intelligence Artificielle"
Inter Editions, Paris 1991
- [HAY 83] F. Hayes-Roth, D.A. Waterman, D.B. Lenat
"Building Expert Systems"
Addison-Wesley, Reading, Massachusettes, 1983
- [HAY 91] R.D. Hayashida, J. Motter, K. McDole
"Turbomachinery Monitoring Systems Capture and Analyse Vibration Data"
IEEE Computer Applications in Power, July 1991, pp.38-43
- [ISE 84] R. Isermann
"Process fault detection based on modelling and estimation methods - A survey"
Automatica, Vol. 20, No.4 1984, pp. 387-404
- [JAU 92] Daniel Jaume, Jacques Proust, André Rault
"De l'AMDEC à la fiabilité prévisionnelle : application à l'industrie automobile"
Revue européenne Diagnostic et sûreté de fonctionnement, Vol. 2 - No 2/1992
- [KAL 87] Prem Kumar Kalra, R.M. Mathur
"Investigations for developing expert system for power system control"
Electric Machines and Power Systems, No.13, 1987
- [KAM 91] S.D. Kaminaris, A.V. Machias, B.C. Papadias
"An intelligent tool for distribution substations troubleshooting and maintenance scheduling"
IEEE Transactions on Power Delivery, Vol.6, No.3, July 1991, pp. 1038-1044
- [KER 89] J.J. Keronen
"An Expert System Prototype for Event Diagnosis and Real-Time Operation Planning in Power System Control"
IEEE Transactions on Power Systems, Vol. May1989
- [KLE 91] F. Kleiner
"Graphics Based Monitoring and Diagnostic Aid System for Large Electric Adjustable Speed Process Drivers", 38th Annual Petroleum and Chemical Industry Conference, IEEE PCIC, Toronto 1991, pp. 85-93
- [LET 90] E.D. Lettner
"Interfacing a portable PC to a microprocessor controlled drive for development work, commissioning and troubleshooting"
Fourth Int. Conf. on Power Electronics and Variable Speed Drives, London, July 1990, pp. 523-528
- [LIM 91] Nikolas Limnios
"Arbres de défaillance"
Edition Hermès, Paris 1991

- [LIU 86] C.-C. Liu, K. Tomsovic
"An expert system assisting decision-making of reactive power/voltage control"
IEEE Transactions on Power Systems, Vol 1, No.3, August 1986
- [MAR 90] J.-C. Martin, C. Tora, D. Lalevée, J. Potage, D. Goulet
"ADEQUAT: Un système d'aide au diagnostic électronique qualifié par arbres de tests", 10ième Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1990, pp. 165-176
- [MAR 91] H.Y. Marathe, C.C. Liu, M.S. Tsai, R.G. Rogers
"An on-line operational expert system with data validation capabilities"
IEEE Transactions on Power Systems, Vol.6, No.3, August 1991
source à revoir
- [MEY 88] Bertrand Meyer
"Object-Oriented Software Construction"
Prentice Hall, 1988
- [MIT 89] J. Mitchell
"Diagnostic Maintenance Expert System for the Hydraulic Subsystem of a Continuous Miner"
IEEE Transactions on Industry Applications, vol. 25, No.5, sept/oct 1989
- [MON 92] Jacky Montmain
"Interprétation qualitative de simulations pour le diagnostic en ligne de procédés continus", Thèse de l'Institut National Polytechnique de Grenoble, 1992
- [MOR 87] J. Morel, B. Ricard, J.P. Tiarri, J.M. David, J.P. Krivine
"DIVA: diagnostic vibratoire des groupes turbo-alternateurs"
SEE- Club 11, Diagnostic et Maintenance par Systèmes Experts- Journées d'études, Paris, octobre 1987
- [NAN 91] E. Nantier, M. Brotz
"Knowledge-based monitoring and diagnosis of rail-guided vehicles"
4th European Conference on Power Electronics and Applications, EPE, Florence 1991
- [NEB 90] G. Nebiacolombo, L. Venzano, M. Mazzarello, P. Pinceti, M. Sepielli
"Holmes: Fault diagnosis and network restoration for industrial power systems"
10ième Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1990, pp.41-55
- [NOR 91] O. Normand, S. Catellani, G. Champenois
"Use of simulation in failure detection and diagnosis of an electromechanical system", 4th European Conference on Power Electronics and Applications, EPE, Florence 1991
- [NOV 89] T. Novak, J.R. Meigs, R.L. Sanford
"Development of an Expert System for Diagnosing Component-Level Failures in a Shuttle Car"
IEEE Transactions on Industry Applications, vol. 25, No.4, 1989, pp. 691-698
- [OGI 91] H. Ogi, H. Tanaka, Y. Akimoto
"Module neural network approach for power system/equipment diagnosis"
Fourth International Conference on Neural Networks, Nîmes, November 4-8, 1991
- [PAG 80] A. Pagès, M. Gondran
"Fiabilité des systèmes"
Edition Eyrolles, Paris 1980
- [PAT 89] R. Patton, P.M. Frank, R. Clark
"Fault Diagnosis in Dynamic Systems - Theory and Applications",
Prentice Hall, 1989

- [PRO 90] C.A. Protopapas, S.D. Kaminaris, A.V. Machias, B.C. Papadias
"An Expert System for Fault Repairing and Maintenance of Electric Machines"
IEEE Transactions on Energy Conversion, Vol.5, No.1, March 1990, pp. 79-83
- [QUE 91] J.M. Quero, J.M. Carrasco, L.G. Franquelo
"A neural controller for quasi-resonant converters"
4th European Conference on Power Electronics and Applications, EPE, Florence
1991
- [REI 87] R. Reiter
"A Theory of Diagnosis from First Principles"
Artificial Intelligence, No. 32, 1987
- [SCH 93] H. Schneider, M. Metz, H. Piquet, H. Foch, J. Courault, O. Lapiere
"Utilization of real-time models for the survey and maintenance of high power
electrotechnical systems",
4th International Conference on Computational Aspects of Electromechanical Energy
Converters and Drives, IMACS TC1, Montréal 1993, pp.39-43
- [SME 91] SMECI "Manuel de Référence, Version 1.54"
Ilog, Gentilly 1991
- [SUH 92] Marie-Christine Suhner, Marc Gabriel, Laurent Claude
"Utilisation de l'AMDEC pour générer la base de connaissances d'un système expert
d'aide au diagnostic. Le cas des systèmes complexes automobiles"
Revue européenne Diagnostic et sûreté de fonctionnement, Vol. 2 - No 2/1992
- [TAL 86] S.N. Talukdar, E. Cardozo, T. Perry
"The Operator's Assistant-- an Intelligent, Expandable Program for Power System
Trouble Analysis", IEEE Transactions on Power Systems, Vol. PWR5-1, No.3,
Aug 1986, pp. 182-187
- [TRI 91] François Trichon
"Modélisation du processus de conception des machines électriques, le système
expert DAMoCLES", Thèse en génie électrique de l'Institut National Polytechnique
de Grenoble, 1991
- [VIL 88] A. Villemeur
"Sûreté de Fonctionnement des Systèmes Industriels"
Édition Eyrolles, Paris 1988
- [VOG 88] C. Vogel
"Génie Cognitif", Masson 1988
- [WAT 86] D.A. Waterman
"A Guide to Expert Systems"
Addison-Wesley, Reading, Massachusettes, 1986
- [WAT 92] K. Watanabe, N. Toyoda
"Diagnosis of Machine Scream - Identification of Standard Number of Bearings and
Diagnosis of the Failure Modes", International Conference on Industrial Electronics,
Control, Instrumentation, and Automation IECON, San Diego 1992, pp. 1058-1063
- [WED 90] H. Wedekind
"Objektorientierung und Vererbung"
Informationstechnik it 32, 2/90, pp. 79-86
- [WEY 89] P. Wey, P. Steiner, C. Jespersen
"Control large motor drives with personal computer", Power, August 1989
- [WOD 90] P. Woda, J.-C. Bumeau
"Système Expert d'aide à la maintenance des équipements de radiodiffusion"
10ième Journées Internationales "Les Systèmes Experts et leurs Applications",

Avignon 1990

- [WOL 86] Bruce F. Wollenberg
"Feasability Study for an Energy Management System Intelligent Alarm Processor"
IEEE Transactions on Power Systems, Vol. PWRS-1, No.2, May1986
- [WOO 91] N. Woodcock, N.J. Hallam, P.D. Picton, A.A Hopgood
"Interpretation of ultrasonic images of weld defects using a hybrid system"
Fourth International Conference on Neural Networks, Nimes, November 4-8, 1991
- [YOO 88a] Wan.C. Yoon, John M. Hammer
"Aiding the Operator During Novel Fault Diagnosis"
IEEE Transactions on Systems, Man, and Cybernetics, Vol.SMC-18, No.1, Jan/Feb
1988, pp. 142-147
- [YOO 88b] Wan.C. Yoon, John M. Hammer
"Deep-Reasoning Fault Diagnosis: An Aid and a Model"
IEEE Transactions on Systems, Man, and Cybernetics, Vol.SMC-18, No.4,
July/Aug 1988, pp. 659-676

Bibliographie

- [ABB 88] K.H. Abbott
"Robust Operative Diagnosis as Problem Solving in a Hypothesis Space"
AAAI88, The 7th National Conference on Artificial Intelligence, Saint Paul,
Minnesota, Aug 1988
- [ABE 88] J. Abel, C. Abel
"Une méthodologie pour le développement des systèmes experts s'appuyant sur la
notion de modèle sous-jacent", 8ieme Journées Internationales "Les Systèmes
Experts et leurs Applications", Avignon 1988
- [AHE 90] M. Ahedo, E. Deschères, J. Moustafiades
"Assistance au diagnostic de pannes d'équipements comportant des axes asservis:
SOCRATE-DIAGNOSTIC, un système expert à vocation pédagogique"
Actes de la Convention IA'91, Paris 1990
- [AHR 91] S. Ahrens, I. Kortaberria, J. Bignon, D. Poupart
"Expert System for Designing Converters for Large Synchronous Machines",
4th European Power Electronics Conference, EPE'91, Florence 1991,
- [AHR 93] S. Ahrens, J. Bignon, D. Poupart
"How to Develop an Expert System for Diagnosis and Maintenance Aid in Large
Electric Variable Speed Drive Applications", 4th International Conference on
Computational Aspects of Electromechanical Energy Converters and Drives,
IMACS-TC1'93, Montreal 1993, pp.
- [AND 89] J.-M. Andrejak
"Evolution of high power drives for high speed induction motors"
EPE Aachen 1989
- [AYE 90] B.E. Ayeb, P. Marquis, M. Rusinowitch
"A new diagnosis approach by deduction and abduction"
G. Gottlob, W. Nejdl: Proceedings of the international Workshop Expert Systems in
Engineering - Principles and Applications, Vienna 1990
- [BAR 81] A. Barr, E. Feigenbaum
"The Handbook of Artificial Intelligence", vol 1,2, William Kaufmann, Inc, 1981/82
- [BOE 90] G. Boella, P. Mussio, P. Mauri, M. Piccoli
"Design of an automatic assistant for failure analysis of linear integrated circuits"
10ieme Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1990
- [BOE 90] G. Boella, P. Mussio, P. Mauri, M. Piccoli
"Design of an automatic assistant for failure analysis of linear integrated circuits"
10ieme Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1990, pp. 137-151
- [BOS 89] B.K. Bose
"Power Electronics - An emerging Technology"
IEEE Transactions on Industrial Electronics Vol. 36, No.3, August 1989
- [BOT 91] L. Bottou
"A formalism for neural networks"
Fourth International Conference on Neural Networks, Nîmes, November 4-8, 1991
- [CAL 88] Calin Alexandru Iosof
"Artificial intelligence techniques in electrical drives"
ICED '88, Proceedings of the International Conference on Electrical Drives,
Poiana Brasov, Romania 1988

- [CAR 89] A. de Carli, F. Venturini
"A first approach to the automatic diagnosis of a machine-tool drive"
EPE, Aachen 1989
- [CHA 88] B.J. Chalmers
"Electric Motor Handbook", Ed. Butterworths, London 1988
- [CHA 90] A. Chapurlet, J.-P. Lalignand
"Système expert de maintenance conditionnelle et d'aide à la conduite d'un compresseur centrifuge", Actes de la Convention IA'91, Paris 1990
- [CHA 90] E.H.P. Chan
"Using Neural Network to Interpret Multiple Alarms"
IEEE Computer Applications in Power, April 1990
- [CHO 91] C. Choulet, A. Haurat, F. Sandroz, M. Tebaa
"Méthode générale d'acquisition de la connaissance experte"
Actes de la Convention IA'91, Paris 1991
- [COR 90] S. Corrias, K-H. Metzger, K. Schweizer, P. Steimer
"Moteur synchrone autopiloté comme entraînement d'un turbocompresseur de 13MW", Revue ABB 7/90
- [DAV 87] J.M. David, J.P. Krivine
"Utilisation de prototypes dans un système expert de diagnostic: le projet DIVA"
7ieme Journées Internationales "Les Systèmes Experts et leurs Applications",
Avignon 1987
- [DAV 90] J.-M. David, J.-P. Krivine
"Explaining reasoning from knowledge level models"
ESAI 90, Schweden, pp. 186-188
- [DEK 90] J. de Kleer, A.K. Mackworth, R. Reiter
"Characterizing Diagnoses"
G. Gottlob, W. Nejdl: Proceedings of the international Workshop Expert Systems in
Engineering - Principles and Applications, Vienna 1990, pp. 1-15
- [DEN 91] T. Denoeux, R. Sobral, T. Langlois, A. Bruchet
"Interprétation de pyrochromatogrammes par réseaux de neurones artificiels"
Actes de la Convention IA'91, Paris 1991
- [DOR 89] J.-L. Dormoy
"Méthodologies pour des systèmes experts de seconde génération"
Journées Francophones de l'Informatique, Nancy, janvier 1989
- [DRE 90] O. Dressler
"Diagnoses as coherent assumption sets"
G. Gottlob, W. Nejdl: Proceedings of the international Workshop Expert Systems in
Engineering - Principles and Applications, Vienna 1990, pp. 47-52
- [DUB 90] L. Dubost
"SEDPAM, un système expert de diagnostic prédictif"
Actes de la Convention IA'91, Paris 1990, pp. 357-377
- [EDW 92] D.G. Edwards
"On-line Diagnosis of Defects in the Stator Winding Insulation Structures of High
Voltage Rotating Machines", International Conference on Electrical Machines,
ICEM'92, Manchester, UK 1992, pp. 1299-1303
- [FAR 91] E. Farges
"Neural networks for real-time automotive diagnosis systems"
Fourth International Conference on Neural Networks, Nîmes, November 4-8, 1991

- [FAU 91] J. Faucher, M. Grandpierre
"Quelques aspects de l'étude de la commande des machines électriques et des convertisseurs statiques à partir des schémas informationnels"
Document interne INPT, Toulouse 1991
- [FEL 91] W. Felloni, V. Lovati
"Design of diagnostic control equipment developed with an expert system, applied to high-speed Italian railways locomotive (ETR 500)"
4th European Conference on Power Electronics and Applications, EPE, Florence 1991
- [FER 89] F. Ferrara, F. Giovannini, E. Paschetta
"An Expert System for packed-switched network monitoring and repair assistance"
9ieme Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1989
- [FIL 92] F. Filippetti, M. Martelli
"Development of Expert System Knowledge Base to On-line Diagnosis of Rotor Electrical Faults of Induction Motors", IEEE Industry Applications Conference, 27th IAS Annual Meeting, Houston, Texas 1992, pp. 92-99
- [FIN 87] Pamela K. Fink, John C. Lusth
"Expert Systems and Diagnostic Expertise in the Mechanical and Electrical Domains"
IEEE Transactions on Systems, Man, and Cybernetics, Vol.SMC-17, No.3, May/June 1987
- [FLE 89] G. Fleischanderl, G. Friedrich, J. Retti
"Model-driven fault localization in audio-routing systems"
9ieme Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1989
- [FOC 91] H. Foch
"Les convertisseurs statiques: du composant à la commande"
Document interne INPT, Toulouse 1991
- [FOR 91] L. Fortuna, A. Gallo, S. Graziani, G. Nunnari, P. Arena
"A neural approach for electromechanical systems modeling"
4th European Conference on Power Electronics and Applications, EPE, Florence 1991
- [FOU 88] R. Fournié, J. Guitton, M. Hérouard, B. Magnon, J.M. Maujean
"Essais d'évaluation et méthodes de diagnostic pour estimer la durée de vie et détecter les défaillances des matériels électriques installés dans les centrales"
EDF Bulletin de la direction des études et recherches - Série B, Réseaux électriques matériels électriques n° 4, 1988
- [FRE 90] H. Freitag
"A generic measurement proposer"
G. Gottlob, W. Nejd: Proceedings of the international Workshop Expert Systems in Engineering - Principles and Applications, Vienna 1990
- [FRU 89] S. Früchtenicht, E. Pittius, H.O. Seinsch
"A diagnostic system for three-phase asynchronous machines"
Fourth International Conference on Electrical Machines and Drives, London 1989
- [GEN 92] G. Gentile, A. Ometto, N. Rotondale, C. Tassoni
"Analysis of Synchronous Motors with Stator Faults", International Conference on Electrical Machines, ICEM'92, Manchester, UK 1992, pp. 1226-1230
- [GIO 90] L. Giordano, A. Martelli
"An abductive characterization of the TMS"
ESAI 90, Schweden

- [GRA 91] R.F. Gray, I.J. King
"User Interface Graphically Improves Generator AI Diagnostics"
IEEE Computer Applications in Power, July 1991
- [GUC 90] T. Guckenbiehl, G. Schäfer-Richter
"SIDIA: Extending prediction based diagnosis to dynamic models"
G. Gottlob, W. Nejd: Proceedings of the international Workshop Expert Systems in
Engineering - Principles and Applications, Vienna 1990
- [HAY 85] Barbara Hayes-Roth
"A Blackboard Architecture for Control", Artificial Intelligence, No. 26, 1985
- [KAR 87] G. Karsai, W. Blokland, C. Biegl, J. Sztipanovits, K. Kawamura, N. Miyasaka, M.
Inui, "Intelligent supervisory controller for gas distribution systems"
Proceedings of the 1987 American Control Conference, Minneapolis, June 1987
- [KLE 89] N. Kleinsorge, U. Putz, H. Stemmler
"Large adjustable speed ac-drives"
EPE Aachen 1989
- [KOH 90] Takeshida Kohda, Koichi Inoue
"Fault Distinguishability of Diagnosis Systems and Selection of Observation Items"
Electrical Engineering in Japan, Vol. 110, No. 6, 1990, pp. 43-52
- [KOH 91] Takeshida Kohda, Satoru Ohki, Koichi Inoue
"Construction of Failure Diagnosis Rules from Diagnosis Tables"
International Conference on Industrial Electronics, Control and Instrumentation,
IECON '91, Kobe, Japan 1991, pp. 137-142
- [KOR 89] A. Korteniemi
"An expert system for fault diagnostics of rotating machines"
Second Symposium on Expert Systems Applications to Power Systems, Seattle,
USA, July 1989
- [LAA 89] H. Lâasri, B. Maître
"Flexibility and Efficiency in Blackboard Systems: Studies and Achievement in
ATOME"
Blackboard Architectures and Applications - Perspectives in Artificial Intelligence"
edited by V. Jagannathan, Academic Press Inc, London 1989
- [LAA 89] H. Lâasri, B. Maître
"Organisation du contrôle dans les architectures de blackboard"
Revue d'intelligence artificielle, vol.3, n° 1/1989
- [LAJ 91] M. Lajoie-Mazenc
"L'alimentation et la commande électroniques des machines synchrones"
Document interne INPT, Toulouse 1991
- [LAT 89] Ph. Lataire, G. Maggetto
"Modern teaching on variable speed induction motor drives"
EPE Aachen 1989
- [LEI 86] D. Leinweber
"Controlling real-time processes on the space station with expert systems"
SPIE, vol. 729 Space Station Automation II, 1986
- [LI 91] X. Li
"What's so bad about rule-based programming", IEEE Software, September 1991
- [MAR 86] M. Marrakchi
"Représentation des connaissances pour l'aide au diagnostic industriel: application
au système expert S.E.DIAG"
Thèse, Université de Valenciennes et du Hainaut-Cambresis, 1986

- [MAR 90] A.J. Marques Cardozo, E.S. Saraiva
"Condition monitoring of variable speed drive systems"
Fourth Int. Conf. on Power Electronics and Variable Speed Drives, London, July 1990
- [MEH 90] Mehrdad Bidjan-irani
"Rule-based Tools for Guaranteeing Design Quality of Integrated Circuits & Systems", 10ieme Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1990, pp. 177-190
- [MEN 88] D. Menexiadis
"Conception d'un système expert d'aide au diagnostic pour les machines tournantes"
Thèse, Université de Valenciennes et du Hainaut-Cambresis, 1988
- [MIL 87] Robert Milne
"Strategies for Diagnosis"
IEEE Transactions on Systems, Man, and Cybernetics, Vol.SMC-17, No.3, 1987, pp. 333-339
- [MIL 91] Robert Milne
"Integration: the key to second generation applications"
IFAC/IMAC '91, pp. 7-9
- [MIT 91] Mitchell D. Lubars
"Representing design dependencies in an issue-based style"
IEEE Software, July 1991
- [NAK 89] S. Nakanishi, H. Hitsuihi
"A few discussion on expert system development for power electronics circuit design", EPE, Aachen 1989
- [NEY 90] A.F. Neyer, F.F. Wu, K. Imhof
"Object-Oriented Programming for flexible Software: Example of a Load-flow"
IEEE Transactions on Power Systems, Vol.5, No. 3, August 1990
- [PAN 90] I.O. Pandelidis, J.F. Kao
"DETECTOR: A knowledge-based system for injection molding diagnostics"
Journal of Intelligent Manufacturing, vol.1, 1990
- [PAW 91] A. Pawlaczyk
"Automatic Diagnostics of H.V. high-power thyristor converters"
4th European Conference on Power Electronics and Applications, EPE, Florence 1991
- [PER 90] R.A. Perez, C.D. Sarmiento, P.J. Victor, S.W. Koh
"A modular expert system design for diagnosing semiconductor wafers"
10ieme Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1990, pp. 153-164
- [PRE 91] B. Premzel, K. Jezernik
"Modern diagnostics support for industrial drive systems"
4th European Conference on Power Electronics and Applications, EPE, Florence 1991
- [REC 87] N. Recrosio, Y. Jegou
"DIONYSOS: Aide au diagnostic de défauts dans les transformateurs"
SEE- Club 11, Diagnostic et Maintenance par Systèmes Experts- Journées d'études, Paris, octobre 1987, pp. 57-60
- [REM 91] W. Remmele, B. Ueberreiter
"Systematic Knowledge Acquisition in Expert Systems"
Siemens Review R&D Special, Spring 1991

- [SAA 89] O. Saarela, H. Markkanen
"Model-based control of power plant diagnostics"
Second Symposium on Expert Systems Applications to Power Systems, Seattle, USA, July 1989
- [SAI 89] J. Saint Michel
"Propulsion by Electric Motor",
3th European Power Electronics Conference, EPE'89, Aachen 1989
- [SCA 87] E.A. Scarl, J.R. Jamieson, C.I. Delaune
"Diagnosis and Sensor Validation through Knowledge of Structure and Function"
IEEE Transactions on Systems, Man, and Cybernetics, Vol.SMC-17, No.3, May/June 1987
- [SED 89] H.G. Sedding, B.A. Lloyd, G.C. Stone, J.M. Braun, J.C. White
"Development of novel instrumentation and expert system concepts in turbine generator condition monitoring"
Fourth International Conference on Electrical Machines and Drives, London 1989, pp. 177-181
- [SET 90] S.K. Sethuraman, P. Sagar
"Knowledge-based system model for a.c. drive control"
Fourth Int. Conf. on Power Electronics and Variable Speed Drives, London, July 1990
- [SHE 90] G.B. Sheblé
"Knowledge Based System Support for Introductory Power System Network Analysis on Personal Computers"
IEEE Transactions on Power Systems, Vol 5, No.1, Feb 1990
- [SOE 90] D.C. van Soest, R.R. Bakker, F. van Raalte, N.J.I. Mars
"Improving Effectiveness of model-based Diagnosis"
10ieme Journées Internationales "Les Systèmes Experts et leurs Applications", Avignon 1990
- [SOT 89] J. Sottile jr, J.L. Kohler
"Techniques for improved predictive maintenance testing of industrial power systems", IEEE Transactions on Industry Applications, vol. 25, No.6, nov/dec 1989
- [SPU 90] G. Spur, S. Weisz
"Application of model-based diagnosis to machine tools"
G. Gottlob, W. Nejd: Proceedings of the international Workshop Expert Systems in Engineering - Principles and Applications, Vienna 1990
- [STR 90] P. Struss, O. Dressler
"Discrepancies - Theories and applications of model-based diagnosis "
invited talk abstract
G. Gottlob, W. Nejd: Proceedings of the international Workshop Expert Systems in Engineering - Principles and Applications, Vienna 1990
- [SUR 91] H. Surmann, B. Kiziloglu, U. Rückert, K. Goser
"Neural nets for defect recognition on masks and integrated circuits: first results"
Fourth International Conference on Neural Networks, Nîmes, November 4-8, 1991
- [TAV 86] P.J. Tavner, B.G. Gaydon, D.M. Ward
"Monitoring Generators and large Motors"
IEEE Proceedings, Vol. 133, Pt.B, No.3, May 1986
- [TAV 89] P.J. Tavner
"Condition monitoring - the way ahead for large electrical machines"
Fourth International Conference on Electrical Machines and Drives, London 1989

- [TAY 89] Tim Taylor, David Lubkeman
"Applications of Knowledge-Based Programming to Power Engineering Problems"
IEEE Transactions on Power Systems, Vol. 4, No.1, Feb 1989
- [TES 90] D.B. Tesch, D.C. Yu, L-M Fu, K. Vairavan
"A Knowledge-Based Alarm Processor for an Energy Management System"
IEEE Transactions on Power Systems, Vol 5, No.1, Feb 1990
- [TRO 88] J. Trost, R. Patzke, H. Frohne
"Diagnose von Läufer-Erdschlüssen in Synchronmaschinen durch Messung von
Zeitverlauf und räumlicher Verteilung des Isolationswiderstandes"
Archiv für Elektrotechnik 71, 1988
- [VAL 91] B. Valiquette, G.L. Torres, D. Mukhedkar
"An expert system based diagnosis and advisor tool for teaching power system
operation emergency control strategies"
IEEE Transactions on Power Systems, Vol.6, No.3, August 1991
- [VEN 91] P. Venet, G. Grellet, G. Rojat
"Identification of significant parameters relative to failure during operation of a
switchmode power supply - application to preventive diagnostic"
4th European Conference on Power Electronics and Applications, EPE, Florence
1991
- [VIL 90] D. Vilain, F. Madar
"Surveillance et diagnostic en ligne d'un four à ciment à l'aide d'un système expert
temps réel", Actes de la Convention IA'91, Paris 1990
- [VOY 87] R. Voyer
"Moteurs de systèmes experts", Eyrolles, Paris 1987
- [ZER 90] K. Zercher
"Constructing decision trees from examples and their explanation-based
generalizations"
G. Gottlob, W. Nejdl: Proceedings of the international Workshop Expert Systems in
Engineering - Principles and Applications, Vienna 1990
- [ZHA 89] Z.Z. Zhang, G.S. Hope, O.P. Malik
"Expert Systems in Electrical Power Systems - A Bibliographical Survey"
IEEE Transactions on Power Systems, Vol.4, No.4, Oct 1989
- [ZOC 90] S.E. Zocholl
"Motor Analysis and Thermal Protection"
IEEE Transactions on Power Delivery, vol. 5, No.3, July 1990



AUTORISATION DE SOUTENANCE

Vu les dispositions de l'arrêté du 30 Mars 1992 relatifs aux Etudes Doctorales

Vu les Rapports de présentations de :

Monsieur Alain NICOLAS

Monsieur Gérard CHAMPENOIS

Monsieur AHRENS Sigurd

est autorisé(e) à présenter une thèse en soutenance en vue de l'obtention
du diplôme de Docteur de l'Institut National Polytechnique de Grenoble,
spécialité "GENIE ELECTRIQUE".

Fait à Grenoble, le 5 novembre 1993

Pour le Président de l'INPG
et par délégation
le Directeur de l'Ecole Doctorale
J.L. LACQUEME

Résumé

L'objectif de cette thèse est de fournir un outil d'aide au diagnostic et à la maintenance d'installations complexes pour aider les ingénieurs dans leur tâche de diagnostic curatif.

L'étude est basée sur la démarche utilisée habituellement par les experts sur un cas industriel réel. Cette démarche a été analysée, et un modèle de connaissances a été développé, en vue de la réalisation d'un système expert.

Le modèle proposé comporte une structuration de la connaissance relative aux entraînements (leurs composants, capteurs, etc...) et une formalisation de la connaissance des experts (règles d'heuristique et stratégie de diagnostic). Cette connaissance s'appuyant sur les observations faites sur l'entraînement (signaux des capteurs, observations du personnel de maintenance), le système expert a été conçu pour être couplé avec un système d'acquisition de signaux et pour fonctionner en parallèle avec l'entraînement.

Pour assurer la modularité et une bonne évolutivité de ce système expert, l'implantation informatique de son prototype a été réalisée autour d'une architecture blackboard. Cette dernière met en oeuvre la structuration et l'exploitation des connaissances.

Mots clés : entraînement électromécanique, diagnostic, système expert, représentation de connaissances, architecture blackboard.

Summary

The aim of this work is to create a tool for diagnostic and maintenance aid for complex industrial drive equipments to help service engineers in their diagnostic task.

This study is based on the experts' usual approach in a real industrial context. This approach has been analysed, and a knowledge model has been developed in order to be integrated into an expert system.

The proposed model contains a structured representation of the drive equipment under diagnosis (drive components, sensors, etc...) and a formalism for expert knowledge representation (heuristic rules, diagnosis strategy). Since diagnostic knowledge is based on observations about the equipment (sensor signals and maintenance personal's observations), the expert system has been designed to be linked to the drive via a data acquisition system.

For modularity and maintainability reasons, the implementation of the prototype expert system is based on a blackboard architecture. This architecture provides a modular structure for data and knowledge representation and the framework for implementing the diagnostic strategy.

Keywords : electrical drive, diagnosis, expert system, knowledge representation, blackboard architecture.

