



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

10
b127 354
S/M2
98/4
2
d'ordre :

Année 1998

THESE

Présentée par

Ayman YOUSSEF

Pour obtenir le titre de

Docteur de l'Université de METZ

Spécialité : Automatique / Production Automatisée

Architecture Distribuée Multi-Experts avec Contrôle Hiérarchique pour le Pilotage des Systèmes de Production

Defendue le 18 juin 1998, devant la Commission d'Examen

Pierre PADILLA	Président Professeur université à L'ENIMetz
Alain HAURAT	Rapporteur Professeur université à L'ESIAnecy
Claude POURCEL	Rapporteur Professeur université à L'EITours
Lilian BARROS	Professeur à l'UTTroyes
Bernard MUTEL	Examineur Professeur université à L'ENSAIStrasbourg
Jacques RICHARD	Examineur Professeur à l'université de NANCY
Michel SPADONI	Co-directeur de thèse Maître de conférences à L'ENIMetz
François VERNADAT	Directeur de thèse Professeur université à METZ

BIBLIOTHEQUE UNIVERSITAIRE DE METZ



022 174513 9

BIBLIOTHEQUE UNIVERSITAIRE - METZ	
N° inv.	19980085
Cote	S/M3 98/4
Loc	Magasin

*" Je pense qu'on ne peut mieux vivre
qu'en cherchant à devenir meilleur,
ni plus agréablement qu'en ayant
conscience de son amélioration"*

"Socrate"

DEDICACES

Il m'est très agréable mais également très difficile d'exprimer ma reconnaissance à toutes celles et tous ceux qui m'ont permis de parvenir à ce stade d'études. D'une part, parce que je ne dois oublier personne, mais également parce que les mots de remerciement sont souvent trop limitatifs pour exprimer mes sentiments.

A mon père

Quoi que je dise, je ne saurais exprimer l'amour et l'admiration que j'éprouve pour toi. Je suis très content d'avoir pu remplir ma mission et te dédier cette thèse de doctorat. Que Dieu te garde.

A ma mère et ma providence

Par vos tendres encouragements et vos innombrables sacrifices, merci Maman, merci Papa, grâce à vous j'ai pu parvenir à ce stade car sans votre amour et votre aide je n'y serais certainement pas parvenu.

A YOUSRA ma femme

Le soleil de ma vie, tu m'avais entouré de ton aide pendant les moments les plus difficiles de ma vie. Tes sacrifices pour moi sont innombrables et je ne saurais les oublier, tu as créé le climat affectueux propice à la poursuite de mes études. Je te Dédie ce travail, expression de ma profonde affectation. Nulle dédicace ne saurait refléter mon admiration, ma profonde reconnaissance et mon amour pour toi et pour mes enfants ALHASSAN et ORNMA.

A mes soeurs et mes frères

Je vous aime toutes et tous.

A Marie-Claire et à Jean-Claude

Je vous aimerai toujours

REMERCIEMENTS

Au terme de ce long et enrichissant travail, je tiens à remercier toutes celles et ceux qui, de près ou de loin, ont contribué à la réalisation de cette thèse.

Que Monsieur François VERNADAT, Professeur à l'université de Metz et directeur de thèse, qui m'a donné l'opportunité de faire ce travail en m'accordant sa confiance soit sincèrement remercié, pour l'intérêt qu'il a témoigné pour cette étude, et pour m'avoir permis, grâce à ses précieux conseils, de mener à bien ce travail.

J'exprime toute ma reconnaissance à Monsieur Michel SPADON, Maître de conférences à l'ENMM, pour tout l'intérêt et le temps qu'il m'a consacré tout au long de ces années de travail. Pour m'avoir aidé tout au long du développement de mes recherches à une époque où le ciel était bas et les nuages très lourds, je le remercie du fond du cœur.

J'aimerais aussi me souvenir de ceux qui m'ont fait l'honneur de faire partie de mon jury de thèse :

- Je suis très reconnaissant à Monsieur Alain HURAT, Professeur à LLP/CESALP, ESAuncey, pour avoir bien voulu se charger de la tâche difficile de rapporteur.*
- J'adresse mes respectueux remerciements à Monsieur Claude POURCEL, Professeur à l'école d'ingénieur de TOURS, pour l'intérêt qu'il a accordé à mon travail en acceptant de le juger en tant que rapporteur.*

Je remercie Madame Lilian BARROS, Professeur à UT Troyes Département GSI, Monsieur Bernard MUTEL, Professeur à l'Ecole Nationale Supérieure des Arts et Industries de Strasbourg, Monsieur le Professeur Jacques RICHARD, UHP-CRAN, Faculté des Sciences, qui ont accepté de participer à mon jury de thèse.

Je remercie vivement Monsieur Pierre PADILLA, Directeur de L'Ecole Nationale d'Ingénieur de Metz, Directeur du LGIPM, de m'avoir accueilli dans son laboratoire, pour l'aide matérielle, dont il m'a fait bénéficier pour sa bonne humeur et sa simplicité, durant les années de recherche. Je le remercie également d'avoir accepté de participer à ce jury.

Je ne terminerai pas sans avoir exprimé des remerciements envers toutes les personnes qui ont contribué à rendre ces dernières années de travail très agréables :

Je tiens à remercier particulièrement Mlle Brigitte FREL, Ingénieur de recherche au sein de l'équipe pour l'aide qu'elle m'a apportée pendant les années de recherches, et sa disponibilité à la finalisation de ce manuscrit.

Je pense ici aux chercheurs de cette jeune équipe avec qui j'ai pu discuter, et qui m'ont permis d'avancer dans mon travail. Que soient remerciés Lyes HAMADI, Yasmina HARANI, je leur souhaite une vie heureuse.

Je tiens à remercier Mlle Mounira HARZALLAH, pour avoir su me motiver et m'encourager durant certains moments difficiles.

Je tiens à remercier Rachid OUBRAHM, mon ami de souffrance, pour sa disponibilité dans les moments où la pression a augmenté, pour avoir accepté la discussion et les échanges d'idées.

Je remercie également Fahmie HAMIDA, Lamia, Michel GARDON, Jérôme BOULENGER, Layla ELKARAMA, Radia ELTAÏB, Berenice DAMASCENO et Yves MARX.

L'ordre de mes remerciements n'a pas d'importance. Tous ceux que j'ai nommé m'ont apporté, à un moment ou à un autre, un soutien décisif.

TABLE DES MATIERES

Introduction Générale

Chapitre I : Problématique de la conduite des systèmes de production	9
Chapitre II : Pilotage d'atelier : Etat de l'art	9
Chapitre III : Pilotage des systèmes de production : analyse du problème et proposition d'une approche	9
Chapitre IV : Modélisation et exploitation des Savoir- Faire	10
Chapitre V : Proposition d'une architecture de conduite : Architecture distribuée avec contrôle hiérarchique	11
Chapitre VI : Conclusion	13

Chapitre I : Problématique de la conduite des systèmes de production

I.1 Introduction	14
I.2 Industries manufacturières	15
I.3 La fonction conduite	17
I.3.1 Interaction "Conduite-Ordonnancement"	17
I.3.2 Interaction "Conduite - Système physique "	18
I.3.3 Relations avec les autres fonctions de l'atelier	18
I.3.4 Caractérisation du type d'atelier	19
I.4 Conclusion	19

Chapitre II : Pilotage d'atelier : Etat de l'art

II-1 Introduction	20
II-2 Structure organisationnelle d'un système de production	22
II-3 Gestion hiérarchisée des décisions	24
II-3.1 Niveau stratégique (long terme)	25
II-3.2 Niveau tactique (moyen terme)	25
II-3.3 Niveau opérationnel (court terme)	25
II-4 Techniques de modélisation organique des fonctions d'une entreprise	26
II-4.1 L'approche GRAI	26
II-4.2 L'approche CIMOSA	28
II-5 Les structures de pilotage et les systèmes multi-agents	30
II-5.1 La structure centralisée	30
II-5.2 La structure coordonnée	31
II-5.3 La structure distribuée	31
II-5.4 La structure distribuée supervisée	31
II-5.5 La structure hiérarchisée	32
II-5.6 Les systèmes holoniques	32

II-6 Les différentes approches de pilotage	34
II-6.1 Pilotage à ordonnancement prévisionnel.....	34
II-6.1.1 Le pilotage à ordonnancement prévisionnel partiel.....	34
II-6.1.1.1 La structure centralisée	35
II-6.1.1.2 La structure hiérarchisée.....	35
II-6.1.1.3 La structure coordonnée	37
II-6.1.1.3.1 Le modèle CODECO	37
II-6.1.1.3.2 Le modèle PCS.....	38
II-6.1.1.4 La structure COSIMA	39
II-6.1.1.5 La structure du Projet ESPRIT 809	41
II-6.1.2 Le pilotage à ordonnancement prévisionnel total.....	41
II-6.1.2.1 Système SYROCO	42
II-6.2 Pilotage sans ordonnancement prévisionnel	43
II-6.2.1 La structure centralisée.....	43
II-6.2.2 La structure distribuée	43
II-6.2.2.1 Le modèle de Bakker.....	43
II-6.2.2.2 Le modèle de Duffie	44
II-6.2.2.3 Le modèle de Tchako	44
II-6.2.3 La structure distribuée supervisée	44
II-6.2.3.1 Le modèle de Kouiss	44
II-6.2.3.2 Le modèle de Tacquard	44
II-6.3 La structure hybride	45
II-7 Synthèse des structures et des approches proposées	45
II-8 Conclusion	46

Chapitre III : Pilotage des systèmes de production : analyse du problème et proposition d'une approche

III.1 Introduction	47
III.2 Système MRP	47
III.3 Analyse de la réactivité	51
III.3.1 Espace virtuel de réactivité.....	53
III.3.2 Niveau de réactivité "AMONT".....	53
III.3.3 Niveau de réactivité "AVALE"	55
III.4 Gestion des réactivités	56
III.5 Proposition d'une approche	59
III.6 Conclusion	62

Chapitre IV : Modélisation et exploitation des Savoir-Faire

IV-1 Introduction	63
IV-2 Présentation du système SIRS	65
IV-3 Modélisation des connaissances : "GRAFCET-M"	65
IV-3.1 Éléments de construction.....	66
IV-3.2 Règles d'évolution.....	67
IV-3.3 Définition formelle du GRAFCET-M	68

IV-3.2 Validation d'un réseau autonome.....	73
IV-3.5 Implantation de la base de connaissances.....	76
IV- 4 Exploitation du système.....	84
IV-4.1 Exemple 1	87
IV-4.1.1 Aspect Capitalisation du Savoir-Faire	87
IV- 5 Exemples d'Applications.....	89
IV-5.1 Application du modèle à la représentation de gammes de fabrication	89
IV-5.2 Application du modèle à la représentation de gammes d'assemblage	91
IV-5.3 Application du modèle à la représentation de gammes d'intervention	92
IV-5.4 Application du modèle à la représentation de nomenclatures	93
IV-5.5 Application	95
IV-6 Conclusion	98

Chapitre V : Proposition d'une architecture de conduite :

Architecture distribuée avec contrôle hiérarchique

V-1 Introduction	99
V-2 Systèmes multi-agents	100
V-2.1 Approches existantes.....	100
V-2.2 Agents	101
V-2.2.1 Caractéristiques d'un agent.....	101
V-2.2.2 Architecture d'un agent cognitif	102
V-2.2.3 Architecture fonctionnelle d'un agent.....	103
V-2.3 Organisation d'un système multi-agents	104
V-2.4 Coopération dans un système multi-agents.....	105
V-2.5 Modes de communication	105
V-3 Systèmes multi-experts.....	106
V-3.1 Principe de fonctionnement	106
V-3.2 Sources de connaissances.....	107
V-3.3 Blackboards.....	109
V-3.4 Contrôle.....	111
V-3.4.1 Résolution de conflits.....	112
V-3.4.2 Focalisation du contrôle	113
V-3.4.3 Représentation du contrôle.....	113
V-3.4.4 Modes de contrôle procédural	114
V-3.4.5 Contrôle hiérarchique	115
V-3.4.6 Mode de contrôle opportuniste.....	116
V-3.4.7 Critères comparatifs	118
V-3.5 Systèmes multi-experts existants	119
V-4 Choix d'un outil d'implantation.....	120
V-5 Présentation de l'outil ATOME (Another TOol for Multi-Expertise)	120
V-5.1 Les blackboards.....	122
V-5.2 Événements	123
V-5.3 Les spécialistes.....	126
V-5.4 Les tâches	128

V-5.4.1 Tâche dirigée par les événements.....	129
V-5.4.2 Tâche dirigée par les règles	130
V-5.4.3 Tâche opportuniste	131
V-5.4.4 Choix d'un type de raisonnement	134
V-5.5 La stratégie.....	135
V-5.6 Le moteur d'inférences.....	136
V-6 Développement de l'application	136
V-6.1 Définition des blackboards.....	136
V-6.1.1 Blackboard "Ressources"	136
V-6.1.2 Blackboard "ProgrammeFabrication"	139
V-6.1.3 Blackboard "Demandes"	145
V-6.1.4 Blackboard "PostesCharge"	147
V-6.1.5 Blackboards "InitSolutionsRCS"	150
V-6.1.6 Blackboards "InitSolutionsSTR"	152
V-7 Définitions de l'architecture du système proposé	153
V-7.1 Définition des spécialistes.....	154
V-7.2 Proposition d'une architecture d'implantation.....	159
Les information manipulées sont de trois types et concernent :	161
V-7.3 Définition du mécanisme de contrôle	161
V-7.3.1 Blackboard "Résumés"	161
V-7.3.2 Définitions des tâches.....	164
V-7.3.3 Définition des événements	166
V-7.3.4 Définition de la stratégie	168
V-7.3.5 Proposition d'une architecture.....	168
V-8 Conclusion	171
Conclusion.....	172
Bibliographie.....	177

Table Des Figures

Figures Introduction Générale

Figure 0-1 : Modélisation du Savoir-Faire d'une famille de gammes	10
Figure 0-2 : SIRS [Système Interactif de Recherche par Séquence]	11
Figure 0-3 : Architecture hiérarchique	12
Figure 0-4 : Distribution hiérarchique du contrôle	12

Figures du chapitre II

Figure II-1 : Naissance de la notion CIM	21
Figure II-2 : Pyramide CIM.....	22
Figure II-3 : Système physique de production.....	23
Figure II-4 : Liens entre le système physique et le système de gestion de production.....	23
Figure II-5 : Les fonctions de pilotage en gestion de production	24
Figure II-6 : Structure décisionnelle à trois niveaux	24
Figure II-7 : Modèle conceptuel GRAI d'un système de production.....	26
Figure II-8 : Schéma de la grille GRAI	27
Figure II-9 : Modèle conceptuel de l'approche CIMOSA	29
Figure II-10 : Différentes structures de pilotage.....	31
Figure II-11 : La structure hiérarchisée	32
Figure II-12 : Architecture générale d'un système holonique	33
Figure II-13 : Les deux types d'ordonnancement prévisionnel.....	35
Figure II-14 : Architecture ORABAID	36
Figure II-15 : Architecture CODECO	37
Figure II-16 : La hiérarchie du centre de pilotage du modèle PCS	38
Figure II-17 : Le modèle conceptuel du centre de pilotage du modèle PCS	39
Figure II-18 : Architecture PAC.....	40
Figure II-19 : Modèle du projet ESPRIT 809 PAC.....	41
Figure II-20 : Schéma de principe de l'architecture SYROCO	42

Figures du chapitre III

Figure III-1 : Activités de gestion d'une entreprise/Fonction Pilotage.....	48
Figure III-2 : Différents niveaux de décision de MRP2	49
Figure III-3 : Plan Industriel et Commercial	50
Figure III-4 : Plan Directeur de Production.....	50

Figure III-5 : Calcul des Besoins Nets.....	51
Figure III-6 : Réactivité "Amont" Réactivité "Avale".....	52
Figure III-7 : Espace virtuel des réactivités.....	53
Figure III-8 : Définition des niveaux de réactivité dans le cadre d'aléas de commande.....	54
Figure III-9 : Définition des niveaux de réactivité dans le cadre d'aléas de production.....	56
Figure III-10 : Réactivité par rapport à une demande "Urgente"	57
Figure III-11 : Gestion de la réactivité à un poste de travail H.C.....	58
Figure III-12 : Gestion de la réactivité à une ressource H.C.	58
Figure III-13 : Gestion de la réactivité à un système de transfert H.C.	59
Figure III-14 : Schéma descriptif de l'approche proposée.....	61

Figures du chapitre IV

Figure IV-1 : Système SIRS	65
Figure IV-2 : Définition des noeuds dans un réseau "GRAFCEM-M"	65
Figure IV-3 : Exemple d'un GRAFCET-M	66
Figure IV-4 : Exemple de GRAFCET-M.....	67
Figure IV-5 : Exemple de représentation matricielle de la fonction Pré.....	68
Figure IV-6 : Représentation matricielle de la fonction d'incidence arrière.....	69
Figure IV-7 : Représentation matricielle des conditions de validation	69
Figure IV-8 : Marquage $M_a = (1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0)$	71
Figure IV-9 : Représentation d'un réseau marqué.....	72
Figure IV-10 : Exemple d'un GRAFCET-M	74
Figure IV-11 : Matrice d'incidence correspondant au GRAFCET-M	75
Figure IV-12 : Structure de base d'un réseau interprété	76
Figure IV-13 : Structure générale de l'expression logique d'une condition.....	77
Figure IV-14 : Syntaxe d'écriture d'une opérante de type "FO"	77
Figure IV-15 : Syntaxe d'écriture d'une opérante de type "PR"	78
Figure IV-16 : Syntaxe d'écriture d'une opérante de type "CM".....	78
Figure IV-17 : Exemple d'application d'une contrainte de séquence CM	79
Figure IV-18 : Syntaxe d'écriture d'une opérante de type "CV".....	80
Figure IV-19 : Exemple d'application d'une contrainte de séquence CV	80
Figure IV-20 : Syntaxe d'écriture d'une opérante de type "CT"	81
Figure IV-21 : Définition des niveaux de passage de valeurs	82
Figure IV-22 : Ensemble des éléments de syntaxe du langage de programmation.....	83
Figure IV-23 : Proposition d'une séquence.....	84
Figure IV-24 : Intégration des modifications	85
Figure IV-25 : Liste d'actions	85
Figure IV-26 : Organigramme des menus	86
Figure IV-27 : Affichage d'une proposition 1 suivi d'une modification 1	87
Figure IV-28 : Affichage d'une proposition 1 suivi des modification 1 et modification 2.....	88
Figure IV-29 : Représentation d'une gamme de fabrication par réseau GRAFCET-M.....	89
Figure IV-30 : Présentation d'une gamme d'assemblage par réseau GRAFCET-M	91
Figure IV-31 : Représentation d'une gamme d'intervention.	92
Figure IV-32 : Représentation d'une nomenclature.	93
Figure IV-33 : Représentation d'une nomenclature par réseau GRAFCET-M.....	94
Figure IV-34 : Guide de butée.....	95
Figure IV-35a : Base de connaissances du Guide de butée	95

Figure IV-35b : Base de connaissances du Guide de butée (Suite 1).....	96
Figure IV-35c : Base de connaissances du Guide de butée (Suite 2).....	97

Figures du chapitre V

Figure V-1 : Structure d'un agent.....	103
Figure V-2 : Architecture fonctionnelle d'un agent	104
Figure V-3 : Communications par partage d'informations	106
Figure V-4 : Analogie architecturale "Système expert - Système à blackboard"	107
Figure V-5 : Architecture d'une source de connaissance.....	108
Figure V-6 : Exemple de structure d'un événement	110
Figure V-7 : Structure d'un blackboard	110
Figure V-8 : Système de contrôle.....	111
Figure V-9 : Premier cas d'une source de conflit.....	112
Figure V-10 : Deuxième cas d'une source de conflit	112
Figure V-11 : Contrôle procédural	114
Figure V-12 : Cycle de base d'un mécanisme de contrôle de type procédural	114
Figure V-13 : Contrôle hiérarchique	115
Figure V-14 : Cycle de base de contrôle du système HASP/SIAP	116
Figure V-15 : Contrôle de mode opportuniste.....	117
Figure V-16 : Cycle de base du système BB-1.....	118
Figure V-17 : Architecture d'ATOME.....	121
Figure V-18 : Boucle de contrôle de base	122
Figure V-19 : Cycle de contrôle de base	122
Figure V-20 : Exemple de critères d'identification du blackboard "Image".....	123
Figure V-21 : Structure d'un événement.....	124
Figure V-22 : Gestion des événements.....	125
Figure V-23 : Caractéristiques d'un spécialiste	126
Figure V-24 : Phases de sélection de spécialistes.....	127
Figure V-25 : Phase de vérification de la pré-condition d'un spécialiste	127
Figure V-26 : Phase d'activation d'un spécialiste	128
Figure V-27 : Caractéristiques d'une tâche	128
Figure V-28 : Structure générale d'une règle de production.....	129
Figure V-29 : Cycle de base du mode dirigé par les événements.....	129
Figure V-30 : Activation d'une tâche dirigée par les événements.....	130
Figure V-31 : Cycle de base du mode dirigé par les règles	131
Figure V-32 : Algorithme d'activation d'une tâche dirigée par les règles	131
Figure V-33 : Calcul d'une priorité d'instance de spécialiste de connaissance.....	132
Figure V-34 : Paramètres associés à un spécialiste	133
Figure V-35 : Définition d'heuristiques.....	133
Figure V-36 : Définition d'une règle d'intégration	133
Figure V-37 : Cycle local d'une tâche opportuniste	134
Figure V-38 : Caractéristique de la stratégie	135
Figure V-39 : Structure d'une règle de production de la stratégie	135
Figure V-40 : Principe de fonctionnement	136
Figure V-41 : Liens inter-niveaux du blackboard ressources	138
Figure V-42 : La présentation des différentes affectations.....	140

Figure V-43 : Ordonnancement des tâches.....	143
Figure V-44 : Les différents liens inter-niveaux	144
Figure V-45 : Blackboard associé à l'exemple du Tableau V-7	147
Figure V-46 : Représentation des postes et de leurs liens	150
Figure V-47 : Les affectations des ressources à partir des gammes	152
Figure V-48 : Les affectations des systèmes de transferts.....	153
Figure V-49 : Proposition d'un architecture d'implantation	160
Figure V-50 : Définition du contenu des agendas de la tâche "Affectations"	165
Figure V-51 : Définition du contenu des agendas de la tâche "Transferts"	166
Figure V-52 : Architecture de contrôle.....	170

Introduction Générale

Chapitre I : Problématique de la conduite des systèmes de production	9
Chapitre II : Pilotage d'atelier : Etat de l'art	9
Chapitre III : Pilotage des systèmes de production : analyse du problème et proposition d'une approche	9
Chapitre IV : Modélisation et exploitation des Savoir- Faire	10
Chapitre V : Proposition d'une architecture de conduite : Architecture distribuée avec contrôle hiérarchique	11
Chapitre VI : Conclusion	13

L'objet de cette thèse concerne la conduite des systèmes de production et porte plus précisément sur la proposition d'une architecture de commande permettant une conduite réactive aux aléas de production. La réactivité doit prendre en compte les aléas internes (panne de machine, rupture de stocks,...) et les aléas externes (commandes urgentes, annulation de commandes) du système de production. Cette architecture, nommée SYCORO (Système de Conduite Réactive par Objectifs), est une architecture distribuée avec contrôle hiérarchique.

Après avoir défini le type de production auquel nous nous intéressons, nous présentons la fonction du pilotage et mettons en évidence la **nécessité d'une double réactivité** du pilotage, par rapport aux commandes urgentes et par rapport au système physique. Ces travaux ont pour objectif le développement d'une **analyse globale des fonctions** liées au pilotage d'un système de production et de permettre une réflexion sur les différentes **modélisations** nécessaires à son implantation. Dans ce cadre, nous présentons une approche et proposons une **architecture d'implantation** prenant en compte la double réactivité qui permet une gestion dynamique du pilotage d'atelier.

La thèse est structurée en cinq chapitres :

Chapitre I : Problématique de la conduite des systèmes de production

Nous présentons la problématique traitée dans le cadre de nos travaux, dans le domaine des industries manufacturières, des productions par lots. Ensuite, nous analysons la fonction de conduite par rapport à l'ordonnancement d'une part, et par rapport au système physique, d'autre part.

Chapitre II : Pilotage d'atelier : Etat de l'art

Nous analysons les différentes approches de modélisation organique de l'entreprise et les différents types de systèmes multi-agents existants dédiés au pilotage d'atelier. Ensuite, nous les classons selon leur structure (centralisée, coordonnée, distribuée, distribuée supervisée, hiérarchisée, holonique) en attribuant à chacune d'elle un ou plusieurs modèles de pilotage. Enfin, nous présentons les différents aspects de ces approches sur lesquels nous tentons de spécifier notre modèle de pilotage.

Chapitre III : Pilotage des systèmes de production : analyse du problème et proposition d'une approche

L'objectif de ce chapitre est de présenter une analyse du problème de la réactivité et d'établir une architecture globale intégrée aux fonctions existantes de l'atelier. Nous définissons la réactivité par rapport au savoir-faire représenté par les "potentialités de l'entreprise", et non par rapport à la vitesse de réaction. En effet, nous mettons l'accent sur la maîtrise du savoir-faire de l'entreprise plutôt que sur les dixièmes de secondes que feraient gagner l'utilisation d'un tel réseau dans le cadre d'un transfert de données. La conséquence de l'application d'une approche par objectifs est le déplacement de certaines décisions liées à l'ordonnancement vers le pilotage. Ce recentrage des décisions sur le pilotage favorise un premier type de réactivité par rapport à une commande et un second type de réactivité par rapport au système physique de production.

Chapitre IV : Modélisation et exploitation des Savoir- Faire

La modélisation et l'exploitation des Savoir-Faire représentent la trace des activités et des compétences d'une entreprise. L'objectif est de mémoriser et de maîtriser les savoir-faire pour qu'ils puissent être réutilisables (Figure 0-1).

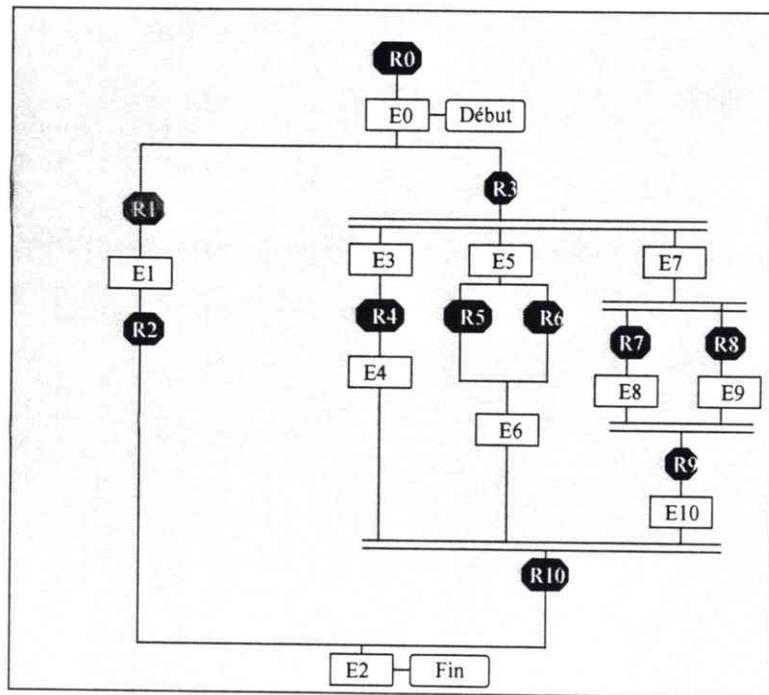


Figure 0-1 : Modélisation du Savoir-Faire d'une famille de gammes

Dans le cadre de nos travaux, nous considérons un savoir-faire "Produits" représenté, d'une part par les nomenclatures des produits, les gammes d'assemblage des produits, les gammes de fabrication des composants et un savoir-faire "Ressources" concernant les moyens et d'autre part représenté, par les gammes de maintenance et les gammes d'intervention.

Il existe trois approches principales permettant d'aborder les savoir-faire :

- l'approche documentaire basée sur l'illustration,
- l'approche système à base de connaissances,
- l'approche système d'information basée sur les notions d'échanges et de mise en valeur des expériences individuelles.

Nous faisons appel à un système à base de connaissances nommé SIRS [Système Interactif de Recherche par Séquence] (Figure 0-2). Ce système est capable de générer, modifier, annuler une ou plusieurs gammes d'opérations à partir de la représentation des connaissances par familles de gammes. Une opération peut être de type : fabrication, assemblage, maintenance, intervention ou représenter le choix d'un composant.

Une famille de gammes se compose d'un ensemble de gammes d'opérations semblables. L'ensemble des gammes de fabrication appartenant à une même famille est représenté par un graphe orienté (figure 0-1). A chaque famille de gammes correspond une base de connaissances. Une gamme est définie comme une liste ordonnée d'opérations. Chaque opération est sélectionnée en fonction d'un ensemble de conditions de fabrication (poste de travail, outillage, nombre de pièces, etc.) et de contraintes géométriques de la pièce. Une proposition de gamme d'opérations est représentée par un chemin dans le graphe.

La base de connaissances est écrite à l'aide d'un langage interne à base de règles de production de type «Si [..., condition,...] Alors [..., Action,...]». La capitalisation du savoir-faire est obtenue à partir des modifications de proposition et ceci grâce à une interface très conviviale.

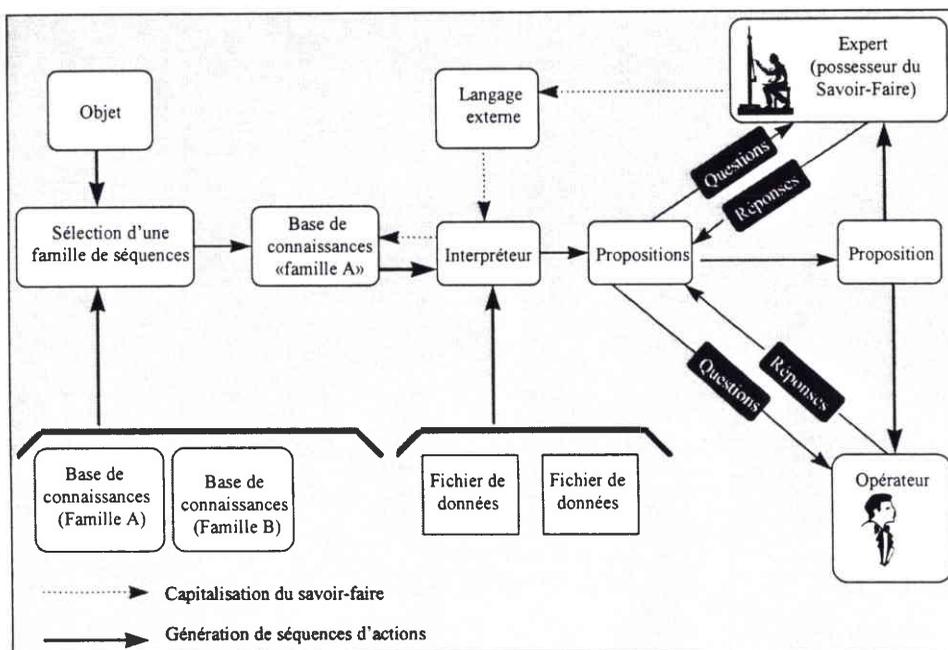


Figure 0-2 : SIRS [Système Interactif de Recherche par Séquence]

Chapitre V : Proposition d'une architecture de conduite : Architecture distribuée avec contrôle hiérarchique

L'architecture classique de fonctionnement d'un atelier est hiérarchisée par niveau de responsabilités (Figure 0-3). En effet, on y retrouve le chef d'atelier, les responsables de secteur, les techniciens et les opérateurs.

Avec ce type d'architecture, chaque niveau possède une partie de la connaissance nécessaire à la bonne marche de la production et les informations y circulent verticalement. Une telle architecture n'est pas propice au développement de la réactivité car il y a une mauvaise exploitation du savoir-faire. En effet, celui-ci est dilué dans la hiérarchie.

C'est pourquoi, du fait des caractéristiques interdisciplinaires des connaissances nécessaires au pilotage d'un système de production, nous proposons une approche dite distribuée qui consiste à répartir le problème à résoudre en plusieurs entités indépendantes. On parle alors de **système multi-agents** et de **système multi-experts**.

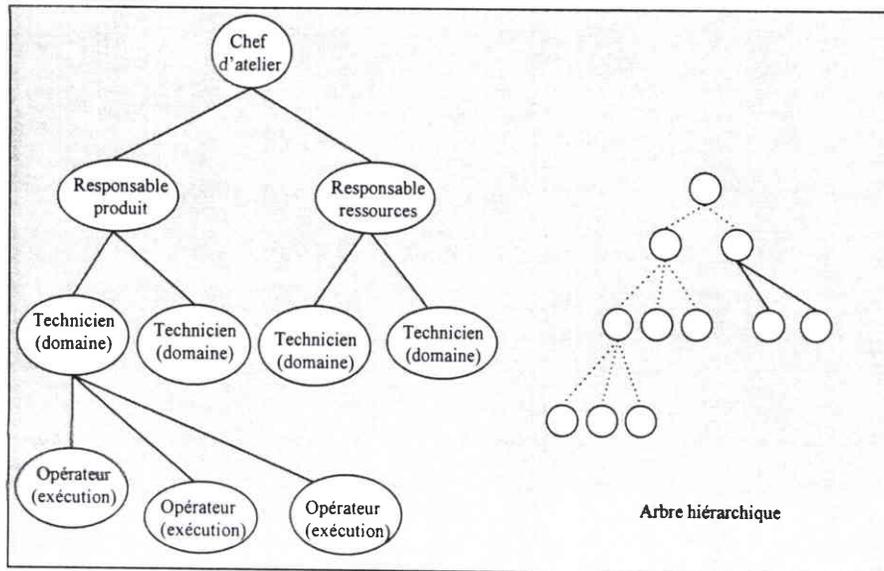


Figure 0-3 : Architecture hiérarchique

Dans ce type d'architecture, les problèmes sont traités à la base et la compétence est distribuée sur l'ensemble des problèmes à traiter. L'ensemble des actions menées par les différentes entités est contrôlé hiérarchiquement sur trois niveaux.

Un premier niveau permet la coordination des actions spécifiques à des domaines particuliers tels que l'affectation de ressources ou le suivi de production. Un deuxième niveau dit de "méta-contrôle" permet de coordonner l'utilisation des entités par rapport aux enjeux globaux de la production. En conséquence, les connaissances ne sont pas dispersées dans les niveaux de la hiérarchie mais situées au plus bas niveau, c'est à dire proches des problèmes à traiter. Le niveau deux ne se préoccupe que du contrôle des actions à mener pour la résolution des problèmes. Le niveau trois prend en charge le contrôle de l'ensemble des décisions stratégiques à prendre permettant de tendre vers les objectifs fixés. C'est ainsi que nous proposons SYCORO, une architecture distribuée avec un contrôle hiérarchique (Figure 0-4).

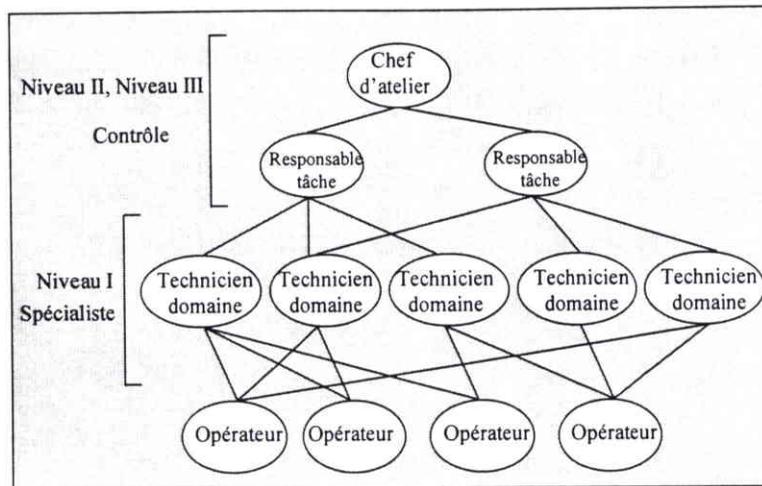


Figure 0-4 : Distribution hiérarchique du contrôle

Ensuite, nous présentons les différents critères permettant la définition de systèmes distribués. Puis, nous présentons les différents types de systèmes multi-agents existants, leur système de contrôle, leur organisation, leur coopération et leur mode de communication. Nous nous attacherons plus particulièrement aux développements des systèmes multi-experts reliés au pilotage d'atelier. Après étude des différentes approches et des caractéristiques définies dans le cadre de notre approche, notre choix s'est porté sur le système ATOME.

Chapitre VI : Conclusion

En conclusion, l'originalité de ce travail consiste à proposer une solution à l'intégration de la réactivité au sein d'un système de pilotage en précisant son domaine d'application, ses avantages, ses inconvénients et les perspectives de développements futurs.

Chapitre I :

Problématique de la conduite des systèmes de production

I.1 Introduction	14
I.2 Industries manufacturières	15
I.3 La fonction conduite	17
I.3.1 Interaction "Conduite-Ordonnancement"	17
I.3.2 Interaction "Conduite - Système physique "	18
I.3.3 Relations avec les autres fonctions de l'atelier.....	18
I.3.4 Caractérisation du type d'atelier.....	19
I.4 Conclusion	19

I.1 Introduction

L'ambition d'une entreprise ne se limite pas seulement à la fabrication d'un produit en fonction des demandes. C'est avant tout et surtout le désir de **satisfaire les besoins des clients**. Ces besoins peuvent être exprimés :

- **Lors de la description du produit** : Dans ce cas, il n'est pas toujours possible pour un client de clarifier ses besoins de manière absolue à partir de spécifications, tout comme pour l'entreprise de satisfaire un client à partir d'un ensemble de spécifications. C'est pourquoi, une entreprise doit s'intéresser à comprendre les préoccupations du client et doit, par conséquent, savoir interpréter un ensemble de spécifications. En effet, il est souvent difficile, voire conflictuel, de vouloir satisfaire les spécifications du client, car le spécialiste du produit est l'entreprise et non le client. Dès lors qu'une entreprise a une connaissance quasi-parfaite du produit, elle doit être en mesure de savoir s'il remplira correctement son rôle, à hauteur des exigences du client. Pour l'exemple, on peut citer le cas de telle entreprise qui fabrique des vitres et des pare-brise pour des constructeurs automobiles. Lors de l'installation des structures et des différents services, il a été décidé d'appliquer un contrôle continu selon les spécifications. Les résultats ont été très décevants et l'existence même de l'entreprise a été mise en cause. Dès lors, la direction a décidé de se préoccuper davantage du produit et de la satisfaction du client, que des seules spécifications. Cette nouvelle approche a permis de passer d'un rendement de 75% à plus de 90%, avec une oscillation forte entre 95% et 99,5%. Le nombre de rebuts est en diminution de 100000 unités pour une année. En conclusion, la satisfaction des clients a été améliorée avec un produit fabriqué à partir du savoir faire de l'entreprise et non exclusivement sur la base de spécifications.
- **Lors d'une demande programmée** : Les besoins sont exprimés dans le cadre de négociations entre le client et le fabricant. Dans ce cas, le client trouve satisfaction à travers le produit mais aussi et surtout à travers des relations développées avec l'entreprise qui débouchent sur du quasi-partenariat.
- **Lors d'une demande non programmée** : Les besoins reposent alors sur une réponse rapide et précise. Dans ce cas, le client doit trouver satisfaction dans une réponse qui correspond au mieux à ses exigences qui peuvent être : le prix, une date au plus tôt, une option supplémentaire. En effet, le client doit comprendre qu'il ne peut pas obtenir satisfaction à n'importe quel prix. Pour cela, il doit définir des priorités dans ses exigences et attendre de l'entreprise un ensemble de propositions. On peut citer l'exemple d'une commande urgente de chaises de couleur jaune avec comme priorité une date au plus tôt à un prix plus élevé. Une première proposition peut être le respect de la date et du prix mais des chaises de couleur verte. Une seconde proposition peut être le respect de la date de la couleur mais un prix différent. En conséquence, l'entreprise doit être capable de fournir un ensemble de propositions respectant l'adéquation «Exigences-Disponibilités des ressources-Potentiel du système physique»

En résumé, il ressort que dans une optique de satisfaction des besoins du client, l'entreprise doit être à même :

- de posséder un savoir-faire et de l'exploiter au mieux.**
- d'être capable de fournir un ensemble de propositions respectant l'adéquation "Exigences clients-Disponibilités des ressources-Potentiel du système physique".**

Attention, l'intégration de commandes urgentes ne doit pas perturber la production en cours. En effet, il nous paraît difficile, voire dangereux, pour l'entreprise de vouloir remettre systématiquement en cause la production en cours. Car cela coûte très cher en prix, ou de qualité et déstabilise les services en charge de la planification. En conséquence, nous proposons que la prise en charge de la commande dite urgente puisse être négociée en collaboration avec l'atelier et ainsi prendre en compte la disponibilité de la production et du savoir faire. En outre, de par la complexité des moyens (ressources et procédés) mis en oeuvre et la versatilité de l'environnement (aléa commandes et aléa physiques), les objectifs à atteindre ne peuvent pas rester figés dans le temps. Cette fluctuation des objectifs doit être intégrée à la production, c'est à dire que le responsable de production doit en permanence vérifier l'adéquation entre l'état des objectifs à l'instant t et les moyens disponibles à cet instant.

Ces besoins de satisfaction de la clientèle dans le monde industriel favorisent l'émergence de nouvelles recherches dans le domaine du pilotage d'atelier. Ces recherches doivent prendre en compte la gestion du savoir-faire et la gestion de ressources humaines et matérielles nécessaires au pilotage d'atelier.

L'objectif de ce travail de thèse est de développer un système de pilotage d'atelier permettant de répondre au mieux à cette préoccupation et ainsi contribuer à l'élaboration de solutions. Ce système est développé au sein d'un projet nommé SYCORO [Système de Conduite Réactive par Objectifs]. Il utilise une architecture distribuée de type multi-experts avec un contrôle hiérarchique.

Le projet SYCORO a pour objectif de prendre en compte les concepts suivants : l'intégration, la flexibilité, la réactivité et la capitalisation des connaissances. Ces termes concernent les moyens et les méthodes de production. Ils s'adressent essentiellement aux productions de type discontinu. En effet, la prise en compte de l'évolution des modes de consommation intéresse en priorité les industries manufacturières.

1.2 Industries manufacturières

La production de biens de nature mécanique concerne des domaines aussi variés que : la construction mécanique, la travail des métaux, l'automobile, l'aéronautique, la construction navale, l'électroménager, etc.

Les industries de transformation mécanique travaillent essentiellement par lots répétitifs. La taille de ces lots tend à diminuer actuellement. Cela entraîne des contraintes particulières qui doivent être prises en compte dès la définition et la mise en oeuvre des systèmes de production.

Pour aider à la mise en oeuvre des systèmes de production, la majorité des entreprises s'équipe de moyens de production pilotés par ordinateur : MOCN, CAO, GPAO, etc. En installant dans les différents sites l'outil informatique et en conservant aux hommes la conduite des opérations, l'entreprise développe une grande capacité de réaction et de polyvalence. L'informatique aide alors à maîtriser toutes les phases de la production dans les délais imposés, prenant en compte des contraintes et garantissant le niveau de qualité. Lorsque les postes de travail communiquent sous la direction d'un logiciel de supervision, l'entreprise est dite intégrée. Dans le cas où les postes de travail communiquent mais restent indépendants, l'entreprise est simplement dite informatisée.

Les lois du marché imposent un certain nombre de contraintes aussi bien aux produits de grande consommation, aux produits industriels et aux biens d'équipement. Ces contraintes doivent être prises en compte au niveau de la production, de son organisation et des moyens à mettre en oeuvre pour les assurer. En effet, la concurrence incite à modifier de manière quasi permanente les produits existants et à les remplacer par d'autres plus performants. A cela, on peut noter que la tendance actuelle est de proposer à la clientèle plusieurs variantes de produits de base ou des options. De la prise en compte de ces nouveaux comportements est né le concept de flexibilité. Ceux-ci permettent ainsi de faciliter une réadaptation et une réaffectation de l'outil de production suivant l'évolution de la demande dans le temps (court et moyen terme).

Parmi les différents types de structures flexibles, on peut citer :

- **les lignes de transfert flexibles** qui se composent de machines, banalisées ou spéciales, programmables. Une ligne de transfert flexible est destinée à la production de pièces morphodimensionnellement proches, appartenant à une même famille. A chaque famille correspond une **gamme standard**. Le pilotage d'une ligne est facilité, car le flux de pièces est linéaire, sans possibilité de conflits de priorité. De plus, le nombre de machines, qui dépend de la famille, est en général très limité, aux environs d'une dizaine.
- **les ateliers flexibles** se composent d'un ensemble de machines à commande numérique, en général une vingtaine, associées à des systèmes de transfert ou de chargement et de déchargement automatique. Un atelier flexible est destiné à la production de pièces variées, morphodimensionnellement différentes, avec des gammes non forcément identiques. Cependant, ces gammes possèdent certaines **opérations identiques** permettant de gérer l'optimisation de l'engagement des machines. Cela suppose un pilotage en temps réel, qui réagit dès l'instant où une machine se libère, en fonction des disponibilités d'usinage des pièces restantes. Donc, la progression des pièces n'est pas prédéterminée.
- **les cellules ou îlots flexibles** qui se composent d'un nombre plus restreint de machines et un seul moyen de transfert, et par là même restent moins complexes qu'un atelier flexible. Le regroupement des machines est réalisé en fonction d'un ensemble d'opérations à exécuter, indépendamment de l'appartenance des pièces à une famille bien déterminée. Au contraire d'un atelier flexible, cette association est liée à la notion d'**opérations spécifiques**. Ainsi, une cellule flexible est caractérisée par la nature des opérations qu'elle permet de réaliser.

1.3 La fonction conduite

La fonction conduite est chargée de réaliser la production prévue et pour cela nécessite un modèle précis de fonctionnement du système à conduire. Celui-ci doit intégrer des représentations diverses telles que : le flux de produits, le flux de matières, le flux d'informations et la définition des ressources. Ce modèle est donc nécessairement multi-représentations, et il faut en gérer la cohérence et l'intégration.

Parmi les fonctions à réaliser par la fonction conduite, on peut citer [Bauer et coll. 94] :

- la planification des ordres de fabrication,
- le lancement de la fabrication,
- la supervision et le contrôle de la fabrication.

La problématique de la fonction conduite est posée par la nécessité d'intégration avec les fonctions d'ordonnancement et d'atelier, ainsi que de disposer d'une réactivité et d'une adaptabilité vu la proximité du système physique de production. **En effet, la conduite se situe à l'intersection de problèmes de gestion prévisionnelle et de fonctionnement du système physique.** Ces deux composantes doivent être prises en compte au niveau de la fonction conduite, dont l'analyse doit nécessairement être menée avec une approche globale du système à conduire.

L'interaction de la conduite avec les fonctions propres d'atelier (opérations, maintenance, qualité, etc.) se traduit par la prise en compte des perturbations du système physique. Pour cela, la conduite doit être dotée de capacités de réaction et d'adaptation pour être à même de prendre des décisions cohérentes et rapides, avec des temps de réponse adaptés aux contraintes du système physique et ne pas perturber les fonctions d'ordonnancement.

La problématique met en évidence l'existence de deux niveaux de réactivité :

- **la réactivité aux aléas externes en amont de la conduite**, qui permet de réagir aux perturbations de la demande (commandes urgentes, modification d'une commande en cours, etc.).
- **la réactivité aux aléas internes en aval de la conduite**, qui permet de réagir aux perturbations issues du système physique (dépannage, casse d'outil, etc.).

Ces deux types de réactivité seront traités plus profondément au chapitre III.

1.3.1 Interaction "Conduite-Ordonnancement"

Les fonctions de conduite et d'ordonnancement ont leurs objectifs propres et doivent rester cohérentes avec l'objectif global de l'entreprise.

La fonction d'ordonnancement permet à la conduite de disposer d'un cadre de décision, dans lequel elle pourra optimiser le lancement selon ses propres critères, en minimisant les risques de divergence. L'horizon de décision est ainsi déterminé par le plan prévisionnel issu de l'ordonnancement. Il doit être inférieur ou égal au temps séparant deux ordonnancements successifs et supérieur ou égal au temps de réponse minimal de l'atelier. Les tâches prévisionnelles et leurs contraintes de réalisation (durée, début au plus tôt, fin au plus tard,

etc.) donnent à la conduite des libertés de décision pour l'affectation d'une tâche prioritaire. La gestion de ces tâches s'effectue en fonction des paramètres réels (état du système, contraintes de maintenance, etc.).

Deux cas peuvent être envisagés :

- les contraintes de délai sont très fortes et le séquençement des opérations complexe. Dans ce cas, l'ordonnancement restreint très fortement le cadre de décision. En conséquence, la conduite remplit des fonctions de distribution du travail et de supervision.
- les contraintes et aléas du système de production sont importants. De ce fait, l'ordonnancement ne peut prévoir précisément le séquençement des tâches. Dans ce cas, la conduite bénéficie d'une marge de manoeuvre importante sans séquençement imposé.

Dans la pratique, le prévisionnel peut estimer correctement 70% à 80% desancements, le reste est fixé à l'atelier. Toutefois, dans le cas où la conduite ne peut effectuer la production dans les délais impartis, un réordonnancement est programmé en accord avec l'ordonnancement, permettant d'assouplir les délais.

En conclusion, les fonctions d'ordonnancement et de conduite ont toutes deux une tâche essentielle pour une optique de réactivité et d'intégration. Une réelle réactivité de l'ensemble implique aussi de donner les moyens à la conduite de jouer un rôle régulateur des perturbations en décentralisant une partie des décisions.

1.3.2 Interaction "Conduite - Système physique "

Le rôle essentiel de la fonction conduite est la prise en charge de la décision finale, qui se traduit en fonction de son niveau d'intervention par : le choix d'un poste de travail, un ordre immédiat d'exécution, etc. Le résultat est la réalisation de listes d'ordres de fabrication complets ou d'opérations élémentaires à exécuter sur une ressource qui peut être une cellule de production ou un poste de travail. Donc, pour atteindre ces objectifs, la fonction conduite doit connaître les capacités réelles du système physique.

Les différents types de conduite existants sont définis en fonction du degré de hiérarchisation, du degré de séquençement des gammes et du niveau d'automatisation du système. Certains systèmes nécessitent une conduite globale de la transformation, d'autres une conduite locale ou par sites (voir II-5 de chapitre II) .

1.3.3 Relations avec les autres fonctions de l'atelier

La nécessité d'une réaction rapide implique une intégration et une coordination des actions de l'ensemble des fonctions de l'entreprise. La définition d'un système de conduite doit inclure la possibilité de prendre en compte différentes contraintes en collaboration avec d'autres fonctions et ainsi d'être en mesure de commander des actions : telles que des modifications de gammes.

Chaque fonction a sa propre représentation, son propre point de vue particulier du système physique de production et donc a un modèle différent.

Pour atteindre l'objectif fixé, la conduite doit assurer la cohérence des différents modèles : description des ressources suivant le point de vue de la production ou de la maintenance, problème de compatibilité des gammes suivant le point de vue méthodes ou conception, par exemple.

1.3.4 Caractérisation du type d'atelier

Dans le cadre de nos travaux, on propose d'introduire une réactivité qui repose sur le savoir-faire de l'entreprise. En conséquence, le choix de tel ou tel type d'atelier ne joue pas un rôle primordial. Toutefois, il apparaît que pour obtenir une plus grande amplitude dans le potentiel de réactivité, il est tout de même souhaitable de manipuler des entités relativement malléables. En conséquence, dans le cadre de nos travaux, l'attention va se porter sur des postes de travail. Que ceux-ci soient disposés en cellule ou autre n'est pas important. Nous rappelons que notre objectif n'est pas d'optimiser l'engagement des postes de travail en faisant appel à un pilotage en temps réel, mais de réagir, à moindre coût, à un aléa. En effet, la réactivité étant obtenue à partir du savoir-faire de l'entreprise, l'objectif est d'adapter la réaction aux disponibilités de l'atelier. C'est à dire que la non-perturbation de la production est prioritaire par rapport à la gestion des aléas. Anisi, la prise en compte d'un aléa peut nous amener à adapter les gammes de fabrication, les gammes d'assemblage et même les nomenclatures. Il nous paraît important de minimiser les perturbations, car elles peuvent déstabiliser les services d'organisation et de planification des besoins.

De manière générale, les aspects de flexibilité, d'adaptabilité sont liés à la structure physique de l'atelier. Par contre, l'aspect réactif nous paraît davantage lié au mode de pilotage. En particulier, dans le cas d'une réactivité basée sur le savoir-faire, on peut parler :

- d'ateliers flexibles réactifs,
- de cellules réactives.

1.4 Conclusion

L'introduction de la problématique des entreprises manufacturières a mis en évidence la nécessité de prendre en compte des notions telles que : l'intégration, la flexibilité, la réactivité et la capitalisation des connaissances. La prise en compte des contraintes du marché de la demande a amené les PME-PMI à développer davantage de flexibilité dans leur système de production. Il s'avère que l'intégration de la notion de flexibilité dans le système de production reste insuffisante. Il est donc nécessaire de pouvoir réagir aux aléas de la demande de production et de l'offre de production. Cela se traduit par la **prise en compte d'une double réactivité : " conduite-ordonnancement " et "conduite-système physique"**. La mise en place d'une réactivité de la conduite sur le système physique de production oblige la conduite à s'informer sur l'état de production à l'instant t prévisionnel du système physique. Pour cela, la conduite s'appuie sur l'ensemble des fonctions de l'atelier.

Chapitre II :

Pilotage d'atelier : Etat de l'art

II-1 Introduction	20
II-2 Structure organisationnelle d'un système de production	22
II-3 Gestion hiérarchisée des décisions	24
II-3.1 Niveau stratégique (long terme)	25
II-3.2 Niveau tactique (moyen terme)	25
II-3.3 Niveau opérationnel (court terme)	25
II-4 Techniques de modélisation organique des fonctions d'une entreprise	26
II-4.1 L'approche GRAI	26
II-4.2 L'approche CIMOSA	28
II-5 Les structures de pilotage et les systèmes multi-agents	30
II-5.1 La structure centralisée	30
II-5.2 La structure coordonnée	31
II-5.3 La structure distribuée	31
II-5.4 La structure distribuée supervisée	31
II-5.5 La structure hiérarchisée	32
II-5.6 Les systèmes holoniques	32
II-6 Les différentes approches de pilotage	34
II-6.1 Pilotage à ordonnancement prévisionnel	34
II-6.1.1 Le pilotage à ordonnancement prévisionnel partiel	34
II-6.1.1.1 La structure centralisée	35
II-6.1.1.2 La structure hiérarchisée	35
II-6.1.1.3 La structure coordonnée	37
II-6.1.1.3.1 Le modèle CODECO	37
II-6.1.1.3.2 Le modèle PCS	38
II-6.1.1.4 La structure COSIMA	39
II-6.1.1.5 La structure du Projet ESPRIT 809	41
II-6.1.2 Le pilotage à ordonnancement prévisionnel total	41
II-6.1.2.1 Système SYROCO	42
II-6.2 Pilotage sans ordonnancement prévisionnel	43
II-6.2.1 La structure centralisée	43
II-6.2.2 La structure distribuée	43
II-6.2.2.1 Le modèle de Bakker	43
II-6.2.2.2 Le modèle de Duffie	44
II-6.2.2.3 Le modèle de Tchako	44
II-6.2.3 La structure distribuée supervisée	44
II-6.2.3.1 Le modèle de Kouiss	44
II-6.2.3.2 Le modèle de Tacquard	44
II-6.3 La structure hybride	45
II-7 Synthèse des structures et des approches proposées	45
II-8 Conclusion	46

II-1 Introduction

Pendant les années de forte consommation, ce sont les entreprises qui dirigeaient le marché à travers leurs programmes de production. Plus récemment, le marché de l'offre s'est transformé en marché de la demande. Aujourd'hui, ce sont les consommateurs qui déterminent l'évolution des produits et de leurs caractéristiques et les entreprises doivent se conformer à leurs désirs.

Ce contexte économique associé à l'avancée des technologies a pour conséquence directe l'augmentation de la concurrence et oblige les entreprises à améliorer en permanence leur niveau de compétitivité.

Pour faire face à cette situation, les entreprises ont adopté des mesures visant à :

- améliorer la qualité des produits,
- élargir la gamme de produits,
- réduire les délais de livraison,
- respecter les délais fixés par le client.

Pour cela, les entreprises ont investi dans de coûteux systèmes automatisés (CAO, CFAO, MRP, etc.) et, en particulier, dans le développement de systèmes de fabrication flexibles. Cependant, les différents moyens mis en oeuvre n'ont permis d'atteindre les objectifs que partiellement. Les expériences ainsi menées ont mis en évidence l'importance de la gestion des informations. En effet, pour améliorer la flexibilité dans une entreprise, la qualité de l'information disponible doit être améliorée. Ceci exige une orientation incontournable vers le traitement intégré des données techniques.

De la même façon, les flux de matières et d'énergie sont abordés comme des problèmes logistiques dans le cadre de la production. On reconnaît aujourd'hui que le flux d'informations doit être traité comme un problème logistique. Ainsi, l'entreprise doit se soucier de répondre à des problèmes tels que :

- accéder à la bonne information,
- accéder à une information conforme aux besoins en quantité et en qualité,
- disposer de la bonne information au bon moment et au bon endroit.

En conséquence, une automatisation effective d'une production présuppose l'action coordonnée de trois fonctions telles que schématisées par la Figure II-1 [Baumgartner *et coll.* 91]:

- automatiser la production,
- améliorer le flux de matières,
- améliorer le flux de l'information.

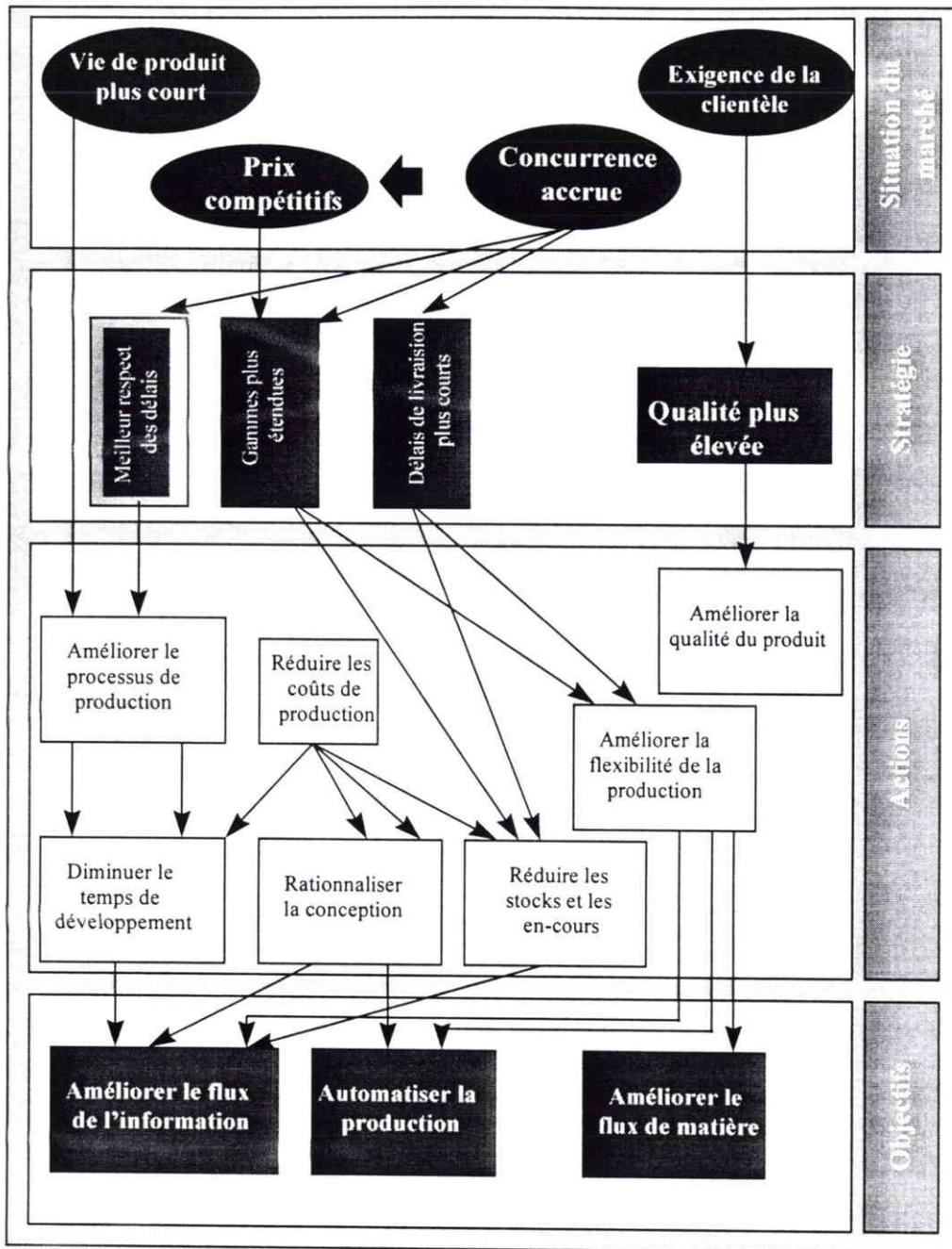


Figure II-1 : Naissance de la notion CIM : "Propositions CIM pour une mise en œuvre de la production"

Les relations entre l'organisation, les techniques d'automatisation et le traitement de l'information doivent être considérées comme un tout qui s'étend à tous les secteurs sans négliger pour autant les compétences et les potentialités des personnels concernés. C'est-à-dire que la prise en compte des techniques de l'information nécessite l'intégration de tous les secteurs de l'entreprise participant à la production.

L'approche du problème de traitement de l'information nécessite la décentralisation des structures traditionnelles et la création de secteurs fonctionnels avec des interfaces claires, afin d'assurer la transparence des fonctions de l'entreprise nécessaires au traitement informatique.

Dans le cadre du concept CIM [AMICE 93 ; Waldner 90], l'entreprise est vue comme un ensemble de fonctions, d'informations et de composants comprenant : les hommes, les machines, les applications, les stockages des données et les réseaux de communication.

L'approche CIM est souvent représentée à partir d'une pyramide (Figure II-2). L'objectif est de mettre en évidence une intégration verticale et horizontale. En effet, initialement l'intégration réalisée était de type horizontal, c'est-à-dire réalisée aux niveaux les plus bas et donc à un faible niveau d'abstraction. La pyramide CIM met en évidence une intégration sur plusieurs niveaux (généralement 6). Le plus haut niveau représente la direction générale de l'entreprise. Le plus bas niveau représente les machines et les équipements.

Le dialogue entre les différents niveaux et la circulation des informations sont abordés par l'utilisation conjuguée des outils de spécification et de conception de systèmes d'information.

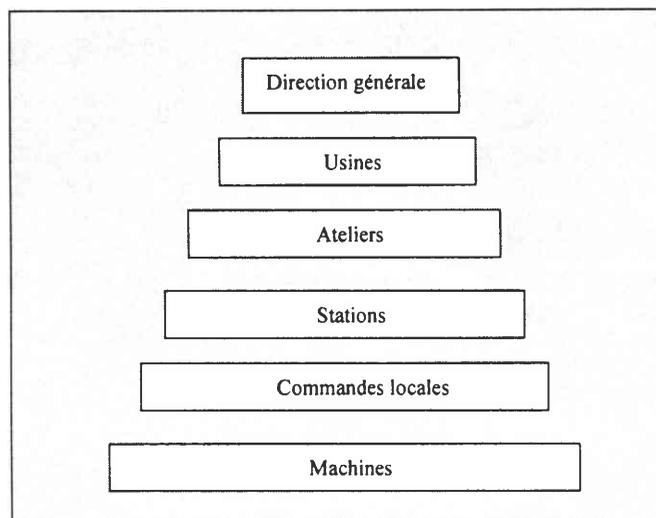


Figure II-2 : Pyramide CIM

Nous allons, dans la suite du document, revenir quelque peu sur la structure organisationnelle d'un système de production afin de mieux situer les différentes approches citées dans cet état de l'art.

II-2 Structure organisationnelle d'un système de production

D'un point de vue systémique, un système de production peut être décomposé en trois parties [Le Moigne 77] :

- **Le système physique**

Le système physique englobe le système opérant (Figure II-3) composé des hommes, des techniques et des flux de matières. Son objectif est d'obtenir des produits finis par la transformation des composants de base (matières premières et produits semi-finis). Il peut être organisé en lignes de fabrication, en ateliers ou par sections.

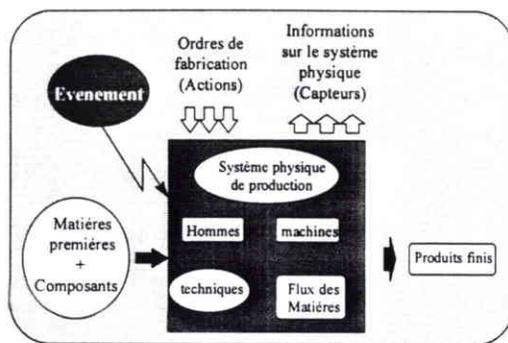


Figure II-3 : Système physique de production

• **Le système informationnel**

Le système informationnel permet de transmettre, traiter et mémoriser les informations nécessaires au système décisionnel. Il sert de liaison entre le système physique, le sous-système décisionnel et le monde extérieur (figure II-4).

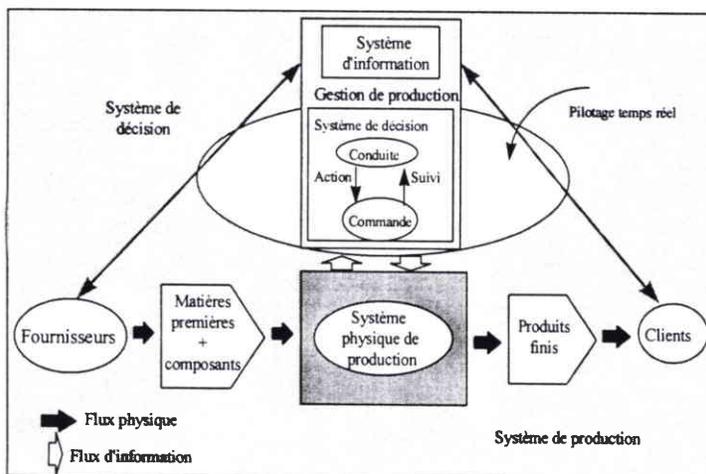


Figure II-4 : Liens entre le système physique et le système de gestion de production

• **Le système décisionnel**

Ce système a pour objectif d'élaborer les décisions, fixant les ordres de gestion transmis au système physique, et agit sur des moyens de production regroupés dans le système physique. On peut dire que son rôle est de piloter la production en transformant les objectifs du marché (objectifs externes) en objectifs opérationnels (objectifs internes à l'entreprise) tout en respectant les contraintes imposées par le système de production (gammes, contraintes technologiques, etc.).

La Figure II-4 synthétise les liens entre le système physique de production, le système de gestion de production, les clients et les fournisseurs.

De plus, on peut distinguer (Figure II-5) le système de pilotage temps réel comme une partie basse du système de décision. Celui-ci contient deux systèmes : le système de conduite de production (superviseur) et le système de commande (envoi des ordres de fabrication) qui ont pour but de piloter le système physique pour qu'il atteigne les objectifs fixés.

Il existe de nombreuses typologies des systèmes de production sans qu'aucune ne soit dominante sur les autres. Celles-ci sont basées sur des critères aussi divers que :

- la continuité de la production (production discrète ou continue),
- la quantité de produits subissant la même transformation (production linéaire, par lots ou unitaire) [Woodward 65],
- le mode de déclenchement de la production (par programme, à la commande ou sur stock) [Giard 88], et
- la complexité et l'incertitude liées aux systèmes de production [Gousty 88].

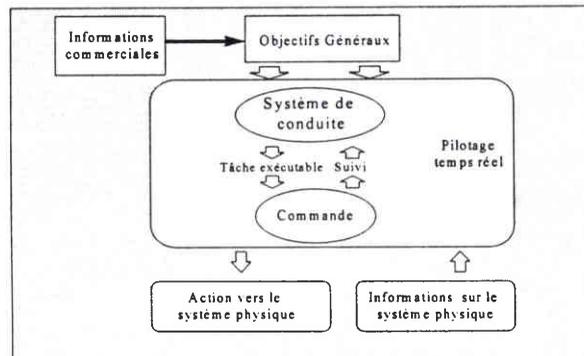


Figure II-5 : Les fonctions de pilotage en gestion de production

Cette grande diversité de type de systèmes de production implique une grande diversité des techniques de gestion de production mises en œuvre. Chaque technique est dédiée à certains types de production plutôt qu'à d'autres. Il y a un grand nombre de modes de gestion de production qui sont adaptés à chaque système de production [Pourcel 86 ; Giard 88 ; Javel 93].

II-3 Gestion hiérarchisée des décisions

On sépare habituellement les décisions en trois niveaux (Figure II-6) : niveau stratégique, niveau tactique et niveau opérationnel en fonction de l'horizon de temps sur lequel ils s'appliquent [Anthony 65 ; Bitran, 89]. Nous allons détailler ces niveaux dans la suite du document.

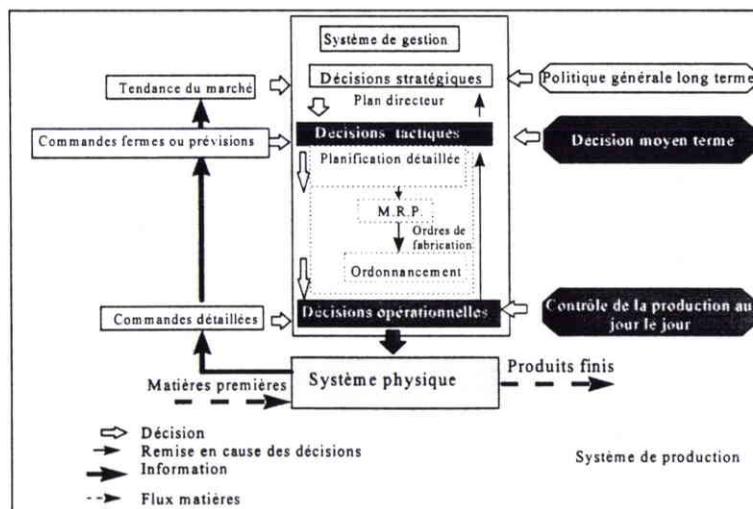


Figure II-6 : Structure décisionnelle à trois niveaux

II-3.1 Niveau stratégique (long terme)

La politique générale ou mission de l'entreprise est décidée à ce niveau. Les décisions qui en découlent concernent la mise au point des installations de production telles que : la taille et l'emplacement de nouvelles usines, l'acquisition d'équipements neufs, le choix de nouvelles gammes de produits, la définition des ressources, etc. Ce sont ces décisions, prises souvent sur un horizon long pouvant s'étendre sur plusieurs années, qui vont être mises en œuvre dans les niveaux inférieurs. Elles indiquent l'orientation des activités de l'entreprise et déterminent les performances demandées aux différents services. Les décisions stratégiques concernent les politiques et les stratégies définies au plus haut niveau de l'entreprise sous forme d'un plan directeur de production.

II-3.2 Niveau tactique (moyen terme)

Les décisions tactiques concernent l'organisation de la fabrication des produits (tant sur le plan matériel que financier) et des ressources (moyens techniques et humains). Cette organisation définit les principaux paramètres de gestion en fonction des prévisions sous forme d'une planification détaillée des besoins nets et d'un ordonnancement prévisionnel.

L'allocation des ressources à des ensembles de produits est effectuée à ce niveau. Les décisions tactiques fixent aussi le plan de production sur un horizon moyen de six à dix-huit mois. Elles nécessitent parfois une agrégation des produits en familles quand le nombre de produits est trop important, ou quand la demande est mal connue au niveau détaillé. Dans ce plan, les quantités à produire par période de planification sont calculées de façon à répondre aux demandes au moindre coût, compte tenu des stocks initiaux disponibles et de divers autres critères qui peuvent venir s'ajouter. Une période de planification varie principalement entre une semaine et un mois. Certaines perturbations, telles que des fluctuations possibles de la demande ou des aléas de production, viennent parfois aussi compliquer le problème.

II-3.3 Niveau opérationnel (court terme)

Les décisions opérationnelles concernent la planification des produits, des moyens et l'exécution des différentes activités de production. Autrement dit, les décisions prises à ce niveau concernent la mise en œuvre des moyens de l'atelier afin de produire, dans les délais, les quantités fixées au niveau supérieur. Ce qui nécessite une désagrégation complète du plan si celui-ci a été construit par famille de produits. Les problèmes que l'on rencontre sont : l'ordonnancement détaillé de la production, le contrôle des stocks, le routage des pièces et l'ordonnancement des moyens de transport entre machines, etc.

Cette décomposition en niveaux décisionnels implique néanmoins de maintenir une approche système. Cela consiste à considérer la production comme un ensemble d'éléments dont on a défini les interrelations. Ces éléments identifiables doivent être situés par rapport à la frontière du système, et nécessitent une définition précise du domaine concerné et de ses relations avec l'environnement.

Il n'y a pas toujours de démarcations très franches entre les différents niveaux, car selon le type d'entreprise, certaines décisions peuvent être prises plus ou moins tôt dans la hiérarchie.

En général, et en particulier dans les systèmes MRP, les décisions à chaque niveau sont prises séquentiellement, du haut vers le bas, et rien n'assure la faisabilité au niveau opérationnel des objectifs décidés au niveau tactique. Ceci explique en partie les retards de livraison et le niveau élevé des en-cours dans l'atelier.

II-4 Techniques de modélisation organique des fonctions d'une entreprise

Parmi l'ensemble des méthodes de modélisation applicables aux organisations des entreprises manufacturières, on retiendra principalement deux approches :

- l'approche systémique : **GRAI**
- l'approche modulaire : **CIMOSA**

II-4.1 L'approche GRAI

D'après les auteurs, l'approche GRAI [Doumeingts 84 ; Roboam 93] s'appuie sur différentes bases théoriques liées :

- à la théorie du système général [Le Moigne 77],
- aux systèmes hiérarchisés, [Mélese 90],
- aux activités discrètes,
- à la gestion de production [Pourcel 86 ; Giard 88].

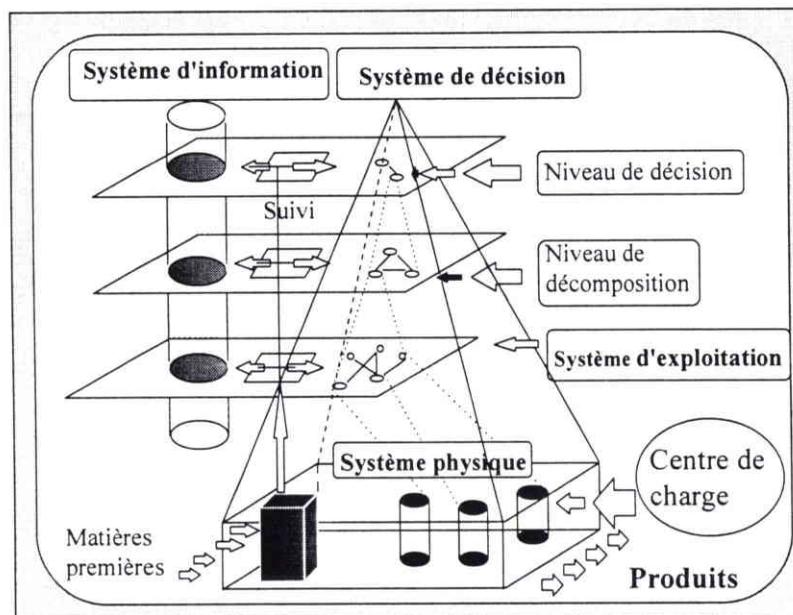


Figure II-7 : Modèle conceptuel GRAI d'un système de production

Le modèle conceptuel de l'approche GRAI reprend l'approche systématique de Le Moigne et considère le système de production suivant quatre sous-systèmes (Figure II-7) :

1. le sous-système physique qui représente le système concret de production ou le système opérant,
2. le sous-système d'exploitation qui recouvre toutes les décisions prises à très court terme. Il fait interface entre le niveau le plus bas du système décisionnel et le système physique,
3. le sous-système d'information qui regroupe l'ensemble des informations circulants ou stockées dans le système de production. Très lié au système de décision, il permet l'échange des informations entre les différents centres de décision et assure la liaison entre les différents systèmes,

4. Le sous-système de décision qui prend les décisions nécessaires à la réalisation des objectifs du système physique en s'appuyant sur les informations fournies par le système d'informations. Il est constitué d'une structure hiérarchisée de centres de décision, dans laquelle un ensemble d'activités de décision ayant même horizon et même période doit être exécuté, suivant les mêmes objectifs donnés par un seul cadre de décision.

L'avantage de cette approche est de situer le système de pilotage dans le système de gestion de production en prenant en compte la définition :

- de l'horizon et de la période qui assurent la synchronisation de la fonction planification à tous les niveaux,
- des différents centres de décision,
- du cadre de décision qui permet de spécifier les dépendances décisionnelles entre les différents centres de décision,
- du module de pilotage qui est un outil informatique associé au centre de décision.

De la décomposition du système de production suivant des critères fonctionnels et temporels, est déduite une représentation graphique à deux dimensions : la grille GRAI des centres de décisions [Doumeingts 84]. Un centre de décision est associé à une fonction et un couple (Horizon / Période), soit une case de la grille. Il est constitué d'activités de décision de différents types et il est relié aux autres centres de décision par des flux informationnels et décisionnels comme le montre la Figure II-8.

Dans le modèle GRAI, le pilotage porte principalement sur les fonctions Planifier (ordres descendants) et Suivre (informations remontantes sur l'état du système physique piloté). Son rôle est d'assurer une boucle de réaction suffisamment rapide pour maintenir le système physique à l'intérieur des objectifs de production fixés. Dans cette boucle, aucune hypothèse n'est faite quant à la structure du système physique (Figure II-8).

Afin de mieux spécifier la boucle de pilotage GRAI et de prendre en compte l'aspect physique, [Bérard 83] a introduit la notion d'activité de pilotage. Ce concept permet de modéliser les centres de décision de bas niveau dans le modèle conceptuel GRAI. Ces centres agissent sur une zone précise du système physique.

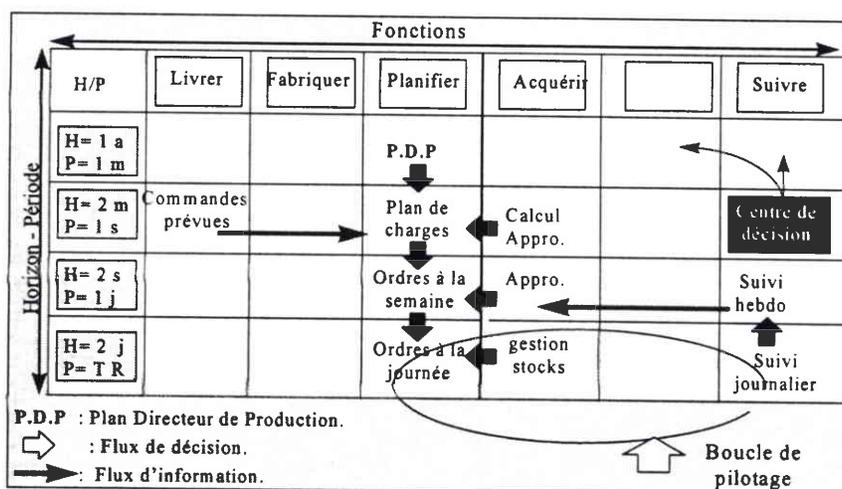


Figure II-8 : Schéma de la grille GRAI

La conduite d'un système de production à l'aide de l'approche GRAI, telle qu'elle est définie dans [Bérard 83], consiste à modéliser le système sous forme d'une hiérarchie d'activités de

pilotage. Chaque activité de pilotage consiste à hiérarchiser un ensemble de boucles simples de pilotage. Cela nécessite des outils capables de couvrir à chaque niveau les besoins des décideurs qui tentent de résoudre leur problème de pilotage en associant des outils de compétences et d'horizons différents. La boucle de rétroaction entre les logiciels de niveaux différents est très mal assurée. D'où une conduite inefficace.

En conclusion, la méthode GRAI ne propose pas d'architecture explicite pour le pilotage d'ateliers. Elle permet de la décrire. Cependant, le concept de pilotage est à la base du modèle GRAI [Breuil 84]. Ce modèle nous permet de :

- situer le problème de pilotage dans le cadre général des problèmes de la gestion de production,
- décrire les éléments de base (activités, flux d'information), et
- mieux spécifier les applications informatiques.

II-4.2 L'approche CIMOSA

CIMOSA (Open Systems Architecture for CIM) [AMICE 89 ; AMICE 93] est une architecture de systèmes ouverts pour la productique développée par le consortium AMICE (European Computer Integrated Manufacturing Architecture) dans le cadre du Programme ESPRIT (Projets 688, 5288 et 7110). Le Consortium AMICE a regroupé plus de vingt et une compagnies de sept pays européens sur sept ans [Vernadat 93].

L'architecture offre un cadre de modélisation de l'entreprise, une infrastructure intégrante et une méthodologie pour l'ensemble du cycle de vie d'un système CIM.

Deux environnements distincts, regroupant l'ensemble des activités d'une entreprise telles que celles réalisées par les entités fonctionnelles, composent cette architecture :

- l'environnement d'ingénierie du système (conception et modification des processus),
- l'environnement opérationnel (exécution des processus par les entités fonctionnelles).

Le cadre conceptuel regroupe une méthode et des outils de modélisation pour aider l'utilisateur à créer le modèle particulier de son entreprise dans l'environnement d'ingénierie. Il se compose d'une Architecture de Référence (fournie par AMICE) et d'une Architecture Particulière propre à l'entreprise. L'Architecture de Référence fournit un langage de modélisation (ou ensemble de modules de base génériques) et des modèles partiels.

Le cadre de modélisation se présente sous forme de trois axes orthogonaux (Figure II-9) :

1. L'axe d'instantiation permet de construire le modèle particulier de l'entreprise à partir de modèles partiels. Ces modèles partiels sont construits à partir des modules de base génériques. Les modules de base génériques sont les primitives élémentaires de modélisation (objets, activités, processus, ressources, etc.).
2. L'axe de dérivation permet de commencer par la modélisation des besoins de l'entreprise (définition des objectifs et des contraintes). Il s'occupe ensuite de la spécification conceptuelle du système (analyse conceptuelle, analyse détaillée, conception du système d'information, évaluation des performances, choix des ressources) et enfin de la description de l'implantation (ressources installées, distribution des fonctions, distribution des informations, gestion des aléas, etc.).

3. L'axe d'élaboration permet de distinguer quatre vues d'un modèle : fonction, information, ressource et organisation. Une vue est un mécanisme d'abstraction permettant de se focaliser sur certains aspects du modèle et d'en ignorer d'autres pour en réduire la complexité.

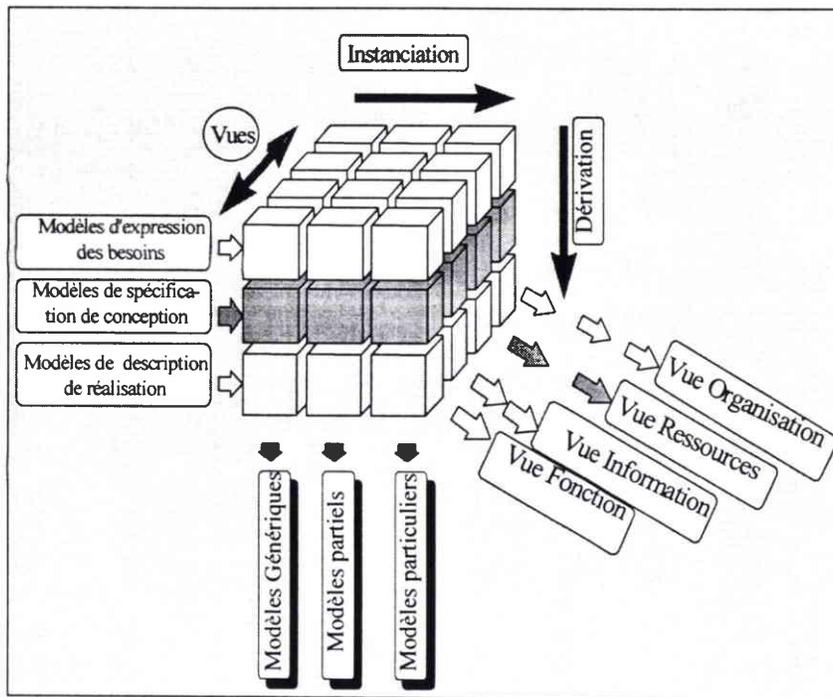


Figure II-9 : Modèle conceptuel de l'approche CIMOSA

Les modèles partiels disponibles peuvent s'appliquer de façon assez générale par exemple à un ensemble d'entreprises ou même à un secteur économique tout entier. Les modèles particuliers sont créés soit en adaptant les modèles partiels aux besoins particuliers, soit directement en particulierisant les modules de base génériques.

A chaque modèle (générique, partiel ou particulier) correspond un niveau d'expression des besoins, un niveau de spécification conceptuelle et un niveau de description de l'implantation. Chacun d'eux offre une vue fonction, une vue information, une vue ressource et une vue organisation.

1. Le niveau d'expression des besoins définit ce qui doit être réalisé par l'entreprise (produits, fonctions, objectifs à atteindre, etc.). Son degré de détails dépend des besoins de l'utilisateur. Comme les niveaux inférieurs, il s'exprime à l'aide du langage fourni par CIMOSA.
2. Le niveau de spécification conceptuelle produit un modèle conceptuel complet de l'entreprise à partir du modèle d'expression des besoins. Il contient plus de détails et peut éventuellement remettre en cause certains choix du niveau précédent. A ce niveau, on peut faire certains choix techniques, tester des solutions, faire des simulations et des analyses de performance. Le temps est pris en compte dans ce modèle.
3. Le niveau de description de l'implantation produit un modèle capable de contrôler l'exécution des opérations de l'entreprise. Il précise les choix faits sur le plan humain et matériel.

A chaque niveau de modélisation, les aspects de l'entreprise pris en compte peuvent être organisés suivant quatre vues :

1. la vue fonction décrit la fonctionnalité du système et l'organisation de son implantation,
2. la vue informations concerne la conception, la spécification et l'analyse du système d'information du système CIM,
3. la vue ressources décrit les éléments matériels, logiciels et humains du système et leur gestion,
4. la vue organisation précise les responsabilités individuelles, les cadres de décision et les niveaux organisationnels de l'entreprise.

CIMOSA a été la première méthodologie à introduire la notion de modélisation et de pilotage des systèmes de production par processus (business process) et non par activité (comme c'est le cas de la méthode GRAI) [AMICE 89]. Il s'agit d'un changement radical d'organisation des systèmes de production visant à abolir les barrières organisationnelles ayant conduit à la création d'îlots d'automatisation et formant un frein au développement de systèmes intégrés [Vernadat 96]. Ainsi, tant CIMOSA que GRAI sont deux méthodes générales permettant de décrire et spécifier un système de production et son système de pilotage, mais ne préconisent à priori aucune architecture de pilotage particulière.

II-5 Les structures de pilotage et les systèmes multi-agents

En général, on distingue deux fonctions pour piloter un système de production. [Bauer et coll. 94 ; Huguet 96 ; Huguet 95 ; Kowk 94] :

- Fonction de planification dont le rôle est d'assurer la programmation d'un ensemble d'actions ou de décisions. Cette gestion est nécessaire à cause de l'incapacité du système de production à gérer la fabrication en temps réel et les changements d'outils et par sa complexité (diversité des ressources, des objectifs et des calculs prévisionnels) [Mercé 87].
- Fonction de conduite en temps réel dont le rôle est de prendre en compte les décisions motivées par les événements dépendants du système de production. Toute la difficulté dans la réalisation d'un système de pilotage réside dans sa capacité à assurer une cohérence des décisions avec des contraintes imposées par la fonction de gestion et par le système physique (pannes, retards, etc.) (Ces contraintes portent sur les objectifs ou les décisions).

Nous résumons dans ce paragraphe les structures des différentes approches de pilotage. Nous décrivons les caractéristiques de chaque structure en analysant à la fois les avantages et les inconvénients des approches utilisant ces structures. Nous citerons les structures centralisées, hiérarchisées, coordonnées, distribuées et distribuées supervisées [Trentesaux 96].

La Figure II-10 donne un aperçu des structures centralisées, coordonnées, distribuées et distribuées supervisées. La structure hiérarchisée sera présentée plus en détails à la fin de ce paragraphe (voir Figure II-11).

II-5.1 La structure centralisée

Cette structure se caractérise par un pilotage localisé (Figure II-10a). Il y a une seule ressource qui supervise. Elle gère la production et réagit aux aléas de production.

II-5.2 La structure coordonnée

Cette structure correspond à un ensemble de structures hiérarchisées où une coopération est possible au même niveau (Figure II-10b). Pour réaliser un ordonnancement prévisionnel, il faut une coopération inter-niveaux. Cette coopération doit améliorer la réactivité et enrichir la qualité de la décision qui permet d'intégrer en temps réel des perturbations.

Pour assurer la cohérence des décisions qui sont gérées par le niveau 1, il faut coordonner l'inter-niveau par la création d'un contrôle qui permet de remettre en cause les décisions du niveau 1. Dans l'industrie, ce niveau est informatisé et en général il intègre la fonction "ordonnancement prévisionnel total".

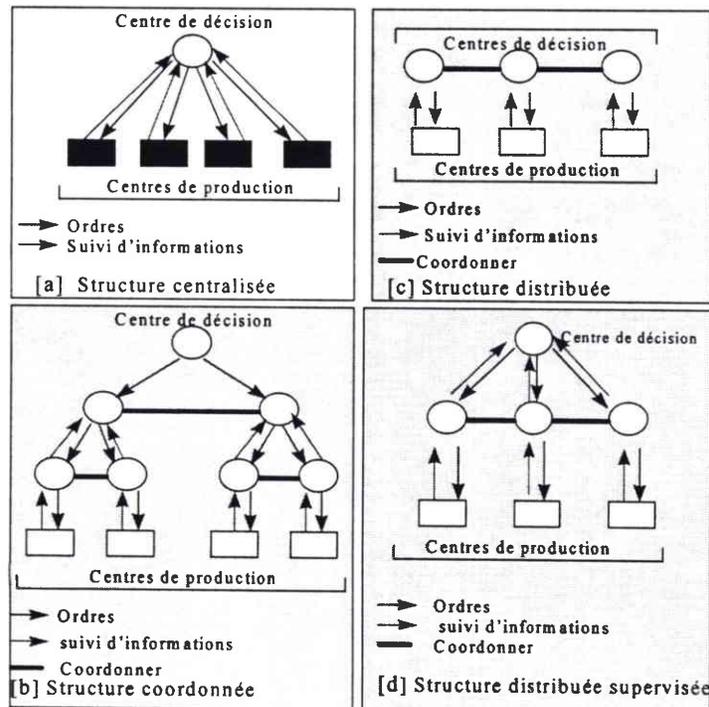


Figure II-10 : Différentes structures de pilotage

II-5.3 La structure distribuée

Dans cette structure, il y a une distribution des décisions sur l'ensemble des ressources pilotant un centre de production (Figure II-10c). Dans ce cas, le contrôle de la structure est rigide et facile à maîtriser mais plus complexe que pour la structure hiérarchisée. Cette structure présente un niveau élevé de réactivité et de flexibilité qui permet difficilement d'anticiper les productions effectives.

II-5.4 La structure distribuée supervisée

Une structure supervisée de pilotage (Figure II-10d) possède une entité qui prend la décision, réagit aux aléas et modifie certaines décisions qui ne respectent pas l'objectif global [Ferber *et coll.* 1988]. Cette structure est un compromis entre les structures distribuées et les structures centralisées. Elle permet une gestion globale et efficace par la centralisation de contrôle d'une part, et une meilleure réaction aux perturbations par la distribution des capacités de décision d'autre part.

II-5.5 La structure hiérarchisée

De nombreuses études ont été réalisées au sujet de cette approche [Gounthier 90 ; Dindeleux 92 ; Tchacko 94]

Dans cette structure, chaque niveau coordonne les unités de pilotage du niveau inférieur jusqu'au niveau le plus bas. Chaque niveau est lié au niveau inférieur et chaque décision est élaborée au niveau où un problème est détecté (Figure II-11).

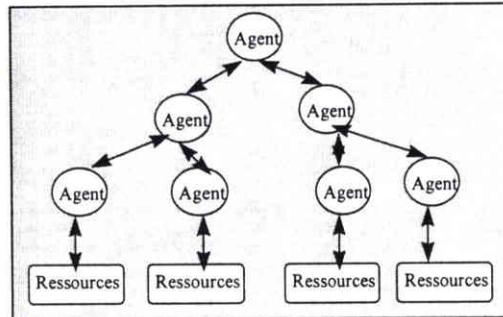


Figure II-11 : La structure hiérarchisée

Les niveaux inférieurs traitent chaque décision comme une contrainte et transmettent en retour une information au niveau supérieur. La gestion temps réel concerne les niveaux ateliers, cellules, postes de travail et équipements.

Dès 1987, une équipe grenobloise s'est intéressée dans le cadre d'un projet EUREKA-FAMOS à l'utilisation des techniques de l'intelligence artificielle, et en particulier des techniques objet et agent, avec les techniques d'optimisation de la recherche opérationnelle pour la conduite des ateliers [Vernadat 87]. Une approche par commande situationnelle avait alors été proposée dans le cadre d'une architecture de pilotage hiérarchisée.

II-5.6 Les systèmes holoniques

L'approche holonique a été introduite par Koestler (1989) pour des organisations sociales et des systèmes s'auto-organisant. C'est une approche originale qui considère un système comme un agglomérat d'unités distribuées et autonomes, notées «holons», qui agissent en tant qu'ensemble d'entités coopérantes. L'application de l'approche holonique envers les systèmes de fabrication fait l'objet d'un des projets de l'Intelligent Manufacturing System (IMS) programme (IMS-HMS 1994).

Un système holonique est une société d'holons qui coopère pour la réalisation d'un but ou objectif. On l'appelle une holarchie. Dans une holarchie, les holons accomplissent leurs tâches de manière autonome mais en même temps, ils connaissent suffisamment tout le reste du système pour contribuer à un rendement optimal global (Figure II-12). Cette approche s'apparente donc à l'approche distribuée.

Un holon est un bloc de construction autonome et coopératif d'un système de fabrication pour transformer, transporter, stocker et/ou valider de l'information ou des objets physiques [Bongaerts et coll. 96]. Un holon se compose, d'une part pour le traitement de l'information et d'autre part du développement physique. Un holon peut représenter un opérateur humain, une

machine ou un logiciel. Enfin, un holon peut aussi faire partie d'une autre holon (IMS-HMS, 1994).

Cette approche est très semblable au concept d'agent utilisé en intelligence artificielle distribuée. L'autonomie d'un holon est définie comme l'aptitude à créer et contrôler l'exécution de ses propres plans et/ou stratégies. La coopération est un processus par lequel un ensemble d'entités créent réciproquement des plans acceptables et exécutent ces plans.

Des exemples d'holons sont les ateliers, les systèmes de transport, les systèmes de contrôle qualité, l'ordonnancement, le processus de planification ou le concepteur de produit. Ces holons peuvent être agglomérés pour former des holons de niveau supérieur comme des départements de planification, des cellules d'assemblage, ou un regroupement de machines. L'idée est d'obtenir une structure de concept générique qui peut s'appliquer à tous les niveaux de l'entreprise (Mathews 96).

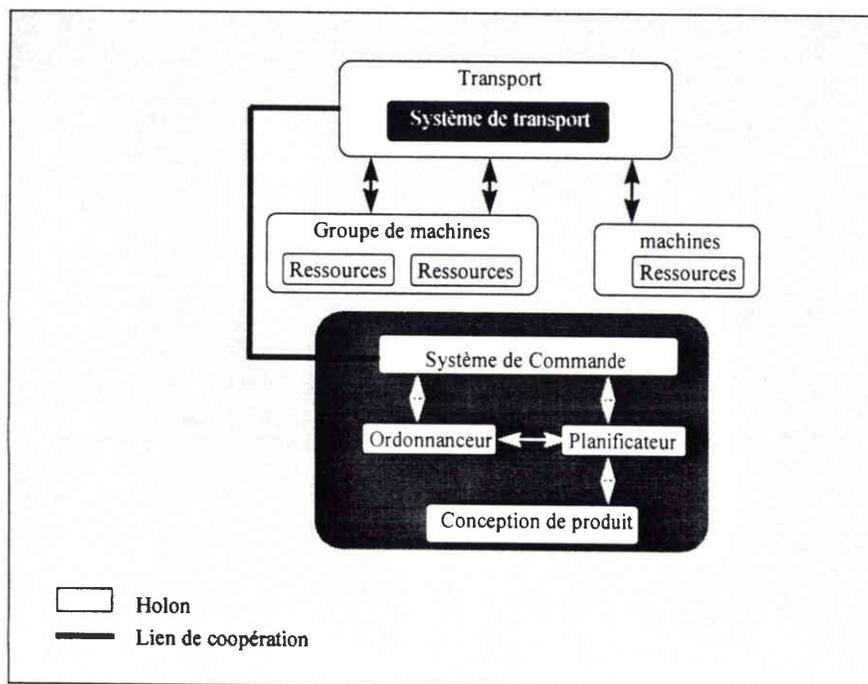


Figure II-12 : Architecture générale d'un système holonique

Cette approche peut être appliquée au contrôle d'une entreprise de fabrication manufacturière. Elle est considérée comme une structure basée sur l'auto-organisation du regroupement d'unités d'organisation en cellules. Ces cellules sont ensuite liées entre elles au niveau supérieur et ainsi de suite. La différence avec CIMOSA vient du fait que, dans l'approche holonique, les unités d'organisation et les cellules sont assimilées à des entités fonctionnelles [Vernadat 96].

Un système de fabrication holonique est une holarchie qui intègre la totalité des activités manufacturières depuis les commandes jusqu'au design, la production et le marketing pour réaliser une entreprise manufacturière (Valckenaers *et coll.* 1994). Bien que ceci ne soit pas le but de l'approche holonique, cette approche peut être aisément implantée ou simulée sur la base de systèmes d'agents.

II-6 Les différentes approches de pilotage

Le pilotage a pour but d'assurer la cohérence de décision entre les ordres issus de la gestion prévisionnelle (planification à court terme) et les actions exécutées au niveau du système de production. Il doit en outre faire face aux contraintes de la gestion prévisionnelle (décisions, objectifs) et aux aléas du système physique (machines en panne, perturbations, retards, etc.).

Ce pilotage affecte le travail des moyens de production afin de respecter le plan et les délais fixés par la planification à moyen terme. Cette activité s'appelle "ordonnancement".

Un ordonnancement est un sous-ensemble du produit cartésien «quoi-où-quand» pour un sous-ensemble de l'ensemble des tâches à réaliser [Parunak 91 ; Gotha 93; Giard 88].

Une tâche est une opération, c'est à dire un ordre de fabrication, qui est liée à des gammes opératoires du produit à réaliser.

Dans ce sens, l'ordonnancement qu'il prévisionnel ou temps réel est réalisé au niveau du système de production. Il est nécessaire de distinguer deux types de pilotage : le pilotage sans ordonnancement prévisionnel et le pilotage à ordonnancement prévisionnel. Dans la suite, nous allons détailler les différentes approches de pilotage avec ou sans ordonnancement prévisionnel par rapport aux différentes structures citées dans les paragraphes précédents.

II-6.1 Pilotage à ordonnancement prévisionnel

L'ordonnancement est défini dans ce cas par le niveau prévisionnel (niveau tactique). Une opération planifiée est une opération appartenant au sous-ensemble des opérations à réaliser, à laquelle est associée une ressource, une date de début et une date de fin de fabrication. Nous distinguons l'ordonnancement prévisionnel partiel de l'ordonnancement prévisionnel total (Figure II-13).

II-6.1.1 Le pilotage à ordonnancement prévisionnel partiel

Un ordonnancement prévisionnel partiel est un sous-ensemble des opérations planifiées qui fait partie de l'ensemble des opérations à réaliser (Figure II-13a). En effet, cet ordonnancement ne concerne qu'une partie des opérations, le reste étant alloué dynamiquement.

Cette approche est structurée selon deux niveaux : le niveau 1 qui génère l'ordonnancement prévisionnel partiel ou total et le niveau 2 qui réalise cet ordonnancement (le niveau 1 est indépendant du niveau 2). L'objectif du niveau 2 est de minimiser les éventuelles remises en question des décisions de niveau 1. La fonction d'ordonnancement prévisionnel doit être intégrée au niveau de pilotage (niveau 2). La fonction de conduite comporte une fonction de séquençage des tâches, dont le rôle est d'établir un ordonnancement partiel portant sur certaines tâches et d'adapter cet ordonnancement en temps réel.

Nous citerons dans ce qui suit, sans rentrer dans les détails, quelques approches de pilotage à ordonnancement prévisionnel partiel trouvées dans la littérature et nous les classerons par rapport à l'ensemble des structures déjà citées.

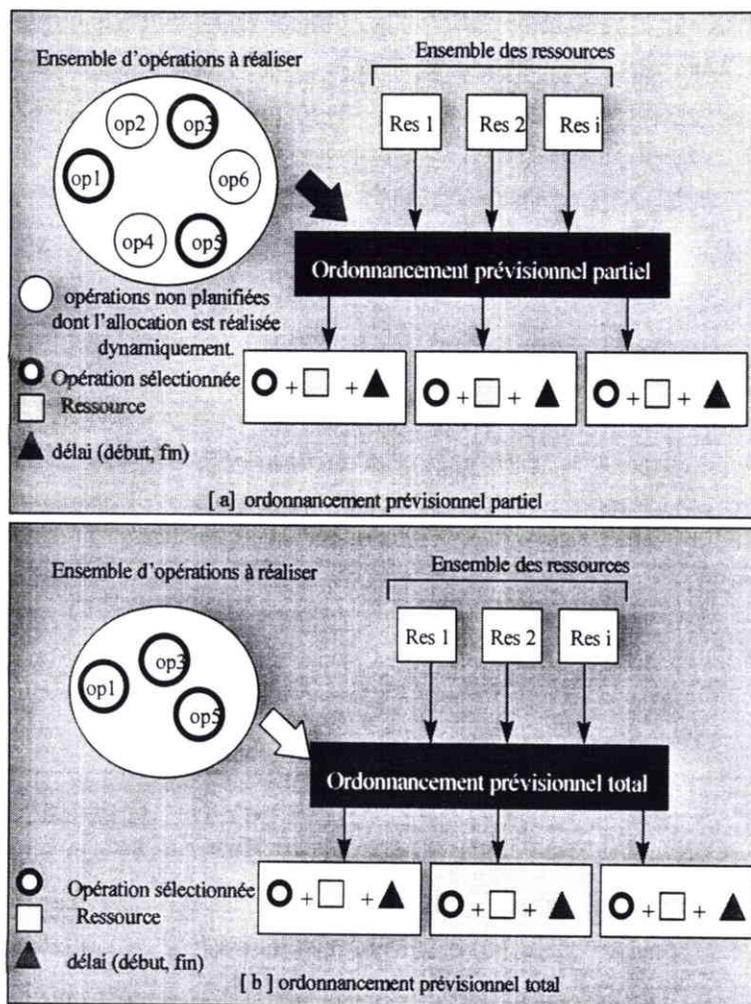


Figure II-13 : Les deux types d'ordonnancement prévisionnel

II-6.1.1.1 La structure centralisée

Le but d'une structure centralisée de pilotage à ordonnancement partiel est de proposer un pilotage temps réel qui prend en compte les dates de réalisation au plus tard des différents ordres de fabrication. On peut citer la méthode "Backward" [Sun 94] qui est utilisée pour le niveau 1, mais qui n'est pas encore exploitée au niveau de pilotage (niveau 2).

L'avantage de cette méthode est qu'elle prend les dates de réalisation au plus tard des ordres de fabrication en temps réel comme une contrainte forte. En résumé, on peut dire que l'intégration d'un ordonnancement prévisionnel partiel implique une complexité croissante de la fonction pilotage. Par ailleurs, la structure centralisée ne semble pas adaptée à un tel ordonnancement.

II-6.1.1.2 La structure hiérarchisée

ORABAID (ORdonnancement d'Atelier Basé sur une AIdé à la Décision) [Roubellat 88] est un logiciel d'ordonnancement temps réel développé au Laboratoire d'Automatique et d'Analyse des Systèmes (L.A.A.S.) à Toulouse à partir de 1980 [Thomas 80].

Cette méthode décompose l'atelier en plusieurs groupes de machines. Chaque groupe traite un ensemble d'opérations (Figure II-14).

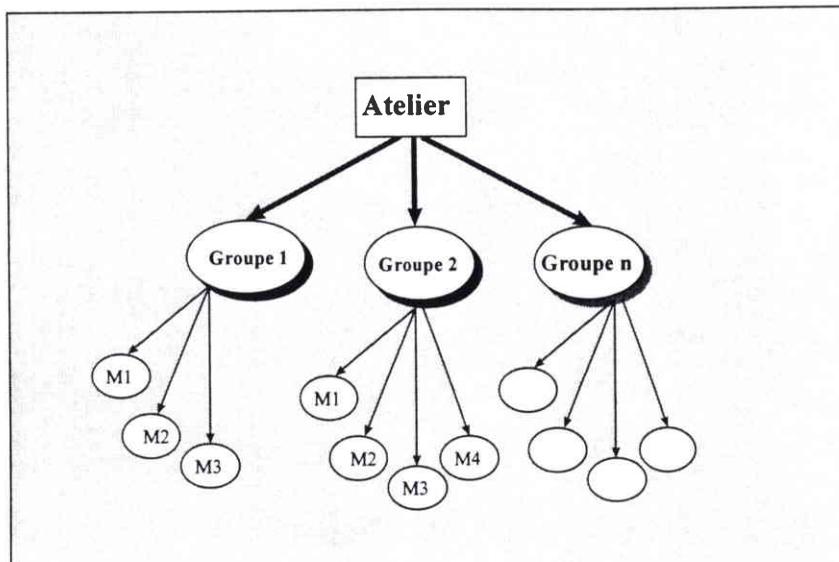


Figure II-14 : Architecture ORABAID

Chaque machine exécute à un moment donné une opération appartenant à l'ensemble des opérations affectées au groupe auquel elle appartient. Dans le but d'augmenter la réactivité, on utilise des groupes de tâches permutable pour l'ordonnancement en temps réel. Pour améliorer la qualité de réactivité temps réel, [Billant 93] a développé une méthode qui se compose en deux phases :

1. La première phase ou phase de prise de décision est déclenchée par les aléas de production (pannes, libérations de ressources, etc.). Elle permet de spécifier les opérations à réaliser au sein d'un groupe d'opérations. Le système de pilotage local doit alors détecter les groupes qui ont engendré de telles perturbations, et par conséquent remettre en question l'ordonnancement. On note deux types de réaction. Les réactions du niveau le plus bas, où les perturbations sur la ressource perturbée sont prises en compte localement dans le groupe d'opérations en cours d'exécution. La perturbation de certaines opérations met en évidence des problèmes de réordonnancement des opérations du groupe qui aide l'utilisateur dans sa prise de décision en temps réel. Le deuxième type de réaction est celui du niveau le plus haut qui intervient dès lors que la séquence du groupe devient non admissible. Cette non-admissibilité est due à l'impossibilité à plus bas niveau de traiter la perturbation concernée. La méthode nécessite alors une remise en cause globale du plan et la recherche d'une nouvelle séquence de groupe.
2. La deuxième phase ou phase d'analyse consiste à chercher à l'aide d'une heuristique un groupe d'ordonnements qui sont regroupés par catégories, selon les degrés de liberté qu'ils autorisent. Dans le cas où deux ordonnancements possèdent le même degré de liberté, ils proposent des groupes d'opérations identiques dont la même permutation ne modifie pas les séquencements des autres opérations.

Dans cette méthode, il n'y a pas de réaction intermédiaire car il est difficile de gérer les cas où une perturbation implique un réordonnement. Par contre, elle permet de réagir aux perturbations que l'on peut classer comme étant de type panne machine ou comme celle du flux.

II-6.1.1.3 La structure coordonnée

Deux méthodes sont considérées pour ce type de structure : le modèle CODECO et le modèle PCS.

II-6.1.1.3.1 Le modèle CODECO

Le modèle CODECO (COnduite DEcentralisée COordonnée d'ateliers) (Figure II-15) a été développé au Laboratoire d'Automatique de Grenoble (L.A.G) [Kallel 85].

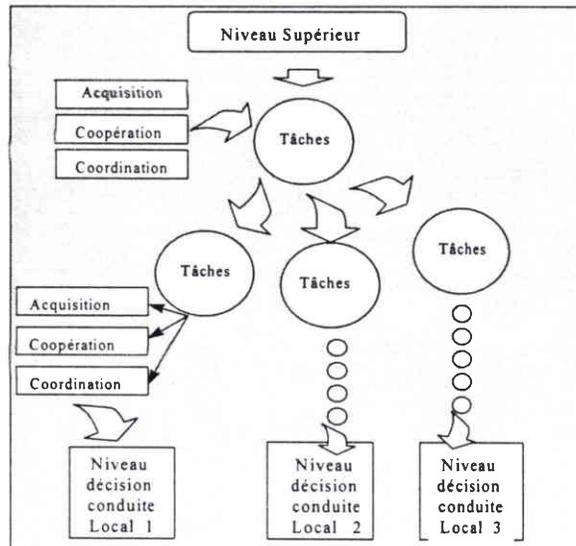


Figure II-15 : Architecture CODECO

Cette méthode permet d'améliorer le processus de pilotage de l'atelier. Elle est considérée comme une extension de la méthode ORABAID en intégrant une solution intermédiaire de réaction basée sur la coopération. Chaque fois qu'un groupe d'opérations perd son caractère de permutableté et que la séquence du groupe devient non admissible, chaque centre de décision tente d'optimiser localement les décisions et les ajuste avec celles des autres centres de décision sous le contrôle d'un coordinateur.

La conduite CODECO se décompose en trois niveaux :

1. Les conduites locales sont chargées du lancement des activités sur la machine. Elles peuvent prendre des décisions qui ne remettent pas en cause les ordres reçus du coordinateur.
2. La coopération des conduites locales représente le niveau intermédiaire proposé. Elle consiste à rechercher une solution principale pour chaque centre de décision indépendamment l'un de l'autre et à la modifier de manière à respecter les contraintes d'interaction.
3. La coordination de l'atelier consiste à élaborer un plan de fabrication.

Parmi les avantages de ce modèle, on peut citer le fait que la solution de coopération introduite permet d'assurer à chaque centre de décision une autonomie locale tout en garantissant la cohérence.

Parmi les inconvénients, on peut mentionner : le problème de suivi qui reste un problème majeur, les aléas et les problèmes liés à leur propagation et le problème de la détermination le degré du liberté concernant les marges allouées à chaque poste de conduite local.

II-6.1.1.3.2 Le modèle PCS

Le modèle PCS (Planification, Conduite, Suivi), décrit par la Figure II-16, est basé sur une approche hiérarchisée du programme d'atelier et du traitement des perturbations [Archimede 91].

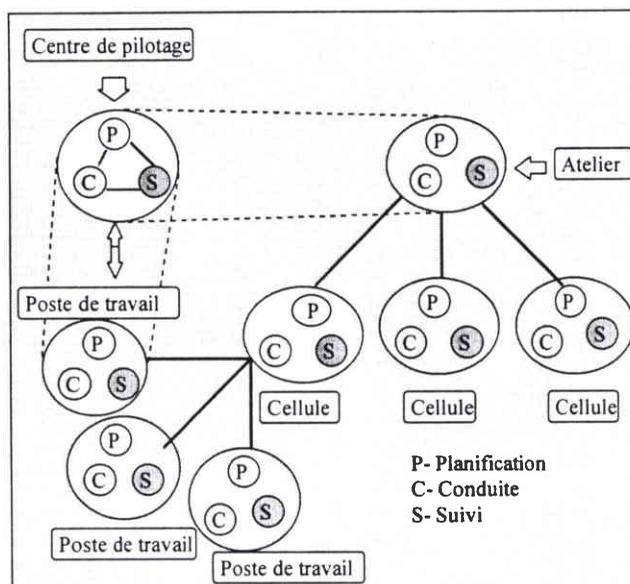


Figure II-16 : La hiérarchie du centre de pilotage du modèle PCS

Le modèle PCS vise à proposer une solution réactive, distribuée et hiérarchisée au problème de pilotage d'atelier. L'aspect réactif se justifie par le souci d'être, d'une part, le plus indépendant possible de l'importance du temps de calcul pour l'obtention d'un plan de fabrication, et d'autre part, de traiter les différentes perturbations survenant en cours d'exécution.

Le centre de pilotage "PCS" s'articule autour de trois systèmes de base présentés par la Figure II-17, capables de s'exécuter en parallèle et associés aux fonctions de planification, de conduite et de suivi. Chaque centre PCS assure les fonctions de planning prévisionnel (acquisition, planification et coordination), de conduite (coopération, réaction et régulation) et de suivi (alimentation et organisation).

Le système de planification est constitué de trois sous-systèmes : l'acquisition qui compare les charges planifiées au potentiel de production, la planification qui permet de résoudre les problèmes généraux de planification, et la coordination qui permet de favoriser la coopération entre ces sous-systèmes.

Le système de conduite est constitué de trois sous-systèmes : la régulation qui assure le traitement des perturbations de types incidents et dérives, la réaction qui agit lorsque le système de régulation est dans l'incapacité de traiter la perturbation, et la coopération qui assure les échanges avec les centres de même niveau.

Le système de suivi est constitué de trois sous-systèmes : la base de données qui assure l'acquisition des informations sur l'état de l'atelier, l'organisation qui assure la mise en oeuvre des procédures de supervision et l'alimentation qui assure l'interface avec le centre de niveau supérieur.

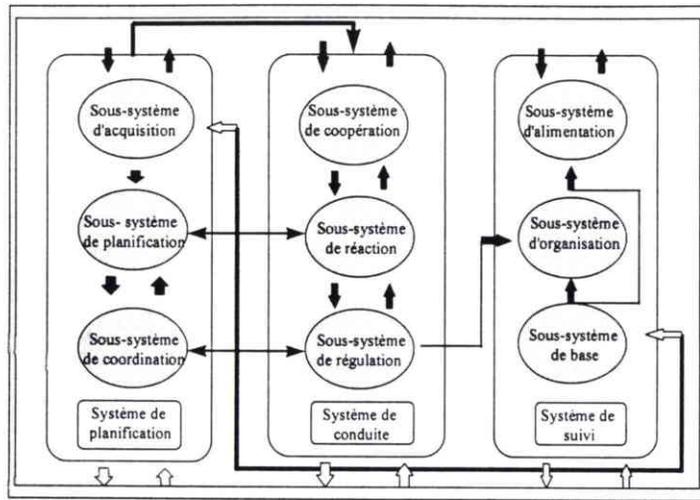


Figure II-17 : Le modèle conceptuel du centre de pilotage du modèle PCS

Le système de conduite a un fonctionnement anonyme par rapport au système de planification jusqu'à réception du nouveau plan. La mise en cause d'un plan sera assurée par le système de conduite qui signifie que le système de conduite contient des outils de replanification.

Cette approche présente comme avantages une solution intéressante au niveau coopération et coordination des niveaux de pilotage, une meilleure intégration de la réactivité au sein de la structure elle-même, et une autonomie décisionnelle qui permet de diminuer le nombre de remises en cause de l'ordonnancement prévisionnel partiel. Par contre, le problème de conduite temps réel n'est abordé que par l'aspect réajustement du plan (coopération avec les systèmes de même niveau, régulation du niveau inférieur). De plus, la propagation des perturbations et des aléas n'est pas maîtrisable facilement.

II-6.1.1.4 La structure COSIMA

L'approche PAC (Production Activity Control) a été développée dans le cadre du projet européen Esprit 477 par le consortium COSIMA (Control System for Integrated Manufacturing) [Bauer *et coll.* 91 ; Browne 88]. Le point central de cette approche est l'architecture PAC (Figure II-18a), qui est dédiée soit au pilotage d'une cellule de production, soit au pilotage d'un atelier constitué de machines organisées en un seul niveau.

L'architecture PAC repose sur cinq fonctions fondamentales :

1. L'ordonnanceur : il tient compte des exigences provenant des ordres de fabrication, collecte des informations du superviseur sur la situation réelle de l'atelier et réalise un plan de fabrication.
2. Le répartiteur : il se charge de coordonner les postes de travail, envoie les ordres d'exécution au niveau inférieur en respectant les ordres des opérations et contrôle les approvisionnements des matières pour les postes de travail.
3. Le transporteur : il s'occupe de l'alimentation des postes en matières ou produits.
4. Le producteur : il s'occupe du lancement de l'exécution et du renvoi au superviseur des informations sur l'état des outils, des machines et des informations sur le chargement, le déchargement et la transformation des pièces.

5. Le superviseur : il se charge de collecter des données issues du système de manutention en temps réel, puis il renvoie ses informations aux modules ordonnancement et répartiteur afin de faciliter la prise de décision.

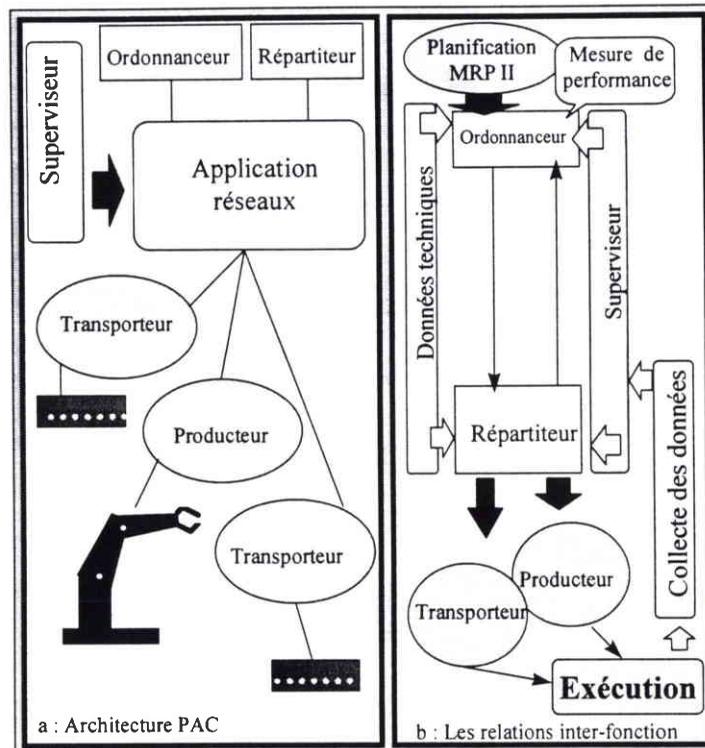


Figure II-18 : Architecture PAC

La Figure (18b) montre la structuration fonctionnelle et précise les données échangées entre les diverses fonctions.

Le système PAC se situe dans le modèle de référence CIM au niveau cellules ou ateliers (Figure 2). Il propose une structuration fonctionnelle du problème du pilotage. Chaque fonction est associée à un module qui communique avec les autres modules à partir d'un réseau d'application. Celui-ci fait circuler les demandes et détermine l'utilisateur capable d'y répondre. Ensuite, il fait parvenir la réponse au demandeur.

On remarque que l'architecture a mis en évidence trois notions importantes :

- l'existence d'un **niveau décisionnel local** pour chaque cellule,
- la **cohérence** ordonnancement/distribution du travail,
- les **indicateurs de performance** permettant une adaptation des règles de gestion en fonction de l'évolution du système de production.

Cette approche permet de traiter l'ordonnancement, la conduite-supervision, le transfert et la gestion des moyens de production.

II-6.1.1.5 La structure du Projet ESPRIT 809

Ce projet avait pour but le pilotage d'un atelier de production de lots de petite quantité, dont l'architecture est très proche de l'architecture PAC comme le montre la Figure II-19 [Pluymb 90]. Grâce à une base de données commune, les différents modules communiquent entre eux.

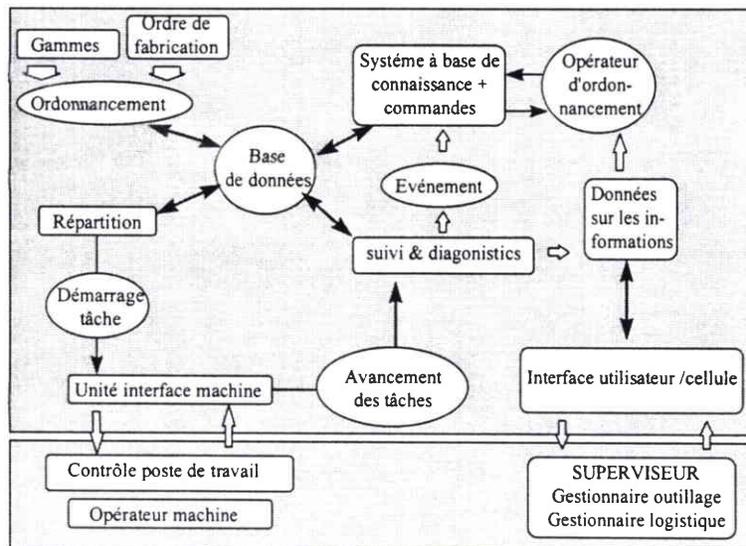


Figure II-19 : Modèle du projet ESPRIT 809 PAC

Les fonctions de ces différents modules sont :

- l'ordonnanceur, qui génère un plan de travail pour la réalisation des ordres de fabrication (gammes) pour une période donnée,
- le répartiteur, qui utilise le plan de travail pour répartir les opérations sur les machines,
- le module de suivi, qui alimente les différents modules sur l'état de l'atelier,
- le système à base de connaissances, qui décide de la date et de la stratégie d'un nouvel ordonnancement.

Le modèle à base de connaissances permet de traiter la perturbation en utilisant une base de données. Il permet à différents modèles de communiquer avec l'avantage d'avoir des interfaces non coûteuses.

Les projet ESPRIT 809 et 477 sont plus particulièrement dédiés au niveau le plus bas de la fonction de pilotage d'atelier où l'ordre de fabrication est directement exécutable.

II-6.1.2 Le pilotage à ordonnancement prévisionnel total

Un ordonnancement prévisionnel total est un sous-ensemble d'opérations à réaliser parmi l'ensemble des opérations qui sont planifiées (Figure II-13b). Ce type de pilotage a été implanté sur des sites industriels depuis quelques années. Ce pilotage a été divisé en deux niveaux : le niveau 1 est responsable de la réalisation des fonctions suivantes : planification, programmation, et ordonnancement (ordonnancement prévisionnel total) ; le niveau 2 a pour but d'éviter la remise en cause de l'ordonnancement établi par le niveau 1 et de réaliser au mieux les tâches planifiées. Le fait de gérer le fonctionnement dégradé est l'un des principaux

avantages de cet ordonnancement, les structures de pilotage dans ce cas sont de type centralisé, car comme nous l'avons dit précédemment, la structure centralisée fournit un contrôle efficace capable de répondre aux contraintes imposées par l'ordonnancement.

II-6.1.2.1 Système SYROCO

Le système SYROCO [Roy 98] propose un modèle de pilotage réactif d'atelier basé sur une plate-forme Multi-agents. Ce système est basé sur une architecture hybride, placée entre les architectures hiérarchiques et centralisées, avec l'utilisation d'un ordonnancement prévisionnel total mais pouvant être remis en cause. Ainsi, une telle conception permet au système d'être rapide et modulaire. Celui-ci exige un flux d'information assez important. Ce système est capable de gérer une production manufacturière perturbable et, *a priori* inconnue. En ce qui concerne l'ordonnancement, il se révèle fiable et parfaitement stable. Il semble que SYROCO soit bien capable de piloter une production en temps réel, grâce à son positionnement entre les résultats de la planification et l'atelier réel existant. En effet, la distribution des constituants sur plusieurs ordinateurs est automatique, ce qui permet de ne pas surcharger un ordinateur. Par contre, le point faible du système est sa connexion réseau entre tous les ordinateurs. Si le réseau tombe en panne, le système s'effondre. De plus, même si le système réagit rapidement aux aléas, le fait de faire des corrections en permanence provoque beaucoup de problèmes dans un environnement industriel non stable. La Figure II-20 fait apparaître les communications et la hiérarchie entre agents, les relations et les communications entre agents.

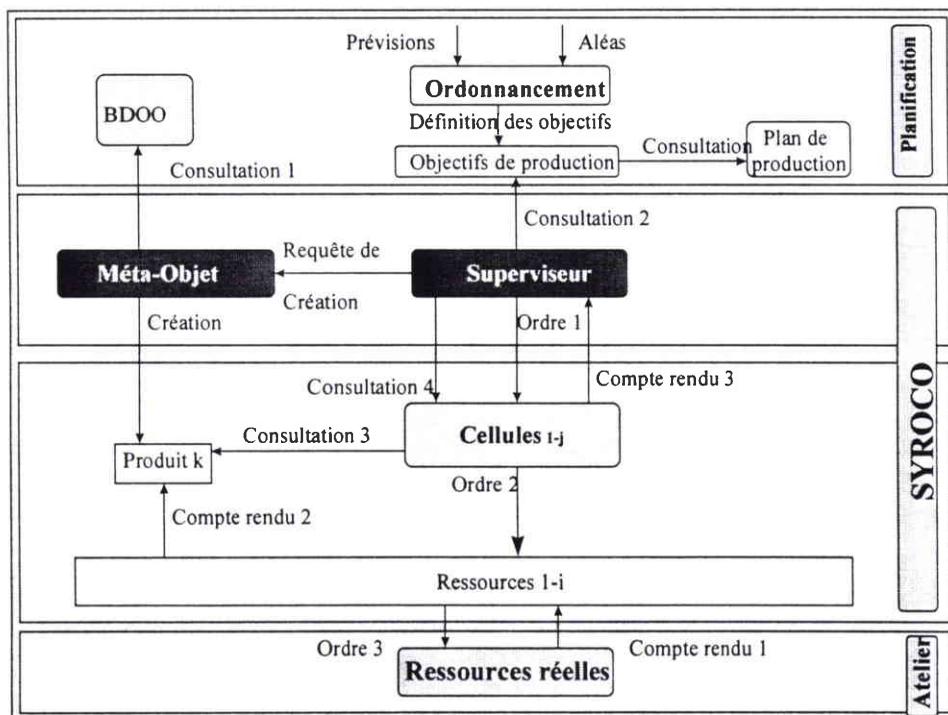


Figure II-20 : Schéma de principe de l'architecture SYROCO

II-6.2 Pilotage sans ordonnancement prévisionnel

Dans ce type de pilotage, le niveau prévisionnel n'élabore aucun ordonnancement. La gestion d'allocation des tâches et des files d'attente se fait dynamiquement, au fur et à mesure de l'évolution du système de production.

Le but ici est de réaliser le programme prévisionnel défini par la fonction programmation. L'allocation des tâches est réalisée de manière dynamique. L'affectation est évaluée par rapport à la qualité de l'ordonnancement réalisé [Merbaki 95].

On note que chacun des trois types de pilotage présentés utilise une des différentes structures connues dans le domaine du pilotage, structures dont nous avons résumé les caractéristiques les plus importantes dans les paragraphes précédents.

II-6.2.1 La structure centralisée

Cette structure met en œuvre un certain nombre d'outils pour la gestion de files d'attente et d'allocation dynamique des tâches. Son avantage est qu'elle s'intéresse aux objectifs globaux (niveau de stock, par exemple). Il est nécessaire de simplifier la structure de production et de préparer les calculs avant et pendant le déroulement du processus de production. On cite ici le modèle de Messaoud [Messaoud 86] qui utilise une méthode multicritères pour réaliser les tâches dans le but de minimiser le temps de cycle. Le modèle a été testé sur un atelier flexible robotisé. Son algorithme est efficace, mais pour pouvoir l'exploiter dans le cadre de production jop-shop, il faut le généraliser.

Un autre modèle qui utilise la structure centralisée est le modèle de [Merbaki 95]. Ce modèle possède des règles de priorité pour la gestion dynamique des files d'attente, en fonction des événements, de l'état courant réel de l'atelier et des objectifs globaux. Une décision dans les files d'attente concerne une machine. Pour prendre cette décision, il faut suivre les étapes suivantes : analyse du système, diagnostic et choix de la règle à appliquer. Le problème qui se pose ici, c'est que cette méthode ne concerne que la gestion de files d'attente de chaque machine. [Merbaki 95] a fait une extension par l'utilisation de la structure multi-agents, ses perspectives vont dans le sens des travaux de [Kouiss *et coll.* 95] présentés ci-après.

II-6.2.2 La structure distribuée

Plusieurs approches ont été proposées, dans le cadre de cette structure. Nous allons en citer quelques unes.

II-6.2.2.1 Le modèle de Bakker

Ce modèle concerne le pilotage distribué DFMS (Distributed Flexible Manufacturing System) [Bakker 88]. Le système est décomposé en deux sous-systèmes, d'une part un ensemble de "Station Managers" SM, d'autre part un ensemble de "Fonctional Models" FM. Chaque SM gère un ensemble de ressources pour : la gestion de files d'attente, l'allocation dynamique des tâches et le contrôle de production. Par contre, chaque FM est responsable de la gestion des outils, des programmes de commande numérique, d'interfacage avec l'opérateur, des bases de données techniques et des convoyeurs.

Les avantages que l'on peut déduire de tels modèles sont : la fiabilité, l'extensibilité et la simplicité. Les inconvénients sont : l'absence d'optimisation globale, et d'optimisation de l'ordonnancement.

II-6.2.2.2 Le modèle de Duffie

Duffie [88] a développé une structure distribuée qui permet l'intégration de l'opérateur humain et qui est capable de s'adapter face à un ensemble d'événements difficilement maîtrisables par le système. Ce modèle utilise la simulation en tant qu'outil de prévision. Il est composé d'entités, elles mêmes composées d'un sous-système de communication et de contrôle pour réaliser l'ensemble des tâches. Les tâches s'effectuent dynamiquement selon deux principes : le principe de l'appel d'offre et le principe selon lequel les tâches sont allouées à la première entité libre. L'un des nombreux avantages de cette méthode est la possibilité d'améliorer le protocole de communication.

II-6.2.2.3 Le modèle de Tchako

Ce modèle présente une structure distribuée s'appuyant sur les concepts de l'intelligence artificielle distribuée [Tchako 94]. Le modèle se compose d'un ensemble d'agents coopératifs appelés SIP (Station Intégrée de Pilotage) comportant cinq sous systèmes (décision, contrôle, communication, interface et information).

Chaque SIP assure le contrôle d'un ensemble de ressources, comme les ressources de production ou les outillages. Ce modèle est applicable à différents types de systèmes comme les job-shop, les lignes de fabrication, etc. Une structure de pilotage distribuée sans ordonnancement prévisionnel favorise la résistance aux perturbations et possède une autonomie de fonctionnement.

Après avoir donné une brève explication des trois modèles de pilotage, on constate qu'une structure distribuée est difficile à réaliser sur tous les sites industriels. L'approche distribuée supervisée tente de résoudre cet inconvénient.

II-6.2.3 La structure distribuée supervisée

Cette structure permet d'éviter les comportements chaotiques qui peuvent survenir dans la structure distribuée, et ainsi, permet d'anticiper les besoins de matériels, de matières premières et de ressources humaines. Le superviseur peut avoir une vision globale sur le processus de production. Nous détaillerons dans ce qui suit deux modèles d'implantation.

II-6.2.3.1 Le modèle de Kouiss

Le modèle de Kouiss [Kouiss *et coll.* 95] présente une structure distribuée en utilisant la notion d'agent. Chaque agent possède un sous-système de connaissance, d'expertise et de communication. Ce modèle utilise les règles de décision pour réaliser les séquences de files d'attente. L'agent applique une règle pour sélectionner une tâche dans sa file d'attente, sa décision est locale et en temps réel. Le rôle d'un agent est de faire respecter les objectifs globaux, et d'imposer si nécessaire une règle particulière à un ensemble d'agents dans le but de respecter ses objectifs.

II-6.2.3.2 Le modèle de Tacquard

[Tacquard 95] présente un modèle de pilotage d'un anneau flexible s'appuyant sur une méthode où seuls les besoins de production sont exprimés en terme de ratio. Cette structure est basée sur un modèle d'agent. Chaque agent possède un sous système de planification, de réaction et de synthèse. L'agent superviseur contrôle le processus global de production et

rétablit la planification sur plusieurs périodes en cas d'aléas. Les entités supervisées gèrent localement les règles de décision [Tacquard *et coll.* 95].

L'inconvénient principal de cette structure est qu'elle est dédiée à la gestion d'anneaux flexibles et n'est pas applicable dans le cas d'un atelier flexible.

Par ailleurs, cette structure établit un compromis important entre la structure distribuée et la structure hiérarchisée pour la structure de pilotage temps réel sans ordonnancement prévisionnel.

II-6.3 La structure hybride

[Trentesaux 96] a développé une structure de pilotage distribuée supervisée intégrant un ordonnancement prévisionnel partiel "pilotage hybride," qui est fondée sur le concept d'intelligence artificielle distribuée, et qui s'appuie sur le modèle distribué de Tchako. Le pilotage hybride proposé par Trentesaux s'articule sur les aspects suivants : une approche multicritère pour la prise de décision au niveau de pilotage, l'influence d'une telle approche au niveau de la réactivité qui permet d'améliorer l'intégration des objectifs (contraintes prévisionnelles) et un compromis efficace en terme de flexibilité et de réactivité temps réel pour assurer une production temps réel.

II-7 Synthèse des structures et des approches proposées

L'analyse des différentes approches et structures de la fonction de pilotage d'atelier présentées précédemment montre que :

- L'approche ORABAID améliore de façon significative la réactivité. Elle introduit, avec la notion de groupes d'opérations permutables, la hiérarchisation dans le traitement des perturbations. Cette hiérarchisation est établie sur deux niveaux : le niveau le plus bas qui propose une réaction locale à la machine et au groupe d'opérations en cours d'exécution, la réaction au niveau haut qui consiste à remettre en cause tout le plan.
- L'approche CODECO montre toute l'importance de la communication directe entre machines pour le pilotage d'atelier. Nous constatons que cette approche propose une amélioration dans le traitement des perturbations avec l'instauration d'une solution intermédiaire de conduite. Cette solution de coopération s'appuie sur les liens informationnels entre machines. Elle se situe entre les deux niveaux pris en compte dans l'approche ORABAID.
- L'approche COSIMA fournit une structuration du centre de pilotage proposée, avec notamment la séparation des fonctions "Ordonnanceur", "Répartiteur" et "Transporteur" qui sous-entend un certain degré de parallélisme dans le fonctionnement interne du centre.
- L'approche PCS propose une solution réactive, distribuée et hiérarchisée du problème de pilotage d'atelier. Cette approche présente une solution intéressante au niveau coopération et coordination des niveaux de pilotage, mais le problème de conduite réelle n'est abordé que par l'aspect réajustement du plan (coopération avec les systèmes de même niveau, régulation du niveau inférieur). Elle présente aussi des solutions intermédiaires de réaction.
- Les structures distribuées présentent un niveau élevé de réactivité, de modularité et de flexibilité, mais il est très difficile d'anticiper les productions effectives, ce qui rend difficile leur mise en œuvre.

- Les structures distribuées supervisées fournissent un compromis intéressant entre les deux structures précédentes dans le sens où la centralisation du contrôle permet une gestion globale efficace et la distribution des capacités de décision permet une meilleure réaction aux perturbations.

II-8 Conclusion

Après avoir défini l'entreprise, son organisation et présenté quelques architectures qui traitent de la modélisation de l'entreprise, nous avons passé en revue quelques approches de pilotage d'atelier. Cette étude nous a permis, entre autre, de situer chaque approche par rapport à la problématique du pilotage d'atelier, d'en préciser les contours et d'énumérer les limites de chacune.

Il s'est avéré que dans la majorité des travaux cités soit l'approche est dédiée à un domaine d'application bien précis, soit qu'elle s'intéresse particulièrement à des systèmes de gestion de production et qu'elle traite de ce fait d'atelier traditionnel.

On note que les principaux problèmes des systèmes multi-agents reposent sur la coopération des agents entre eux. En effet, pour éviter d'effectuer deux fois la même opération, les agents doivent coopérer sans se gêner, ce qui implique une notion de priorité afin de gérer au mieux leurs fonctions. Il faut ensuite rester critique par rapport aux solutions proposées, car deux sources de connaissances différentes, pourront avoir des solutions qui divergent, à cause, d'une part, de la diversité des informations et de leur incertitude, et d'autre part, de par leur évolution dans le temps.

Chapitre III :

**Pilotage des systèmes de production :
analyse du problème et proposition d'une
approche**

III.1 Introduction	47
III.2 Système MRP	47
III.3 Analyse de la réactivité	51
III.3.1 Espace virtuel de réactivité.....	53
III.3.2 Niveau de réactivité "AMONT"	53
III.3.3 Niveau de réactivité "AVALE"	55
III.4 Gestion des réactivités	56
III.5 Proposition d'une approche	59
III.6 Conclusion	62

III.1 Introduction

Comme nous l'avons mentionné au chapitre I, la réactivité est au cœur même de l'approche proposée. En conséquence, il nous apparaît utile de revenir dans un premier temps, sur la définition de la notion de réactivité ainsi que sur les différents types de réactivité. Ensuite, nous développerons une analyse globale de la fonction conduite au sein des activités des entreprises manufacturières. En effet, nous avons déjà notés que l'analyse de la fonction conduite doit nécessairement être menée en harmonie avec les objectifs globaux du système de production. Cette analyse nous permet de mieux cerner la fonction pilotage en identifiant à la fois les activités et les entités physiques ou informationnelles qui caractérisent cette fonction.

Pour être réactives, les entreprises doivent s'appuyer sur des données qui reflètent la réalité de leur quotidien et l'état de leur système de production. Les systèmes de GPAO qui sont pour la plupart de structure MRP et, en particulier leur module de suivi de production, ont pour finalité de répondre à ce besoin. Pour cela, les systèmes existants recherchent des propositions de pilotage prenant en compte le court terme vécu dans l'atelier, à partir des flux d'informations du système productif. Cependant, la nécessaire flexibilité des entreprises impose des changements de programmes de fabrication relativement soutenus. En effet, les données de charge sont particulièrement versatiles et les données de capacité ne sont plus des constantes. En conséquence, il est difficile de garder un sens dans les décisions de replanification et de proposer un ensemble d'indicateurs de pilotage, représentant l'évolution du travail, qui leur soient cohérent. L'adéquation charge/capacité au niveau du programme directeur de production impose un suivi des postes contraintes de capacité. Il est donc important d'avoir une vision globale du système pour être en mesure de bien distinguer les postes contraintes [Roder 94]. Dans notre approche, nous proposons de gérer l'affectation des ressources au niveau de l'atelier, avec l'introduction du Plan d'Affectation des Ressources.

La prise en compte globale des problèmes d'affectation des ressources augmente la capacité de réactivité du système de production mais complique quelque peu le pilotage de l'atelier.

La circulation d'informations est particulièrement importante dans le cadre du traitement de la réactivité. Dans un souci d'efficacité, il nous paraît important de traiter la réactivité par niveau et au plus près des couches physiques. Cela permet de minimiser les perturbations. Pour cela, on distingue deux types de réactivité : la réactivité dite "amont" et la réactivité dite "aval".

III.2 Système MRP

La flexibilité et les démarches dites de qualité totale ont introduit des bouleversements dans l'organisation du travail auxquels les systèmes tels que le MRP classique ne peuvent plus répondre [Thomas 97]. Dans le cadre de nos travaux, nous considérons un système de gestion de production (Figure III-1) de type MRP2 (**M**anufacturing **R**esource **P**lanning) [Toomey 96].

De manière générale, lorsque l'on souhaite étudier un système, il est important de représenter ses fonctions, ses relations, ses interfaces et surtout ce qu'il réalise. Pour cela, il existe un certain nombre de méthodes de modélisation permettant de définir les activités d'un système comme GRAI ou CIMOSA. Pour parvenir à cette modélisation, il faut aborder l'analyse du système de façon globale en ce concentrant sur les activités du système [Vernadat 96]. Cette analyse permet de détecter les invariants statiques et dynamiques et de les déduire les uns des

autres. La démarche consiste, à partir de la connaissance d'une donnée et d'un événement, puis à définir les changements d'états et leurs conséquences.

Comme nous nous limitons à la description fonctionnelle des tâches et de leurs interactions pour définir le système de conduite, nous avons retenu la méthode SADT [Lissandre 87].

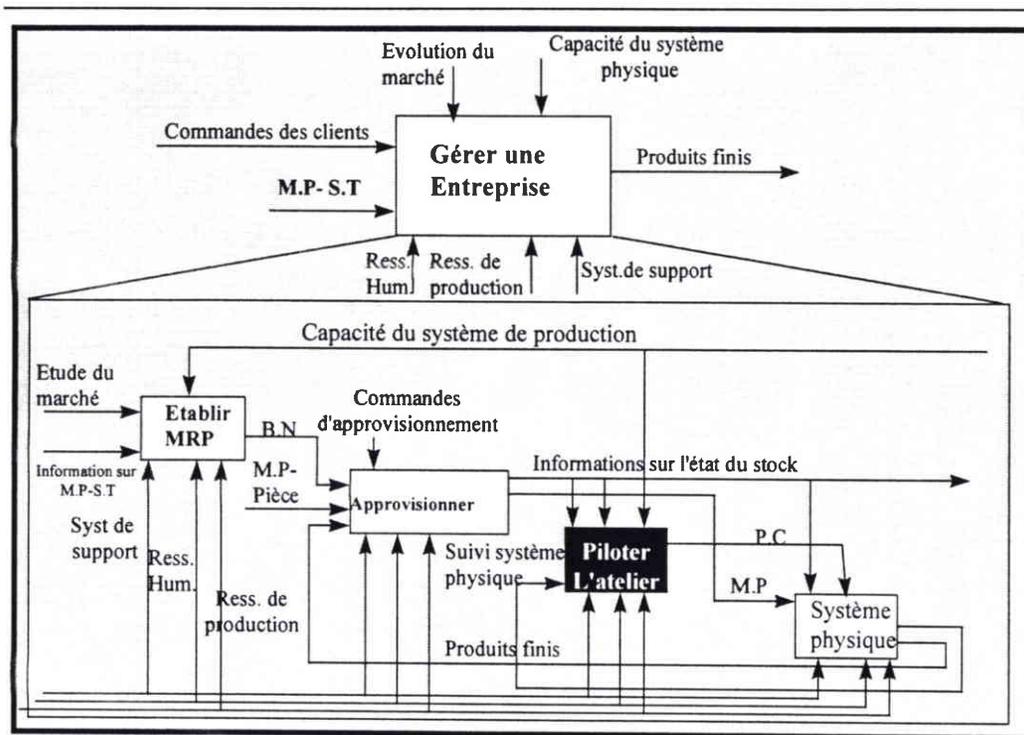


Figure III-1 : Activités de gestion d'une entreprise / Fonction Pilotage

Le concept MRP est né de la mise en évidence de la répartition des besoins en deux types fondamentaux [Orlicky 75] :

- **les besoins indépendants**, qui représentent les produits finis. Ils proviennent de l'extérieur de l'entreprise et sont générés par le marché.
- **les besoins dépendants**, qui représentent les sous-ensembles, les composants et matières premières entrant dans la composition du produit. Ils proviennent de l'intérieur de l'entreprise et sont induits par les besoins indépendants.

Le principe général de MRP2 repose sur une planification de la production construite à partir de la demande (prévisions commerciales, commandes clients) et de l'exécution. Ce principe est exprimé de la manière suivante : "Les besoins indépendants ne peuvent être qu'estimés sur la base des prévisions. Les besoins dépendants peuvent et doivent être calculés".

C'est un outil de communication entre les diverses fonctions de l'entreprise, notamment la fonction commerciale et la fonction de production. Il permet la gestion des besoins sur trois niveaux (Figure III-2). Chaque niveau est validé par un calcul des charges et des capacités.

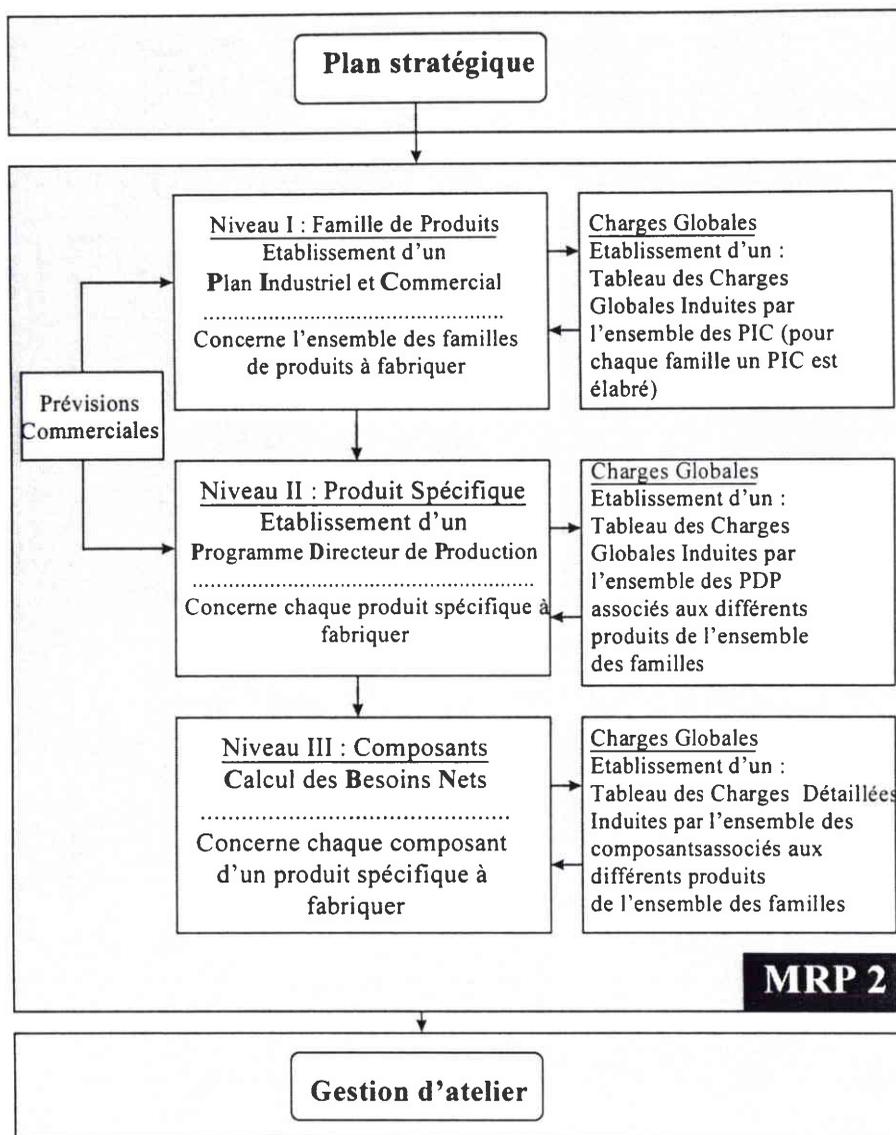


Figure III-2 : Différents niveaux de décision de MRP2

Le Plan Industriel et Commercial ou PIC (Figure III-3) est établi au plus haut niveau par famille de produits (Remarque : la maîtrise du PIC impose un nombre limité de familles compris entre 5 et 20). L'objectif est de permettre un cadrage global de l'activité et donc d'anticiper globalement les problèmes potentiels, notamment une inadéquation entre la capacité de l'entreprise et la charge induite par les besoins commerciaux. Le caractère global se retrouve dans la durée des périodes prises en compte, de 1 mois à plusieurs années. L'horizon dépend du délai total des produits et de l'acquisition des équipements.

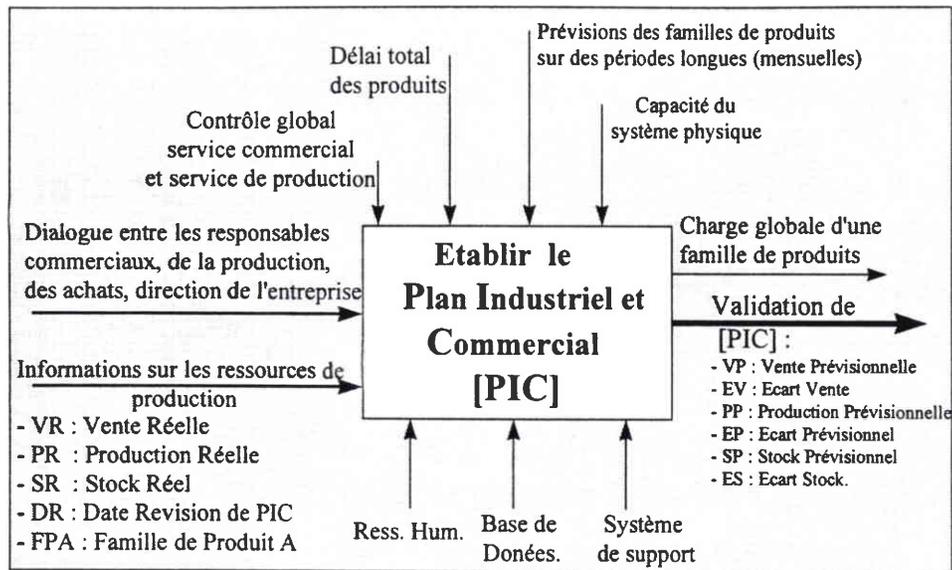


Figure III-3 : Plan Industriel et Commercial

Le Programme Directeur de Production ou PDP (Figure III-4) est établi au niveau intermédiaire entre la fonction commerciale et la fonction production. Il permet de définir précisément l'échéancier des quantités à produire pour chaque produit fini et aide le gestionnaire à anticiper les variations commerciales. Il permet de concrétiser le PIC et sa période est généralement de l'ordre du mois.

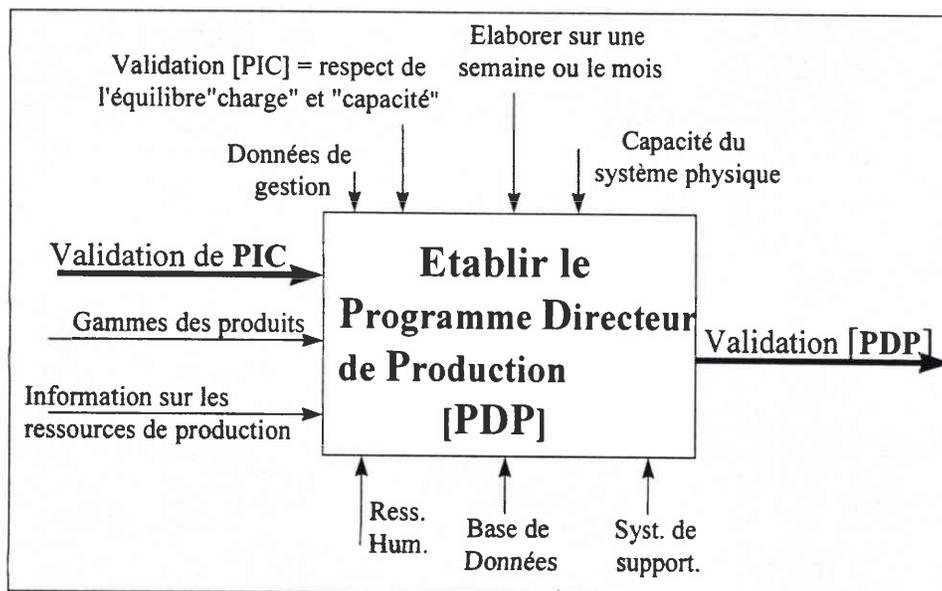


Figure III-4 : Plan Directeur de Production

Le Calcul des Besoins Nets (Figure III-5) fournit les dates d'approvisionnement et de lancement de fabrication de tous les composants. Il vérifie en outre la cohérence des dates de livraison et des dates de besoins, notamment si les besoins changent ou se décalent dans le temps.

Les résultats du calcul des besoins nets sont :

- les ordres proposés, c'est à dire des lancements prévisionnels en fabrication ou des approvisionnements prévisionnels,
- les actions particulières à mener, proposées au gestionnaire dans le but d'une bonne gestion de la production prévue (lancer, avancer, reporter un ordre de fabrication).

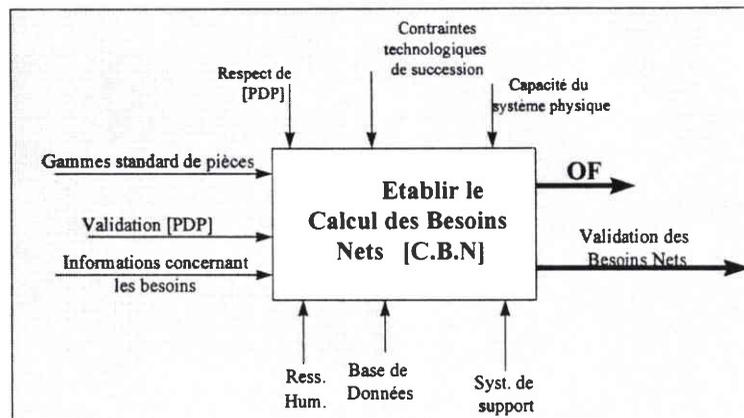


Figure III-5 : Calcul des Besoins Nets

III.3 Analyse de la réactivité

Le domaine d'application "Pilotage réactif d'atelier" nécessite une précision importante qui concerne les termes "réactif" et "aléa". Nous définissons un aléa comme un événement susceptible de perturber la production. Dans le cadre de nos travaux, nous distinguons des types d'aléa :

- Les aléas dits de "gestion", on peut citer :
 - un retard de traitement d'une commande,
 - une erreur dans la rédaction d'une commande,
 - un manque de personnel.
- Les aléas dits de "Commande", on peut citer :
 - une modification de nomenclature,
 - une modification de date,
 - une modification de quantité.
- Les aléas dits "physiques", on peut citer :
 - une panne de machine,
 - un dérèglement inopiné du système,
 - une casse d'outillage,
 - un opérateur absent.

Les aléas dits de "Commande" regroupent les modifications de commandes déjà programmées et l'intégration d'une commande urgente. Le traitement d'un aléa de gestion se traduit soit par la création d'une nouvelle commande, soit par la modification d'une commande déjà enregistrée.

Dans le cadre de nos travaux, nous ne prenons pas en compte que les aléas de commande. En effet, nous considérons que le traitement des aléas de gestion n'est pas des ressort de pilotage d'atelier.

Nous définissons la réactivité par rapport au savoir-faire représenté par les "potentialités de l'entreprise" et non par rapport à la vitesse de réaction. En effet, nous mettons l'accent sur la maîtrise du savoir-faire de l'entreprise plutôt que sur les dixièmes de secondes que ferait gagner l'utilisation d'un tel réseau dans le cadre d'un transfert de données. La qualité de la décision doit primer sur la rapidité de transfert d'informations.

L'ensemble de ces termes est regroupé dans la figure III-6 :

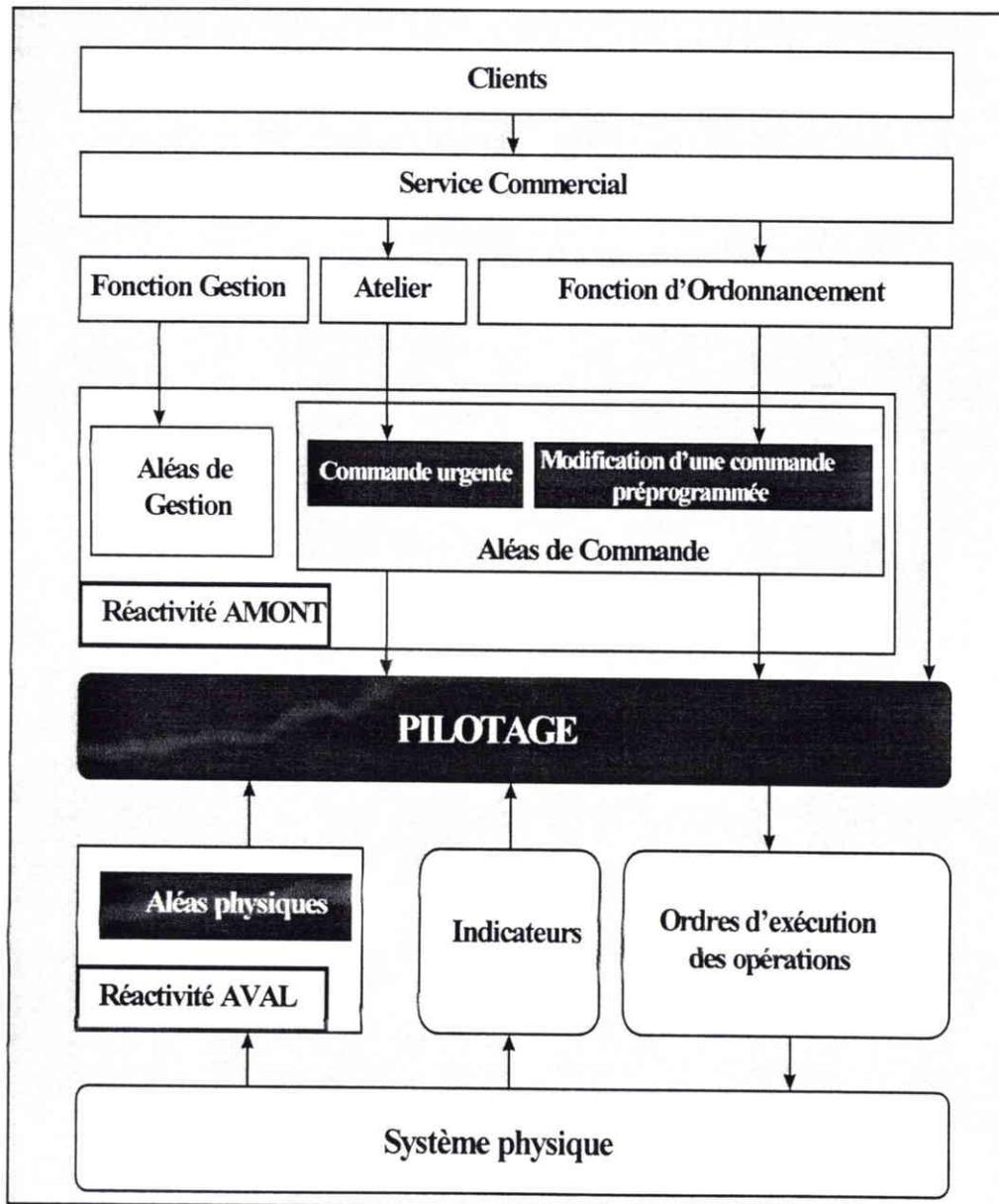


Figure III-6 : Réactivité "Amont" et Réactivité "Aval"

III.3.1 Espace virtuel de réactivité

La réactivité est développée dans un "**Espace virtuel de réactivité**" représenté par l'ensemble des savoir-faire concernant les objets manipulés (produits, composants).

Dans le cas où le système n'est pas à même de régler le problème au niveau du poste, la recherche de solution doit s'effectuer au niveau de l'atelier à partir du plan d'Affectation des Ressources $PAR(i)$. Si le problème persiste et que la volonté de traiter l'aléa s'affirme, le système doit être capable de proposer de nouvelles gammes de fabrication $GF_i(CA_m)$, gammes d'assemblage $GA_j(NM(P_x))$ et éventuellement proposer une autre nomenclature $NM(P_x)$. L'espace de recherche de solutions peut être représenté et obtenu à partir d'ensemble d'itérations imbriquées (Figure III-7).

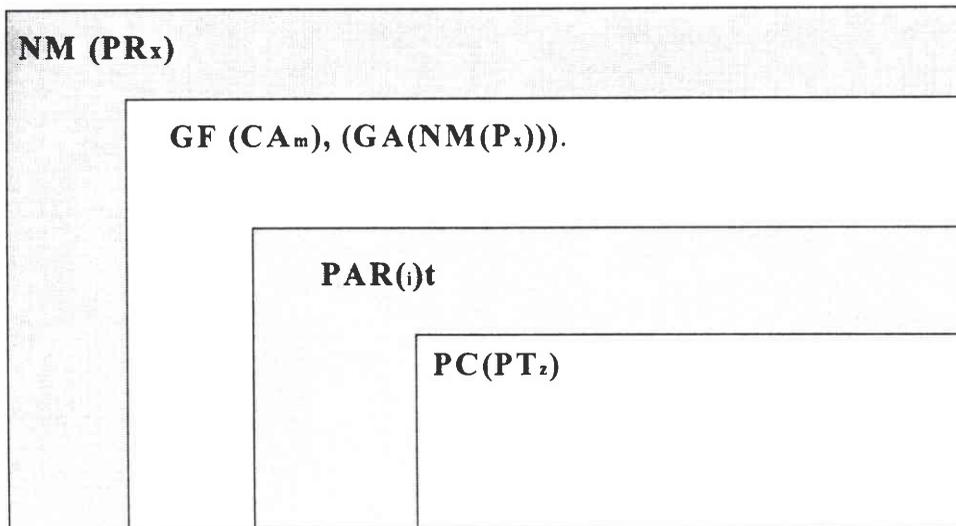


Figure III-7 : Espace virtuel des réactivités.

III.3.2 Niveau de réactivité "AMONT"

La réactivité dite "amont" a pour objectif de réagir aux aléas de commande. Parmi ceux-ci, nous pourrions citer : autre coloris, augmentation de la puissance, modification de la date de livraison, modification du nombre de produits, etc. Dès l'apparition de l'aléa, il nous paraît important d'identifier rapidement le produit impliqué et de définir le niveau auquel il va être nécessaire de réagir.

En conséquence, nous avons introduit des niveaux de réactivité différents. Ce qui permet de gérer les modifications afférentes aux différents types d'aléas (Figure III-8).

Nous définissons quatre niveaux d'intervention :

1. Le niveau I le plus bas est tout simplement celui qui représente la non intervention. C'est-à-dire que l'impact des aléas de commande n'a aucune incidence sur la production. Parmi les aléas susceptibles d'induire ce type de modification, on peut citer le changement de couleur.
2. Le niveau II de réactivité correspond aux modifications qui n'affectent que la fabrication de composant. Parmi les aléas susceptibles d'induire ce type de modification, on peut citer la variation du nombre de produits. Dans ce cas, la réactivité peut concerner la

modification de gammes de fabrication. La réactivité à ce type d'aléas intervient au niveau de l'échéancier issu du calcul des besoins nets. En conséquence, la première action à entreprendre concerne la remise en cause du dernier échéancier. Celle-ci doit être effectuée en fonction des nouvelles données (aléas), mais aussi en fonction de l'état d'avancement de la production.

3. Le niveau III de réactivité correspond aux modifications qui affectent les nomenclatures des produits. Parmi les aléas susceptibles d'induire ce type de modification, on peut citer un changement de caractéristique, telle que la puissance d'un moteur. Dans ce cas, la réactivité concerne la modification de la nomenclature. En conséquence, la première action à entreprendre concerne la remise en cause du dernier programme de production. La remise à jour du PDP doit être effectuée en fonction des nouvelles données (aléas), mais aussi en fonction de l'état d'avancement de la production.
4. Le niveau IV le plus haut correspond à l'impossibilité d'intégration des aléas de commande. La décision doit être prise dans le cadre politique de l'entreprise. Une des conséquences est la remise à jour du plan industriel et commercial.

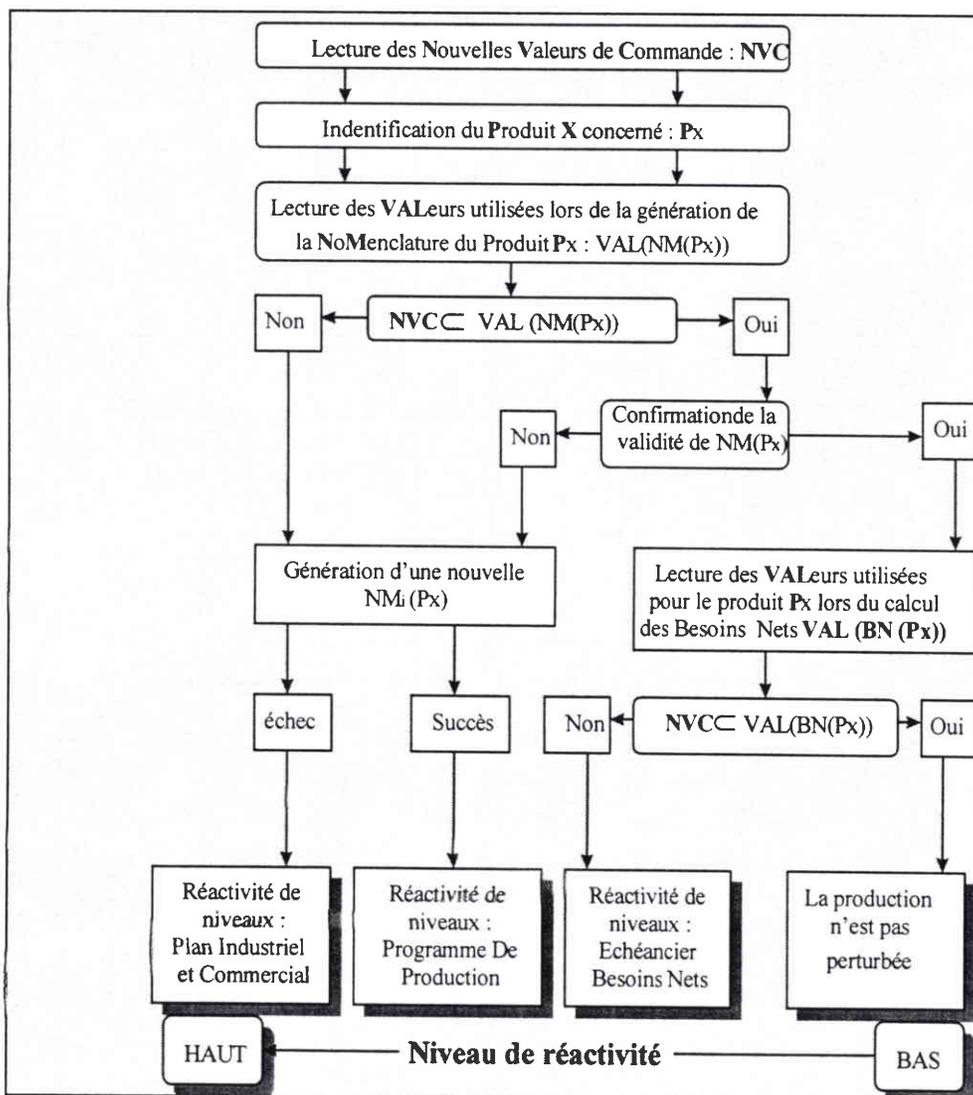


Figure III-8 : Définition des niveaux de réactivité dans le cadre d'aléas de commande

Dans le cadre de nos travaux, nous considérons que la fonction pilotage doit prendre en charge la réactivité de niveau "Besoins Nets" mais aussi la réactivité de niveau "Programme De Production" (Figure III-8).

Attention, il n'est pas noté que la génération du programme de production et le calcul des besoins nets soient du ressort de la fonction pilotage, mais elle doit contribuer à leur élaboration. Cette participation est importante, car la fonction pilotage peut renseigner de l'état de la production à l'instant t . L'objectif est de perturber la production au minimum, donc de trouver l'adéquation entre l'état de la production à l'instant t et le traitement de l'aléa de commande.

III.3.3 Niveau de réactivité "AVAL"

L'objectif de cette réactivité est de réagir aux aléas intervenant en cours de production. Parmi ceux-ci, nous pouvons citer : machine en marche dégradée, transfert de pièces en panne, outil détérioré, opérateur accidenté, etc. Dès l'apparition d'un l'aléa, il nous paraît important d'identifier rapidement le produit impliqué et de définir le niveau auquel il va être nécessaire de réagir.

Nous définissons deux niveaux d'intervention :

- le niveau poste de travail, qui concerne les problèmes internes à celui-ci. Parmi les aléas susceptibles d'induire ce type de modification, nous pouvons citer : panne de la machine, bris d'outil, etc. Dans ce cas, la réactivité concerne le plan de charge. Dès l'instant où le problème induit par l'aléa de production ne peut plus être solutionné au niveau poste de travail, celui-ci doit être pris en charge au niveau atelier.

- le niveau atelier, qui concerne les problèmes à traiter globalement en considérant l'ensemble des postes de travail. Parmi les aléas susceptibles d'induire ce type d'intervention, on peut citer : la réaffectation des ressources, problème de transfert, etc.

On considère que les problèmes d'affectation des ressources doivent être traités conjointement avec les problèmes de transfert. En effet, si une ressource se trouve être disponible pour réaliser une opération et si le transfert des pièces ne peut pas être effectué, cette ressource ne peut pas être sélectionnée.

Dans le cadre de nos travaux, nous considérons que la fonction pilotage doit prendre en charge la réactivité de niveau "poste" ainsi que la réactivité de niveau "Atelier" (Figure III-9).

Quel que soit le type de réactivité à traiter, le niveau d'intervention est une conséquence de la comparaison des valeurs des variables utilisées lors de la dernière action de pilotage et des valeurs générées par l'aléa.

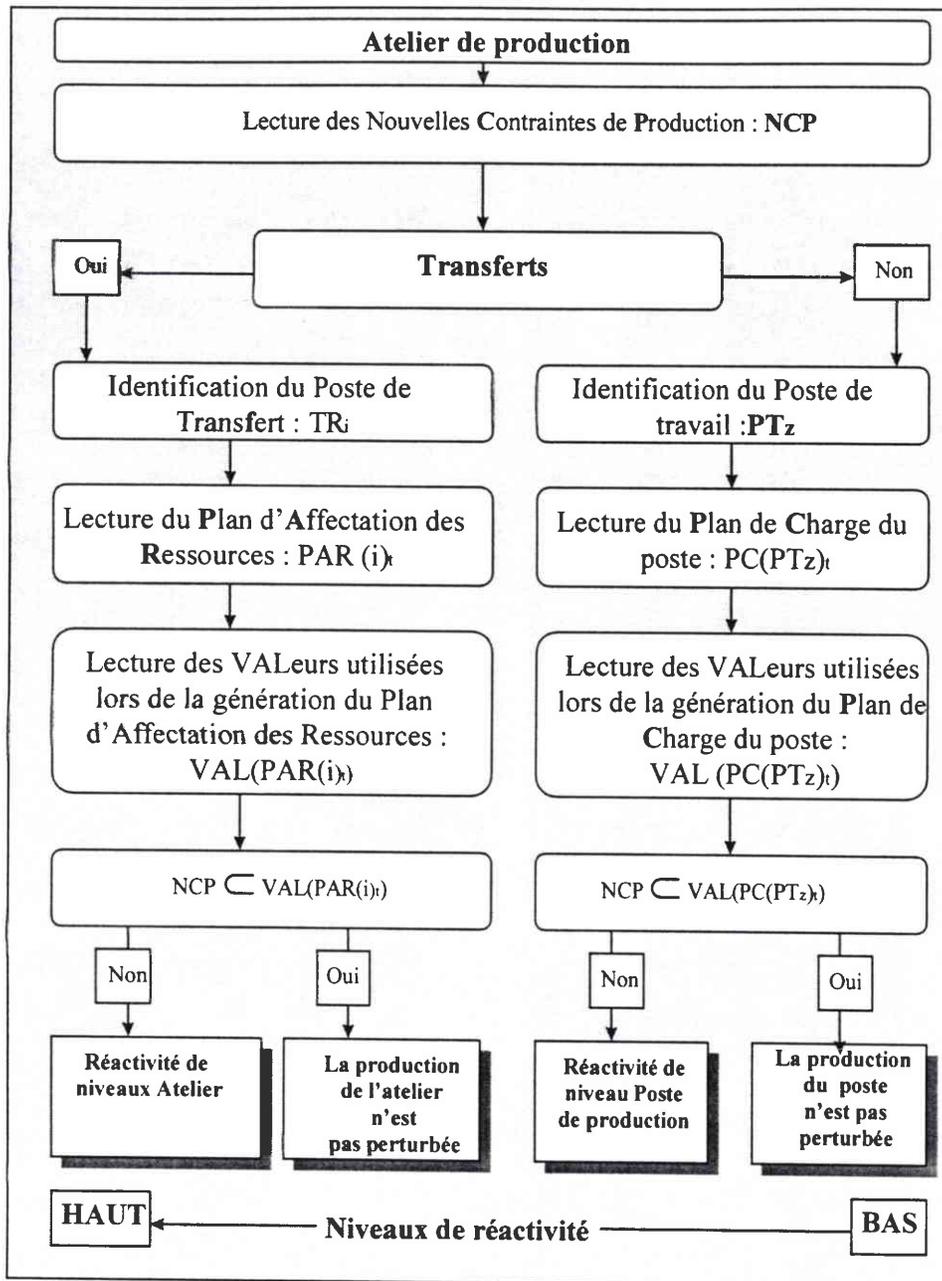


Figure III-9 : Définition des niveaux de réactivité dans le cadre d'aléas de production

III.4 Gestion des réactivités

Après avoir introduit les différents niveaux de réactivité, il est nécessaire de définir la manière dont elle est gérée. L'objectif est de solutionner le problème avec comme priorité de respecter la politique de gestion de réactivité.

En effet, cette politique peut être :

- de minimiser les perturbations occasionnées à la production,
- de minimiser les coûts de production, quitte à perturber les plans établis,
- de charger davantage un poste de travail plutôt qu'un autre, etc.

Dans le cadre de nos travaux nous proposons de prendre en compte quatre sortes de réactivités :

- réactivité à une commande "Urgente" (Figure III-10),
- réactivité à un poste "Hors Service" (Figure III-11),
- réactivité à une ressource d'équipement "Hors Service" (Figure III-12),
- réactivité à un système de transfert "Hors Service" (Figure III-13).

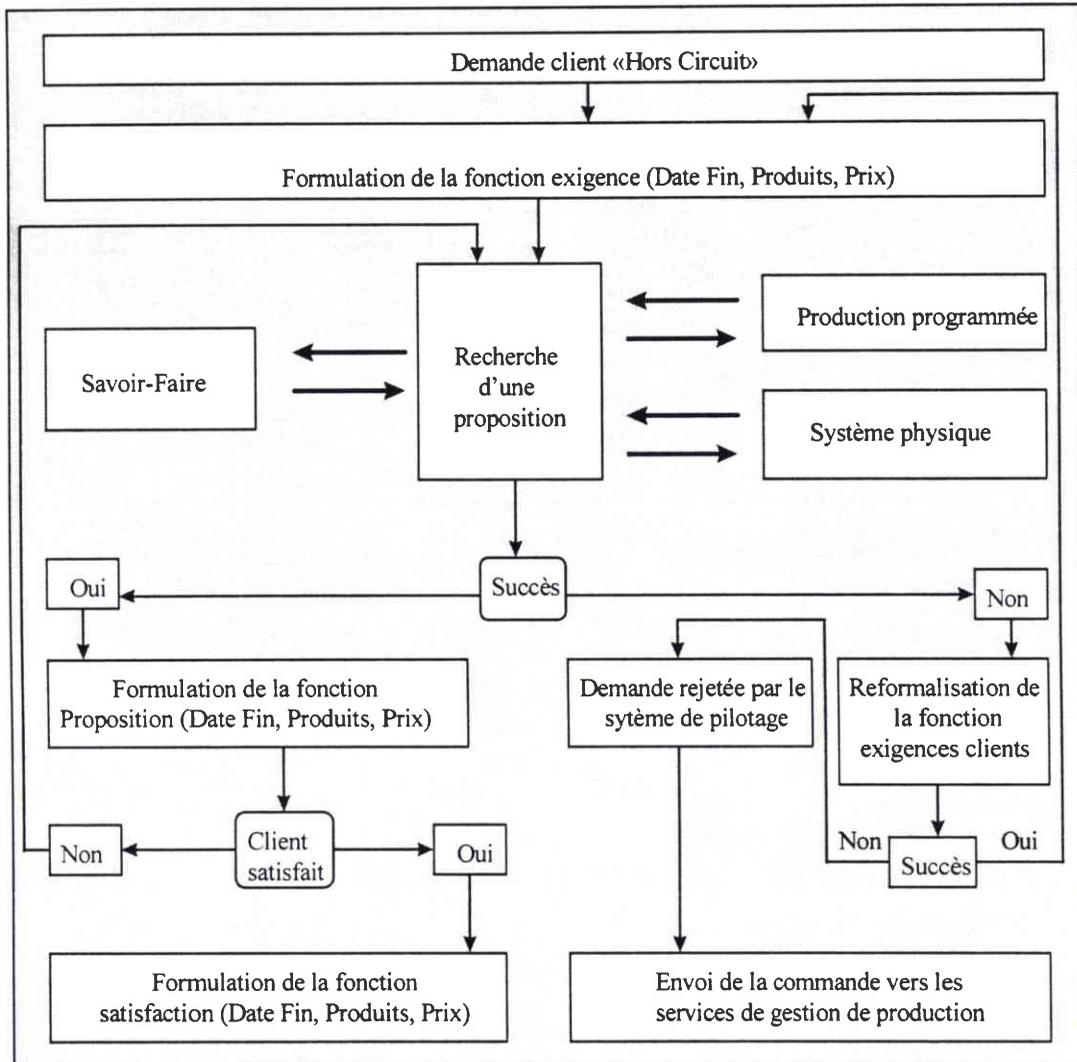


Figure III-10 : Réactivité par rapport à une demande "Urgente"

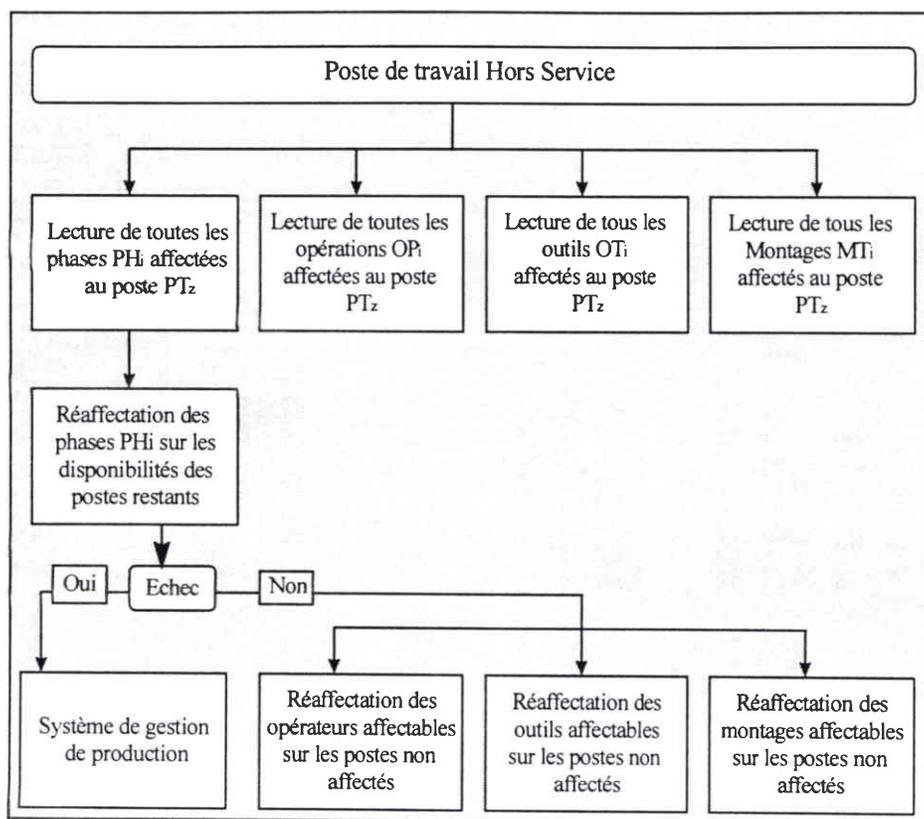


Figure III-11 : Gestion de la réactivité à un poste de travail H.C.

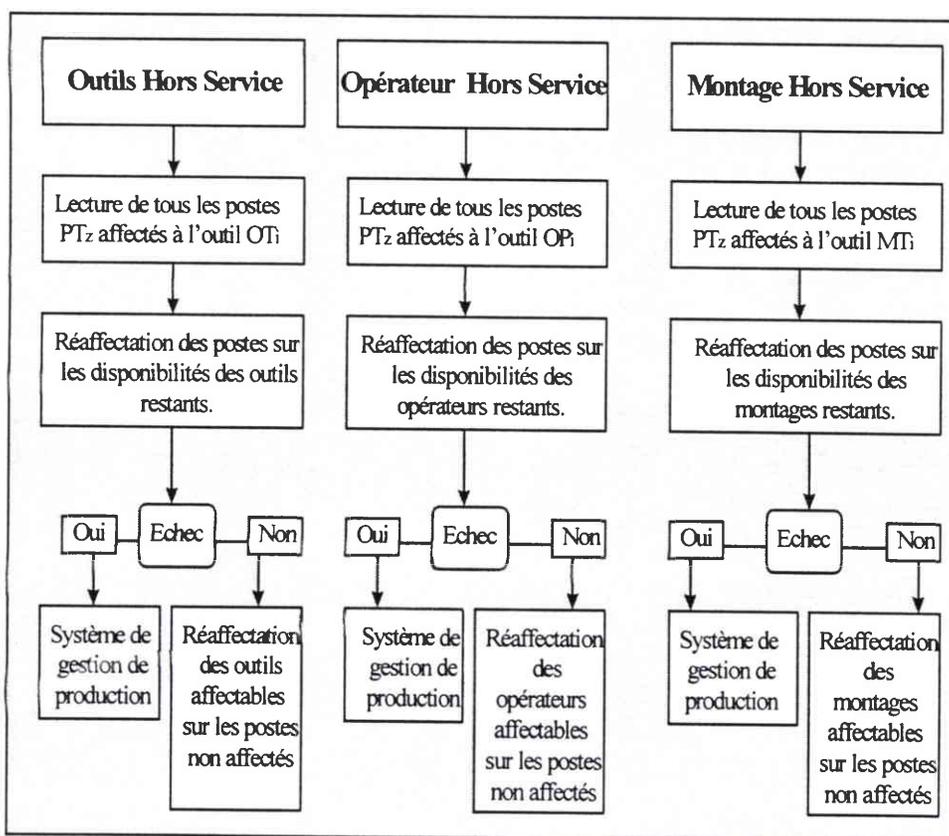


Figure III-12 : Gestion de la réactivité à une ressource H.C.

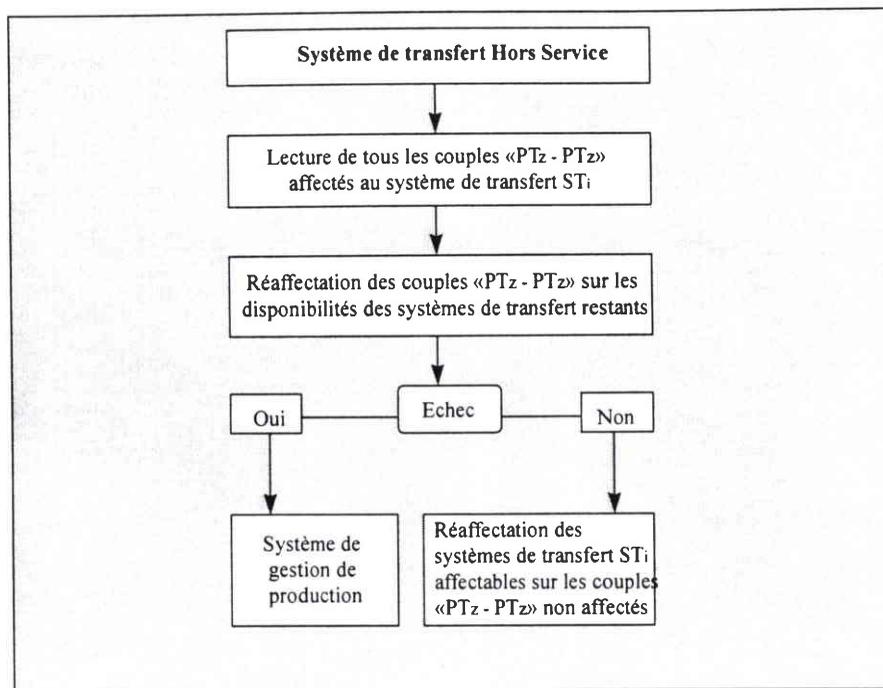


Figure III-13 : Gestion de la réactivité à un système de transfert H.C.

III.5 Proposition d'une approche

L'approche que nous proposons (Figure III-14) repose sur deux points importants :

- le plan objectif,
- le plan d'affectation des ressources.

Le plan objectif regroupe l'ensemble des opérations de production nécessaires à la réalisation du programme de production. Il est établi à partir des gammes de fabrication des composants cités dans l'échéancier des besoins nets et des composants issus des gammes d'assemblage des produits cités dans le programme de production. On note que la génération des gammes de fabrication, des gammes d'assemblage et des nomenclatures reste à la charge du système de pilotage quelle que soit l'origine de la demande.

Cette filiation est justifiée par les raisons suivantes :

- lors de la génération de gamme, on utilise des données techniques et des informations de gestion qui influent sur la gestion du système physique de production ou qui sont issues du système physique de production.
- de part l'objectif de réactivité visé, il est important que l'ensemble de ces données techniques et informations de gestion soient gérées par le système de pilotage.

Le plan d'affectation des ressources représente la répartition des postes et des ressources d'équipements sur les différentes opérations existantes dans le plan objectif. Dès lors que l'ensemble des propositions est réceptionné, le système de pilotage opère l'affectation définitive des opérations et donc génère le plan de charge définitif pour chaque poste de travail. Cette affectation prend en compte l'adéquation entre les disponibilités proposées par les ressources.

L'affectation des opérations étant terminée, le système de pilotage envoie chaque plan de charge aux différents postes. Dès lors, il reste à exécuter les différentes opérations qui s'y rapportent et à organiser un suivi d'exécution.

L'activité de suivi est réalisée à partir des informations représentées dans chacun des Plans de Charge.

Ces informations concernent essentiellement :

- la charge d'un poste de travail nécessaire à l'exécution d'une phase. Ces informations concernent la définition des opérations de la gamme de fabrication à exécuter, la définition des outillages ainsi que les opérations particulières spécifiques au poste de travail pour aider à réaliser une opération de la gamme.
- les actions d'intervention nécessaires au dépannage du poste de travail. Ces informations sont regroupées dans une Gamme d'Intervention.
- les actions de maintenance du poste de travail. Ces informations sont regroupées dans une Gamme de maintenance.

Dès l'instant qu'une phase affectée à un poste de charge est terminée sur un poste de travail, l'information remonte au niveau du Plan d'Affectation des Ressources et celui-ci est mis à jour.

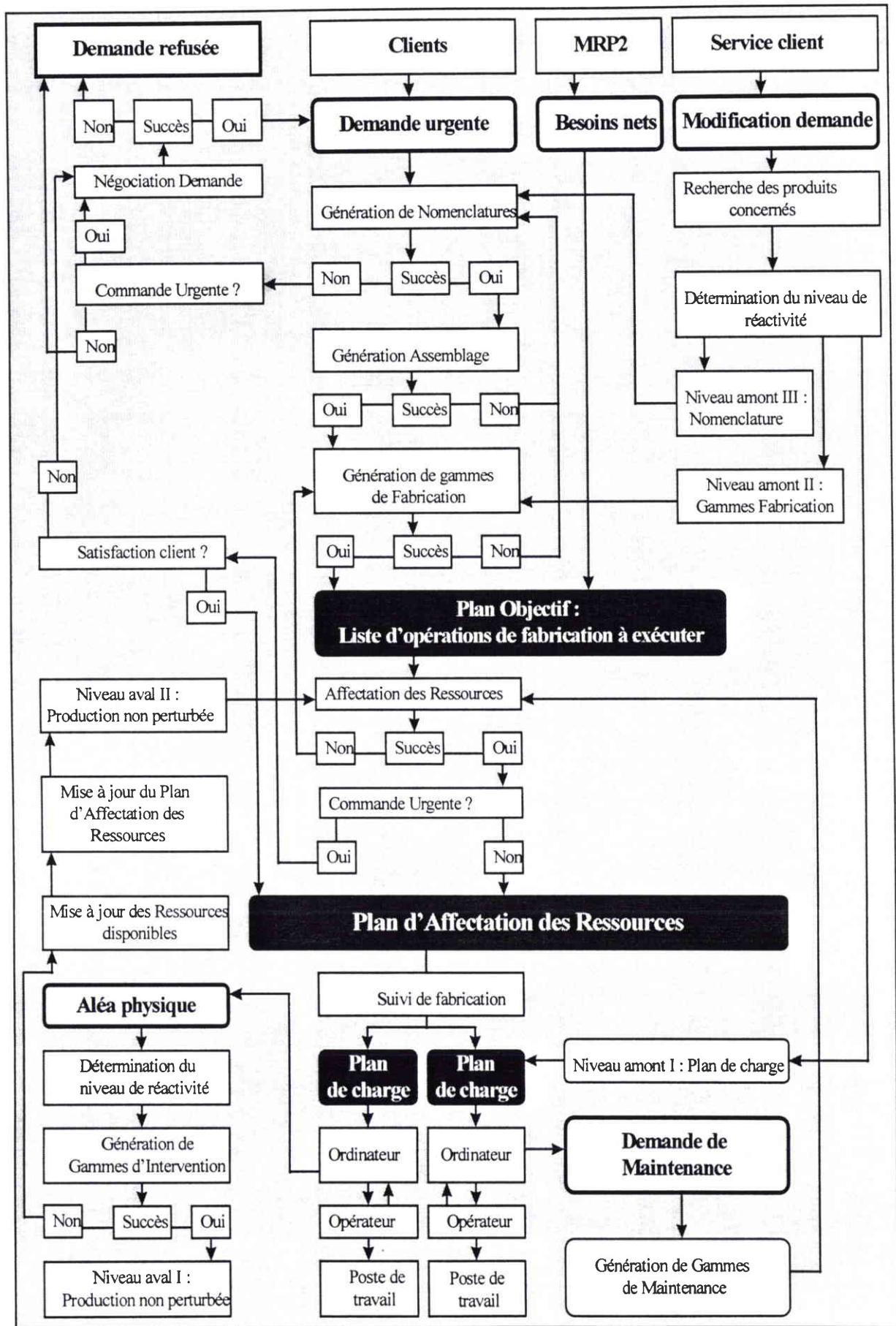


Figure III-14 : Schéma descriptif de l'approche proposée

III.6 Conclusion

Nous considérons que l'ordonnancement et, plus précisément, la planification des tâches issues de la voie "normale" ne doit pas être remis en cause. L'établissement des différentes planifications a généré des ordres de fabrication, des commandes de matières, des autorisations de congés pour le personnel, etc. Il paraît donc difficile, voire risqué, de remettre en cause l'organisation pré-programmée d'un atelier. Dans le cas où l'importance de la commande l'exige, ce n'est plus un problème qui doit être traité au niveau du pilotage.

L'application de la double réactivité doit permettre de répondre de manière optimale aux aléas de commande et de production. Elle permet de faire appel au savoir-faire de l'entreprise et d'adapter au mieux les moyens existants et disponibles du système physique à la demande.

La mise en oeuvre de la réactivité telle que nous l'avons introduite nécessite l'accès à l'ensemble des connaissances de type "Savoir-Faire" de l'atelier. On distingue deux familles de connaissances "Savoir-Faire" :

- **le savoir-faire lié aux produits** qui regroupe l'ensemble des nomenclatures, des gammes d'assemblage et des gammes de fabrication.
- **le savoir-faire lié aux postes de travail** qui regroupe l'ensemble des gammes d'intervention et des gammes de maintenance de chaque type de postes de travail.

Nous développons les détails de la représentation et de la modification des savoir-faire dans le prochain chapitre.

Chapitre IV :

Modélisation et exploitation des Savoir-Faire

IV-1 Introduction.....	63
IV-2 Présentation du système SIRS	65
IV-3 Modélisation des connaissances : "GRAFCET-M"	65
IV-3.1 Éléments de construction.....	66
IV-3.2 Règles d'évolution.....	67
IV-3.3 Définition formelle du GRAFCET-M	68
IV-3.2 Validation d'un réseau autonome.....	73
IV-3.5 Implantation de la base de connaissances.....	76
IV- 4 Exploitation du système.....	84
IV-4.1 Exemple 1	87
IV-4.1.1 Aspect Capitalisation du Savoir-Faire	87
IV- 5 Exemples d'Applications.....	89
IV-5.1 Application du modèle à la représentation de gammes de fabrication	89
IV-5.2 Application du modèle à la représentation de gammes d'assemblage	91
IV-5.3 Application du modèle à la représentation de gammes d'intervention	92
IV-5.4 Application du modèle à la représentation de nomenclatures	93
IV-5.5 Application	95
IV-6 Conclusion	98

IV-1 Introduction

Un des problèmes majeurs des entreprises est la perte de leur savoir-faire. En effet, de part le contexte économique qui pousse les entreprises à se séparer des collaborateurs proches de la retraite et à sous traiter des services, elles finissent par se déposséder de leur savoir-faire. Cette situation a amené de plus en plus d'entreprises à vouloir se doter de systèmes capables de mémoriser et d'exploiter leurs expériences.

Dans notre approche nous considérons le savoir-faire en terme d'actions dans l'exercice d'une activité par opposition au savoir qui peut être transmis indépendamment de la réalité d'une activité. En conséquence, le savoir-faire peut être associé au processus de production d'un objet. Il représente les capacités d'une personne à réaliser une activité et sa compétence dans cette activité. Le savoir-faire est donc mis en oeuvre lors de la réalisation de l'activité.

Dans le cadre de nos travaux, nous nous intéressons plus particulièrement au savoir-faire utile au pilotage d'atelier. Ces connaissances sont liées à la fabrication des produits et à l'utilisation des ressources. Elles concernent plus spécifiquement les produits, la génération de nomenclatures, de gammes d'assemblage, de gammes de fabrication et pour les ressources la génération de gammes de maintenance et de gammes d'intervention. Toutes ces connaissances ont la particularité de pouvoir être représentées comme une liste séquentielle d'actions à exécuter. En effet, les activités au sein d'une organisation peuvent être décrites en termes de choix et de successions d'étapes et d'actions. Elles doivent prendre en compte l'ordonnancement et la mise en oeuvre des étapes (parallélismes, séquences), les entrées sorties, les actions à mener et les contrôles à opérer au niveau de chaque étape.

Attention, de part l'aspect formalisation de la compétence d'un expert afin de la partager, nous ne nous situons pas dans le domaine des systèmes experts et de l'intelligence artificielle. En effet, nous ne cherchons pas à automatiser des raisonnements de l'expert. Nous cherchons à fournir des modèles formant des cadres méthodologiques adaptés à l'expression des savoir-faire.

Parmi les approches utilisées pour aborder les savoir-faire, on peut citer [Maret 97] :

- l'approche documentaire basée sur l'illustration,
- l'approche système à base de connaissances basée sur la simulation,
- l'approche système d'information basée sur les notions d'échange et de mise en valeur des expériences individuelles.

De part la particularité des connaissances à manipuler (Etape/Transition), les modèles de réseaux paraissent bien adaptés. En effet, la démarche consiste à considérer qu'un noeud représente une étape et qu'une transition représente le passage d'une étape E_i à une étape E_{i+1} .

Parmi les modèles existants on peut cités les représentations :

La représentation par RdP [David *et coll.* 87], dont le principal avantage est l'existant d'outils mathématiques permettant de les manipuler. les réseaux sont surtout utilisés dans le domaine de la gestion tel que l'ordonnancement.

La représentation par Grafcet, dont le principal avantage est son succès dans le monde industriel (API)

Mais ni l'un ni l'autre n'offre la possibilité d'une exploitation, du savoir-faire dans un langage quasi naturel.

Dans le cadre de nos travaux, nous proposons un système à base de connaissances "SIRS" (Système Interactif de Recherche de Séquences) [Spadoni 87]. Ce système utilise un modèle de représentation dérivé des réseaux de Pétri, mais avec des règles d'évolutions simplifiées. En effet, il est important que le modèle puisse être implanté à partir d'un langage quasi-naturel. Il n'existe pas et il nous paraît très compliqué d'implanter le modèle RdP de manière simple et conviviale. Par contre, le système SIRS permet l'implantation du modèle à partir d'un langage de programmation simple sur des règles de type Si.....Alors..... . Pour cette raison, il a été introduit un modèle noté "GRAF CET-M" basé sur une structure simple de règles d'évolution simplifiées (pas de boucles, pas de synchronisation).

Parmi les possibilités offertes, on peut citer :

- représentation graphique des connaissances,
- intégration de la notion de famille de gammes,
- base de connaissances écrite à partir d'un langage interne évolutif très simple d'utilisation,
- la capitalisation du savoir-faire à partir d'un interface très simple d'utilisation,
- manipulation de types de données élémentaires ainsi que de matrices, très utiles en cas d'utilisation de code morphodimensionnel,
- manipulation de séquences composées de sous-séquences,
- interpréteur permettant la recherche de toutes les solutions existantes.

En raison de l'importance de la représentation du savoir-faire, nous rappelons dans ce chapitre les points essentiels de SIRS. Aux travaux déjà effectués, nous avons ajouté le développement de la formalisation ainsi qu'un algorithme contribuant à la validation de la base de connaissances.

Nous rappelons que le système SIRS portait à l'origine le nom de SAGGA (Système Automatique de Génération de Gammes par Apprentissage). Celui-ci étant trop restrictif, il a été décidé de ne plus parler de gamme mais de séquence et de le rebaptiser Système Interactif de Recherche de Séquences.

IV-2 Présentation du système SIRS

Le système SIRS (Figure IV-1) permet le développement d'une recherche interactive de séquence d'opérations, concernant un objet, à partir d'une famille de séquences. Pour cela, il sélectionne la famille d'appartenance de l'objet et charge la base de connaissances correspondante dans sa mémoire de travail. Ensuite, il développe la recherche de séquences en s'appuyant sur des données stockées dans des fichiers et sur un jeu de questions réponses. Chaque proposition peut être modifiée à partir d'une interface et ainsi contribuer à la capitalisation du savoir-faire de la base de connaissances.

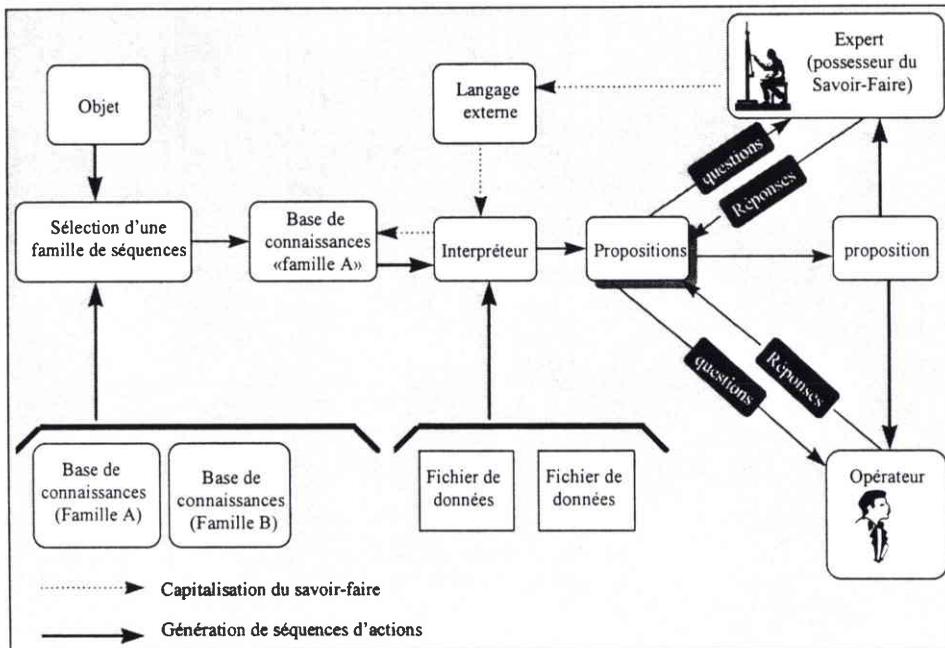


Figure IV-1 : Système SIRS

IV-3 Modélisation des connaissances : "GRAF CET-M"

La représentation des connaissances repose sur un modèle de type réseau "GRAF CET-M" (GRAPhe Fonctionnel de Commande Étape-Transition-Modifié) (Figure IV-5). Ce modèle s'appuie sur la notion d'étapes et de transitions (Figure IV-2) comme le fait la représentation "GRAF CET", la différence se situe au niveau des règles d'évolution et des éléments de construction.

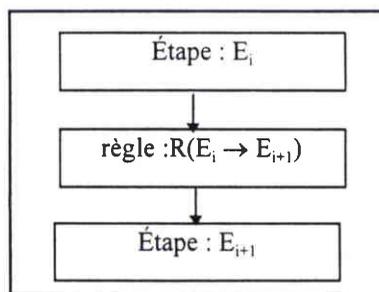


Figure IV-2 : Définition des noeuds dans un réseau "GRAF CET-M"

IV-3.1 Éléments de construction

La présentation des différents éléments de construction du GRAFCET-M est introduite dans la figure IV-3. On y recense cinq types de transition :

- séquentielle,
- divergence de "OU" exclusif,
- convergence de "OU",
- divergence de "ET",
- convergence de "ET".

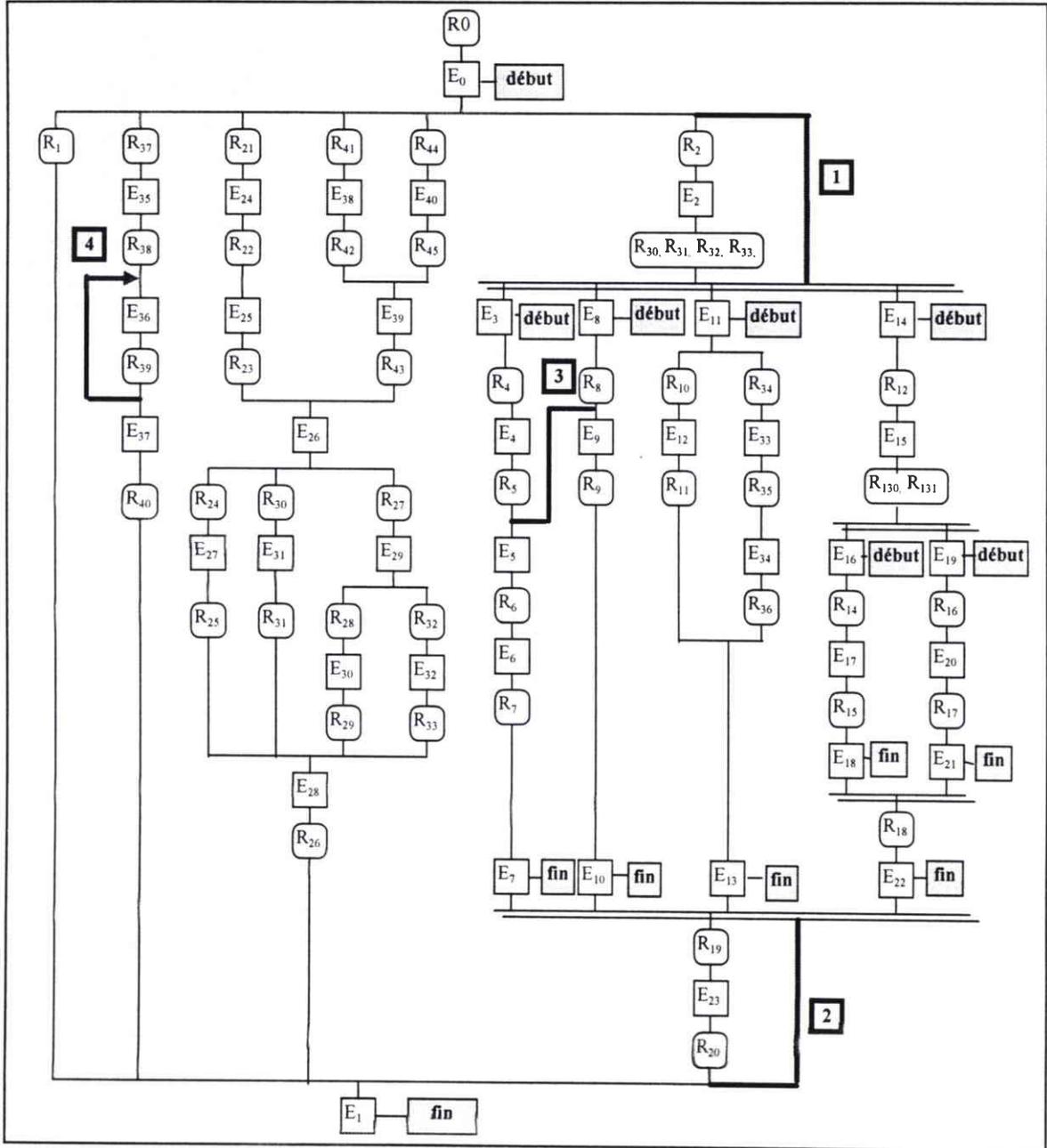


Figure IV-3 : Exemple d'un GRAFCET-M

Il existe un certain nombre de configurations interdit. Celles-ci sont repérées en gras sur la figure IV-3 :

- les configurations 1 et 2 indiquent qu'une transition de type OU et une transition de type ET ne peuvent pas se suivre et inversement,

- la configuration 3 indique qu'aucun lien ne doit relier des séquences d'un bloc ET,
- la configuration 4 indique que l'utilisation de boucle est interdite.

IV-3.2 Règles d'évolution

L'évolution à travers le réseau Grafcet-M est assujettie à un ensemble de règles d'évolution tel que :

- **Règle 1** : L'étape associée à l'action [DEBUT] est automatiquement activée lors de l'initialisation.
- **Règle 2** : Une transition est soit validée, soit non validée. Elle est validée dès lors que toutes les étapes immédiatement précédentes soient actives. Elle ne peut être franchie que lorsqu'elle est validée et que la réceptivité associée à la transition soit vraie.
- **Règle 3** : Le franchissement d'une transition entraîne l'activation de toutes les étapes immédiatement suivantes mais ne désactive pas les étapes déjà activées. En effet, l'objectif est de générer une liste d'actions et non d'exécuter des actions.
- **Règle 4** : Lors du franchissement d'une transition de divergence "ET" toutes les étapes immédiatement suivantes, correspondant au début de chaque séquence, sont activées. Mais l'évolution à l'intérieur du bloc "ET" est effectuée séquence par séquence.
- **Règle 5** : L'activation de l'étape associée à l'action [FIN] entraîne automatiquement l'affichage du résultat, c'est à dire la liste ordonnée des actions étapes franchies. Le résultat peut-être représenté par une séquence d'actions et de sous séquences

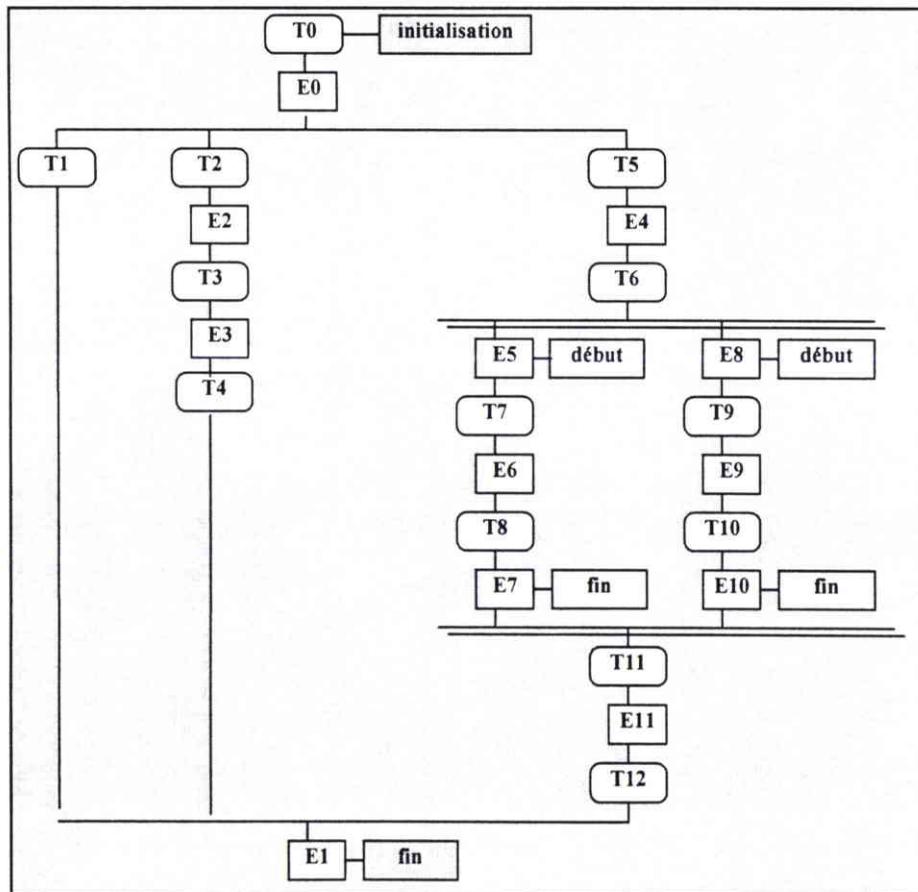


Figure IV-4 : Exemple de GRAFCET-M

IV-3.3 Définition formelle du GRAFCET-M

Un GRAFCET-M est un quadruplé noté : $R = (E, T, \text{Pré}, \text{Post})$

Avec :

- $E = \{E_0, E_1, \dots, E_m\}$, ensemble fini d'étapes avec $m = \text{card}(E)$
- $T = \{T_0, T_1, \dots, T_n\}$, ensemble fini de transitions avec $n = \text{card}(T)$

Les ensembles E et T sont disjoints et on note : $E \cap T = \emptyset$.

- **Pré** est la fonction d'incidence avant. Elle est relative à une transition T_j du couple (E_i, T_j) . En effet, elle informe sur le type d'arc placé en amont de la transition T_j et définit ainsi l'arc permettant la progression entre une étape E_i et une transition T_j .

On note : $\text{Pré} : E * T \rightarrow \{A, E, O, \emptyset\}$

Avec :

- A = progression de type "séquentiel",
- E = progression de type "ET",
- O = progression de type "OU exclusif",
- \emptyset = il n'existe pas de progression.

La représentation matricielle de la fonction d'incidence amont Pré du réseau de la Figure IV-4 est donnée par la Figure IV-5. Les absences de liens, notées \emptyset , n'ont pas été représentées. Pré (E_i, T_j) représente la possibilité et le type de progression de $E_i \rightarrow T_j$. Ainsi Pré $(E_4, T_6) = \{A\}$ indique qu'il existe une possibilité de progression de type séquentiel entre l'étape E_4 et la transition T_6 .

	T ₀	T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈	T ₉	T ₁₀	T ₁₁	T ₁₂
E ₀		O	O			O							
E ₁													
E ₂				A									
E ₃					A								
E ₄							A						
E ₅								A					
E ₆									A				
E ₇												E	
E ₈										A			
E ₉											A		
E ₁₀												E	
E ₁₁													A

Figure IV-5 : Exemple de représentation matricielle du la fonction Pré

- **Post** est la fonction d'incidence arrière. Elle est relative à une transition T_j du couple (T_j, E_i) . En effet, elle informe sur le type d'arc placé en aval de la transition T_j et définit ainsi l'arc permettant la progression entre une transition T_j et une étape E_i .

On note : $\text{Post} : E * T \rightarrow \{A, E, O, \emptyset\}$

La représentation matricielle de l'application d'incidence arrière Post du réseau de la Figure IV-4 est donnée par la Figure IV-6. Les absences de liens, notées \emptyset , n'ont pas été représentées.

	T ₀	T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈	T ₉	T ₁₀	T ₁₁	T ₁₂
E ₀	A												
E ₁		O			O								O
E ₂			A										
E ₃				A									
E ₄						A							
E ₅							E						
E ₆								A					
E ₇									A				
E ₈							E						
E ₉										A			
E ₁₀											A		
E ₁₁												A	

Figure IV-6 : Représentation matricielle de la fonction d'incidence arrière

Post (E_i, T_j) représente la possibilité de progression de T_j → E_i. Ainsi Post (E₄, T₅) = {A}, indique qu'il existe une possibilité de progression de type séquentiel entre la transition T₆ et l'étape E₄.

Le GRAFCET-M ainsi défini, nous introduisons les différents éléments permettant sa manipulation et en particulier la notation d'une équation d'état du réseau :

- **W** est la matrice d'incidence qui représente les conditions de validation. La représentation matricielle de la matrice d'incidence W de la Figure IV-10 est donnée par la Figure IV-8. On note : $\forall E_i \in E, \forall T_j \in T, W(E_i, T_j) = \text{Post}(E_i, T_j) - \text{Pré}(E_i, T_j)$

	T ₀	T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈	T ₉	T ₁₀	T ₁₁	T ₁₂
E ₀	A ⁺	O ⁻	O ⁻			O ⁻							
E ₁		O ⁺			O ⁺								O ⁺
E ₂			A ⁺	A ⁻									
E ₃				A ⁺	A ⁻								
E ₄						A ⁺	A ⁻						
E ₅							E ⁺	A ⁻					
E ₆								A ⁺	A ⁻				
E ₇									A ⁺			E ⁻	
E ₈							E ⁺			A ⁻			
E ₉										A ⁺	A ⁻		
E ₁₀											A ⁺	E ⁻	
E ₁₁												A ⁺	A ⁻

Figure IV-7 : Représentation matricielle des conditions de validation

Remarques :

Les différents types d'arcs peuvent ainsi être définis avec plus de précision.

A^- = progression de type "séquentiel",

E^- = progression de type "divergence de ET",

O^- = progression de type "divergence de OU exclusif",

A^+ = progression de type "séquentiel",

E^+ = progression de type "convergence de ET",

O^+ = progression de type "convergence de OU exclusif".

- **M** est la matrice de marquage d'un réseau R représenté par le couple (R, M). Le marquage M d'un réseau est une fonction notée :

$$M : E \rightarrow \{1, 0\}$$

Avec :

1 = étape active,

0 = étape non active,

Si $e = \text{card}(E)$, \vec{M} est un vecteur à e composantes,

M^R , domaine de marquage du réseau R,

M_x^R , marquage x du réseau R,

$M_x^R(E_i) = 1$ indique que l'étape E_i du marquage x concernant le réseau R est sélectionnée.

Nous présentons ci dessous un exemple de matrice de marquage du réseau de la Figure IV-8 :

$$M^R = (E_0, E_1, E_2, E_3, E_4, E_5, E_6, E_7, E_8, E_9, E_{10}, E_{11})$$

$$M_x^R = (1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0)$$

$$M_x^R(E_2) = (1)$$

- **S** est une séquence de franchissements représentée par la suite des transitions T_a, T_b, \dots permettant le passage d'un marquage M_x à un marquage M_y , avec passage par les différents marquages intermédiaires.

$$\text{On note : } M_x \xrightarrow{Ta} M_a \xrightarrow{Tb} M_b \xrightarrow{Tr} M_y$$

$$M_x \xrightarrow{S} M_y$$

Séquence Si du réseau **R** permettant le passage d'un marquage M_x à un marquage M_y .

$$S_i^{R, Mx \rightarrow My} = (T_a, T_b, \dots, T_r)$$

Exemple de séquence de franchissements du réseau de la Figure IV-9 :

$$S_2^{R, Mo \rightarrow Mm} = (T_0, T_5, T_6)$$

- \vec{S}_i représente le vecteur caractéristique associé à la séquence de franchissements S_i dont les composantes sont les occurrences de chaque transition T_i dans la séquence S_i .

Si $t = \text{card}(T)$, \vec{S}_i est un vecteur à t composantes.

Exemple de vecteur caractéristique associé à la séquence de franchissements S du réseau de la Figure IV-9 :

$$\vec{S}_2 = (1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0)$$

- **Equation d'état d'un réseau** permettant le passage d'un marquage M_x à un marquage M_y .

Soit :

- une séquence $S_i^{R, M_x \rightarrow M_y}$, séquence de franchissements i du réseau R telle que :

$$M_x \xrightarrow{S_i} M_y$$

- une matrice **binaire** P telle que : $P(i,j) = 1$ pour tout $W(i,j) \neq \emptyset$, sinon $P(i,j) = 0$

$$M_y = M_x + [P \times \vec{S}_i]$$

Exemple :

- Soit le marquage $M_a = (1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0)$ décrit dans la Figure IV-8 :

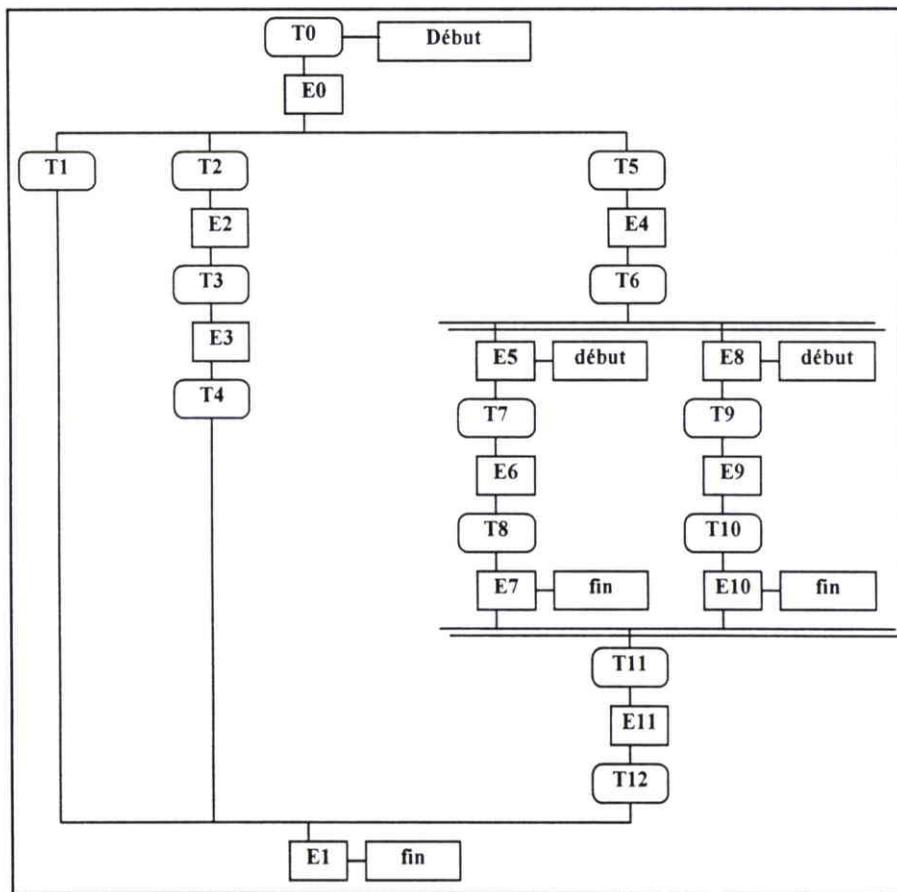


Figure IV-8 : Marquage $M_a = (1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0)$.

- Soit la séquence (Figure IV-9) de franchissements :

$$S_2^{R, Ma \rightarrow Mb} = (T_7, T_8, T_9) :$$

- On calcule :

$$M_a + [P \wedge \vec{S}_2] = M_b$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \vee \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \wedge \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \vee \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

On obtient le réseau marqué suivant (Figure IV-9) :

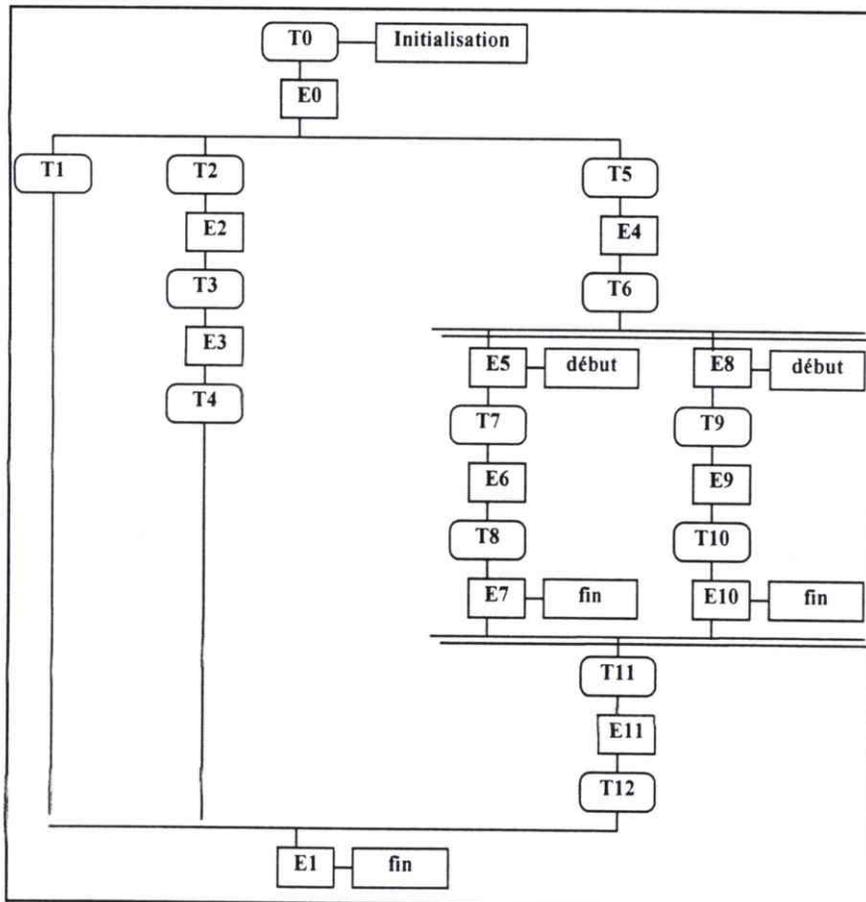


Figure IV-9 : Représentation d'un réseau marqué

Notations :

- T_j , ensemble des étapes d'entrée de T_j tel que : ${}^{\circ}T_j = \{E_i \in E / \text{Pré}(E_i, T_j) \neq \emptyset\}$
- T_j° , ensemble des étapes de sortie de T_j tel que : $T_j^{\circ} = \{E_i \in E / \text{Post}(T_j, E_i) \neq \emptyset\}$
- ${}^{\circ}E_j$, ensemble des transitions d'entrée de E_j tel que : ${}^{\circ}E_j = \{T_j \in T / \text{Post}(T_j, E_j) \neq \emptyset\}$
- E_j° , ensemble des transitions de sortie de E_j tel que : $E_j^{\circ} = \{T_j \in T / \text{Pré}(E_j, T_j) \neq \emptyset\}$

IV-3.2 Validation d'un réseau autonome

Un réseau autonome est un réseau dont le fonctionnement n'est soumis à aucune influence externe. La mise en oeuvre d'un système ne peut être envisagée qu'après avoir vérifié qu'il ne subsiste aucune erreur et aucune ambiguïté dans la spécification des fonctions qui le composent. C'est pourquoi toute implantation d'une base de connaissances doit être précédée par une validation du modèle. L'algorithme est basé sur la reconstruction du réseau à partir de la matrice d'incidence. Il permet d'afficher à l'écran toutes les séquences existantes dans le réseau.

Exemple :

La Figure IV-10 montre un exemple de GRAFCET-M. La matrice d'incidence de ce réseau est donnée par la Figure IV-11.

A partir de la matrice d'incidence de la Figure IV-11, le logiciel affiche à l'écran la liste des séquences générées suivante :

- S 1 = (0, 1)
- S 2 = (0, 2, (3, 4, 5, 6, 7) (8, 9, 10) (11, 12, 13),(14, 15,(16, 17, 18),(19, 20, 21), 22), 23, 1)
- S 3=(0, 2,(3 ,4 ,5 ,6 ,7),(8, 9, 10),(11, 33, 34, 13),(14, 15,(16, 17, 18),(19, 20, 21),22), 23,1)
- S 4 = (0, 24, 25, 26, 27, 28, 1)
- S 5 = (0, 24, 25, 26, 29, 30, 28, 1)
- S 6 = (0, 24, 25, 26, 29, 32, 28, 1)
- S 7 = (0, 24, 25, 26, 31, 28, 1)
- S 8 = (0, 35, 36, 37, 1)
- S 9 = (0, 38, 39, 26, 27, 28, 1)
- S10 = (0, 38, 39, 26, 29, 30, 28, 1)
- S11 = (0, 38, 39, 26, 29, 32, 28, 1)
- S12 = (0, 8, 39, 26, 31, 28, 1)
- S13 = (0, 40, 39, 26, 27, 28, 1)
- S14 = (0, 40, 39, 26, 29, 30, 28, 1)

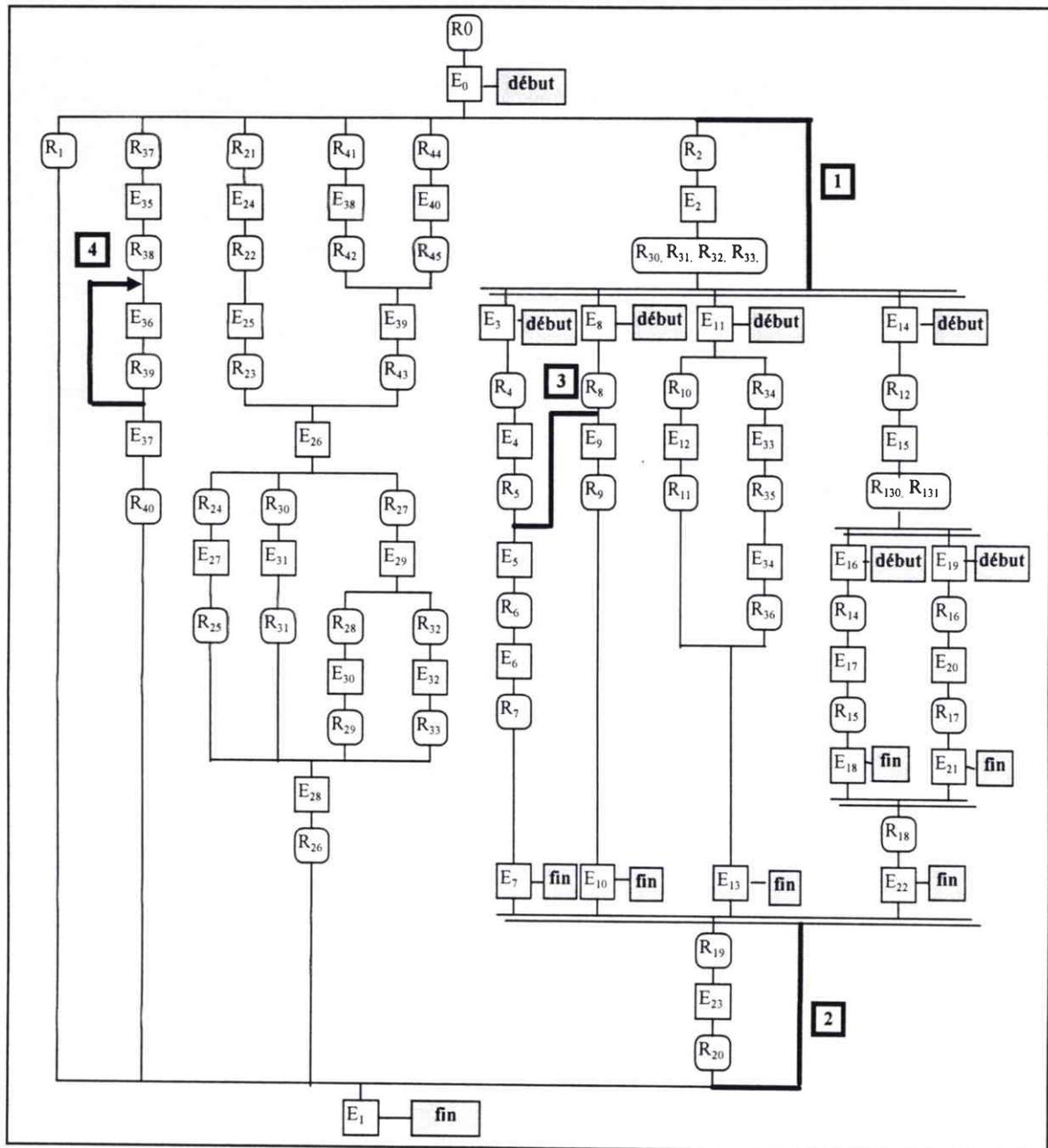


Figure IV-10 : Exemple d'un GRAFCET-M

IV-3.5 Implantation de la base de connaissances

Un réseau est dit interprété dès lors qu'on lui associe des conditions ou actions utilisant des informations externes au système.

Une recherche de séquence dans un réseau interprété est effectuée à partir de BUT.

Soit $\{M_1^R, \dots, M_x^R, \dots, M_X^R\}_{B_i}$ l'ensemble des marquages dont l'étape feuille contient un but B_i .

Soit le marquage courant du réseau R, noté M_{CR}^R , correspondant à l'état de la recherche à l'instant (t).

Si : $M_x^R \subset M_{CR}^R \Rightarrow$ le but B_i est validé.

Si : $E \in \bar{S}_i$ telle que $M_{CR}^R = M_x^R + [W^R \times \bar{S}_i] \Rightarrow$ le but B_i est validé.

L'exploitation d'un réseau interprété requiert une grande souplesse. La structure de base d'un réseau interprété est construite à partir des "actions" et "conditions" que l'on associe respectivement aux étapes et transitions comme le montre la Figure IV-12.

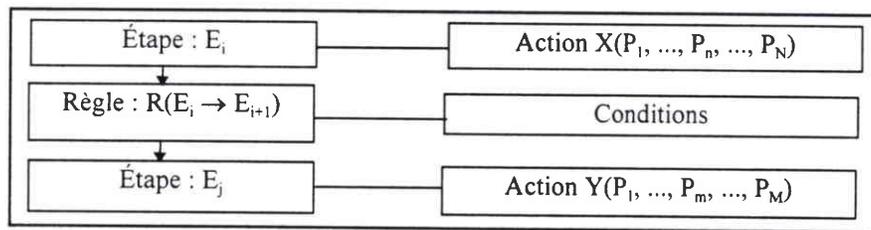
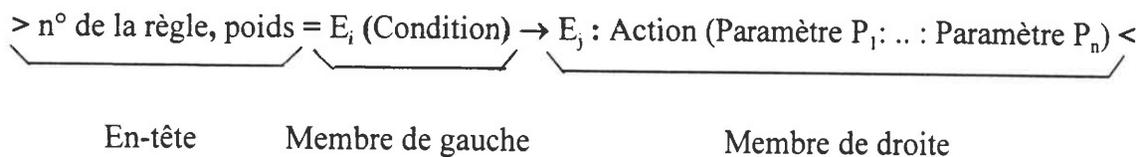


Figure IV-12 : Structure de base d'un réseau interprété

La progression dans un réseau interprété est guidée par les résultats lors de la phase de validation des parties conditions traitées. En conséquence, de part l'exigence d'une grande souplesse et une structure de type CONDITION/ACTION, il est fait appel à une représentation de type règles de production. Un réseau est donc implanté à partir d'un ensemble de règles de production formant une base de connaissances.

Une règle de production est utilisée pour représenter un arc reliant une étape E_i à une étape E_j . L'ensemble des éléments du langage de programmation est récapitulé par la Figure IV-22. La structure générale d'une règle est la suivante :



- **La partie condition** est représentée sous la forme d'une expression logique composée d'opérandes et d'opérateurs. La syntaxe est donnée par la suivante (Figure IV-13) :

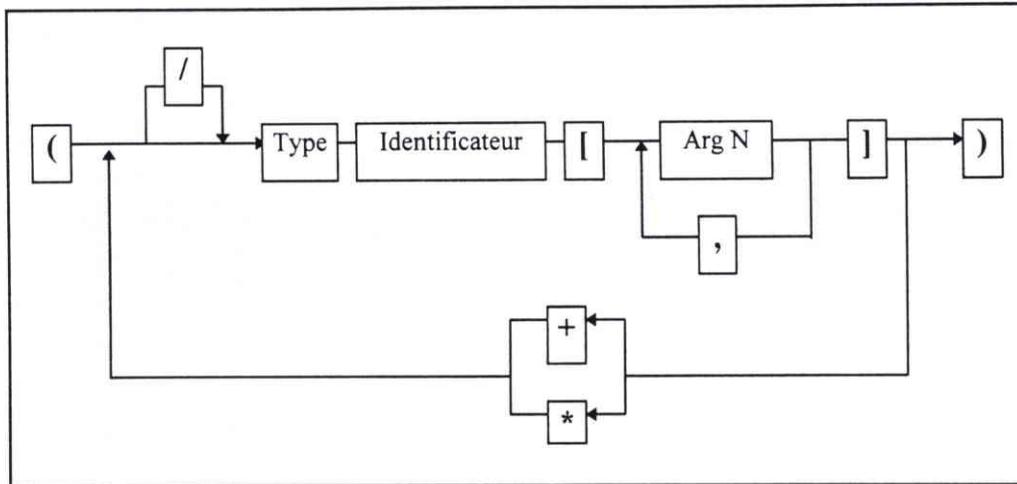


Figure IV-13 : Structure générale de l'expression logique d'une condition

Le langage permet de manipuler :

- trois types d'opérateurs :

- l'opérateur de négation noté : "/",
- l'opérateur OU inclusif noté : "+",
- l'opérateur ET noté : "*".

- cinq types d'opérandes :

- une opérande de type **FO**nction logique notée "FO" qui renvoie une valeur "VRAI" ou "FAUX". La valeur renvoyée est le résultat de l'exécution d'un sous programme. L'ensemble des fonctions logiques est regroupé dans le fichier "FONCTION.BIN".

La syntaxe est donnée par la Figure suivante (Figure IV-14) :

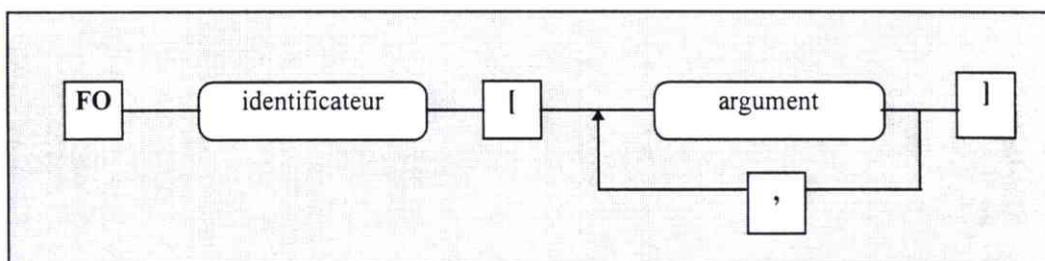


Figure IV-14 : Syntaxe d'écriture d'une opérande de type "FO"

- une opérande de type **PR**édicat notée "**PR**" qui vérifie une relation entre plusieurs arguments et renvoie la valeur "VRAI" si la relation est vérifiée, sinon renvoie la valeur "FAUX". L'ensemble des fonctions logiques est regroupé dans le fichier "PREDICAT.BIN".

La syntaxe est donnée par la Figure suivante (Figure IV-15) :

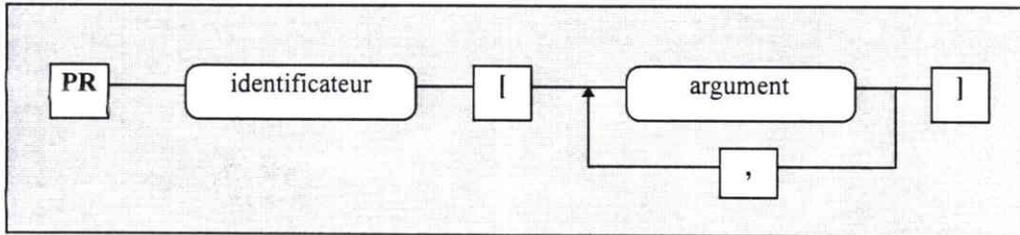


Figure IV-15 : Syntaxe d'écriture d'une opérande de type "PR"

- Une opérande de type Contrainte de séquence sur la partie aMont de la transition courante notée "CM". Cette opérande permet de vérifier si l'action A, notée dans la partie argument, a été sélectionnée en amont de la transition courante étudiée. La vérification de l'existence d'une action A est considérée comme un "but" à valider.

Soit le marquage courant du réseau R, noté M_{CR}^R , correspondant à l'état de la recherche à l'instant t .

Soit la partie argument de l'étape E notée Arg_E

Soit l'étape E telle que $A \subset Arg_E$

On note :

$$CM [A] \text{ est égale à "VRAI" si : } \{ A \subset Arg_E, E \subset M_{CR}^R \}$$

L'action est écrite élément par élément avec la possibilité de remplacer un élément par un filtre noté "?".

La syntaxe est donnée par la Figure suivante (Figure IV-16) :

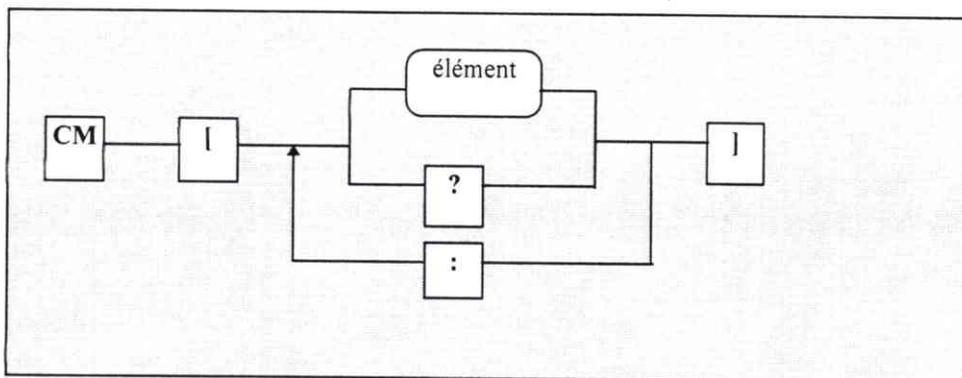


Figure IV-16 : Syntaxe d'écriture d'une opérande de type "CM"

Dans l'exemple de la Figure IV-17, lors de la validation de la condition associée à la transition T9 seul le chemin trait en gras est validé. En effet, la validation de la condition est assujettie à la valeur VRAI de la contrainte de séquence $CM[? : Z : ? : Y : ?]$, qui ne peut être validée que par la sélection de l'étape E6.

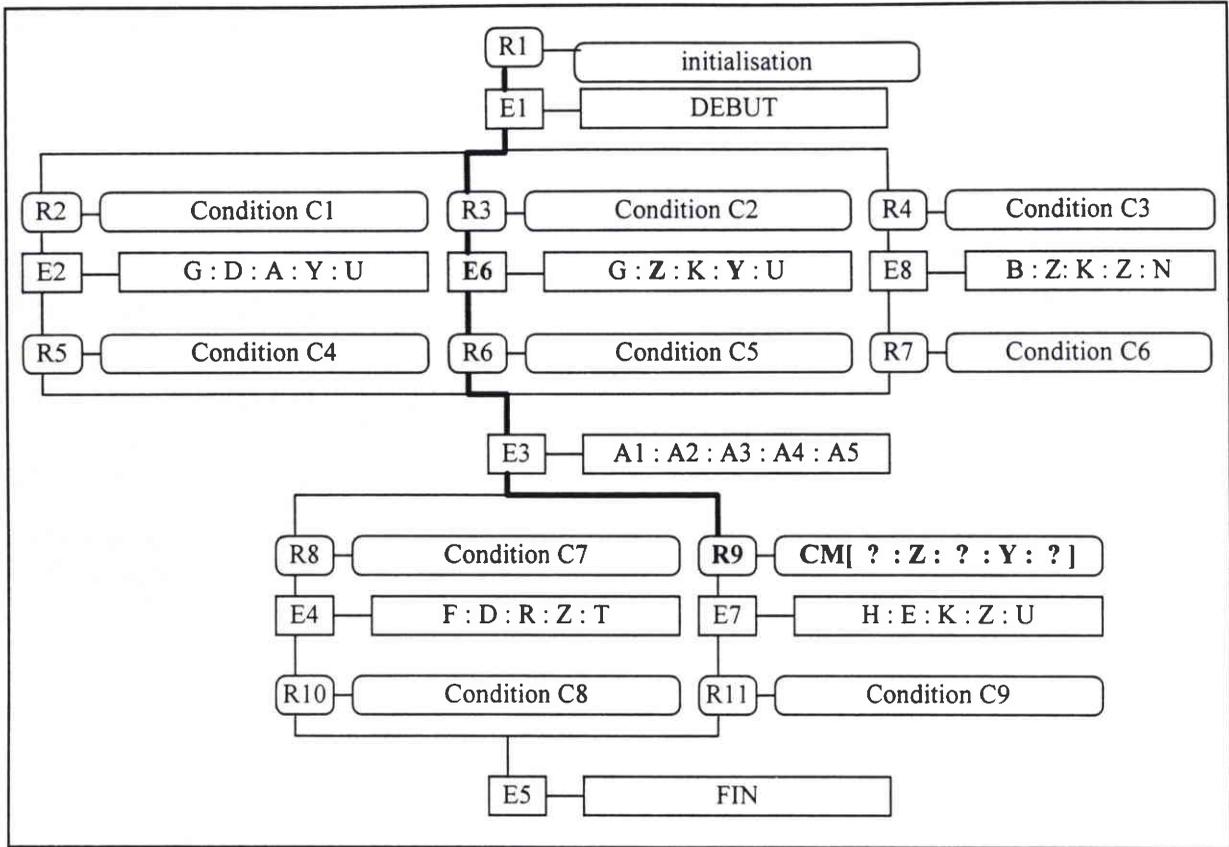


Figure IV-17 : Exemple d'application d'une contrainte de séquence CM

- Une opérande de type Contrainte de séquence sur la partie aVale de la transition courante notée "CV". Cette contrainte vérifie si l'action A, notée dans la partie argument, a été sélectionnée en aval de la transition courante étudiée. Comme pour l'opérateur "CM", la vérification de l'existence d'une action A est considérée comme un "but" à valider.

Soit le marquage courant du réseau R, noté M_{CR}^R , correspondant à l'état de la recherche à l'instant t.

Soit la partie argument de l'étape E notée Arg_E

Soit l'étape E telle que $A \subset Arg_E$

Soit le marquage $M_{CR'}^R$ tel que $E \subset M_{CR'}^R$

Soit la séquence $S_i^{R, M_{CR}^R \rightarrow M_{CR'}^R}$ permettant le passage $M_{CR}^R \rightarrow M_{CR'}^R$

On note :

$$CV[A] \text{ est égale à "VRAI" si : } \{ A \subset Arg_E, E \subset M_{CR'}^R, M_{CR'}^R = M_{CR}^R + [P \times \bar{S}_i] \}$$

L'action est écrite élément par élément avec la possibilité de remplacer un élément par un filtre noté "?".

La syntaxe est donnée par la Figure suivante (Figure IV-18) :

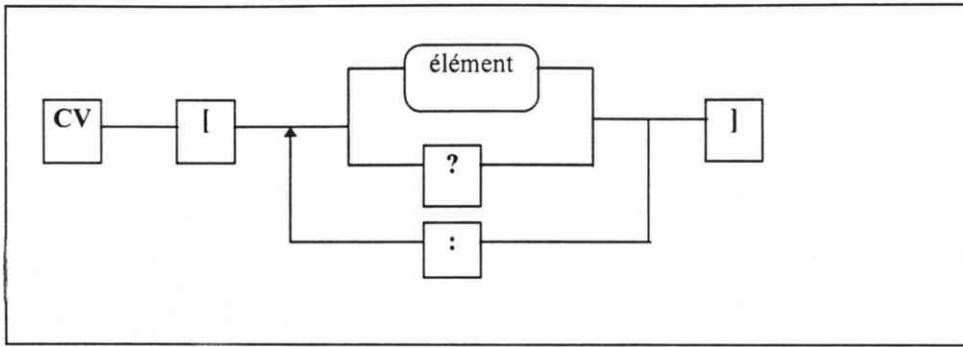


Figure IV-18 : Syntaxe d'écriture d'une opérante de type "CV"

Dans l'exemple de la Figure IV-19, lors de la validation de la condition associée à la transition T4, seul le chemin trait en gras est validé. En effet, la validation de la condition est assujettie à la valeur VRAI de la contrainte de séquence CV[H : ? : ? : ? : ?], qui ne peut être validée que par la sélection de l'étape E7.

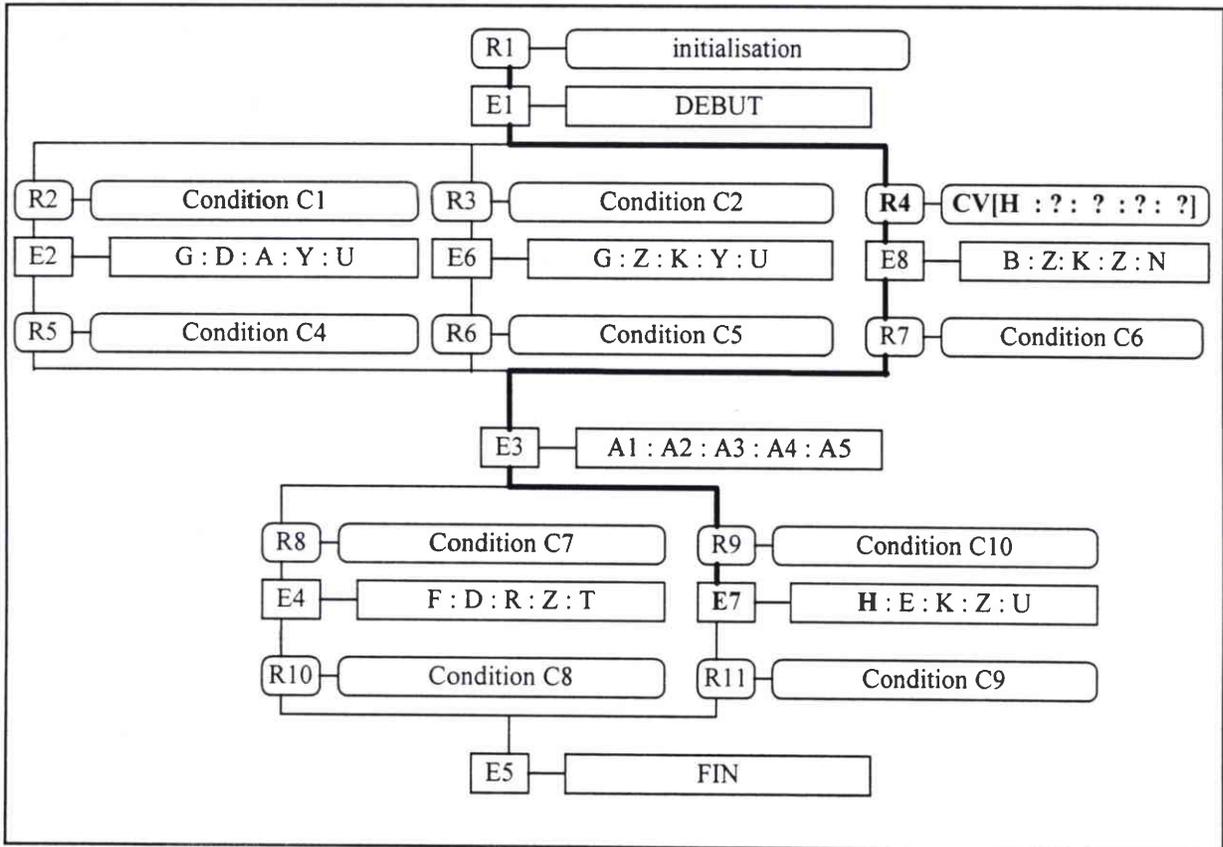


Figure IV-19 : Exemple d'application d'une contrainte de séquence CV

- Une opérante de type Contrainte de séquence sur la Totalité de la séquence devant être sélectionnée. Cette contrainte vérifie si l'action A, notée dans la partie argument.

Soit le marquage de fin de recherche du réseau R, noté M_{Efn}^R ,

Soit la partie argument de l'étape E notée Arg_E

Soit l'étape E telle que $A \subset \text{Arg}_E$

On note : $\text{CT}[A]$ est égale à "VRAI" si : $\{ A \subset \text{Arg}_E, E \subset M_{Efm}^R \}$

L'opérande "CT" peut être définie à partir des opérantes "CV" et "CM" :

On note : $\text{CT}[A]$ est égale à "VRAI" si : $\{ A \subset \text{Arg}_E, \text{CT} = \text{CM}[A] + \text{CV}[A] \}$

L'action est écrite élément par élément avec la possibilité de remplacer un élément par un filtre noté "?".

La syntaxe est donnée par la suivante (Figure IV-20) :

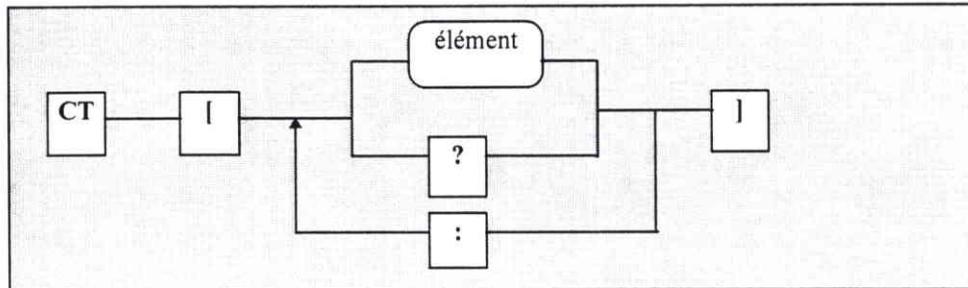


Figure IV-20 : Syntaxe d'écriture d'une opérande de type "CT"

- Les parties arguments des opérandes permettent la manipulation de valeurs.

Il existe **trois types de passage** de valeurs suivant **trois niveaux** (Figure IV-21) :

- **par constante**, dans ce cas la valeur est notée directement et en clair au **niveau de la règle** (ex : règle n°10 la valeur 20 (Figure IV-21)). Une valeur est définie à ce niveau dès lors qu'elle n'est pas appelée à évoluer.

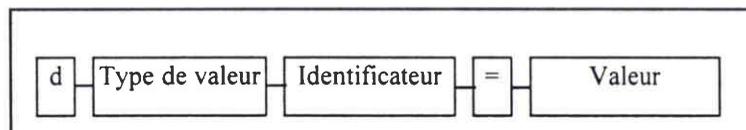
Syntaxe d'écriture : les valeurs sont notées en clair.

Attention, le passage par constante n'autorise que l'utilisation de type élémentaire (E=Entier, C = Chaîne de caractères) et non les types structurés (M = Matrice).

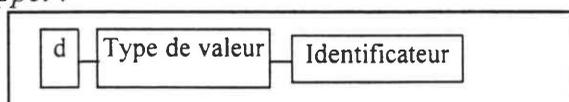
- **par déclaration**, dans ce cas la valeur est notée dans la partie déclaration de la base de connaissances (ex : règle n°11 la valeur dE6 (Figure IV-21)). Une valeur est définie au **niveau de la base de connaissances** dès lors qu'elle est appelée à évoluer mais reste spécifique à la base de connaissances, donc à la famille.

A ce niveau, tous les types de valeurs sont autorisés (E = Entier, R = Réel, C = Chaîne de caractères, M = Matrice).

Syntaxe de définition :



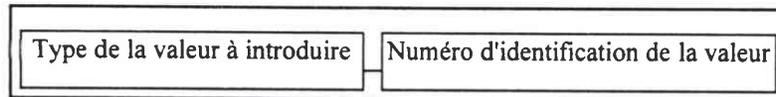
Syntaxe d'appel :



- **par variables**, dans ce cas la valeur est introduite par l'utilisateur au **niveau de la recherche**. Dès lors qu'elle est liée à la recherche de séquence à l'instant (t), elle reste vraie pour une même recherche. A ce niveau, le système affiche à l'écran la question associée à la valeur et l'utilisateur entre la valeur. Le numéro de l'identificateur correspond au numéro de la question à poser. L'ensemble des questions est stocké dans un fichier "Fichier.dat".

A ce niveau, tous les types de valeurs sont autorisés (E =Entier, R =Réal, C =Chaîne de caractères, M =Matrice).

Syntaxe d'écriture :



Syntaxe d'appel :

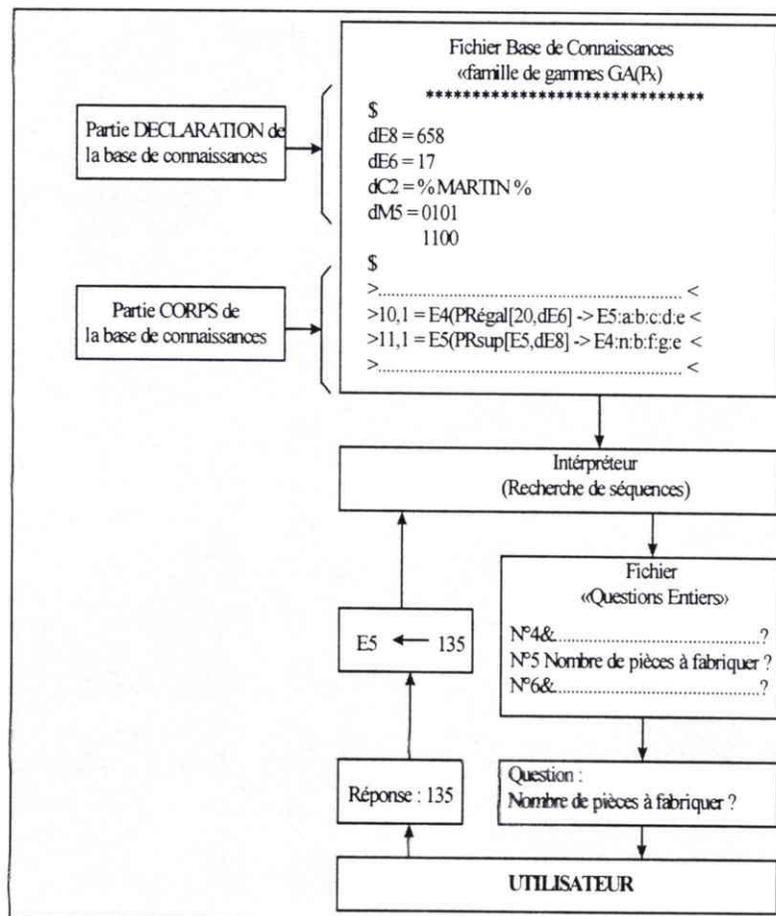
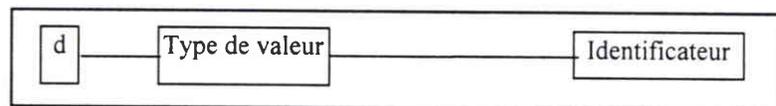


Figure IV-21: Définition des niveaux de passage de valeurs

IV- 4 Exploitation du système

L'exploitation de la base de connaissances consiste à manipuler l'ensemble des règles et des éléments représenté par le modèle GRAFCET-M et générer une ou plusieurs propositions de séquences d'actions.

Un des objectifs du systèmes SIRS est de minimiser l'intervention de l'utilisateur. Pour cela une interface est utilisée (Figure IV-26), permettant de guider l'utilisateur. Lors du lancement du système, l'utilisateur introduit le nom de la base de connaissances sur laquelle la recherche doit être développée. Ensuite la base de connaissances est chargée en mémoire centrale et le système demande à l'utilisateur si la recherche est sous contrainte d'actions (Figure IV-25) non désirables. La recherche étant lancée, l'utilisateur répond à un certain nombre de questions et le système affiche obligatoirement une proposition (Figure IV-23). L'utilisateur peut soit accepter la proposition, et dans ce cas la recherche est terminée, soit demander une autre proposition ou alors modifier (Figure IV-24) la proposition et d'où capitaliser le savoir-faire.

Syntaxe d'une proposition :

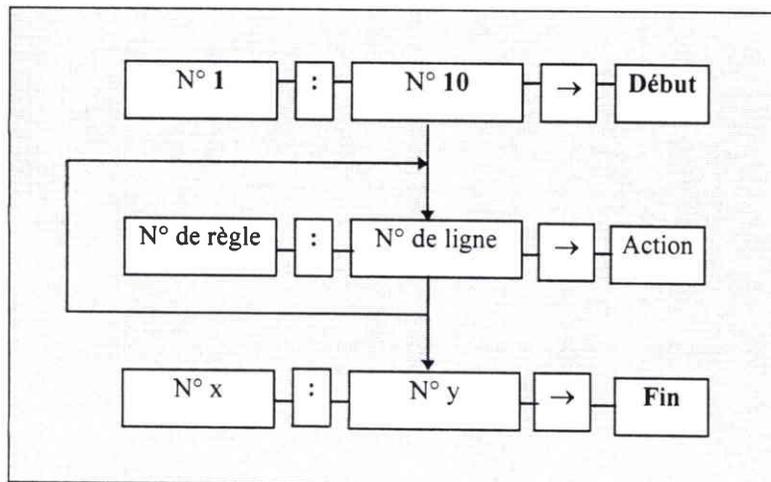


Figure IV-23 : Proposition d'une séquence

Syntaxe de la structure de modification (Figure IV-24) :

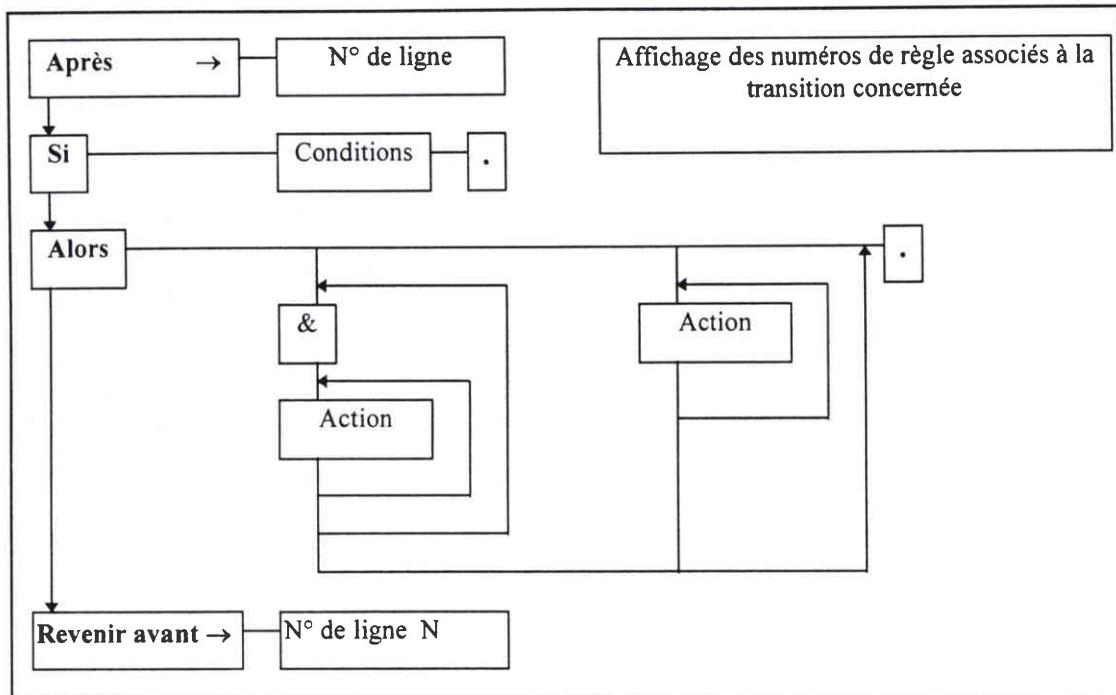


Figure IV-24 : Intégration des modifications

Syntaxe d'écriture d'une liste d'actions proscrites (Figure IV-25) :

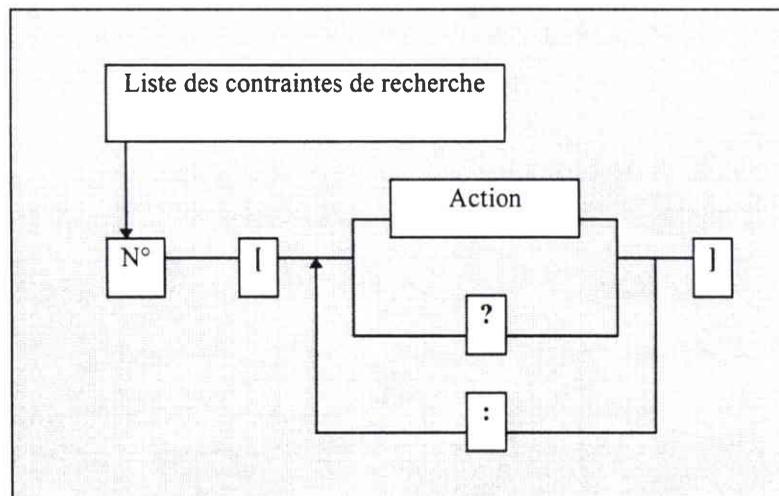


Figure IV-25 : Liste d'actions

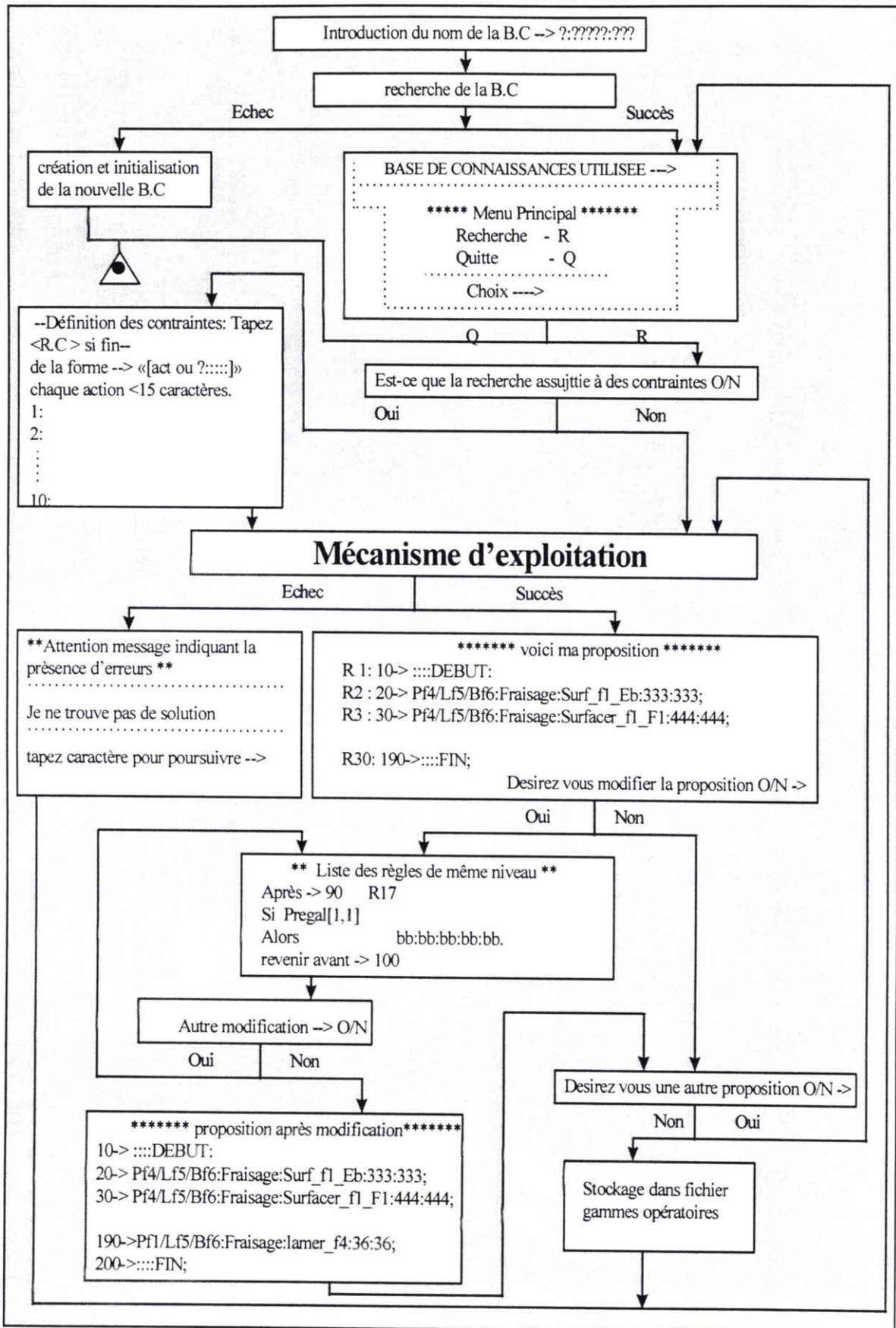


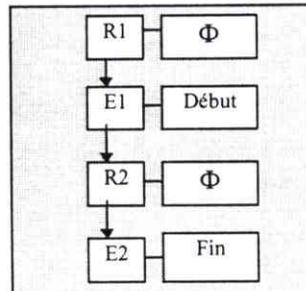
Figure IV-26 : Organigramme des menus

IV-4.1 Exemple 1

IV-4.1.1 Aspect Capitalisation du Savoir-Faire

L'objectif est de mettre en évidence le développement de la base de connaissances à partir de modifications. Pour cela nous présentons différentes étapes concernant alternativement : des propositions, des modifications et des mises à jours de la base de connaissances représentée par GRAFCET-M. La recherche est développée à partir d'une base de connaissances qui n'existe pas, elle est initialisée par le système.

- Etape 1 : Chargement et initialisation de la base de connaissances.



- Etape 2 : Affichage d'une proposition 1, modification 1 et mise à jour de la base de connaissances (Figure IV-27).

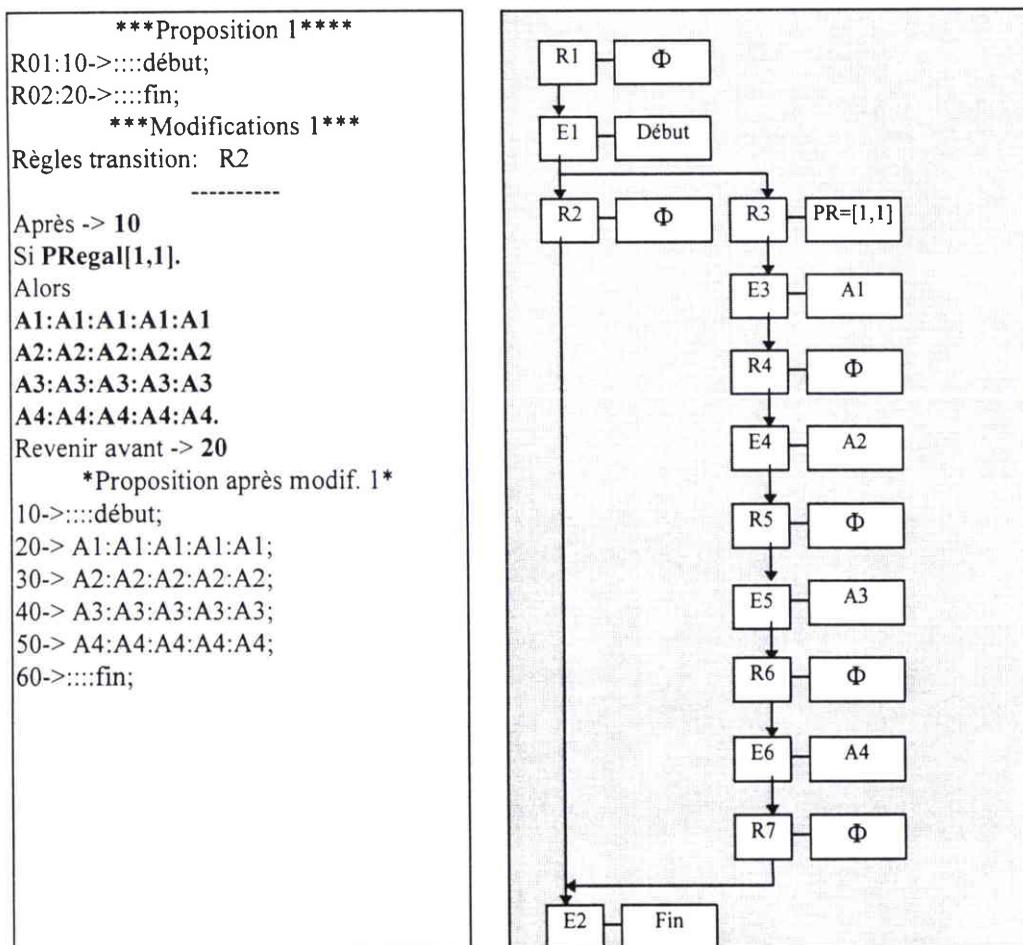


Figure IV-27 : Affichage d'une proposition 1 suivi d'une modification 1

- Etape 3 : Affichage d'une proposition 1, modification 1, modification 2 et mise à jour de la base de connaissances (Figure IV-28).

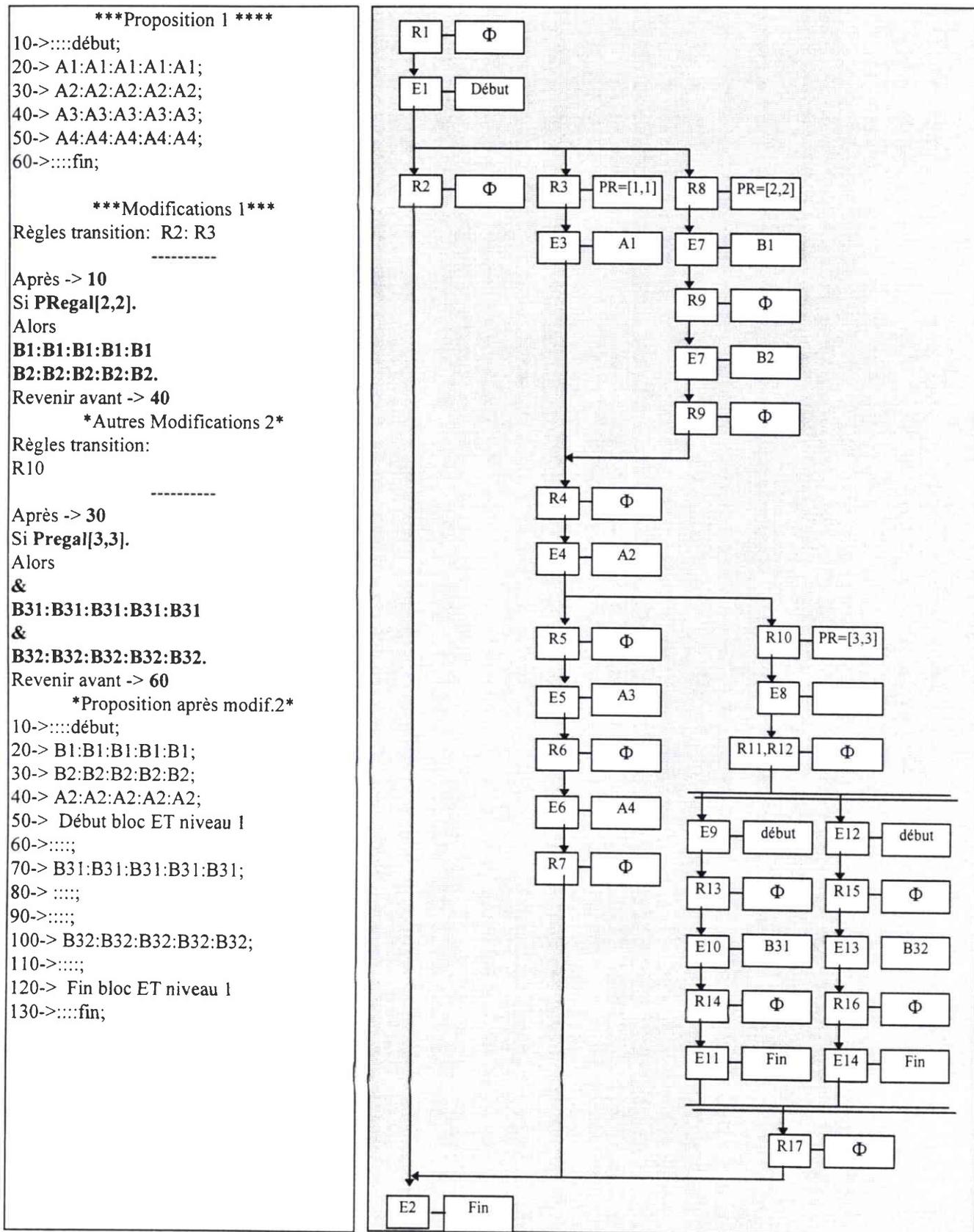


Figure IV-28 : Affichage d'une proposition 1 suivi des modification 1 et modification 2

IV- 5 Exemples d'Applications

IV-5.1 Application du modèle à la représentation de gammes de fabrication

Une gamme de fabrication spécifie l'ensemble des opérations de production nécessaire à la fabrication d'une pièce. Elle est notée : $GF_i(CA_m)$ Gamme de Fabrication i du ComposAnt m . La Figure IV-29 donne l'exemple d'un réseau associé à une telle gamme.

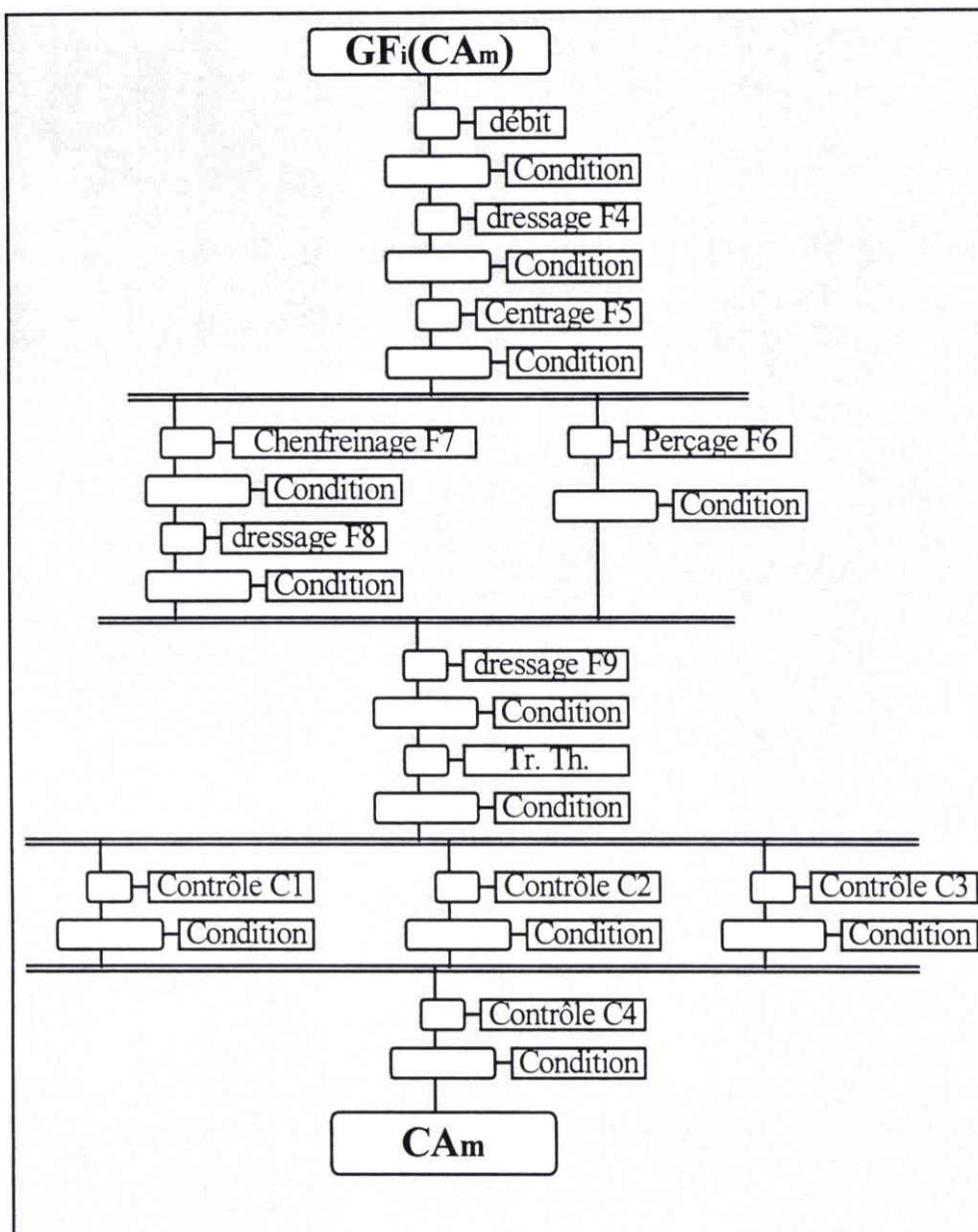


Figure IV-29 : Représentation d'une gamme de fabrication par réseau GRAFCET-M

Dans cet exemple nous considérons une gamme comme une suite ordonnée d'opérations. L'ensemble des opérations exécutées sans discontinuité sur un même poste de travail est appelé phase.

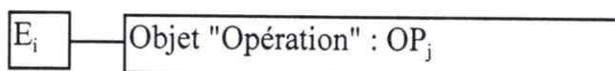
On peut définir une classe "gamme" de la manière suivante :

Définition Classe : "Gamme"	
(
identification de la gamme	: identification de la gamme
identification de la commande	: identification de la commande concernée
date de livraison	: date de livraison de la commande
nombre de produit	: liste des postes pouvant être sélectionnés
liste des opérations	: ensemble ordonné d'opérations de fabrication
)	

De même, nous définissons une classe "opération" :

Définition Classe : "Opération"	
(
identification de l'opération	: identification de l'opération
numéro de l'opération	: identification le numéro de l'opération
type du poste	: identification le type du poste
liste du poste sélectionnable	: liste du poste sélectionnable
type montage pièces	: identification le type montage pièces
liste montage pièces sléctionnable	: liste montage pièces sléctionnable
type montage outil	: identification le type montage outil
liste montage outils sléctionnable	: liste montage outils sléctionnable
type d'outil	: identification le type d'outil
liste d'outil sélectionnable	: liste d'outil sélectionnable
durée de l'opération	: liste de date de début et date de fin
liste des profils opérateur	: la capacité et son savoir-faire
paramètres liés à la qualité demandé	:
paramètres liés à l'exécution de l'opération	: liste paramètres liés à l'exécution de l'opération
outils de contrôle	: liste des outils de contrôle
mode de contrôle	: paramètres liés au mode de contrôle
opération précédente	: Liste des opération précédente
opération suivante	: Liste des opération précédente
)	

En conséquence, la partie action d'une étape est représentée par un objet opération :



Exemples d'opérations : tournage, dressage, centrage, perçage,..... .

IV-5.2 Application du modèle à la représentation de gammes d'assemblage

Une gamme d'assemblage spécifie les composants et les contraintes de montage d'un produit (de succession, de réglage, fonctionnelles, etc.).

Elle est notée : $GA_h(PR_x)$, Gamme d'Assemblage h associée au **PR**oduit x . La Figure IV-30 reproduit un réseau associé à une gamme d'assemblage.

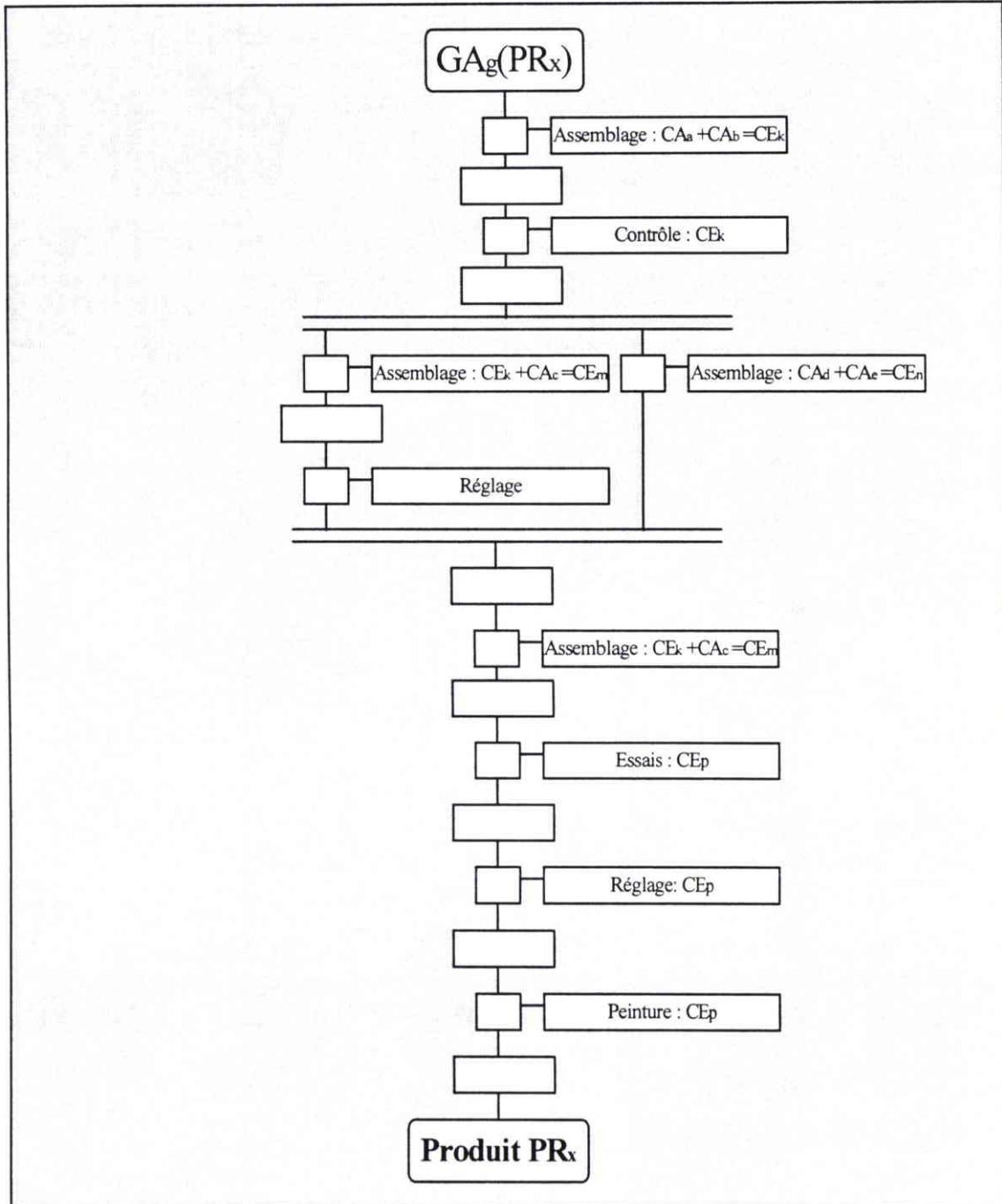


Figure IV-30 : Présentation d'une gamme d'assemblage par réseau GRAFCET-M

IV-5.3 Application du modèle à la représentation de gammes d'intervention

A chaque poste de travail est associé une gamme d'intervention générale qui concerne la maintenance répétitive régulière et des gammes d'interventions suivant les différents types de pannes prévus.

Elle est notée : **GI(PT_z)** Gamme de d'Intervention sur le PosTe z. La Figure IV-31 représente une gamme d'intervention générale qui concerne la maintenance répétitive régulière

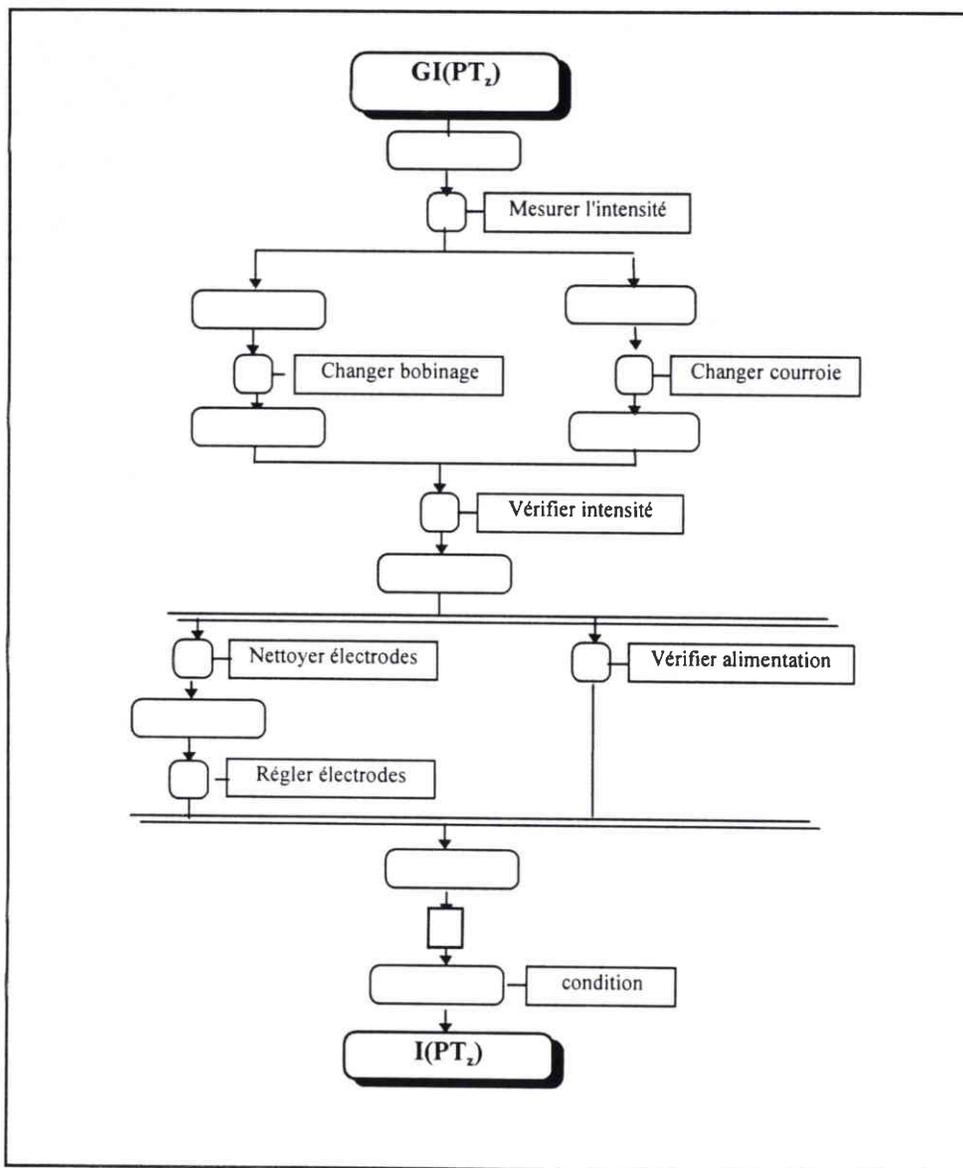


Figure IV-31 : Représentation d'une gamme d'intervention générale de la maintenance répétitive régulière.

IV-5.4 Application du modèle à la représentation de nomenclatures

Une nomenclature regroupe l'ensemble des composants d'un produit.

Elle est notée : $NM_g(PR_x)$, NoMenclature g associée au PR oduit x . Les Figure IV-32 et IV-33 donnent un exemple d'une nomenclature et d'une représentation d'une nomenclature par réseau GRAFCET-M.

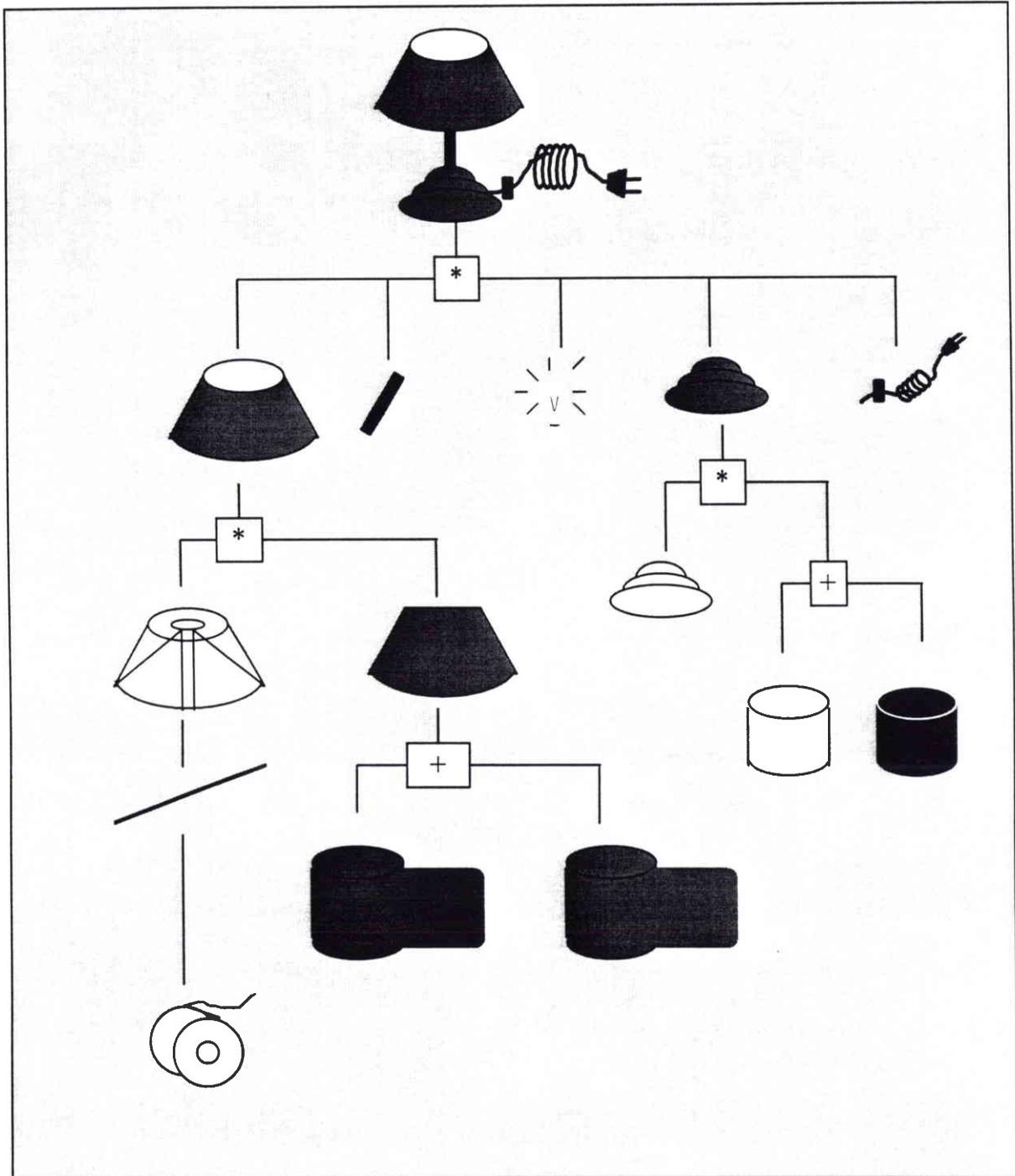


Figure IV-32 : Représentation d'une nomenclature.

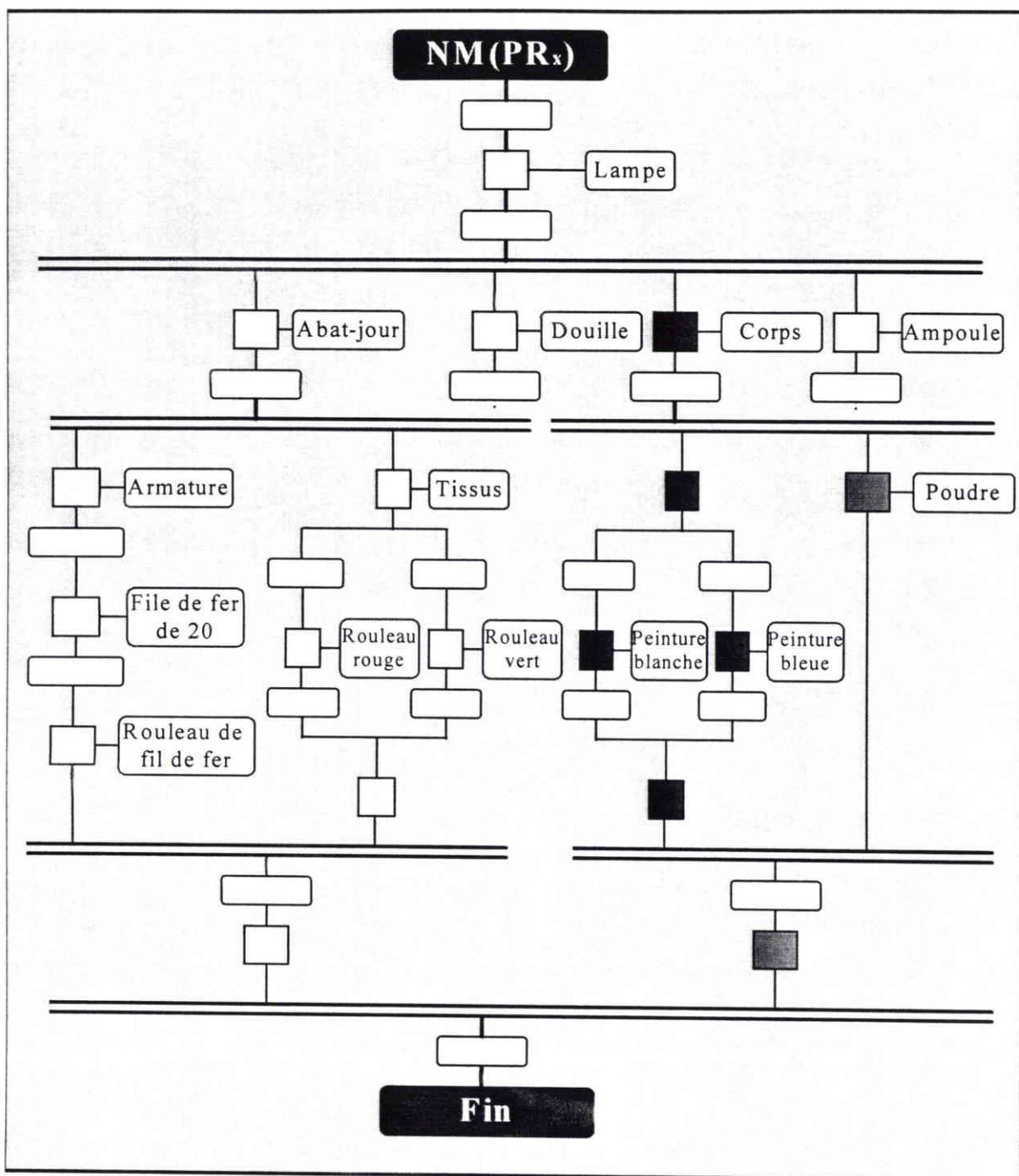


Figure IV-33 : Représentation d’une nomenclature par réseau GRAFCET-M associée à un produit x.


```

----- Suite 1 de la base de connaissances -----
> 1,1= ( )-> E1 ::::DEBUT<
> 2,1= e1 ( )-> E2 ::::FIN<
> 3,1= e1 ( PRegal[1,1] )-> E3 :Pf4/Lf5/Bf6:Fraisage:Surf_fl_Eb:333:333<
> 4,1= e3 ( )-> E4 :Pf4/Lf5/Bf6:Fraisage:Surfacer_fl_Fi:444:444<
> 5,1= e4 (/PRinc[M1,dM2] )-> E6 :Pf1/Lf2/Bf3:Fraisage:Surf_fs_Fi:555:555<
> 6,1= e4 ( PRinc[M1,dM2] )-> E5 :Pf4/Lf5/Bf6:Fraisage:Rain_fl_Eb:666:666<
> 7,1= e5 ( )-> E7 :Pf4/Lf5/Bf6:Fraisage:Rain_fl_Fi:777:777<
> 8,1= e7 ( )-> E8 :Pf4/Lf5/Bf6:Fraisage:Rain_fl_Fi:888:888<
> 9,1= e33 ( )-> E9 ::Controle:::<
> 10,1= e6 ( )-> E33 ::Ebavurage:::<
> 11,1= e9 ( PRinf[E1,10]*PRinc[M1,dM3] )-> E13
:Pf1/Vf4/Bf6:Tournage:Dres_f6_Eb:11:11<
> 12,1= e9 ( PRsup[E1,5]*PRinc[M1,dM3] )-> E10
:Pf1/Lf5/Bf3:Fraisage:Surf_f3f6_Simul:12:12<
> 13,1= e13 ( )-> E14 :Pf1/Vf4/Bf6:Tournage:Dres_f6_Fi:13:13<
> 14,1= e14 ( FOquest[1] ) E15:Pf1/Vf4/Bf6:Tournage:Ales_f6_Eb:14:14<
> 15,1= e14 ( /FOquest[1] ) E16:Pf1/Vf4/Bf6:Tournage:Percer:15:15<
> 16,1= e15 ( )-> E17 :Pf1/Vf4/Bf6:Tournage:Aleser_1/2Fi:16:16<
> 17,1= e16 ( )-> E17 :Pf1/Vf4/Bf6:Tournage:Aleser_1/2Fi:17:17<
> 18,1= e17 ( )-> E18 :Pf1/Vf4/Bf6:Tournage:Aleser_Fi:18:18<
> 19,1= e18 ( )-> E19 :Pf1/Vf4/Bf6:Tournage:Chanfreiner:19:19<
> 20,1= e19 ( )-> E20 :Ct.Lg.int//Bf6:Tournage:Dres_f3_Eb:20:20<
> 21,1= e20 ( )-> E21 :Ct.Lg.int//Bf6:Tournage:Dres_f3_Fi:21:21<
> 22,1= e21 ( )-> E22 :Ct.Lg.int//Bf6:Tournage:Chanfreiner:22:22<
> 23,1= e32 ( EF[?:?:Rain_fl_Eb:?:?] )-> E23 :Pf1/Lf2/Bf6:Percage:Percer_f4:23:23<
> 24,1= e10 ( FOquest[1] )-> E12 :Pf1/Lf5/Vf4:Percage:Aleser_Fi:24:24<
> 25,1= e10 ( /FOquest[1] )-> E11 :Pf1/Lf5/Vf4:Percage:Percer_f6:25:25<
> 26,1= e11 ( )-> E12 :Pf1/Lf5/Vf4:Percage:Aleser_Fi:26:26<
> 27,1= e32 ( /EF[?:?:Rain_fl_Eb:?:?] )-> E30 :Pf1/Lf5/Bf6:Percage:Percer_f4:27:27<
> 28,1= e23 ( )-> E24 :Pf1/Lf2/Bf6:Percage:Lamer_f4:28:28<
> 29,1= e24 ( )-> E25 :Pf1/Lf2/Bf3:Percage:Percer_f5:29:29<
> 30,1= e31 ( )-> E2 ::::FIN<
> 31,1= e25 ( )-> E26 :Pf1/Lf2/Bf6:Percage:Lamer_f5:31:31<
> 32,1= e26 ( )-> E27 :Pf1/Lf2/Bf6:Percage:Tarauder_f5:32:32<
> 33,1= e27 ( )-> E28 :Pf1/Lf2/Bf6:Percage:Percer_f1:33:33<
> 34,1= e28 ( )-> E29 :Pf1/Lf2/Bf6:Percage:Tarauder_f1:34:34<
> 35,1= e29 ( )-> E2 ::::FIN<
> 36,1= e30 ( )-> E31 :Pf1/Lf5/Bf6:Percage:Lamer_f4:36:36<
> 37,1= e22 ( )-> E32 ::Controle:::<
> 38,1= e12 ( )-> E32 ::Controle:::<
> 39,1= e8 ( )-> E33 ::Ebavurage:::<
> 40,1= e33 ( PRinc[M1,dM3] )-> E34 :::::<
> 41,1= e34 ( )-> E35 :::::<
> 42,1= e35 ( )-> E36 :Pf1/Lf5/Bf3:Fraisage:Surf_f4_Fi:42:42<
> 43,1= e36 ( /FOquest[3] )-> E37 :Pf1/Lf5/Bf3:Fraisage:Rain_f4_Eb:43:43<
> 44,1= e37 ( )-> E38 :Pf1/Lf5/Bf3:Fraisage:Rain_f4_Fi:44:44<
> 45,1= e38 ( )-> E39 :::::<
> 46,1= e34 ( )-> E40 :::::<

```

Figure IV-35b : Base de connaissances du Guide de butée (Suite 1)

```

----- Suite 2 de la base de connaissances -----
>47 ,1= e40 ( ) -> E41 :Ct_Lg_ext/Bf3:Fraisage:Surf_Chap_fl:47:47<
>48 ,1= e41 ( /FOquest[3] ) -> E42 :Ct_Lg_ext/Bf3:Fraisage:Rain_Chap_fl:48:48<
>49 ,1= e42 ( ) -> E43 :Ct_Lg_ext/Bf3:Fraisage:Lamer_f4:49:49<
>50 ,1= e43 ( ) -> E44 :Ct_Lg_ext/Bf3:Fraisage:Percer_f4:50:50<
>51 ,1= e44 ( ) -> E45 :::::<
>52 ,1= e39 * e45 ( ) -> E9 ::Contrôle:::<
>53 ,1= e9 ( EF[?:?:Rain_Chap_fl:?:?]*PRsup[E1,5] ) -> E46
:Pf1/Lf5/Bf3:Assemblage:CorpsChapClav:53:53<
>54 ,1= e46 ( ) -> E10 :Pf1/Lf5/Bf3:Fraisage:Surf_f3f6_Simul:54:54<
>55 ,1= e9 ( EF[?:?:Rain_Chap_fl:?:?]*PRinf[E1,10] ) -> E47
:Pf1/Lf5/Bf3:Assemblage:CorpsChapClav:55:55<
>56 ,1= e47 ( ) -> E13 :Pf1/Vf4/Bf6:Tournage:Dres_f6_Eb:56:56<
>57 ,1= e41 ( FOquest[3] ) -> E48 :Ct_Lg_ext/Bf3:Fraisage:Percer_f4_Pig:57:57<
>58 ,1= e48 ( ) -> E43 :Ct_Lg_ext/Bf3:Fraisage:Lamer_f4:58:58<
>59 ,1= e36 ( FOquest[3] ) -> E49 :Pf1/Lf5/Bf3:Fraisage:Percer_f4_Pig:59:59<
>60 ,1= e49 ( ) -> E39 :::::<
>61 ,1= e9 ( EF[?:?:Percer_f4_Pig:?:?]*PRinf[E1,10] ) -> E50
:Pf1/Lf5/Bf3:Assemblage:CorpsChapPig:61:61<
>62 ,1= e50 ( ) -> E13 :Pf1/Vf4/Bf6:Tournage:Dres_f6_Eb:62:62<
>63 ,1= e9 ( EF[?:?:Percer_f4_Pig:?:?]*PRsup[E1,5] ) -> E51
:Pf1/Lf5/Bf3:Assemblage:CorpsChapPig:63:63<
>64 ,1= e51 ( ) -> E10 :Pf1/Lf5/Bf3:Fraisage:Surf_f3f6_Simul:64:64<
>65 ,1= e32 ( PRegal[1,1] ) -> E52 :::::<
>66 ,1= e52 ( ) -> E53 :::::<
>67 ,1= e53 ( ) -> E54 :vvvv:bbbb:bbb:bbb<
>68 ,1= e54 ( ) -> E55 :nnn:bbb:bbb:bbb<
>69 ,1= e55 ( ) -> E56 :::::<
>70 ,1= e52 ( ) -> E57 :::::<
>71 ,1= e57 ( ) -> E58 :bbbb:bbb:bbb:bbb<
>72 ,1= e58 ( ) -> E59 :::::<
>73 ,1= e56 * e59 ( ) -> E31 :Pf1/Lf5/Bf6:Percage:Lamer_f4:36:36<
>74 ,1= e9 ( PRegal[1,1] ) -> E60 :::::<
>75 ,1= e60 ( ) -> E61 :::::<
>76 ,1= e61 ( ) -> E62 :bbbb:b:b:b<
>77 ,1= e62 ( ) -> E63 :n:n:n:n<
>78 ,1= e63 ( ) -> E64 :::::<
>79 ,1= e60 ( ) -> E65 :::::<
>80 ,1= e65 ( ) -> E66 :c:c:c:c<
>81 ,1= e66 ( ) -> E67 :::::<
>82 ,1= e64 * e67 ( ) -> E11 :Pf1/Lf5/Vf4:Percage:Percer_f6:25:25<
>83 ,1= e3 ( PRegal[3,3] ) -> E68 :bbbb:bbb:bbb:bbb<
>84 ,1= e68 ( ) -> E6 :Pf1/Lf2/Bf3:Fraisage:Surf_f5_Fi:555:555<
>85 ,1= e3 ( PRegal[1,1] ) -> E69 :gbvghf:ff:ff:jj:jj<
>86 ,1= e69 ( ) -> E6 :Pf1/Lf2/Bf3:Fraisage:Surf_f5_Fi:555:555<

```

Figure IV-35c : Base de connaissances du Guide de butée (Suite 2)

IV-6 Conclusion

La gestion des savoir-faire constitue un enjeu stratégique dans les organisations pour les années à venir. Savoir et savoir-faire sont les deux aspects indissociables de la connaissance et leur partage constitue une perspective stratégique au sein des entreprises. Ils représentent la trace des activités et des compétences d'une entreprise. Tant qu'ils ne sont pas maîtrisés, ils constituent un **capital fragile** car ils ne sont ni partageables, ni persistants, c'est-à-dire réutilisables en l'absence de leur détenteur (retraite, maladie, démission, mutation). Par contre, s'ils sont maîtrisés, ils deviennent réutilisables. Cette réutilisation permet d'améliorer la qualité des tâches menées, par accumulation de l'expérience, elle peut être utilisée pour mettre en valeur les individus par leur compétences. En effet, le savoir-faire d'une entreprise représente l'ensemble des compétences individuelles et sa mémoire collective. Le savoir-faire pour une personne correspond à ses connaissances méthodologiques.

Le savoir-faire est difficile à capturer, à formaliser. Comment peut-on réutiliser ses savoir-faire. Quelle architecture et quelles fonctionnalités doit avoir un système logiciel pour gérer ces éléments, c'est-à-dire les rendre effectivement existants au sein d'un groupe, partageables et réutilisables. Il s'agit de rendre ses savoir-faire réellement exploitables en action, c'est-à-dire dans l'exercice des tâches.

Nos travaux se placent dans le cadre de l'activité de l'ingénierie pour le pilotage d'atelier. Cette activité est menée par des ingénieurs, des cadres, des techniciens et des opérateurs. Elle est caractérisée par la double tâche d'affectation des ressources et du suivi de fabrication.

En outre, elle produit un certain nombre de documents intermédiaires ou finaux destinés à formaliser, partager, et faire valider les éléments d'informations à chaque étape. Cet aspect de la problématique n'a pas été abordé dans le cadre de nos travaux.

Dans le cadre de nos travaux, nous avons utilisé le système à base de connaissances SIRS qui permet la mémorisation et la gestion de savoir-faire. Pour cela, il utilise le modèle GRAFCET-M implanté à partir d'un langage de programmation évolutif. La capitalisation du savoir-faire est obtenue lors de modification de propositions grâce à une interface.

Nous rappelons qu'il ne s'agit pas de chercher à concevoir des systèmes automatiques se substituant aux acteurs (systèmes experts), mais de proposer, dans une approche système d'informations un cadre permettant à chacun de décrire ses savoir-faire et permettant à tous de les réutiliser.

Chapitre V :

**Proposition d'une architecture de conduite :
Architecture distribuée avec contrôle
hiérarchique**

V-1 Introduction	99
V-2 Systèmes multi-agents	100
V-2.1 Approches existantes.....	100
V-2.2 Agents	101
V-2.2.1 Caractéristiques d'un agent.....	101
V-2.2.2 Architecture d'un agent cognitif	102
V-2.2.3 Architecture fonctionnelle d'un agent.....	103
V-2.3 Organisation d'un système multi-agents	104
V-2.4 Coopération dans un système multi-agents.....	105
V-2.5 Modes de communication	105
V-3 Systèmes multi-experts	106
V-3.1 Principe de fonctionnement	106
V-3.2 Sources de connaissances.....	107
V-3.3 Blackboards.....	109
V-3.4 Contrôle.....	111
V-3.4.1 Résolution de conflits.....	112
V-3.4.2 Focalisation du contrôle	113
V-3.4.3 Représentation du contrôle.....	113
V-3.4.4 Modes de contrôle procédural	114
V-3.4.5 Contrôle hiérarchique	115
V-3.4.6 Mode de contrôle opportuniste.....	116
V-3.4.7 Critères comparatifs	118
V-3.5 Systèmes multi-experts existants	119
V-4 Choix d'un outil d'implantation	120
V-5 Présentation de l'outil ATOME (Another TOol for Multi-Expertise)	120
V-5.1 Les blackboards.....	122
V-5.2 Événements	123
V-5.3 Les spécialistes.....	126
V-5.4 Les tâches.....	128
V-5.4.1 Tâche dirigée par les événements.....	129
V-5.4.2 Tâche dirigée par les règles	130
V-5.4.3 Tâche opportuniste	131
V-5.4.4 Choix d'un type de raisonnement	134
V-5.5 La stratégie.....	135
V-5.6 Le moteur d'inférences	136
V-6 Développement de l'application	136
V-6.1 Définition des blackboards.....	136
V-6.1.1 Blackboard "Ressources"	136
V-6.1.2 Blackboard "ProgrammeFabrication"	139
V-6.1.3 Blackboard "Demandes"	145
V-6.1.4 Blackboard "PostesCharge"	147
V-6.1.5 Blackboards "InitSolutionsRCS"	150
V-6.1.6 Blackboards "InitSolutionsSTR".....	152

V-7 Définitions de l'architecture du système proposé.....	153
V-7.1 Définition des spécialistes.....	154
V-7.2 Proposition d'une architecture d'implantation.....	159
Les information manipulées sont de trois types et concernent :	161
V-7.3 Définition du mécanisme de contrôle	161
V-7.3.1 Blackboard "Résumés"	161
V-7.3.2 Définitions des tâches.....	164
V-7.3.3 Définition des événements	166
V-7.3.4 Définition de la stratégie	168
V-7.3.5 Proposition d'une architecture.....	168
V-8 Conclusion.....	171

V-1 Introduction

L'augmentation constante du volume des connaissances humaines engendrée par les avancées technologiques et les nouvelles méthodes de travail induit la diminution du **champ** des compétences de chaque individu. De ce fait, cela entraîne l'accroissement de la spécialisation nécessaire au développement d'une activité professionnelle de haut niveau. Cette évolution des sciences et des techniques tend à supprimer la notion d'expert interdisciplinaire et encourage des **interactions** plus fréquentes et plus complexes entre spécialistes.

Dans le cadre de la conduite d'une production, on fait appel à un ensemble d'activités humaines pour lesquelles une **approche centralisée** s'avère insuffisante. La synchronisation du processus de conduite impose la connaissance de données sur la situation dans le domaine de ces activités. De plus, la résolution d'un problème posé fait intervenir des connaissances issues de spécialités différentes.

Les connaissances ainsi manipulées ont un caractère :

- **pluridisciplinaire**, qui existe à travers les multiples fonctions nécessaires à la maintenance, la qualité, au fonctionnement des systèmes mécaniques, etc.
- **interdisciplinaire**, qui existe à partir des liens nécessaires entre les différentes informations et compétences, dont la coopération est cruciale pour une utilisation optimale.

Le passage d'un comportement **individuel** (système unique) à un comportement **collectif** (multi-systèmes) fait l'objet de nombreux travaux et reste du domaine de l'**intelligence artificielle distribuée** (I.A.D.) [Huhns 87]. L'objectif est la résolution d'un problème (diagnostic, planification d'actions, ...) en **distribuant l'expertise** sur un groupe d'éléments qui évolue dans un même environnement et règle les éventuels conflits.

De manière générale, l'intelligence artificielle classique s'intéresse à la modélisation du comportement d'un seul élément, et l'intelligence artificielle distribuée s'intéresse à l'ensemble des comportements intelligents issus de l'activité de plusieurs éléments. Ceux-ci restent autonomes et interagissent dans un environnement commun, avec comme objectif la résolution d'un problème.

La problématique liée au développement et à l'application de l'intelligence distribuée s'attache à résoudre des problèmes tels que [Muller 96] :

- la mobilisation de la connaissance, le problème de sa répartition sur les différents agents et l'exploitation des résultats,
- la génération de plans d'actions avec prise en compte des agents compétents,
- la gestion des conflits entre les agents et le maintien de la cohérence des décisions et des plans d'actions,
- la communication et l'interaction entre les agents (langage, protocole) avec prise en compte des conséquences,
- l'organisation, l'architecture et la coopération des agents et des résultats.

Parmi les systèmes existants, on relève deux approches différentes [Labidi 93] :

- **l'approche multi-agents**, dans ce cas, chaque élément du système est appelé agent et l'on parle de système multi-agents.

- **l'approche multi-experts**, dans ce cas chaque élément du système est appelé source de connaissances et l'on parle de système multi-experts.

V-2 Systèmes multi-agents

Un système multi-agents se compose d'un ensemble **d'agents autonomes** organisés autour d'un **système de communication** et dont l'objectif est la résolution de problèmes distribués.

Pour cela, il s'appuie sur une **structure en réseaux** qui permet la communication entre plusieurs agents et des objets associés à un environnement. Son fonctionnement est assujéti à une **organisation sociale** et fait appel à des **mécanismes de contrôle et de prise de décision**.

V-2.1 Approches existantes

Les différentes approches utilisées dans le développement d'applications existantes peuvent être classées en deux catégories et différenciées par le degré de détail des connaissances des agents impliqués. Cette caractéristique est appelée la "**granularité**" et exprime la complexité des fonctionnalités de l'agent.

Il existe deux principaux types d'approches :

1. Les **approches réactives** [Brooks 89 ; Ferber 90 ; Ferber 88 ; Sari 92], dont les agents sont dits réactifs. Cette approche est fondée sur le principe **qu'il n'est pas nécessaire que chaque agent soit individuellement intelligent pour parvenir à un comportement global intelligent**. C'est à dire que des mécanismes simples de réaction aux événements (stimulus/action) **peuvent faire émerger des comportements correspondant aux objectifs**. Du fait de la simplicité, donc de la **très forte granularité de ces agents (à grain fin)**, il est nécessaire d'en augmenter le nombre. D'après Books [89], plusieurs milliers de micro-robots identiques, travaillant ensemble sur une tâche donnée, peuvent être plus efficaces qu'un seul gros robot spécialisé. Si l'un d'entre eux tombe en panne, le relais est pris par d'autres et le fonctionnement global n'est pas condamné. La construction de ces petits agents simples, dotés d'une autonomie réelle, ne serait pas plus difficile que la fabrication des circuits intégrés. Un autre exemple d'application est celui développé par l'équipe MIRIAD du LAFORIA concernant le trafic aérien [Sari, 92]. L'objectif est de coordonner les trajectoires d'un grand nombre d'avions, chaque avion étant considéré comme un agent réactif autonome.

L'analogie classique d'une société d'agents réactifs est celle de la fourmilière ou de la ruche d'abeilles.

On note toutefois, que la complexité des systèmes réactifs nécessite le développement de nouvelles théories dans les domaines de la coopération ou de la communication ainsi que d'un **nouveau concept "l'émergence"**. Une fonctionnalité qui émerge est une fonctionnalité qui résulte d'interaction pouvant concerner : le comportement, la stratégie, le contrôle, etc.

2. Les **approches cognitives** [Dieng 90] dont les agents sont dits "cognitifs". Cette approche est fondée sur la **coopération d'agents capables à eux seuls d'effectuer des opérations complexes**. Le système se compose d'un nombre limité d'agents disposant d'une faible **granularité (gros grain)**. C'est à dire, que chacun d'eux dispose d'une capacité de raisonnement sur une base de connaissances liée au domaine d'application qui l'intéresse, avec une aptitude à traiter des informations relatives à l'environnement de travail, et en particulier à la gestion des interactions.

Les points fondamentaux de ces deux approches peuvent être récapitulés de la manière suivante (Tableau V-1) :

Approche réactive	Approche cognitive
Pas de représentation explicite de l'environnement.	Représentation explicite de l'environnement.
Pas d'historisation.	Prise en compte de l'historique.
Forte granularité (grain fin).	Faible granularité (gros grain).
Grand nombre d'agents.	Petit nombre d'agents.
Agents non intelligents.	Agents intelligents.
L'intelligence émerge des interactions de l'ensemble des agents.	L'intelligence est intrinsèque à chaque agent.
La perte d'un agent n'est pas significative.	La perte d'un agent est préjudiciable.

Tableau V-1 : Les approches réactive et cognitive

V-2.2 Agents

Un agent peut être défini comme une entité physique ou non, capable d'agir sur elle même et sur son environnement, dont il dispose d'une représentation partielle [Ferber 88]. Les actions générées sont directement liées à sa connaissance, à ses interactions avec les autres agents ainsi qu'à ses observations sur l'environnement.

Le concept d'agent est développé suivant deux aspects :

- **social**, défini par les mécanismes et les connaissances attachés aux activités de groupe d'agents.
- **individuel**, défini par les mécanismes et les connaissances attachés aux rôles et fonctionnement interne de l'agent.

V-2.2.1 Caractéristiques d'un agent

Un agent est caractérisé par un ensemble de paramètres caractérisés en trois classes:

- **Caractéristiques de définition:**

Celles-ci sont utilisées pour situer et définir la place de l'agent dans le domaine d'utilisation, c'est à dire l'application. On peut citer : son rôle, sa spécialité, ses objectifs, ses croyances, ses capacités (décisionnelles, de communication, d'apprentissage).

- **Caractéristiques d'interface :**

Celles-ci sont utilisées pour expliquer des actions vis-à-vis de l'environnement de l'agent. Elles peuvent être liées :

- **à l'environnement**, représentation que l'agent se fait de son environnement et de lui-même.

Ex: l'état de sa production par rapport à l'ensemble des activités de l'atelier.

- **à la structure**, détermination des composants de l'agent.

Ex : vitesse, flux, stock.

- **au comportement**, prise en compte des caractéristiques environnementales dans le comportement de l'agent.

Ex: opérateur, cahier des charges.

- **Caractéristiques de fonctionnement :**

Celles-ci permettent de définir les différents modes de fonctionnement que l'on exprime à partir de quelques notions.

On peut citer :

- **L'intentionnalité** [Searle 90 ; Searle 92 ; Tuomela et cool. 88] qui est définie par la déclaration explicite des buts et des moyens. Une intention exprime les actions à effectuer par un agent pour atteindre un but. Un agent intentionnel est un agent guidé par les buts.

- **La rationalité** [Newell 82 ; Pol 92] qui consiste à sélectionner la meilleure action pour atteindre l'objectif. Un agent rationnel dispose de critères d'évaluation de ses actions, lui permettant d'en sélectionner la meilleure, et ainsi de justifier sa décision.

- **L'engagement** [Becker 60 ; Bond 90] après avoir construit un plan pour atteindre un but, l'agent se donne les moyens d'y parvenir et donc "s'engage" à accomplir les actions pour satisfaire ce but. La notion d'engagement est essentielle dans le cadre de la coopération entre plusieurs agents. Un agent est dit coopératif lorsqu'il planifie ses actions par coordination et négociation avec les autres agents.

- **L'adaptabilité** qui est définie par la capacité d'un agent de contrôler ses aptitudes (communication, comportement) en fonction de l'agent avec lequel il interagit. L'adaptabilité d'un agent fixe son degré de flexibilité.

V-2.2.2 Architecture d'un agent cognitif

L'architecture se compose d'un ensemble de connaissances nécessaires au bon fonctionnement du système multi-agents. Ces connaissances intéressent les trois points essentiels que sont :

- **l'environnement**,

- **le problème à résoudre**,

- **la coopération** liée à l'importance des interactions entre les agents.

Pour répondre à ces exigences, un agent est structuré en cinq zones (Figure V-1) :

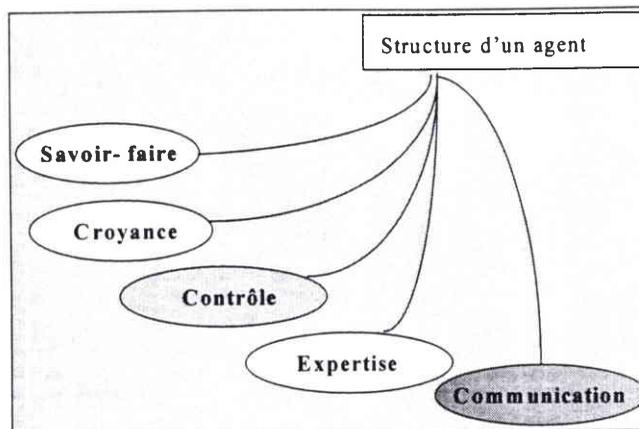


Figure V-1 : Structure d'un agent

- le **savoir-faire** représente les connaissances issues de la compétence de l'agent. Celui-ci permet au système global de sélectionner un agent en fonction de ses compétences, pour une tâche donnée.
- la **croyance** représente les connaissances de l'agent sur lui-même et les autres. Ces connaissances étant propres à chaque agent, elles représentent l'idée que celui-ci se fait de lui-même et des autres agents, et de ce fait, sont nécessairement non objectives, d'où l'appellation "croyance" et non pas "certitude".
- le **contrôle** représente les connaissances définies par les buts, les intentions, les plans et les tâches que l'agent possède.
- l'**expertise** représente les connaissances directement liées à la résolution du problème.
- la **communication** regroupe les informations et les connaissances concernant le protocole de communication avec les autres agents.

V-2.2.3 Architecture fonctionnelle d'un agent

L'architecture fonctionnelle (Figure V-2) d'un agent se compose de quatre fonctions principales.

- **fonction perception** : qui permet l'acquisition d'informations de type proprioceptif, qui concernent l'agent et de type extéroceptif qui concernent l'état de l'environnement.
- **fonction communication** : mise en oeuvre par l'ensemble des capacités d'interaction avec les autres agents (négociation, etc.).
- **fonction décision et planification** : le contexte distribué génère des disponibilités d'interactions et donc des possibilités de négociations qui autorisent une gestion locale des conflits et une planification dynamique.

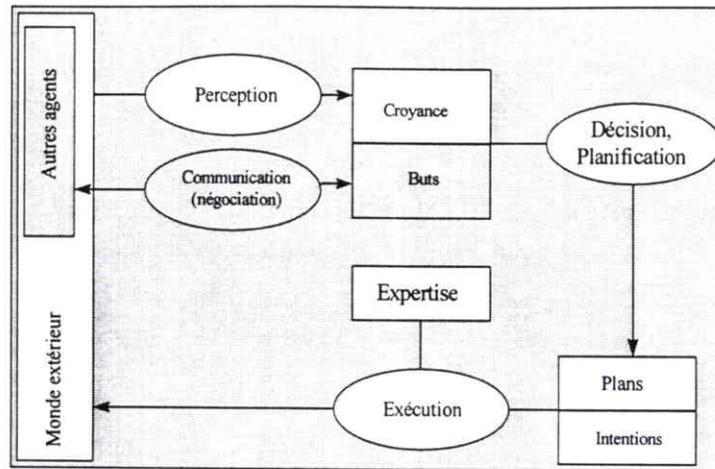


Figure V-2 : Architecture fonctionnelle d'un agent

Pour cela, il est possible d'appliquer deux approches différentes :

- Une première approche où chaque agent construit son propre plan en coordination avec d'autres agents.
- Une deuxième approche qui consiste à mettre en place une planification centralisée permettant l'élaboration d'un plan global, les conflits étant gérés par un superviseur. L'aspect décision consiste à ordonner la liste des buts en fonction des informations de perception.
- **fonction exécution** : c'est l'application proprement dite du plan sur l'état de l'environnement.

A partir des processus de perception du monde extérieur et d'interactions avec les autres agents, l'agent établit un ou plusieurs buts. Ensuite, à partir du processus de décision, il sélectionne le but à satisfaire en priorité.

V-2.3 Organisation d'un système multi-agents

L'organisation d'un système multi-agents permet d'en définir sa nature, sa responsabilité, ses besoins en ressources d'informations et les liens de communication entre les agents du système.

Une structure organisationnelle est différenciée par :

- le type de décentralisation,
- le type d'agent,
- le mode de coopération entre les agents.

Les recherches en intelligence artificielle distribuée s'orientent vers l'adaptation des modèles sociologiques aux systèmes multi-agents. Parmi les structures existantes, on peut citer les structures [Mintzberg 86] :

- **simples**, organisations pyramidales centralisées,
- **bureaucratiques**, organisations décentralisées verticalement et horizontalement,
- **adhocratiques**, organisations fonctionnant par ajustement mutuel entre des petits groupes.

Toutefois, on note d'après Bond [90] qu'il n'existe que deux structures d'organisation :

- **la structure horizontale** dans laquelle tous les agents sont au même niveau, sans relation de type maître-esclave. Ce type de structure est bien adapté dans le cas d'agents ayant des spécialités différentes et travaillant pour la résolution d'un même problème.
- **la structure verticale** dans laquelle les agents sont structurés par niveau et traitent le problème à résoudre suivant différents niveaux d'abstraction. Sur chacun des niveaux, on peut retrouver localement une structure horizontale. Dans une telle structure, l'agent reçoit le problème à résoudre d'un agent hiérarchiquement supérieur et le décompose en sous-problèmes. Puis, il y a tentative de régler le problème au niveau local, en collaboration avec les autres agents de même niveau. Les sous-problèmes non résolus sont transmis aux agents de niveau inférieur.

V-2.4 Coopération dans un système multi-agents

La coopération [Durfee *et coll.* 89 ; Cammarata *et coll.* 83] est un aspect très important dans les systèmes multi-agents. En effet, une résolution distribuée d'un problème est le résultat de l'interaction coopérative entre les agents. Pour cela, un agent doit disposer de certaines qualités caractérisées par la capacité :

- de mettre à jour le modèle du monde environnant,
- d'intégrer des informations venant d'autres agents,
- d'interrompre un plan pour aider d'autres agents,
- de déléguer une tâche qu'il ne peut résoudre, à un autre agent dont il connaît les compétences.

Il existe quatre modèles de coopération [Vailly 88] :

- **La coopération par partage** : l'ensemble des agents coopère par partage de tâches et de résultats, avec la possibilité de prendre en compte localement les plans des autres agents.
- **La coopération par commande** : un agent dit "supérieur" décompose un problème en sous-problèmes et le distribue à d'autres agents. Lorsque ceux-ci ont résolu ces sous-problèmes, les différentes solutions partielles sont renvoyées à l'agent dit "supérieur".
- **La coopération par appel d'offre** : un agent dit "supérieur" décompose le problème en sous-problèmes et diffuse la liste parmi les autres agents. Ensuite chaque agent, en fonction de ses compétences, envoie une offre à l'agent dit "supérieur" qui, en fonction de celles-ci, distribue les sous-problèmes.
- **La coopération par compétition** : un agent dit "supérieur" décompose le problème en sous-problèmes et en diffuse la liste. Chaque agent, en fonction de ses compétences, choisit un ou plusieurs sous-problèmes. Ensuite, les résultats sont envoyés à l'agent dit "supérieur" qui, après sélection, retient les meilleurs.

V-2.5 Modes de communication

On relève deux modes de communication [Vernadat 92] :

- **La communication par échange de messages** : dans lequel les agents sont directement connectés les uns aux autres. Dès lors qu'un agent connaît la connaissance d'un autre agent, il

peut entrer en communication avec lui. Un exemple d'utilisation de communication par échange de messages est représenté par les **systèmes acteurs**.

- **La communication par partage d'informations** : dans laquelle les agents communiquent à partir d'une structure de données partagée (Figure V-3). Celle-ci regroupe les connaissances relatives à la résolution du problème à tout moment et, de ce fait, est représentative de l'état courant du problème.

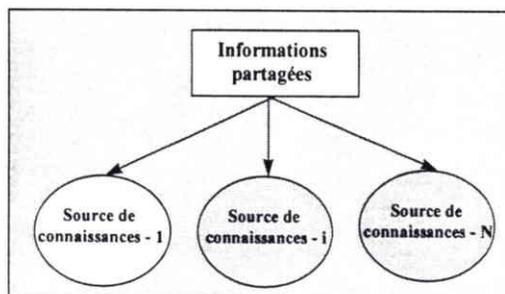


Figure V-3 : Communications par partage d'informations

Ce mode de communication est très utilisé dans la conception de **système multi-experts**. Sa mise en œuvre est principalement réalisée à partir de systèmes à base de **blackboard**.

V-3 Systèmes multi-experts

Dans un système où la communication entre agents n'est pas directe, mais réalisée par l'intermédiaire d'une structure de données partagée, il est fait appel à des agents appelés sources de connaissances. On parle alors de systèmes multi-experts.

V-3.1 Principe de fonctionnement

Un système multi-experts est composé d'un ensemble d'entités intelligentes autonomes et d'une base de données centrale appelée blackboard [Hayes-Roth 84].

La base de données est structurée hiérarchiquement et représente la zone de partage de données pour les différentes entités. Les entités peuvent alors communiquer à travers les données qu'elles y lisent, écrivent ou modifient. Initialement, un blackboard représente le problème à résoudre, à partir duquel les entités interviennent de manière autonome. C'est ainsi que les actions de l'ensemble des entités font évoluer le problème vers une solution. En conséquence, **la solution est construite de manière incrémentale et opportuniste selon l'état de la résolution en cours**.

Il existe deux modes d'accès au blackboard :

- **le mode parallèle** qui concerne les implantations parallèles, dont les entités ont la possibilité d'accéder à tout moment au blackboard,

- **le mode séquentiel** qui concerne les implantations monoprocesseur, chaque entité intervient de manière exclusive. Dans ce cas, il est nécessaire de faire intervenir un moniteur permettant de gérer le cas où plusieurs entités sont éligibles. Pour établir son choix, le moniteur peut faire appel à des critères tels que : premier arrivé, premier sélectionné, ou priorité à l'entité possédant le plus d'actions à réaliser sur le blackboard. Ces critères constituent des stratégies

de résolution que le moniteur sélectionne à l'initialisation du système ou qu'il développe en cours d'exécution. **Le moniteur est aussi appelé le système de contrôle.**

Le principe fondamental sur lequel repose le modèle blackboard est la séparation de la connaissance. La solution du problème dépend de la contribution de chacun des spécialistes des domaines. Elles est constituée d'éléments d'information fournis par chaque spécialiste et dont l'assemblage forme la solution. En conséquence, il est nécessaire de s'appuyer sur une structure pour organiser ces éléments de solution. Un exemple d'organisation peut être une représentation hiérarchique du problème, dont la racine représente l'objectif à atteindre. Chaque niveau de la hiérarchie possède ses propres modes de description.

En introduisant un raisonnement sur plusieurs niveaux d'abstraction, le modèle blackboard permet de réduire l'espace de recherche et donc sa combinatoire.

Le principe de fonctionnement peut être rapproché de celui des systèmes experts. L'architecture classique des systèmes experts est constituée d'une base de règles, d'une base de faits et d'un moteur d'inférence (Figure V- 4)

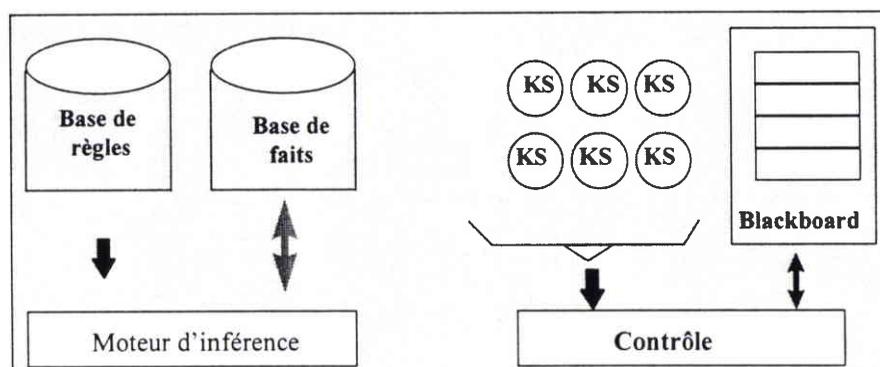


Figure V-4 : Analogie architecturale "Système expert - Système à blackboard"

Dans une architecture classique de système expert, le moteur d'inférence exploite la base de règles en fonction de l'état courant de la base de faits et génère de nouveaux faits qu'il place dans la base de faits.

Le modèle blackboard est une architecture particulière de système expert, modulaire, multi-expert lui permettant de traiter des applications plus complexes. La connaissance est séparée en plusieurs modules indépendants appelés sources de connaissances ou KS (Knowledge Source). Celles-ci partagent une base de données structurées (blackboard) qui constitue la base de faits du système. Le contrôle joue le rôle du moteur d'inférence, c'est à dire qu'il gère l'activité des sources de connaissances et la cohérence du blackboard. C'est ainsi que le modèle blackboard permet la coopération entre plusieurs domaines de connaissance. Ces sources de connaissances sont indépendantes et peuvent de ce fait être de types différents déclaratif ou procédural.

Cette possibilité de faire coopérer plusieurs sources de connaissances permet de compenser l'imprécision des données et la disponibilité des connaissances. Une conséquence est l'utilisation de connaissances redondantes sur le même problème permettant ainsi d'affiner la solution et la rendre **opportuniste**.

V-3.2 Sources de connaissances

L'expertise nécessaire à la résolution du problème est contenue dans les sources de connaissances. Chacune d'elles correspond à un domaine de compétence bien précis dont

l'objectif est de coopérer à la résolution du problème. Les actions d'une source de connaissances ont pour conséquence la modification, la destruction ou la création d'éléments de solution dans le blackboard.

Les sources de connaissances sont indépendantes et n'ont aucun lien les unes avec les autres. Elles ne communiquent que par l'intermédiaire de la base de données commune, le blackboard. Elles sont coopérantes dans le sens où chacune d'elles apporte un élément de solution. Les communications entre les sources de connaissances se font exclusivement par lecture ou écriture dans le blackboard. Chacune d'elles renferme une parcelle de la connaissance nécessaire à la résolution du problème. L'objectif est de faire coopérer l'ensemble des compétences et aboutir à la résolution du problème.

Une source de connaissance se compose de trois parties (Figure V- 5) :

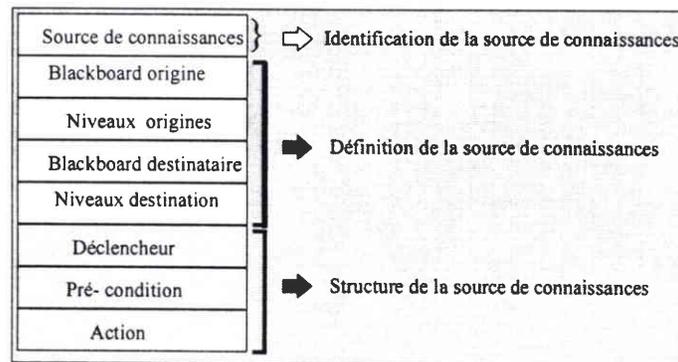


Figure V-5 : Architecture d'une source de connaissance

- **Partie identification** : Elle permet de nommer la source de connaissances, donc de l'identifier.
- **Partie définition** : Elle permet de définir les conditions nécessaires à l'application de la source de connaissances ainsi que l'action à exécuter.

On y retrouve des informations concernant :

- **Le blackboard d'entrée** dans lequel la source de connaissances va rechercher l'information événementielle. Cet événement est appelé "déclencheur de la source de connaissances".
- **Les blackboards destinataires** dans lesquels la source de connaissances exécute les différentes actions définies par les conclusions de ses règles.
- **Les niveaux origines** du blackboard d'entrée, à partir desquels la source de connaissances peut se déclencher. Ceux-ci sont définis comme les niveaux de son déclencheur. Le déclencheur d'une source de connaissances ne se réfère qu'à un seul événement, mais en fonction du contexte celui-ci peut appartenir à des niveaux différents. En conséquence, la source de connaissances se déclenchera dès l'apparition d'un événement déclencheur actif à l'un de ses niveaux origines.
- **Les niveaux de destination** des blackboards destinataires, dans lesquels la source de connaissances applique ses actions.
- **Partie structure** : Elle est utilisée pour définir le cadre d'utilisation de la source de connaissances.

Celle-ci a un format de type : "**condition - action**"

- **La condition** est divisée en deux sous parties dans lesquelles on trouve :

Un déclencheur, représenté par un prédicat basé sur un événement qui permet de tester la pertinence de la source de connaissances. Le test du déclencheur ne concerne que l'apparition de nouveaux événements dans le blackboard. **Dès lors que le déclencheur est VRAI, la source de connaissance est dite "déclenchable"**. L'événement qui a provoqué la valeur VRAI du prédicat est appelé déclencheur de la source de connaissances.

La pré-condition d'une source de connaissances est représentée par un prédicat qui teste la faisabilité de la source de connaissances. Ce prédicat peut concerner l'ensemble des événements des blackboards du domaine et éventuellement du contrôle, dans le cas d'un contrôle opportuniste. **Dès lors que la pré-condition est vraie, la source de connaissance est dite "activable"**. Les événements qui ont rendu vraie la pré-condition de la source de connaissances sont appelés "événement activateurs".

- **L'action**, qui se compose de procédures ou d'une liste de règles de production. Une procédure peut être un appel à une fonction externe écrite dans un langage quelconque. Une règle de production est formulée dans un format "prémisse-conclusion". Dès lors que la prémisse est réalisée, son activation entraîne la réalisation de la conclusion, qui se traduit par l'exécution d'une action dans le blackboard.

V-3.3 Blackboards

Ceux-ci sont des structures globales de données permettant de stocker de façon organisée les différents faits ou hypothèses (appelés aussi noeuds), émis lors du processus de résolution du problème. Un blackboard est découpé en différentes zones appelées niveaux qui sont propres à chaque application et pour lesquelles des fonctions de manipulation des hypothèses sont automatiquement définies. Il peut être considéré comme une mémoire temporaire de travail. A chaque instant, il reflète l'état d'avancement de la solution.

Un noeud peut être considéré comme une classe, et les hypothèses constitués les instances de cette classe. En général, les éléments sont représentés à partir d'objets reliés entre eux par des liens de hiérarchie ou de dépendance. Le blackboard peut alors être considéré comme un réseau d'hypothèses. Chacune de ces hypothèses participe à l'activation de certaines sources de connaissances qui agissent sur les différents niveaux du blackboard.

Les niveaux hiérarchiques du blackboard correspondent aux différents niveaux d'abstraction de l'analyse du problème. Toutefois, la grande majorité des systèmes existants utilisent un autre axe de structuration (horizontal), dont la dimension dépend du niveau hiérarchique considéré (temps, position).

Le blackboard d'un domaine enregistre les différents éléments de solution représentés par des événements. Dans le modèle du blackboard, la notion d'événement est inhérente. Il mémorise ce qui s'est passé dans le blackboard mais ne contient pas d'informations détaillées sur le contenu de ce dernier. Les événements sont utilisés par le mécanisme de contrôle pour détecter et gérer l'activation des sources de connaissances intéressées par les dernières mises à jour de la solution courante. Ils permettent ainsi au contrôleur d'éviter un parcours combinatoire du blackboard et d'analyser uniquement ce qui s'est passé dans celui-ci.

Un événement est une structure qui regroupe deux types d'informations :

- **les hypothèses**, qui représentent les éléments de solution créés par les différentes sources de connaissances.

- **les événements à déclencher**, qui informent sur les actions menées sur le blackboard. Par exemple un événement peut indiquer le niveau du blackboard dans lequel un changement a été effectué, le nom de l'hypothèse sur lequel a porté le changement, le type de changement (modification, création, suppression d'une hypothèse) et sur le contexte dans lequel il a été généré (la source de connaissances responsable de l'action, la date d'exécution de l'action).

Un élément de solution est donc une structure complexe composée de plusieurs champs (Figure V-6). Cette structure enregistre l'hypothèse générée qui peut être définie par des champs tels que : fait, contrainte, ... , indique le contexte de création défini par des champs tels que : créateur, cycle, ..., et établit des relations de dépendance entre les différents événements par la notion de champ père, champ fils ou autres. Chaque structure indique en outre le niveau auquel l'événement est enregistré.

On note que le nombre et la définition de ces différents champs sont laissés à l'initiative du concepteur. C'est à dire que le champ peut être lui-même un objet structuré dont la définition de ses champs internes est fonction de l'application.

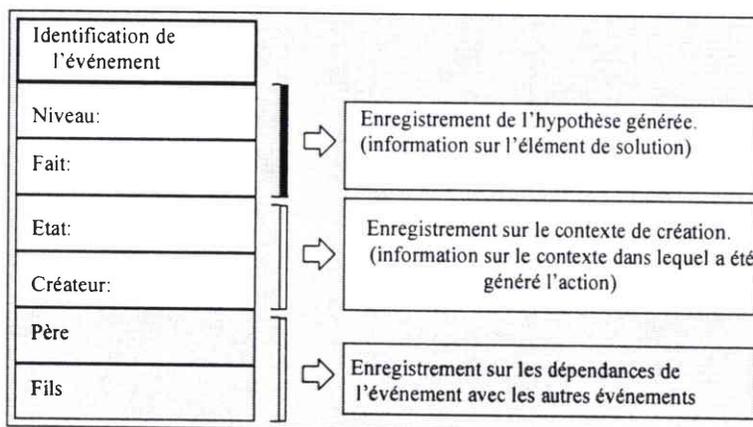


Figure V-6 : Exemple de structure d'un événement

Ainsi, un blackboard est composé d'un ensemble d'événements présentés de manière hiérarchique et structurés par niveaux d'abstraction (Figure V- 7).

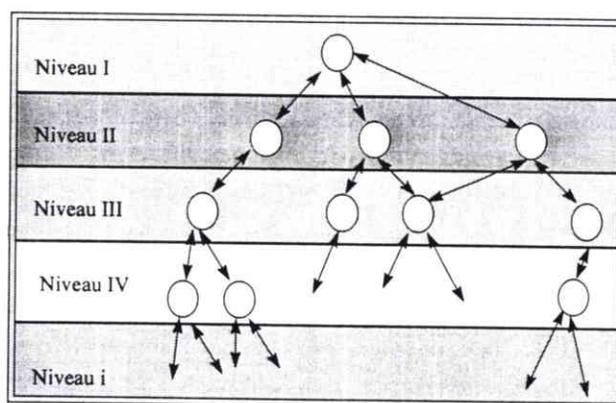


Figure V-7 : Structure d'un blackboard

Lors de la résolution d'un problème, tous les événements seront stockés dans un niveau d'abstraction précis et auront une structure dépendante de ce niveau. Développer une

application consiste à créer des instances de structures d'événement en affectant à chaque champ la valeur qui convient.

V-3.4 Contrôle

Le contrôle (Figure V-8) est la composante clef d'un système de blackboard. Les caractéristiques générales du modèle blackboard ont donné lieu à différentes implantations dont la différence essentielle réside dans le type de contrôle utilisé. **En effet, l'activité des sources de connaissances est dirigée par les événements, qui représentent les évolutions intervenant dans le blackboard.** Chaque changement qui intervient dans le blackboard constitue un événement qui, en la présence d'autres informations du blackboard, peut satisfaire la condition d'une ou plusieurs sources de connaissances. Etant donné que plusieurs sources de connaissances peuvent avoir leurs conditions satisfaites simultanément et se retrouver en compétition pour exécuter leurs actions, il est nécessaire de gérer leur activation. **Le mécanisme de gestion de l'activation des sources de connaissances est appelé contrôle.** Celui-ci a pour objectif de régler les conflits entre les différentes sources de connaissances activables, et d'en sélectionner une en fonction de critères prédéfinis. Le problème du contrôle peut être exprimé de la manière suivante [Maitre 89] : A chaque instant du processus de résolution du problème, quelle action le système doit-il exécuter. **Il permet ainsi d'exprimer explicitement des stratégies de résolution du problème.**

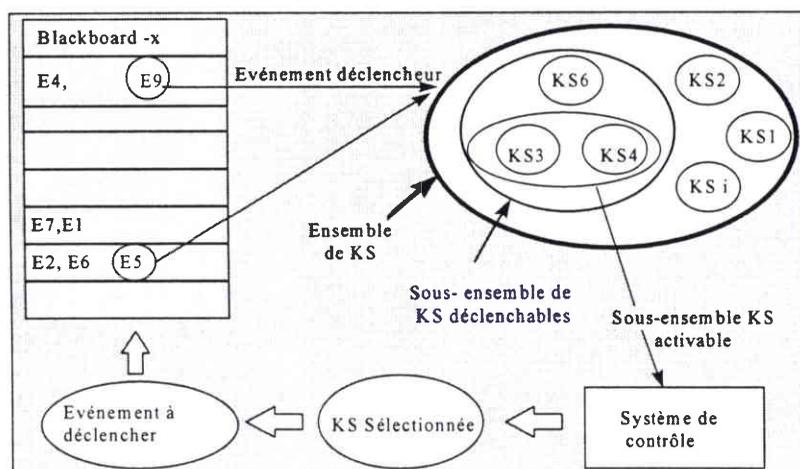


Figure V-8 : Système de contrôle

La gestion du fonctionnement global du système nécessite la coordination de la coopération entre les différentes sources de connaissances. Pour ce faire, le système contrôle l'évolution de la solution courante mémorisée dans le blackboard et oriente les activités du système. En général, les informations présentes dans le blackboard intéressent plusieurs sources de connaissances et génèrent ainsi une source de conflits.

Parmi les sources de conflits on peut citer :

- Les cas où plusieurs sources de connaissances veulent travailler sur les mêmes données (Figure V-9).

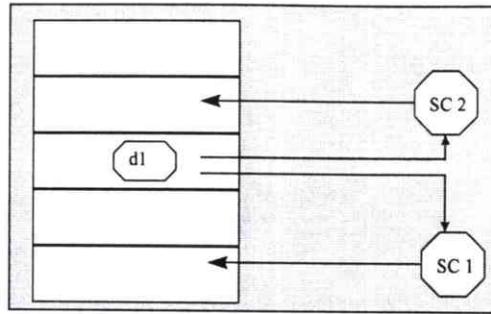


Figure V-9 : Premier cas d'une source de conflit

Les sources de connaissances SC1 et SC2 sont en conflit car elles sont intéressées par la même donnée pour générer une action différente.

- Les cas où plusieurs données rentrent en compétition. Les sources de connaissances SC1 et SC2 sont en conflit (Figure V-10), car elles traitent deux données compétitives. En effet, le traitement d'une donnée peut conduire à annuler ou à améliorer les effets du traitement de la seconde :

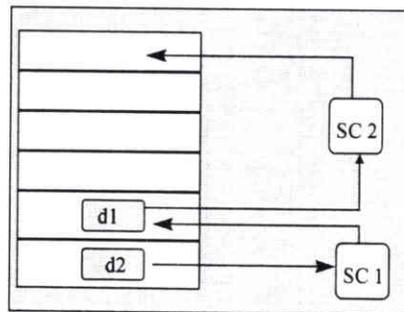


Figure V-10 : Deuxième cas d'une source de conflit

La qualité de la solution finale, c'est-à-dire l'état final du blackboard, et le temps de résolution du problème dépendent du nombre de données à traiter et de l'ordre dans lequel elles sont traitées.

V-3.4.1 Résolution de conflits

La résolution de conflits entre plusieurs sources de connaissances peut être traitée par deux approches différentes :

- **Approche par AGENDAS (HEARSAY-II, BB-1) :**

A chaque source de connaissances est associée des paramètres tels que : efficacité, fiabilité, coût, etc. Ces paramètres sont généralement dépendants de la configuration du blackboard et sont, par conséquent, susceptibles d'évoluer en fonction de l'état d'avancement de la résolution du problème. A chaque cycle de résolution du problème, le mécanisme de contrôle détecte les sources de connaissances activables. Ensuite, il crée des instanciations de sources de connaissances et associe ainsi à chaque source de connaissances les changements du blackboard. Les instanciations de source de connaissances sont placées dans une liste d'attente appelée AGENDA. La sélection de la meilleure source de connaissances est réalisée à partir d'un ensemble d'heuristiques.

- Approche par META-SOURCES DE CONNAISSANCES (HASP/SIAP, CRYNALIS) :

Une méta-source de connaissances se charge de coordonner les activités d'un sous-ensemble de sources de connaissances et ainsi lui fournit un contrôle local. Pour gérer l'ensemble des méta-connaissances, il est fait appel à une source de méta-connaissances qui fournit alors un contrôle global.

V-3.4.2 Focalisation du contrôle

A chaque instant du processus de résolution d'un problème, le système dispose en général de plusieurs actions en compétition. Le résultat du choix d'une action est appelé focalisation de contrôle ou focalisation d'attention.

Si la prochaine action que doit effectuer le mécanisme de contrôle consiste à sélectionner la prochaine région (niveau) du blackboard à traiter, alors la focalisation de contrôle est une région (HEARSAY-II, BB-1).

Si la prochaine action que doit effectuer le mécanisme de contrôle consiste à sélectionner la prochaine source de connaissances, alors la focalisation de contrôle est une source de connaissances (HASP/SIAP).

La focalisation de contrôle peut être également la prochaine région du blackboard à traiter et la prochaine source de connaissances à activer (CRYNALIS, ATOME).

V-3.4.3 Représentation du contrôle

La représentation du contrôle désigne l'ensemble des structures de contrôle ainsi que les modules de contrôle qui opèrent sur ces structures pour gérer la conduite du système.

Il existe deux types de représentation :

- **Représentation implicite du contrôle** (HEARSAY-II) qui permet la sélection d'une instance de source de connaissances à activer à partir d'un agenda d'instances de sources de connaissances activables, grâce à un ensemble d'heuristiques. Ces heuristiques prennent en compte l'état actuel du blackboard, les niveaux d'entrée et de sortie de chaque instance de source de connaissances, ainsi que les résultats attendus pour chacune d'elles [Haton 91].

- **Représentation explicite du contrôle** (HASP/SIAP, CRYNALIS, BB-1, ATOME) qui permet la sélection d'une instance de source de connaissances à activer à partir de stratégies spécifiées explicitement lors de la mise en oeuvre du système. Dans HASP/SIAP, CRYNALIS et ATOME, chaque source de connaissances de contrôle est chargée de coordonner directement les activités d'un sous-ensemble des sources de connaissances du domaine. Dans BB-1, les sources de contrôle opèrent sur un second blackboard, appelé blackboard de contrôle, afin de gérer d'une façon indirecte les activités de tout l'ensemble des sources de connaissances du domaine.

Une représentation implicite du contrôle ne permet pas une évolution ou une maintenance aisée du contrôle. En effet, la stratégie de sélection est prédéfinie lors de l'écriture de l'algorithme et ne permet pas de s'adapter à une situation non prévue. Cette contrainte rend ce mode de représentation totalement dépendant de l'application sans possibilité de s'adapter à l'état courant de la solution.

Une représentation explicite du contrôle présente l'avantage d'être modulaire, facile à faire évoluer et à maintenir. En effet, les connaissances liées au contrôle peuvent être mises à jour sans remettre en cause systématiquement les connaissances liées au domaine et vice-versa.

V-3.4.4 Modes de contrôle procédural

Le mode de contrôle procédural (Figure V-11) fait appel à une représentation implicite. Il consiste à intégrer toutes les connaissances du contrôle dans un programme. Ce programme est un ordonnanceur (Figure V-12) dont le rôle est de sélectionner la prochaine action à exécuter en fonction de l'état global du blackboard et des heuristiques de contrôle.

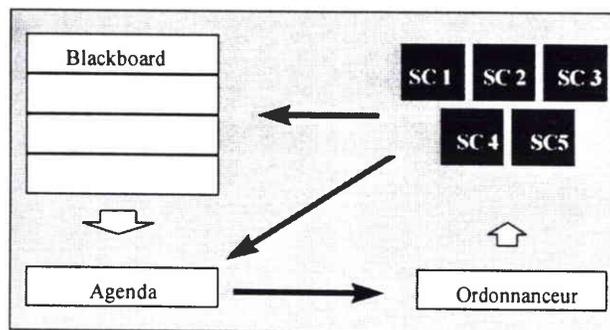


Figure V-11 : Contrôle procédural

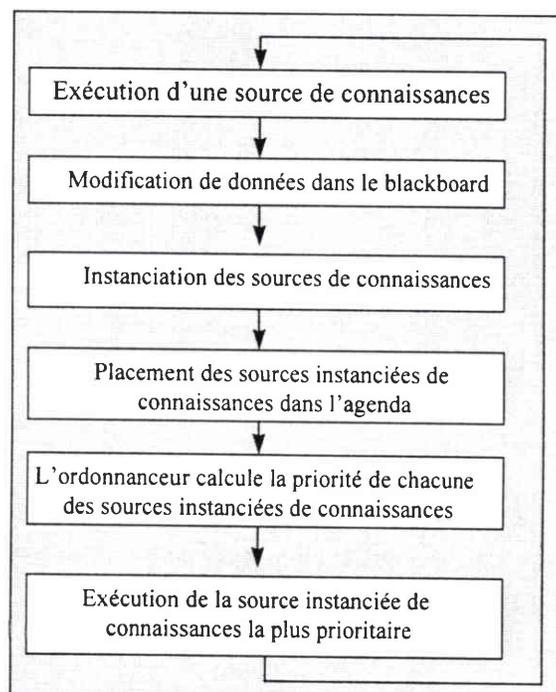


Figure V-12 : Cycle de base d'un mécanisme de contrôle de type procédural

Le contrôle procédural a été implanté pour la première fois dans le système HEARSAY-II utilisé pour la reconnaissance et la compréhension de la parole. Le blackboard est divisé en sept niveaux d'abstraction : signal paramètre, segment, syllabe, mot, séquence de mots, phrase et interface de base de données.

V-3.4.5 Contrôle hiérarchique

Dans un contrôle hiérarchique toutes les sources de connaissances, qu'elles soient du domaine ou du contrôle, sont représentées sous forme de sources de connaissances organisées en plusieurs niveaux hiérarchiques (Figure V-13). Le niveau le plus bas de la hiérarchie est constitué par les sources de connaissances du domaine. Le niveau le plus haut est constitué d'une seule source de contrôle. Les niveaux intermédiaires sont constitués de sources de connaissances de contrôle spécifique à chaque sous domaines.

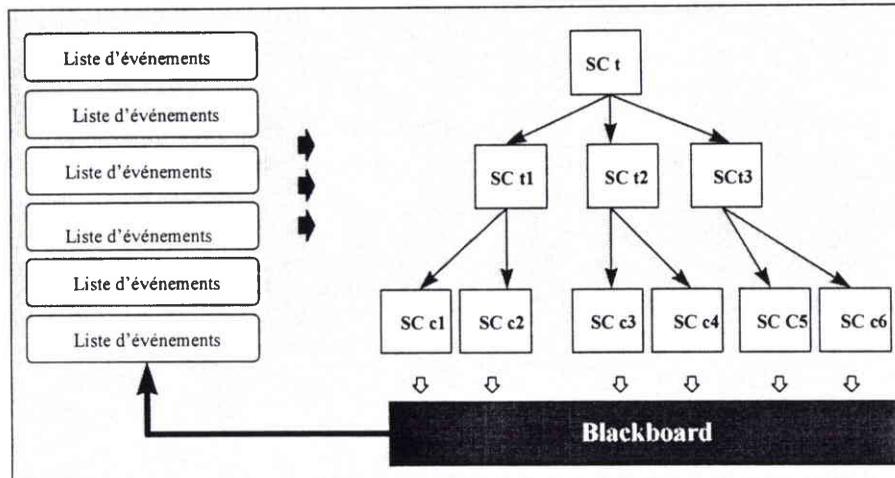


Figure V-13 : Contrôle hiérarchique

Le découpage hiérarchique du blackboard suivant différents niveaux de méta-connaissance augmente la modularité. Néanmoins, la hiérarchie établie impose de connaître un ordre de séquençement des sources de connaissances, ce qui confie une certaine rigidité au système. Ce mode de contrôle ne peut être utilisé que dans les applications où l'ordre d'activation des sources de connaissances peut être prédéterminé.

Parmi les différents types d'application, on peut citer le système HASP/SIAP dont l'objectif est d'interpréter des signaux sous-marins afin de reconnaître les sources de bruit les ayant engendrés et d'en déduire ainsi les navires présents dans la zone sous surveillance. Le blackboard est composé de six niveaux : segments, raies, harmoniques, sources émettrices de bruits, bâtiments et flottes.

Le mécanisme de contrôle est composé de quatre structures de contrôles manipulées par plusieurs modules de contrôle organisés en deux niveaux : le niveau stratégie et le niveau des gestionnaires de sources de connaissances.

Les quatre structures de contrôle sont des listes dont les éléments sont créés par des actions des sources de connaissances dans le blackboard. Les quatre listes sont :

- Liste d'événements, qui recueille tous les changements du blackboard résultant des actions des sources de connaissances.
- Liste d'événements attendus, qui répertorie l'ensemble des événements attendus dans le blackboard. En effet, le système dispose de connaissances a priori au sujet des activités prévues dans la région surveillée.
- Liste de problèmes, qui mémorise les types de problèmes rencontrés par certaines sources de connaissances durant leur activation.

- Liste d'événements temporels, mémorise les événements temporels composés d'une date, d'un ensemble de sources de connaissances et d'un contexte de travail. Les sources de connaissances énumérées seront exécutées à la date et dans le contexte spécifié.

Les modules de contrôle sont disposés sur deux niveaux :

- Au niveau le plus élevé Figure un seul module de contrôle «la stratégie", qui sélectionne en fonction du contenu des quatre listes définies auparavant, une gestion de niveau inférieur.
- Au niveau le plus bas, Figurent les gestionnaires des sources de connaissances.

Le cycle de contrôle est le suivant (Figure V-14) :

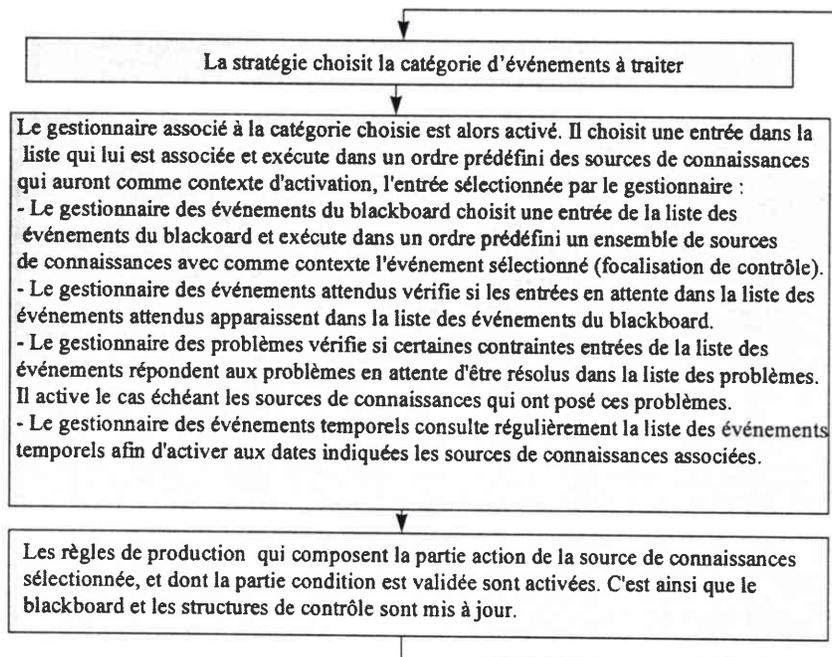


Figure V-14 : Cycle de base de contrôle du système HASP/SIAP

V-3.4.6 Mode de contrôle opportuniste

Le mode de contrôle opportuniste est mis en oeuvre par un ensemble de sources de connaissances ayant comme domaine d'application le contrôle et agissant sur un blackboard de contrôle. Le rôle des sources de connaissances de contrôle est de construire et d'adapter le plan de commande à suivre pour gérer le système. Les sources de connaissances, que celles-ci soient du domaine ou du contrôle, sont toutes concourantes et traitées par un même ordonnanceur (Figure V-15). Ce mode de contrôle permet d'exprimer explicitement des heuristiques de résolution. Lors de la résolution d'un problème, les heuristiques de contrôle s'adaptent dynamiquement à l'état courant de la solution. L'uniformité des représentations de connaissances lui confère une grande clarté, par contre elle peut s'avérer trop complexe pour des applications simples. Ce mode de contrôle peut être efficace lors de la phase de développement d'une application.

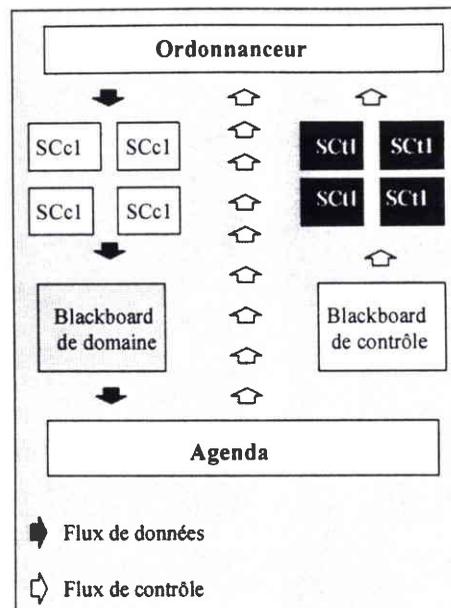


Figure V-15 : Contrôle de mode opportuniste

Parmi les différents types d'applications, on peut citer le système BB-1 qui est un outil d'aide au développement de systèmes à multi-sources de connaissances développé au KSL (Knowledge Systems Laboratory) [Austin 62].

Le blackboard du domaine est défini par le concepteur de l'application et reste directement lié à l'application. Les différents niveaux du blackboard de contrôle sont prédéfinis et restent indépendants de l'application en cours de réalisation. Le nombre de niveaux dépend de la complexité de l'application.

On peut citer trois niveaux de base :

- Le niveau stratégie qui contient différentes stratégies compétitives, complémentaires ou indépendantes.
- Le niveau objectif qui mémorise les différents objectifs qui implantent et réalisent une ou plusieurs stratégies. Généralement, une stratégie est composée d'une séquence d'objectifs. Un objectif peut apparaître dans plusieurs stratégies.
- Le niveau heuristique qui enregistre les heuristiques utilisées pour décrire la façon dont les différents coefficients, associés à une action, doivent être évalués ou combinés. Ces coefficients caractérisent l'action (ex: coût, fiabilité).

En fonction de la complexité de l'application, on peut rajouter des niveaux tels que : tactique, sous-stratégie.

On distingue deux types de sources de connaissances de contrôle :

- **génériques**, prédéfinies et donc indépendantes de l'application. Elles permettent l'implantation d'une stratégie donnée en une suite d'objectifs.
- **spécifiques**, à définir par le concepteur et donc dépendantes de l'application. Elles permettent de prendre des décisions de contrôle adaptées au domaine d'application en fonction de l'évolution de la solution courante.

Le mécanisme de contrôle utilise quatre agendas des instances de sources de connaissances de contrôle qu'il met régulièrement à jour.

Les instances de sources de connaissances sont partagées en fonction de leur état qui peut être :

- "sélectionnées",
- "exécutables",
- "rejetées",
- "exécutées".

Seuls les agendas des sources sélectionnées et exécutables sont pris en compte par le processus de résolution du plan de contrôle. Les agendas des sources rejetées et exécutées sont utilisés par le mécanisme explicatif.

Le cycle de contrôle doit être exécuté jusqu'à ce que les conditions d'arrêt soient rencontrées ou que l'agenda des exécutables soit vide (Figure V-16).

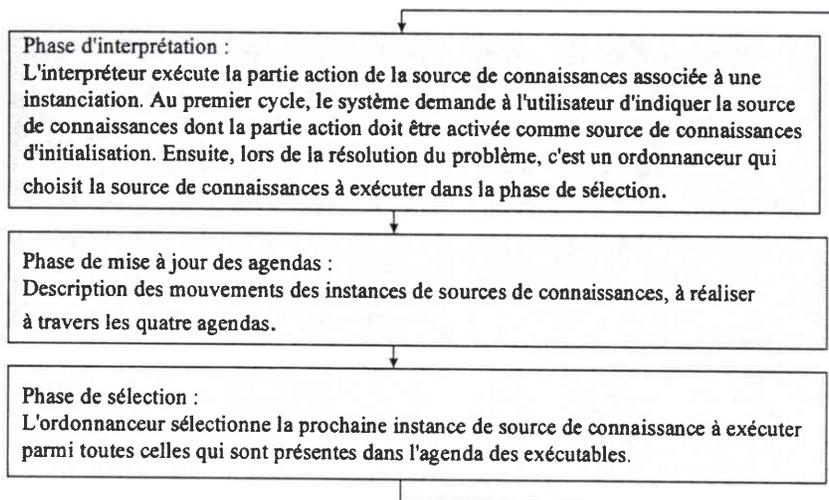


Figure V-16 : Cycle de base du système BB-1

V-3.4.7 Critères comparatifs

Les différents types de contrôle dans les architectures de type blackboard peuvent être classés à partir de quatre critères fondamentaux :

- Efficacité du système : Un système fondé sur le modèle du blackboard est efficace lorsqu'il est capable de sélectionner rapidement la ou les prochaines actions à exécuter. Dans le cas du contrôle hiérarchique, les séquences d'actions sont prédéfinies et par conséquent ne nécessitent aucun calcul de priorité, donc le système peut être considéré comme efficace.
- Uniformité du système : Un système est uniforme lorsqu'il traite le problème de contrôle au même niveau et de la même manière que celui du domaine. Dans le cas du contrôle opportuniste, les sources de connaissances du domaine et du contrôle utilisent le même formalisme et sont manipulées par un système indépendant.
- Souplesse du système : Un système est souple lorsqu'il est facilement adaptable à l'évolution du domaine, c'est à dire que des composantes peuvent facilement lui être ajoutés et que le contrôle s'adapte à l'évolution de la solution du problème courant. Ce compromis existe dans un contrôle de type opportuniste.

- Clarté du système : La clarté d'un système intéresse l'architecture globale du système par une représentation simple des stratégies et de l'organisation des connaissances. Les architectures à base de contrôle hiérarchique fournissent une représentation structurée des connaissances et des stratégies faciles à formaliser.

V-3.5 Systèmes multi-experts existants

Les caractéristiques générales du modèle blackboard ont donné lieu à différentes implantations, parmi lesquelles on peut citer :

- le système HEARSAY-II [Erman *et coll.* 80] a été développé à l'université de Carnegie Mellon dans le cadre de la conception de parole, conception d'image, conception de signal.
- le système VISION [Hayes 87] a été développé à l'Université du Massachusetts dans le cadre du traitement d'images naturelles,
- le système RC2 [Gallagher 88] a été développé aux laboratoires de Marcoussis CGE dans le cadre de la reconnaissance de la parole.
- le système OPM [Hayes 79] a été développé à la université de STANFORD. Le système OPM a été appliqué au domaine de la détermination de la structure des protéines et dans le domaine génie civil.

A cela, il faut également citer le développement de générateurs de systèmes multi-experts. Les premières architectures ont été développées à partir de la généralisation d'applications.

On peut citer :

- Hearsay-III, issu du système Hearsay-II,
- BB1, issu du système OPM,
- GBB, issu du système VISION.

Par ailleurs, sont apparus en Europe:

- ATOME, développé au CRAN à NANCY,
- BLONDIE, développé à Netherlands PTT,
- TRAM, développé au LETI - CEA.

Le Tableau V- 2 récapitule les caractéristiques de quelques systèmes existants :

	Hearsay III	AGE	BB1	GBB	ATOME
Université d'origine	Carnegie Mellon	Stanford	Stanford	Massachusetts Amherst	Nancy CRIN- INRIA
Langage	AP3	Inter Lisp	Inter Lisp CommonLisp	Common Lisp	Common Lisp C++
Blackboard	2	1	2 et plus	plusieurs	plusieurs
KS condition	prédicats	système règles	Trigger pré- condition	Prédicats	Trigger pré-condition
KS action	Fonction quelconque	système règles	système règles	Fonction	Fonction regles
Contrôle	Opportuniste	Hiérarchique	Opportuniste	Opportuniste	Hiérarchique

Tableau V- 2 : Récapitulatif des caractéristiques de quelques systèmes existants

V-4 Choix d'un outil d'implantation

L'approche proposée dans SYCORO :

- fait appel à une grande quantité de connaissances issues, pour l'essentiel, des expériences passées de l'expert donc de nature heuristique. Il est beaucoup plus question de **savoir-faire** que de méthodologie rigoureuse le savoir-faire structuré par famille
- comporte net aspect de **multi-expertises**. En effet, Le pilotage est partagé en trois sous-problèmes "Affectation Ressources" "affectation de transferts et "suivi" et chacun sous-problèmes fait appel à un nombre de spécialistes.
- fait apparaître une **démarche de l'expert non linéaire**. C'est-à-dire qu'à tout moment certaines parties du raisonnement peuvent être remises en cause et nécessiter des retours arrières.

De part les critères énoncés ci-dessus, nous avons opté pour la coopération de **plusieurs agents "intelligents"**. Cette coopération est réalisée par le partage d'informations à travers **plusieurs zones de travail** communes aux différents agents. Pour mener à bien ce type de coopération, il est nécessaire de faire appel à un mécanisme de contrôle capable de gérer de manière efficace et souple l'interaction entre ces agents. Nous avons opté pour une **structure de contrôle hiérarchique**, caractérisée par un ensemble de sources de connaissances de contrôle hiérarchisées.

Les caractéristiques choisies : agents intelligents, zones blackboards et contrôle hiérarchique offrent une bonne synthèse entre une approche distribuée et une approche hiérarchique. Elles permettent la mise en place d'une architecture modulaire et flexible, et de ce fait bien adaptée aux problèmes posés par la conduite réactive des ateliers de production telle qu'elle a été introduite dans le chapitre. En outre, elles permettent un fonctionnement asynchrone des modules et la possibilité d'une implantation parallèle.

Parmi les outils énoncés dans le Tableau V-2 récapitulatif, il ressort que le système ATOME offre des caractéristiques intéressantes pour l'implantation d'une architecture capable de supporter et de valider l'approche proposée, en particulier la possibilité de manipuler plusieurs blackboards.

V-5 Présentation de l'outil ATOME (Another TOol for Multi-Expertise)

Le système ATOME est un environnement d'aide au développement de système à multi-sources de connaissances et repose sur le modèle du blackboard [Laasri 89]. En tant qu'outil d'aide au développement de système d'intelligence artificielle à multi-sources de connaissances, il fournit un ensemble de directives permettant de représenter et d'organiser les données à manipuler ainsi que de définir l'ensemble des sources de connaissances intervenant dans le processus de résolution d'un problème particulier.

La séparation entre l'expertise de contrôle et l'expertise du domaine permet une modification plus aisée des différentes stratégies de résolution à mettre en oeuvre lors de la phase de mise au point. Dans ce but, ATOME fournit un **mécanisme de contrôle hiérarchique** composé de sources de connaissances de contrôle appelées "tâches", et d'une méta-source de connaissances de contrôle appelée "stratégie". Chacune de ces sources de connaissances est introduite sous forme d'une base de règles de production. ATOME, de ce fait, peut être considérée, en fonction de l'expertise qu'il renferme, comme "système expert de contrôle".

L'architecture du système de contrôle se compose de trois niveaux et intègre deux modes de raisonnement. Un premier niveau dit "stratégique" constitué d'une seule source de connaissances de contrôle. Un deuxième niveau inférieur dit "local" constitué d'autant de sources de connaissances de contrôle, appelées «tâches», que de sous domaines disponibles. Le troisième niveau regroupe l'ensemble des sources de connaissances, appelées «spécialistes». Chaque spécialiste est donc affectée à un sous domaine d'application relatif à un travail spécifique.

En résumé, ATOME dispose de trois types de sources de connaissances :

1. les spécialistes qui regroupent les connaissances liées au domaine,
2. les tâches qui fournissent un contrôle local en gérant les spécialistes,
3. la stratégie qui propose un contrôle global du système en gérant les tâches.

Les spécialistes sont les seules sources de connaissances qui accèdent aux blackboards. Les tâches disposent d'une structure de contrôle local composée de listes d'événements. A chaque tâche est associée une liste d'événements. La stratégie accède à une seconde structure de contrôle constituée par des résumés de blackboards. En conséquence, l'architecture d'ATOME utilise deux niveaux hiérarchiques de contrôle avec des possibilités de raisonnement guidés par les règles, guidés par les événements et opportunistes (Figure V-17).

Le fonctionnement global du contrôle peut être décrit à partir d'une boucle (Figure V-18) et d'un cycle (Figure V-19)

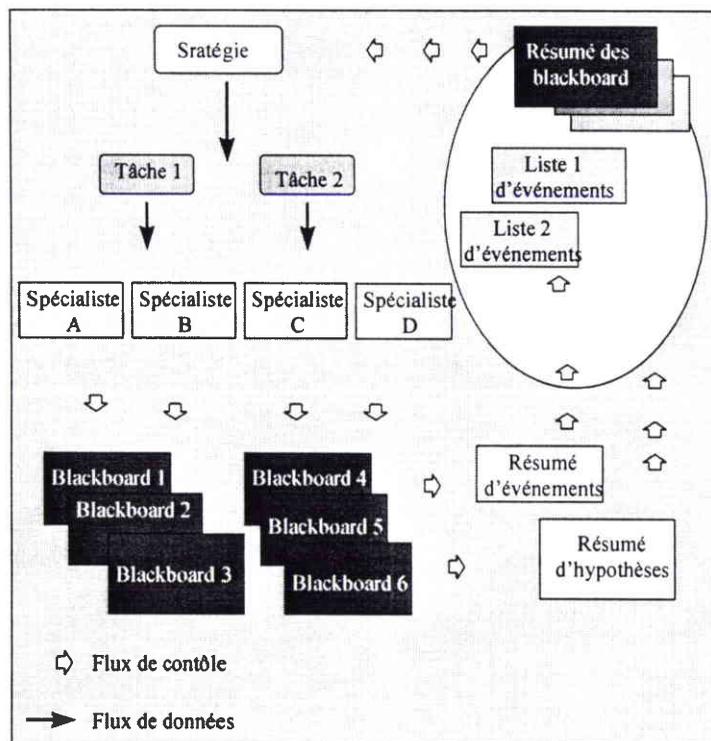


Figure V-17 : Architecture d'ATOME

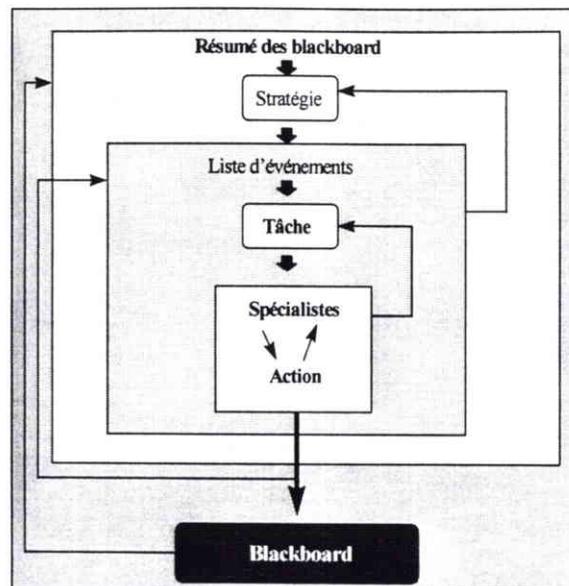


Figure V-18 : Boucle de contrôle de base

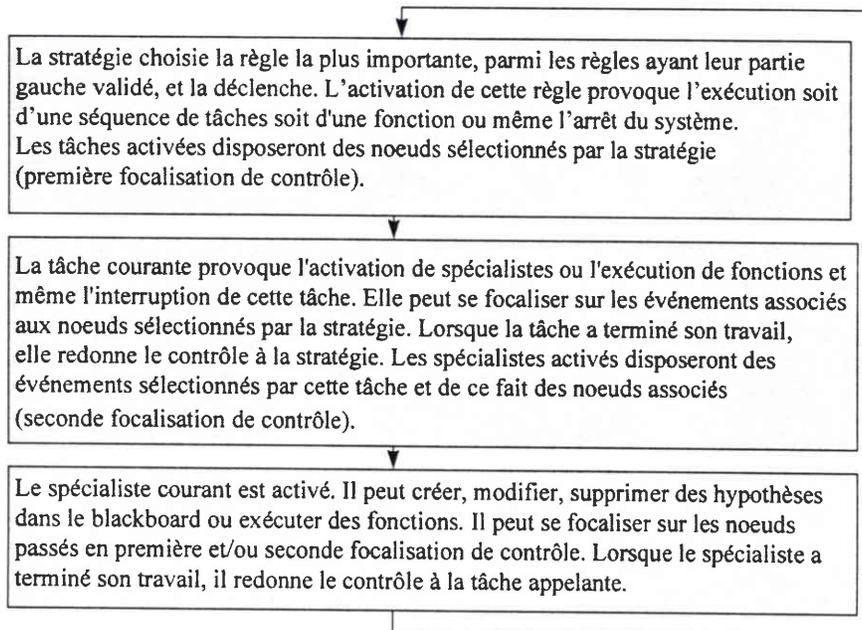


Figure V-19: Cycle de contrôle de base

V-5.1 Les blackboards

Le système ATOME permet l'utilisation de plusieurs blackboards dans le développement d'une application. Chaque noeud est constitué d'un ensemble d'attributs qui peut être de deux types :

- Un attribut de type valeur permet de stocker une ou plusieurs données. Il peut avoir plusieurs valeurs alternatives, provenant d'une ou plusieurs sources de connaissances et être affecté d'un coefficient de vraisemblances.
- Un attribut de type lien permet de créer des relations réciproques entre différents noeuds qu'ils soient de même nature ou non. Ceci, afin de maintenir la symétrie des relations entre

les hypothèses. Il est constitué d'une liste d'hypothèses qui sont en relation entre elles. Celles-ci peuvent être issues d'un même niveau ou être issues de niveaux différents.

Les actions de mise à jour d'un blackboard sont de quatre types :

- création d'une hypothèse dans un niveau du blackboard,
- suppression d'une hypothèse du blackboard,
- modification d'attributs ou de liens d'une hypothèse du blackboard avec conservation de leurs anciennes valeurs,
- remplacement d'attributs ou de liens d'une hypothèse du blackboard en écrasant leurs anciennes valeurs.

Les filtres permettant de construire les résumés des blackboards relèvent de l'expertise d'une application donnée. Ils permettent de vérifier sous une forme déclarative les conditions sous lesquelles les hypothèses d'un certain niveau d'un blackboard peuvent être placées dans le résumé correspondant.

Parmi l'ensemble des hypothèses utilisées, certaines sont constamment évolutives et d'autres moins sollicitées voire plus utilisées. Cependant, elles ne doivent pas être supprimées dans la mesure où elles contiennent de l'information potentielle encore exploitable. Pour cela, il est fait appel à un mécanisme de gestion des hypothèses permettant de détecter les hypothèses anciennes et non productives des blackboards, et de les déplacer dans une mémoire à long terme. Chaque blackboard susceptible de contenir de telles hypothèses se voit attribuer une mémoire à long terme appelée "**blackboard long terme**". Chaque blackboard à long terme reste accessible en lecture par les spécialistes du système et ainsi l'information n'est pas perdue. Cependant ces accès doivent être formulés explicitement par les spécialistes.

Les critères d'identification du potentiel d'utilisation des hypothèses (Figure V-20) sont à définir par l'utilisateur et sont représentés par :

- le cycle à partir duquel les opérations de nettoyage seront lancées,
- la période (nombre de cycles) d'exécution de l'opération de nettoyage,
- le nombre de cycles durant lesquels une hypothèse n'a pas été traitée.

Identificateur de blackboard	Début du cycle	Période	Durée non manipulée
Image	20	5	10

Figure V-20 : Exemple de critères d'identification du blackboard "Image".

V-5.2 Événements

Les événements représentent des messages envoyés au mécanisme de contrôle par les spécialistes pour le prévenir de leurs actions dans les blackboards. De part ses connaissances et le contexte de la solution courante, le spécialiste est le seul capable de juger l'importance de ses actions et donc de décider s'il est nécessaire d'alerter le mécanisme de contrôle en engendrant des événements. En effet, un même type d'action dans un blackboard peut engendrer, ou non, un événement selon le spécialiste à l'origine de cette action et selon le contexte ayant conduit cette action. Dès lors que l'action est jugée digne d'intérêt pour le

mécanisme de contrôle, le spécialiste lui associe un coefficient d'importance permettant de le situer par rapport à un classement dans la liste des événements.

Le calcul de ce coefficient est exprimé sous une forme d'importance relative telle que :

- l'action qui vient d'être effectuée est plus importante que toute autre action de même niveau,
- l'action qui vient d'être effectuée est plus importante que toute autre action de niveau X,
- de part l'action effectuée, l'événement doit être classé à la plus grande priorité,
- de part l'action effectuée, l'événement doit être classé à la plus petite priorité.

Les types d'action qui engendrent un événement sont relatifs à la **création**, à la **modification** au **remplacement** ou à la **suppression** d'une hypothèse. Toutefois, l'action qui consiste à supprimer une hypothèse doit être traitée différemment des autres actions. Le problème lié à la création d'événements pour des suppressions d'hypothèses concerne la focalisation de contrôle. En effet, lorsqu'un événement quelconque permet l'activation d'un spécialiste, l'hypothèse associée à cet événement est passée en seconde focalisation de contrôle au spécialiste. Cependant, lorsqu'une suppression d'hypothèse est demandée dans un blackboard, l'hypothèse n'est pas seulement retirée du blackboard, elle est aussi détruite.

En conséquence, lorsque la suppression d'une hypothèse est demandée :

- l'hypothèse est retirée du blackboard,
- tous les événements associés à cette hypothèse sont retirés de la liste d'événements,
- l'hypothèse n'est détruite physiquement que si sa suppression n'engendre aucun événement ou lorsque l'événement engendré est traité.

Un événement est caractérisé par (Figure V-21) :

- le niveau du blackboard dans lequel l'action est effectuée,
- le noeud ou hypothèse sur lequel a porté le changement,
- le type de l'action effectué (création, modification, remplacement),
- la liste des attributs valeurs mis à jour qui ont été créés ou modifiés,
- la liste des attributs liens mis à jour qui ont été créés ou modifiés.

<p>Nom : Identification de l'événement.</p> <p>Niveau : Niveau de l'hypothèse sur laquelle porte les deux événements synthétisés.</p> <p>Noeud : Identification de l'hypothèse.</p> <p>Type : Type d'action concerné : soit modification soit remplacement.</p> <p>Attribut valeur : Regroupe l'ensemble des attributs valeurs des deux événements synthétisés.</p> <p>Attribut lien : Regroupe l'ensemble des attributs liens des deux événements synthétisés.</p>

Figure V-21 : Structure d'un événement

Une liste d'événements participe à la représentation de ce qui s'est passé dans les blackboards ou de ce qui n'a pas encore été traité. A chaque tâche est associée une liste d'événements à laquelle sont associés deux niveaux de filtrage d'événements.

- Un premier filtre est utilisé lors de la création d'événements par les spécialistes, car ne sont créés que les événements jugés nécessaires à la tâche. Un même événement successible d'intéresser plusieurs tâches est créé dans toutes les listes d'événements correspondantes. Pour cela, le système fait appel à un module de propagation (Figure V-22) des événements vers les tâches, dont le rôle est de filtrer les événements envoyés par les spécialistes afin de les propager vers les listes associées aux tâches intéressées. Le filtrage est effectué à partir des connaissances fournies par les tâches quant aux types d'informations susceptibles de les intéresser.

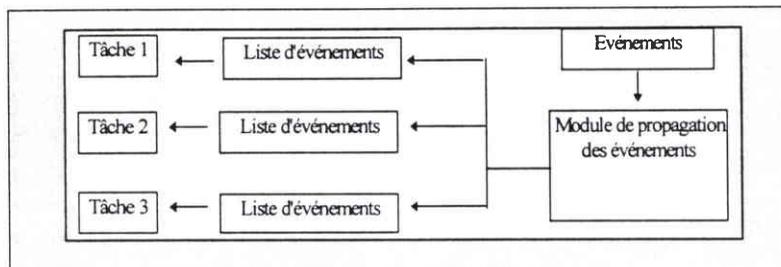


Figure V-22 : Gestion des événements

- Un deuxième filtre est utilisé lors de l'exploitation de la liste par la tâche, car la tâche sélectionne en priorité les événements pertinents.

Un événement utilisé lors de l'activation d'une tâche est retiré de la liste d'événements qui lui est associée.

Dans une même liste d'événements, il peut exister deux événements semblables. Pour traiter ce problème, le système utilise la notion de synthèse d'événements d'une même liste d'événements. Lorsqu'un événement est placé dans la liste d'événements d'une tâche, le système compare cet événement à ceux déjà existants dans cette liste. Il recherche si ce dernier peut ou non être synthétisé avec l'un d'entre eux.

Pour cela, il vérifie si les deux événements :

- concernent la même hypothèse,
- représentent le même type d'action.

Les actions intéressées sont la modification ou le remplacement. En effet, on ne crée ou ne supprime qu'une seule fois une hypothèse. Le problème de la synthèse d'événements ne se pose pas pour la création et la suppression d'événement.

L'événement qui résulte de la synthèse regroupe les différents attributs de chacun et est placé dans la liste au rang de l'élément le mieux classé (Voir Figure V-21).

Attention, le rôle d'une synthèse d'événements est de diminuer le nombre d'événements dans les listes d'événements locales aux tâches et ainsi de regrouper les informations faisant référence à des mêmes types d'actions sans perdre de l'information. C'est pourquoi une synthèse d'événements n'est pas possible entre deux événements se rapportant l'un à une modification, l'autre à un remplacement d'une même hypothèse, car ces deux actions n'ont pas

le même comportement. La modification conserve les anciennes valeurs des attributs tandis que le remplacement écrase ces valeurs.

V-5.3 Les spécialistes

Les spécialistes désignent les sources de connaissances propres au domaine de l'application. Ce sont des modules indépendants, experts dans un sous-domaine particulier. Chaque spécialiste renferme la connaissance propre à apporter une solution à une partie du problème. Le rôle d'un spécialiste est de résoudre un sous problème particulier en fonction de l'état des blackboards. Pour cela, ils utilisent les faits déjà existants dans les blackboards, avec la possibilité d'en créer, d'en modifier ou d'en supprimer et ainsi d'apporter leur participation à la résolution du problème. Dans ATOME, ils peuvent être de natures très diverses (fonction C++, système expert ou base de règles).

Un spécialiste est caractérisé par (Figure V-23).

<p>Nom : Identificateur de spécialiste</p> <p>Variables locales : Dont le rôle est de permettre la spécification du filtre du spécialiste. A chaque variable locale est associée différentes fonctions de filtrage. On retrouve trois types de variables :</p> <ul style="list-style-type: none"> • Variables de contrôle : Ces variables sont définies dans le cas où l'on fait appel à des heuristiques (mode opportuniste règle d'intégration). Chaque variable est affectée d'une valeur qui peut être constante ou variable en fonction de l'évolution du contexte. • Variables de précondition ou de contexte : Ces variables sont liées lors de la phase de vérification de la pré-condition du spécialiste et restent accessibles lors de la phase d'activation. Elles sont donc manipulées aussi bien dans sa pré-condition que dans sa base de règles. • Variables d'action : Ces variables sont liées lors de la phase d'activation du spécialiste et sont manipulées dans la base de règles. <p>Pré-condition : Liste des conditions (tests) à vérifier par le spécialiste pour passer de l'état présélectionné à l'état activable.</p> <p>Base de règles de production : Qui constitue le corps du spécialiste,</p> <p>Mode de fonctionnement : Qui stipule le type d'interprétation de la base de règle qui peut être : simple, multiple ou cyclique.</p>
--

Figure V-23 : Caractéristiques d'un spécialiste

De part son expertise, un spécialiste est capable de spécifier à l'avance le type de données du blackboard qui l'intéresse. En effet, le spécialiste peut définir un filtre sur les hypothèses du blackboard qui ont provoqué son activation et qui lui sont, de ce fait, passées en focalisation d'attention par le mécanisme de contrôle. **La définition d'un filtre lui permet à chaque activation, de déterminer son contexte de travail.** Celui-ci est représenté par un sous-ensemble de noeuds du blackboard permettant au spécialiste d'orienter ses activités vers les régions de données qu'il doit traiter. La manipulation de tels spécialistes peut être décomposée en trois phases :

1 - Phase sélection :

Lorsque le mécanisme de contrôle sélectionne un ou plusieurs spécialistes afin de les activer, il leurs signale en fait que le contexte actuel leurs est propice. On dit que le mécanisme de contrôle "réveille" les spécialistes susceptibles d'être intéressés par les derniers changements

effectués dans les blackboards. C'est ensuite aux spécialistes de juger et de vérifier que toutes les conditions sont requises pour leurs permettre effectivement de travailler.

Ces conditions sont exprimées dans la partie pré-condition de chaque spécialiste sous la forme de tests sur l'état des blackboards ou d'une partie des blackboards (Figure V-24). Si ces conditions sont vérifiées, le spécialiste passe de l'état pré-sélectionné à l'état activable.

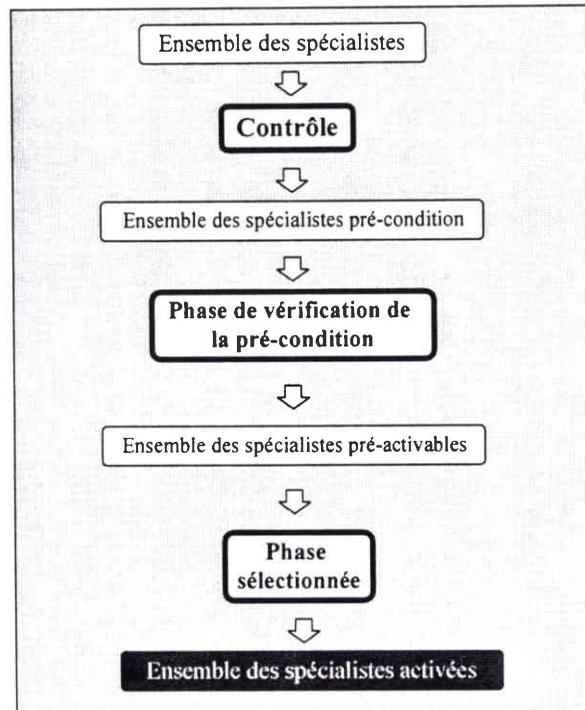


Figure V-24 : Phases de sélection de spécialistes

2 - La phase de vérification (Figure V-25) de la pré-condition d'un spécialiste est effectuée en deux étapes :

- création d'un contexte préliminaire obtenu par filtrage des blackboards ou sur une partie des blackboards. Ce contexte préliminaire représente un sous ensemble du contexte de travail du spécialiste. Il contient la vue partielle de l'état des blackboards nécessaire au spécialiste pour vérifier sa pré-condition. Le spécialiste peut en effet limiter les hypothèses des blackboards qui ont permis sa pré-sélection. Ces hypothèses lui sont passées en focalisation d'attention par le mécanisme de contrôle.
- test de la validité de la pré-condition .

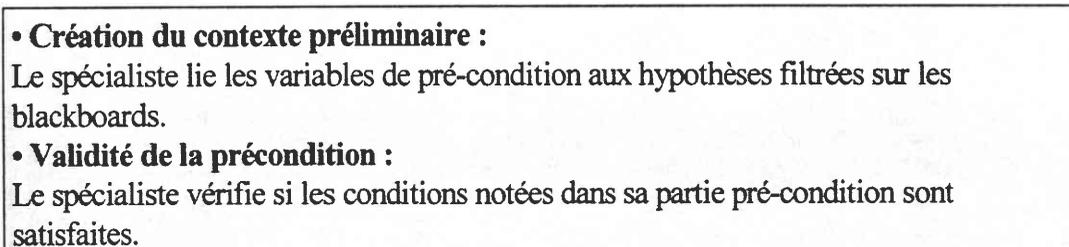


Figure V-25 : Phase de vérification de la pré-condition d'un spécialiste

3 - La phase d'activation (Figure V-26) d'un spécialiste est composée de deux phases :

- création de son contexte de travail,
- exécution du corps du spécialiste.

- **Création du contexte de travail :**
Le spécialiste lie ses variables d'action aux hypothèses filtrées sur les blackboards. Les variables de pré-condition ont déjà été évaluées et restent accessibles durant toute la phase. L'ensemble des variables ainsi liées constitue le contexte de travail du spécialiste.
- **Activation de(s) règle(s) :**
L'étape activation est liée au mode de fonctionnement utilisé :
 - **Simple :** Activer la règle déclenchable la plus importante. Le choix est fonction du coefficient associé à chaque règle déclenchable.
 - **Multiple :** Activer toutes les règles déclenchables en un seul passage. Le système n'effectue qu'un seul parcours de la base de règles.
 - **Cyclique :** Activer toutes les règles déclenchables jusqu'à saturation. Le système boucle sur le mode simple.

Figure V-26 : Phase d'activation d'un spécialiste

V-5.4 Les tâches

Les tâches fournissent des contrôles locaux en fonction des événements survenus dans les blackboards et constituent le niveau de contrôle le plus bas. Chaque tâche est une méta-connaissance dont le rôle est de gérer l'ordre et l'instant d'intervention d'un sous ensemble de spécialistes placé sous son contrôle. Une tâche est composée d'un ensemble de règles de production et dispose d'une liste d'événements locale lui permettant d'avoir une vue de l'ensemble des changements du blackboard qu'elle n'a pas encore traité (Figure V-27) et qui sont susceptibles de l'intéresser.

Une tâche est caractérisée par les éléments suivants :

- Nom :** Identificateur de la tâche
- Variables locales :** Attention, ces variables ne peuvent pas être comparées aux variables locales des spécialistes. En effet, elles ne constituent pas le contexte de travail principal de la tâche. Celui-ci est caractérisé par l'état de la liste des événements au moment de l'activation de la tâche. Dans le cas de la tâche, les variables locales permettent de mémoriser une valeur ou un calcul intermédiaire. Elles sont visibles par les spécialistes qui sont sous la responsabilité de la tâche.
- Base de règles :** Corps de la tâche.
- Mode de raisonnement :** Informe si la tâche est dirigée par les événements, par les règles ou opportuniste. On rappelle qu'une tâche opportuniste prend en compte l'activation de ses règles et de ce fait n'est pas assujettie à un mode de fonctionnement.
- Mode de fonctionnement :** Il est défini en fonction du mode de raisonnement.
- Tâche dirigée par les événements : Le mode de fonctionnement de la tâche peut être de type simple, multiple, cyclique-simple ou cyclique-multiple.
 - Tâche dirigée par les règles : Le mode de fonctionnement de la tâche peut être simple, multiple ou cyclique.

Figure V-27 : Caractéristiques d'une tâche

Le regroupement des spécialistes sous une même tâche peut être effectué suivant deux critères :

- l'existence de conflits : dans ce cas, elles sont dites compétitives. Dès l'instant où plusieurs spécialistes sont amenés à travailler sur les mêmes données, il y a conflit.
- l'existence d'une coopération : dans ce cas, elles sont dites coopératives. Dès l'instant où les résultats de l'une sont utilisés comme données d'entrée pour l'autre spécialiste, il y a coopération.

Un même spécialiste peut apparaître sous plusieurs tâches à la fois.

Le comportement global de la tâche consiste essentiellement à lire des informations dans la liste des événements et à activer en conséquence les spécialistes adéquats. Pour cela, elle s'appuie sur un ensemble de règles de production (Figure V-28). La partie gauche de la règle teste la présence d'événement dans la liste des événements et l'état des variables locales et globales. La partie droite contient un ou plusieurs spécialistes à exécuter de façon séquentielle ou opportuniste selon le mode de raisonnement de la tâche.

SI E5 et E2 existent ALORS SP5 , SP 8 , SP1

Figure V-28 : Structure générale d'une règle de production

V-5.4.1 Tâche dirigée par les événements

Le mode de raisonnement dirigé par les événements consiste à sélectionner l'événement le plus important de la liste d'événements et de rechercher la ou le(s) règle(s) déclenchable(s) (Figure V- 29).

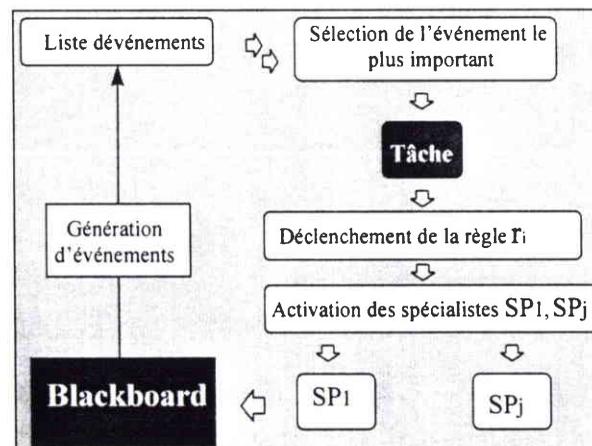


Figure V-29 : Cycle de base du mode dirigé par les événements

Chaque déclenchement de règle provoque l'activation des spécialistes qui lui sont associés. L'activation d'un spécialiste génère des actions dans les blackboards et de ce fait de nouveaux événements sont engendrés. Ainsi, une tâche dirigée par les événements réagit à chaque changement important dans les blackboards. Le ou les spécialistes activés peuvent disposer de l'hypothèse à l'origine de leur activation. Cette hypothèse est fournie par la tâche sous forme de focalisation de contrôle.

A chaque déclenchement de règle, les spécialistes sont sollicités pour travailler séquentiellement.

Lors de l'activation d'une tâche dirigée par les événements, le système retire, au fur et à mesure de leur utilisation, ceux qui ont permis l'activation des règles (Figure V-30). Lors de l'activation de la tâche, le rôle du système est de réagir aux actions les plus importantes dans le blackboard et de les traiter rapidement. Ainsi, lorsque ces actions ont été prises en compte, elles peuvent être retirées de la liste des événements.

1 : La tâche sélectionne l'événement le plus important non encore traité par celle-ci.
2 : Si aucune règle n'est activable avec cet événement
 Alors le marquer comme traité et revenir en 1.
 Sinon, cas où le mode de fonctionnement de la tâche est :

- Simple :

21 : Activer la première règle déclenchable avec cet événement. Un coefficient d'importance est associé à chaque règle. Les règles à coefficients égaux sont classées dans l'ordre de leur définition.
22 : Retirer l'événement traité de la liste des événements.

- Multiple :

21 : Activer toutes les règles déclenchables en un seul passage avec cet événement. Le système n'effectue qu'un seul parcours de la base de règles.
22 : Retirer l'événement traité de la liste des événements.

- Cyclique-simple :

21 : activer la première règle déclenchable avec cet événement.
22 : Retirer cet événement de la liste des événements.
23 : Retourner en 1.

Boucle sur le mode simple pour chaque événement de la liste des événements.

- Cyclique multiple:

21 : Activer toutes les règles déclenchables en un seul passage, avec cet événement.
22 : Retirer cet événement de la liste.
23 : Retourner en 1.

Boucle sur le mode multiple pour chaque événement de la liste des événements.

Figure V-30 : Activation d'une tâche dirigée par les événements

V-5.4.2 Tâche dirigée par les règles

Le mode de raisonnement dirigé par les règles (Figure V-31) consiste à sélectionner la règle la plus importante. Dans ce mode de raisonnement, à chaque règle est affectée un coefficient d'importance. Après avoir sélectionné la règle affectée du coefficient le plus élevé, la tâche consulte la liste d'événements et cherche à valider la règle sélectionnée. En conséquence, une tâche dirigée par les règles traite les événements par groupes, ceux qui satisfont la partie gauche d'une même règle sont traités ensemble, indépendamment de leur importance.

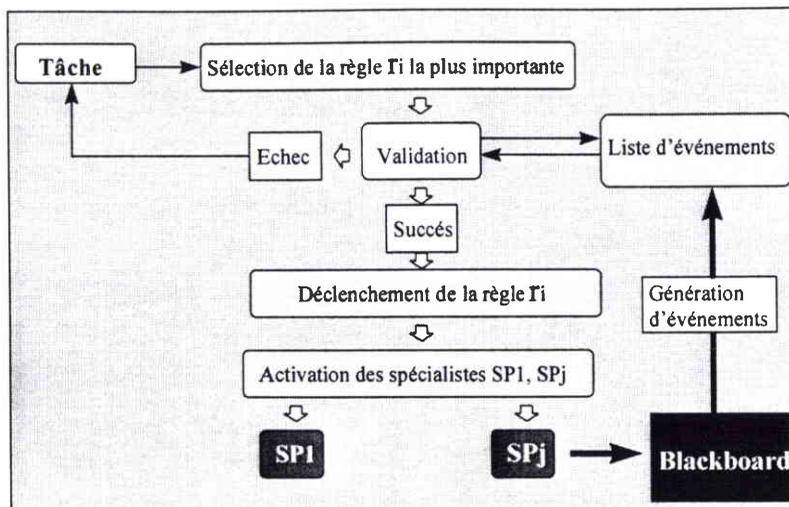


Figure V-31 : Cycle de base du mode dirigé par les règles

A chaque déclenchement de règle, les spécialistes sont sollicités pour travailler séquentiellement.

Lors de l'activation d'une tâche dirigée par les règles, le rôle du système est d'essayer d'activer des règles importantes de la tâche en recherchant les événements appropriés (Figure V-32). Il n'y a aucune raison qui justifierait leur suppression de la liste des événements à la fin de l'activation de la tâche. Toutefois, afin que le système ne néglige pas les actions récentes en s'attardant sur d'anciennes actions et pour limiter la liste des événements, ils seront retirés de la liste à la fin du cycle au cours duquel la tâche a été activée.

1 : A partir de l'état courant de la liste d'événements, rechercher toutes les règles déclenchables.

2 : Si le mode de fonctionnement est :

- **Simple** : Activer la règle déclenchable la plus importante. Pour cela un coefficient d'importance est associé à chaque règle. Les règles à coefficients égaux sont classées dans l'ordre de leur définition.
- **Multiple** : Activer toutes les règles déclenchables en un seul passage. Le système n'effectue qu'un seul parcours de la base de règles.
- **Cyclique-simple** : Activer toutes les règles déclenchables jusqu'à saturation. Le système boucle sur le mode multiple pour chaque événement de la liste des événements.

Figure V-32 : Algorithme d'activation d'une tâche dirigée par les règles

V-5.4.3 Tâche opportuniste

Le mode de raisonnement opportuniste permet une recherche globale sur l'ensemble des événements de la liste d'événements et la classification des spécialistes potentiels par ordre de priorité. En effet, une tâche opportuniste recherche pour chacun des événements de la liste d'événements les règles déclenchables de la tâche. Ensuite, la tâche déclenche chaque règle qui a pour conséquence de réveiller les spécialistes constituant les parties droites de chacune d'eux. Les spécialistes éveillés sont alors placés dans un agenda sous la forme d'instances de

spécialistes. C'est pourquoi la règle d'une tâche est aussi appelée le "TRIGGER" des spécialistes contenus dans sa partie droite. Elle permet ainsi d'en spécifier le déclenchement.

Après avoir parcouru entièrement la liste des événements, le tâche retire de cette liste les événements qui ont permis le réveil des spécialistes et recherche dans l'agenda les spécialistes activables, c'est-à-dire ceux dont la pré-condition est vérifiée. L'ensemble des spécialistes activables constitue des activités potentielles. L'ensemble des instances de spécialistes activables est retiré de l'agenda des "sélectionnés" pour être placé dans l'agenda des "exécutables". L'agenda des instances de spécialistes exécutables regroupe l'ensemble des activités potentielles en concurrence pour travailler.

La tâche doit alors sélectionner une instance de spécialiste dans l'agenda des "exécutables" et l'exécuter. La sélection s'appuie sur la priorité affectée aux instances de spécialiste, calculée à partir d'un ensemble d'heuristiques et de la règle d'intégration qui indique comment prendre en compte ces heuristiques (Figure V-33).

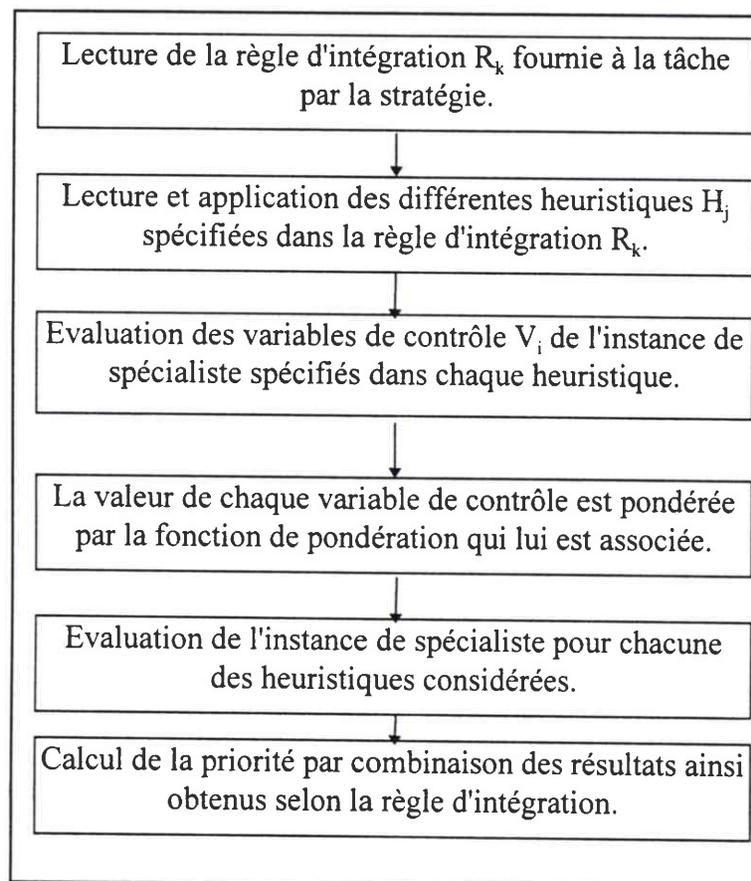


Figure V-33 : Calcul d'une priorité d'instance de spécialiste de connaissance

Les heuristiques sont basées sur des paramètres attribués à chaque spécialiste permettant de noter la pertinence (ex : efficacité, coût) du spécialiste pour une application donnée. Ces paramètres sont définis sous la forme de **variables de contrôle** (Figure V-34). Chaque variable de contrôle est donc affectée d'une valeur qui pourra être constante ou variable en fonction de l'évolution du contexte dans lequel le système évolue. En résumé, une heuristique est définie (Figure V-35) par l'ensemble des variables de contrôle qu'elle souhaite prendre en

compte et par le poids qu'elle leur affecte. Son but est de favoriser ou défavoriser l'influence de certains paramètres dans le calcul des priorités des instances de spécialiste. Les heuristiques à prendre en compte sont spécifiées par la règle d'intégration.

Variables de contrôle	Valeurs
Coût	80
Efficacité	F(blackboards)

Figure V-34 : Paramètres associés à un spécialiste

Heuristiques	Variables de contrôle
H1	(efficacité : 100)
H2	(coût : 010)

Figure V-35 : Définition d'heuristiques

Les heuristiques à prendre en compte et la règle à appliquer pour évaluer la priorité des instances de spécialiste sont spécifiées par la règle d'intégration. Son rôle est de favoriser ou de défavoriser certaines heuristiques. Pour cela elle se compose d'un ensemble d'heuristiques à appliquer à l'instance de spécialiste et par le poids qu'elle leur affecte. Un exemple simple serait de définir une règle d'intégration qui considérerait l'instance la plus efficace comme la plus prioritaire (Figure V-36). Une même tâche peut être activée plusieurs fois avec des règles d'intégration différentes.

Règles d'intégration	Heuristiques
R1	(H1: 100)

Figure V-36 : Définition d'une règle d'intégration

Ensuite, la tâche active le spécialiste associé à l'instance de spécialiste sélectionnée en lui fournissant l'événement responsable de son réveil, en tant que focalisation de contrôle. L'arrêt de la tâche est provoqué lorsque l'agenda des instances de spécialistes exécutables est vide. En résumé, chaque cycle local à une tâche opportuniste (Figure V-37) consiste à déterminer toutes les activités potentielles de la tâche, puis à exécuter l'activité potentielle classée la plus prioritaire.

En résumé, chaque cycle local à une tâche opportuniste consiste à déterminer toutes les activités potentielles de la tâche, puis à exécuter l'activité potentielle classée la plus prioritaire.

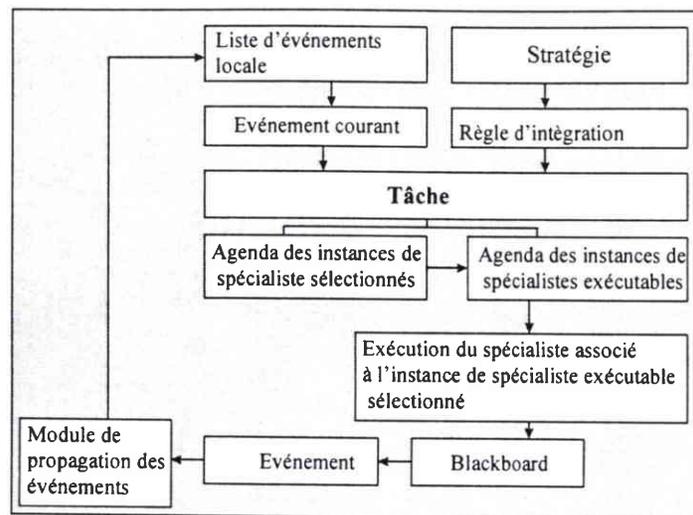


Figure V-37 : Cycle local d'une tâche opportuniste

A chaque cycle local de la tâche le système vérifie la validité des pré-conditions des instances de spécialiste. Cette vérification appelée «maintenance des agendas" est confiée à un système de maintenance. Dès lors que la pré-condition d'une instance de spécialiste n'est plus valide, celle-ci est retirée de l'agenda des «exécutables" pour être placée dans l'agenda des sélectionnés. Pour aider à cette maintenance et ainsi limiter le nombre d'instances manipulées dans les agendas, il a été introduit une caractéristique supplémentaire dans la définition d'un spécialiste. Cette caractéristique permet de définir les conditions de rejet, c'est-à-dire d'exprimer les situations pour lesquelles un spécialiste en association avec l'événement qui l'a réveillé n'est plus capable de fournir un travail et par conséquent **doit être retiré définitivement des agendas**. Attention, ces conditions de rejet ne sont pas redonnantes par rapport à la pré-condition d'un spécialiste. En effet, la validité d'une pré-condition permet de passer une instance de spécialiste d'un agenda à une autre **mais pas de la retirer définitivement des agendas**. Ces conditions de rejet sont exprimées sous la forme de tests sur l'état des blackboards et intégrées au filtre associé à la pré-condition du spécialiste et détermine ainsi son contexte préliminaire. Les variables de pré-condition qui permettent de définir le filtre sur les blackboards pour construire ce contexte préliminaire sont appelées **variables de contexte**.

V-5.4.4 Choix d'un type de raisonnement

Le choix d'un type de raisonnement est lié au type d'événement à traiter et à la possibilité d'expertise.

Une tâche dirigée par les événements permet de traiter des événements au fur et à mesure de leur apparition et de leur importance. Par contre, une tâche dirigée par les règles est plus adaptée au contrôle de spécialistes déclenchables avec plusieurs événements en contexte, indépendamment de leur importance. Les tâches dirigées par les événements ou par les règles sont plus efficaces que les tâches opportunistes, mais exigent une spécification plus précise et plus fine de l'expertise, car le contexte et l'ordre d'exécution des spécialistes doivent être précisés.

Les tâches opportunistes sont plus souples dans la mesure où l'expertise ne déclare que le contexte d'intervention des spécialistes, leur ordre d'activation étant déterminé

dynamiquement en fonction des paramètres associés aux spécialistes, des heuristiques et des règles d'intégration à appliquer.

V-5.5 La stratégie

La stratégie constitue le niveau supérieur du contrôle. Elle a pour rôle de **fournir un contrôle général de la résolution du problème** en s'appuyant sur des résumés de l'état des blackboards. Elle supervise le déroulement du processus de résolution du problème qui consiste à sélectionner les points particuliers du problème à traiter en fonction de l'avancement de la résolution du problème. C'est ainsi qu'elle détermine les régions des blackboards sur lesquelles le système doit se focaliser. Pour cela, elle s'appuie sur une vue globale (résumés des blackboards) de la solution courante et coordonne les activités des tâches.

Pour permettre de tester différentes configurations du système, il est possible d'associer plusieurs sources de connaissances de type stratégie à une application ATOME. Il est évident qu'elles ne pourront être utilisées simultanément.

Une source de connaissances de type stratégie est caractérisée (Figure V-38) essentiellement par des variables locales manipulées et par les prémisses des règles internes à la stratégie. L'ensemble de ces règles est défini dans la base de règles de la stratégie.

Nom : Identificateur de la stratégie.

Variables locales : Attention, ces variables ne peuvent pas être comparées aux variables locales des spécialistes. En effet, elles ne constituent pas le contexte de travail principal de la tâche. Celui-ci est caractérisé par le résumé des blackboards. Dans le cas de la tâche, les variables locales permettent de mémoriser une valeur ou un calcul intermédiaire.

Base de règles : Corps de la stratégie.

Mode de fonctionnement : Il peut être simple, multiple ou cyclique.

Figure V-38 : Caractéristique de la stratégie

Les connaissances de la stratégie sont exprimées sous forme de règles de production (Figure V-39). La partie gauche teste la présence d'hypothèses dans les résumés des blackboards et l'état des variables locales à la stratégie ou globale au problème. La partie droite contient une ou plusieurs tâches à exécuter séquentiellement quand la règle est déclenchée. Dans le cas où l'une des tâches à activer est opportuniste alors une règle d'intégration lui est affectée.

Si H1 et H5 Alors T1/R5 et T6 et T1/R2

Interprétation :

Si les hypothèses H1 et H2 existent dans les résumés des blackboards, Alors exécuter la tâche T1 (type opportuniste associée à la règle d'interprétation R5) ensuite la tâche T6 et enfin la tâche T1 (type opportuniste associée à la règle d'interprétation R2).

Figure V-39 : Structure d'une règle de production de la stratégie

C'est ainsi que la stratégie fournit aux tâches les hypothèses ayant permis leur activation, en tant que focalisation de contrôle. Cette focalisation permet aux tâches d'orienter éventuellement leurs travaux vers des événements portant sur ces hypothèses, par exemple.

V-5.6 Le moteur d'inférences

ATOME fait appel à un moteur d'inférences d'ordre 1, paramétrable, et disposant d'un mécanisme d'unification qui permet d'appliquer la connaissance à des sous-ensembles de noeuds des blackboards. Pour cela, les règles comportent en prémisse une partie condition classique et une partie sélection dont la satisfaction s'exprime par l'existence de noeuds sélectionnés.

Le principe de fonctionnement (Figure V-40) est le suivant :

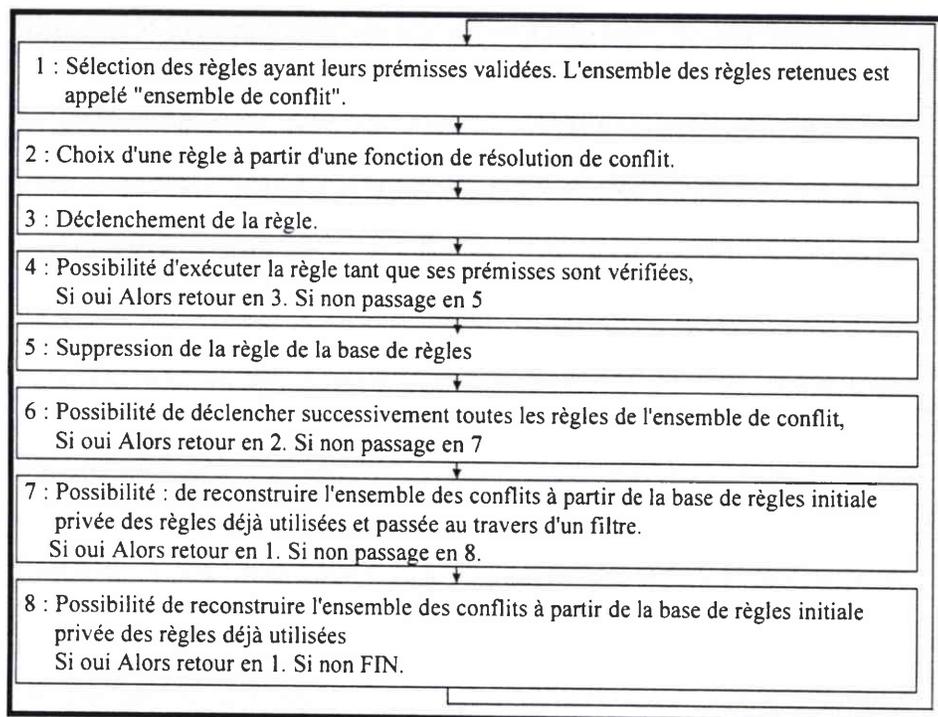


Figure V-40 : Principe de fonctionnement

V-6 Développement de l'application

V-6.1 Définition des blackboards

Après avoir défini le problème à résoudre et les différents objets manipulés, la première étape dans le développement d'une application ATOME est de définir les blackboards manipulés.

V-6.1.1 Blackboard "Ressources"

Le blackboard "Ressources" regroupe l'ensemble des informations concernant le potentiel de production de l'atelier. Les ressources concernent les opérateurs, les postes de travail et leur

équipement respectif existant dans l'atelier. Il permet l'identification et la caractérisation de chacune d'elles ainsi qu'une hiérarchisation entre les différentes ressources.

(Définition du Blackboard "Ressources")	
(Définition niveau "Ressources.Transferts"	
identification	: identification du transfert TR_j
état	: disponible ou non disponible
coût	: coût par unité de temps
liste d'opérateurs autorisés	: liens avec le niveau "Ressources.Opérateurs"
liste des postes départ autorisés	: liens avec le niveau "Ressources.Postes"
liste des postes arrivée autorisés	: liens avec le niveau "Ressources.Postes"
)	
(Définition niveau "Ressources.Postes"	
identification	: identification du poste de travail PT_z
état	: activité ou inactivité
coût	: coût par unité de temps
liste d'opérateurs autorisés	: liens avec le niveau "Ressources.Opérateurs"
liste d'outils autorisés	: liens avec le niveau "Ressources.Outils"
liste de montages d'outils autorisés	: liens avec le niveau "Ressources.MontagesOutils"
liste de montages de pièces autorisés	: liens avec le niveau "Ressources.MontagesPièces"
)	
(Définition niveau "Ressources.Opérateurs"	
identification	: identification de l'opérateur OE_i
date de début d'inactivité	: date
durée d'inactivité	: temps
coût	: coût horaire
liste des postes de travail autorisés	: liens avec le niveau "Ressources.Postes"
)	
(Définition niveau "Ressources.Outils"	
identification	: identification de l'outil OT_b
état	: disponible ou non disponible
coût	: coût par unité de temps
liste de postes autorisés	: liens avec le niveau "Ressources.Postes"
)	
(Définition niveau "Ressources.MontagesOutils"	
identification	: identification du montage MT_c
état	: disponible ou non disponible
coût	: coût par unité de temps
liste de postes autorisés	: liens avec le niveau "Ressources.Postes"
)	
(Définition niveau "Ressources.MontagesPièces"	
identification	: identification du montage MT_c
état	: disponible ou non disponible
coût	: coût par unité de temps
liste de postes autorisés	: liens avec le niveau "Ressources.Postes"
)	

Exemple de distribution des liens inter-niveaux du blackboard "Ressources" :

- Compatibilité des ressources opérateurs, outils de fabrication, montages d'outils et montages de pièces sur les différents postes de travail (Tableau V-3).

	OE1	OE2	OE3		OT1	OT2	OT3	OT4		MO1	MO2		MP1	MP2
PT1	*		*		*	*				*			*	
PT2		*	*			*	*				*		*	
PT3		*				*	*	*			*		*	*

Tableau V-3 : Compatibilité des ressources sur les différents postes

- Compatibilité des systèmes de transferts entre les différents couples "PosteDépart-PosteArrivée" (Tableau V-4). Le sens des flèches indique les liens entre les postes de départ vers les postes arrivés

ST1	PT1	PT2	PT3
PT1			→
PT2			→
PT3	↑	↑	

ST2	PT1	PT2	PT3
PT1		→	
PT2	↑		
PT3			

Tableau V-4 : Compatibilité des systèmes de transfert

- Représentation des différents liens inter-niveaux du blackboard ressources (Figure V-41) :

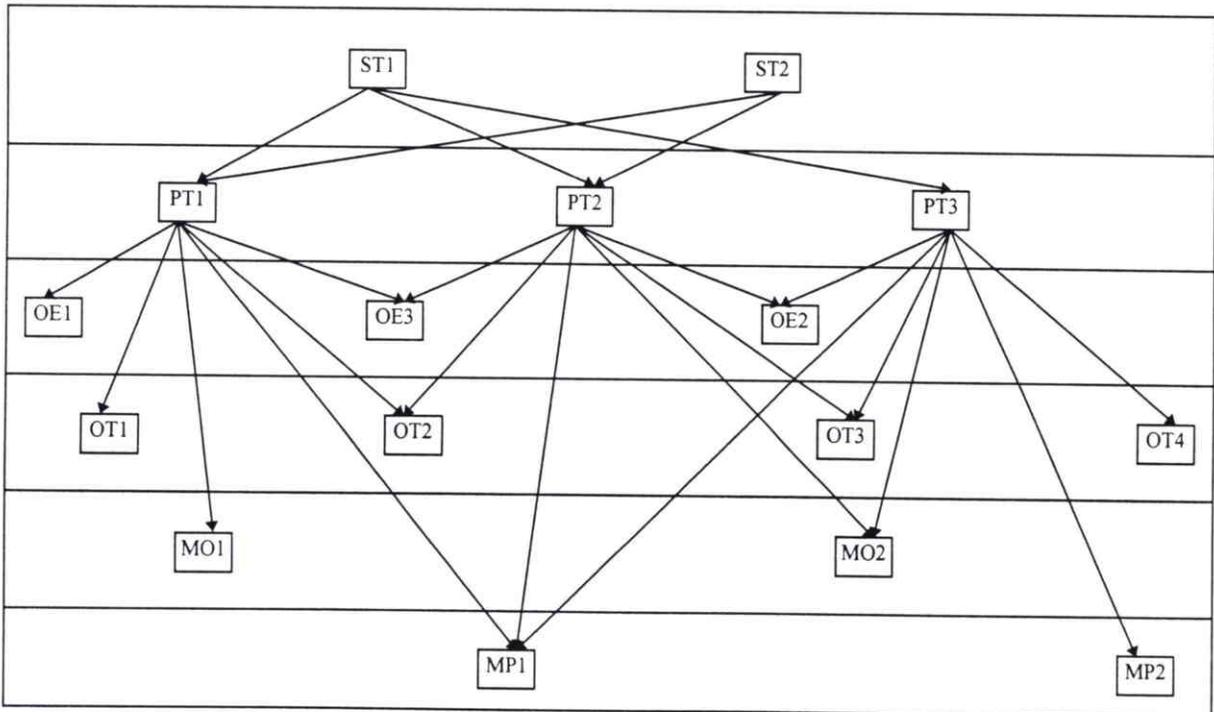


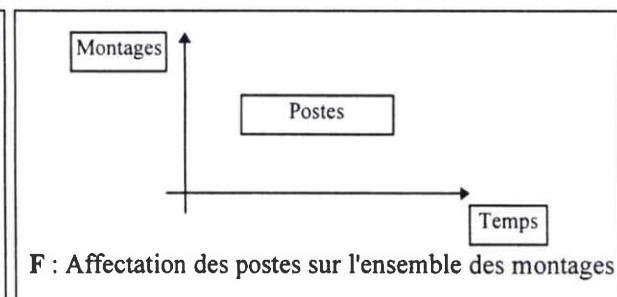
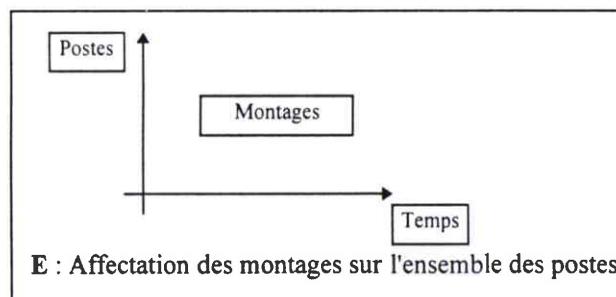
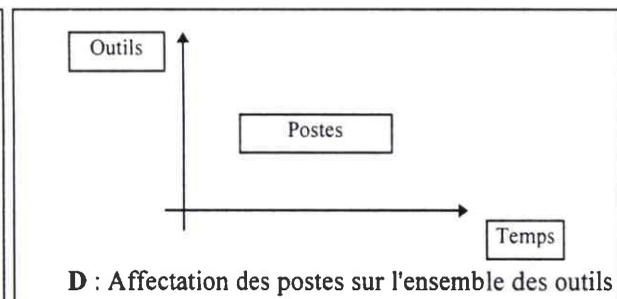
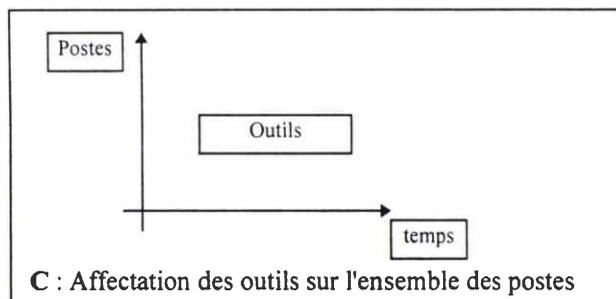
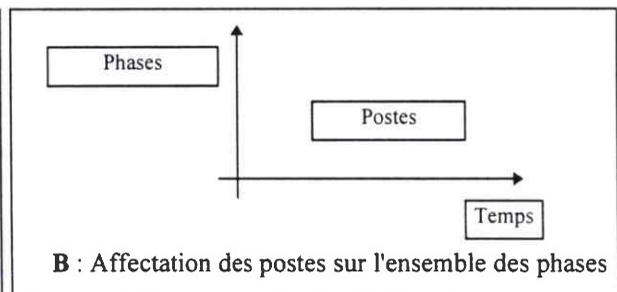
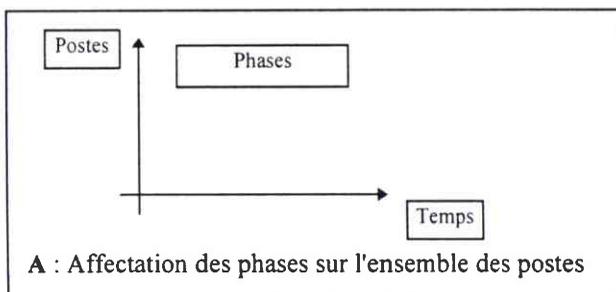
Figure V-41 : Liens inter-niveaux du blackboard ressources

V-6.1.2 Blackboard "ProgrammeFabrication"

Le blackboard "ProgrammeFabrication" regroupe l'ensemble des informations concernant les affectations des différentes ressources. Ces informations doivent aider à l'élaboration de programmes d'utilisation des moyens de production et permettre ainsi une vision globale de l'utilisation des ressources existantes. Le problème posé concerne l'affectation des ressources par rapport au temps, caractérisée par des dates de début et des dates de fin. En conséquence, la méthode GANTT est bien adaptée au problème posé car les affectations de ressources sont relativement stables [Javel 93].

Nous rappelons qu'une représentation de type Gantt utilise un système d'axes où les abscisses représentent le temps et les ordonnées les objets à affecter dans le temps. Le début du segment est calé sur la date de début de l'affectation et la fin du segment sur la date de fin d'affectation. Ce type de graphe a l'avantage d'être facile à lire, mais il a l'inconvénient de ne pas représenter les enchaînements des opérations d'affectation. C'est pourquoi, nous avons introduit des attributs de type lien intra-niveau nous permettant ainsi la représentation de l'ensemble des cas précité.

Suivant l'affectation réalisée la présentation est différente (Figure V-42) :



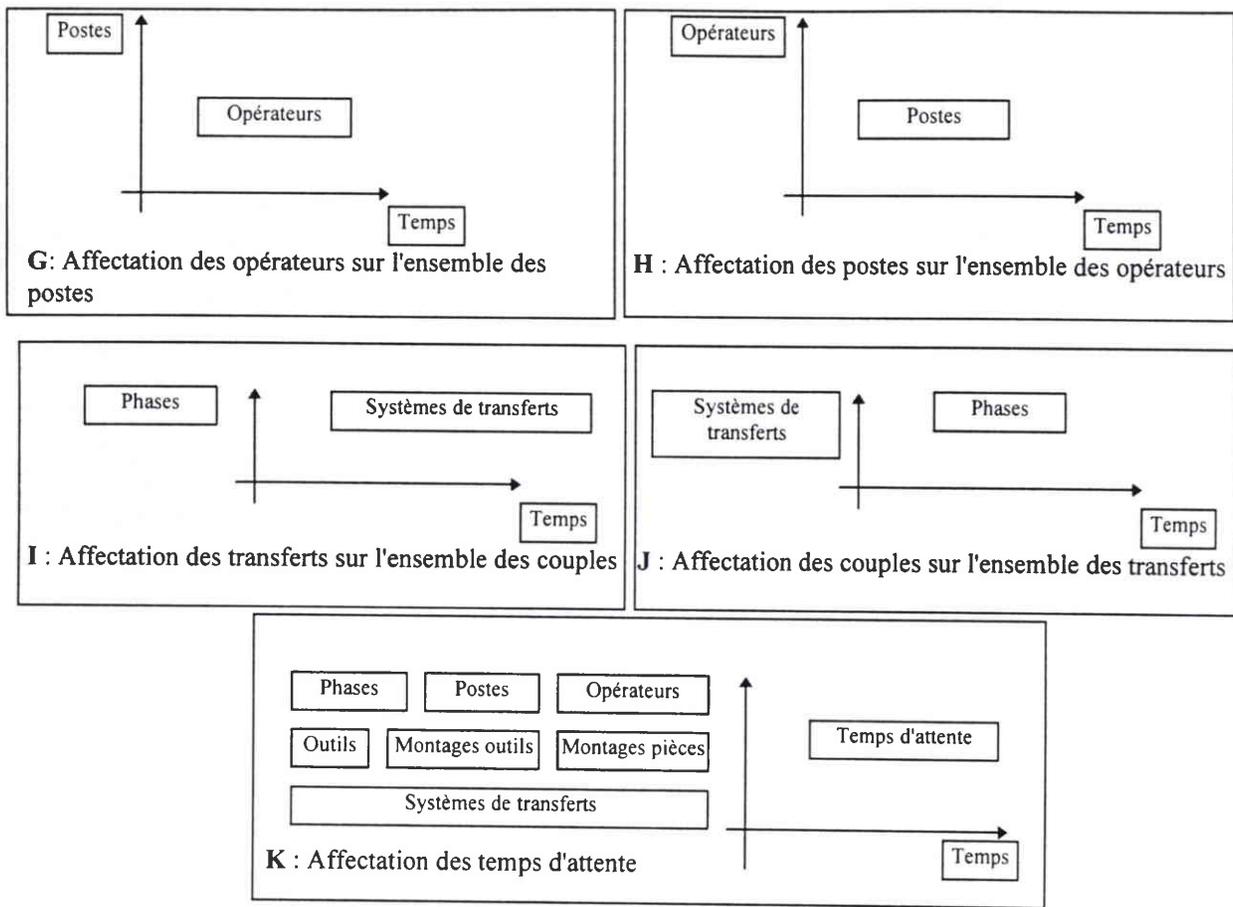


Figure V-42 : La présentation des différentes affectations

Les présentations des cas A , C , E , G permettent de traiter le cas d'un ou plusieurs postes notés hors service. En effet, à partir de chacun des postes on accède directement à l'ensemble des phases, des outils, des montages et des opérateurs qui s'y rapporte. Les phases peuvent ainsi être réaffectées et les ressources libérées.

La présentation B permet de traiter le cas de la suppression d'une ou plusieurs phases. En effet, à partir de cette présentation, on accède directement aux postes et indirectement aux ressources allouées. L'ensemble des postes et des ressources est ensuite libéré.

Les présentations des cas D, F, H, I, J permettent de traiter le cas de la suppression d'une ou de plusieurs ressources. En effet, à partir de chacune d'elles on accède directement aux postes qui s'y rapportent et on réaffecte de nouvelles ressources de remplacement. On remarque que les systèmes de transferts sont affectés aux phases et non aux postes. En effet, les besoins d'un transfert existent dès lors qu'une phase existe et non pas par la seule présence d'un poste de travail. A chaque phase est affectée le système de transferts qui doit être mis en oeuvre en fin d'exécution de la phase.

La présentation K permet de gérer les temps d'attente de la même manière qu'une affectation de ressources. Dans le cadre de nos travaux, nous considérons qu'une phase est en attente d'exécution et une ressource est en attente de travail. Dans tous les cas, il est nécessaire de minimiser le paramètre attente. Attention, lorsqu'une phase est mise en attente ce n'est pas obligatoirement qu'elle est en attente qu'un poste se libère mais aussi qu'une ressource quelconque se libère

nous avons introduit des attributs de type lien intra-niveau nous permettant ainsi la représentation de l'ensemble des cas précité comme le suivent :

(Définition du Blackboard "ProgrammeFabrication")

(Définition niveau "ProgrammeFabrication.Transferts"

identification de l'affectation AFF-TR : identification de l'affectation
 phase départ PH_i : lien avec le niveau "ProgrammeFabrication.Phases"
 poste arrivée PT_z : lien avec le niveau "ProgrammeFabrication.Transferts"
 identification du moyen de transfert : identification du moyen de transfert TR_y
 liste des transferts de remplacement : liste des transferts capables de réaliser le transfert
 date de début d'affectation : date
 durée du transfert : temps
 coût de l'affectation : coût horaire
 pointeur en avant : pointeur en avant sur noeud avec même "transfert"
 pointeur en arrière : pointeur en arrière sur noeud avec même "transfert"
)

(Définition niveau "ProgrammeFabrication.Phases"

identification de la phase : identification de la phase PH_i
 commande : identification de la commande CM_k
 identification du système de transfert départ pièces : lien vers le niveau "ProgrammeFabrication.Transferts"
 identification du poste PT_z : lien vers le niveau "ProgrammeFabrication.Postes"
 date de début : date
 date de fin : temps
 pointeur en avant : pointeur en avant sur noeud de la phase suivante
 pointeur en arrière : pointeur en arrière sur noeud de la phase précédente
 temps d'attente : lien vers le niveau "ProgrammeFabrication.Attentes"
)

(Définition niveau "ProgrammeFabrication.Attentes"

identification de l'affectation AFF-AT : identification de l'affectation
 date de début : date
 date de fin : temps
 identification de la phase : lien avec le niveau "ProgrammeFabrication.Phases"
 identification du poste PT_z : lien vers le niveau "ProgrammeFabrication.Postes"
 identification de l'opérateur OE_i : lien vers le niveau "ProgrammeFabrication.Opérateurs"
 identification du poste OT_i : lien vers le niveau "ProgrammeFabrication.Outils"
 identification du poste MT_i : lien vers le niveau "ProgrammeFabrication.MontagesOutils"
 identification du poste MP_i : lien vers le niveau "ProgrammeFabrication.MontagesPièces"
 pointeur en avant : pointeur en avant sur noeud suivant de la même ressource
 pointeur en arrière : pointeur en arrière sur noeud précédent de la même ressource
)

(Définition niveau "ProgrammeFabrication.Postes"

identification de l'affectation AFF-PT : identification de l'affectation
 identification de la phase PH_i : lien vers le niveau "ProgrammeFabrication.Phases"
 identification du transfert ST_i alimenation : lien vers le niveau "ProgrammeFabrication.Transferts"
 identification du poste : identification du poste PT_z
 liste des postes de remplacement : liste des postes capables de réaliser la phase
 date de début d'affectation : date
 date de fin d'affectation : temps
 coût de l'affectation : coût horaire
 pointeur en avant : pointeur en avant sur noeud avec même "poste"
 pointeur en arrière : pointeur en arrière sur noeud avec même "poste"
 temps d'attente : lien vers le niveau "ProgrammeFabrication.Attentes"
)

(Définition niveau "ProgrammeFabrication.Opérateurs"	
identification de l'affectation AFF-OE	: identification de l'affectation
identification du poste PT_z	: lien vers le niveau "ProgrammeFabrication.Postes"
identification de l'opérateur	: identification de l'opérateur OE_j affecté au poste PT_z
liste des opérateurs de remplacement	: liste des opérateurs capables de réaliser la tâche
date de début d'affectation	: date
date de fin d'affectation	: date
coût de l'affectation	: coût horaire
pointeur en avant	: pointeur en avant sur noeud avec même "Opérateurs"
pointeur en arrière	: pointeur en arrière sur noeud avec même "Opérateurs"
temps d'attente	: lien vers le niveau "ProgrammeFabrication-Attentes"
)	
(Définition niveau "ProgrammeFabrication.Outils"	
identification de l'affectation AFF-OT	: identification de l'affectation
identification du poste PT_z	: lien vers le niveau
"ProgrammeFabrication.Postes"	
identification de l'outil	: identification de l'outil OT_j affecté au poste PT_z
liste des outils de remplacement	: liste des outils capables de réaliser l'opération
date de début d'affectation	: date
date de fin d'affectation	: date
coût de l'affectation	: coût horaire
pointeur en avant	: pointeur en avant sur noeud avec même "Outils"
pointeur en arrière	: pointeur en arrière sur noeud avec même "Outils"
temps d'attente	: lien vers le niveau "ProgrammeFabrication-Attentes"
)	
(Définition niveau "ProgrammeFabrication.MontagesPièces"	
identification de l'affectation AFF-MP	: identification de l'affectation
identification du poste PT_z	: lien vers le niveau
"ProgrammeFabrication.Postes"	
identification du montage MP_y	: identification du montage MP_y affecté au poste PT_z
liste des montages de remplacement	: liste des montages capables de positionner la pièce
date de début d'affectation	: date
date de fin d'affectation	: date
coût de l'affectation	: coût horaire
pointeur en avant	: pointeur en avant sur noeud avec même "MontagePièces"
pointeur en arrière	: pointeur en arrière sur noeud avec même "MontagePièces"
temps d'attente	: lien vers le niveau "ProgrammeFabrication-Attentes"
)	
(Définition niveau "ProgrammeFabrication.MontagesOutils"	
identification de l'affectation AFF-MO	: identification de l'affectation
identification du poste PT_z	: lien vers le niveau
"ProgrammeFabrication.Postes"	
identification du montage MO_y	: identification du montage MO_y affecté au poste PT_z
liste des montages de remplacement	: liste des montages capables de positionner la pièce
date de début d'affectation	: date
date de fin d'affectation	: date
coût de l'affectation	: coût horaire
pointeur en avant	: pointeur en avant sur noeud avec même "MontageOutils"
pointeur en arrière	: pointeur en arrière sur noeud avec même "MontageOutils"
temps d'attente	: lien vers le niveau "ProgrammeFabrication-Attentes"
)	

- **Tableau d'affectation de ressources à des gammes (Tableau V-5) :**

Informations gammes de fabrication				Affectations des ressources					
Gammes	Opérations	Durées	Phases	Postes	Transferts	Opérateurs	Outils	Montages d'outils	Montages de pièces
GF A	OP100A	20	PH10A	PT5	ST5	OE1	OT1	MO1	MP3
A	OP101A	15	PH10A	PT5	ST5	OE1	OT1	MO2	MP3
A	OP102A	15	PH10A	PT5	ST5	OE1	OT2	MO4	MP3
A	OP200A	30	PH20A	PT7	ST3	OE5	OT4	MO1	MP1
A	OP300A	20	PH30A	PT5	ST3	OE1	OT2	MO1	MP5
GF B	OP100B	40	PH10B	PT4	ST5	OE5	OT5	MO3	MP1
GF C	OP100C	30	PH10C	PT7	ST5	OE4	OT4	MO4	MP2
C	OP101C	10	PH20C	PT4	ST1	OE5	OT5	MO4	MP2

Tableau V-5 : Affectation de ressources aux opérations des gammes

- **Tableau d'affectation des systèmes de transfert établis à partir des phases programmées (Tableau V-6) :**

Transferts	Durées	Phases départs	Postes arrivées
ST1		Chargement Magasin	PT5
ST5		PH10A	PT7
ST3		PH20A	PT5
ST3		PH30A	Dégorgement Magasin
ST1		Chargement Magasin	PT4
ST5		PH10B	Dégorgement Magasin
ST1		Chargement Magasin	PT7
ST5		PH10C	PT4
ST1		PH20C	Dégorgement Magasin

Tableau V- 6 : Affectation des systèmes de transfert

- **Ordonnancement des tâches (Figure V-43) :**

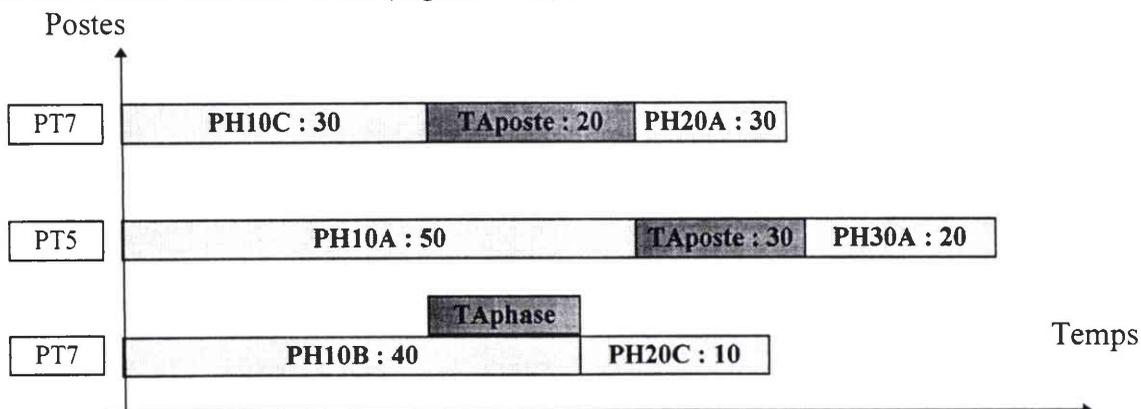


Figure V-43 : Ordonnancement des tâches

• **Représentation des différents liens inter-niveaux et intra-niveaux** (Figure V-44) :

(Remarque : il n'a été reporté que les temps d'attente poste et phase)

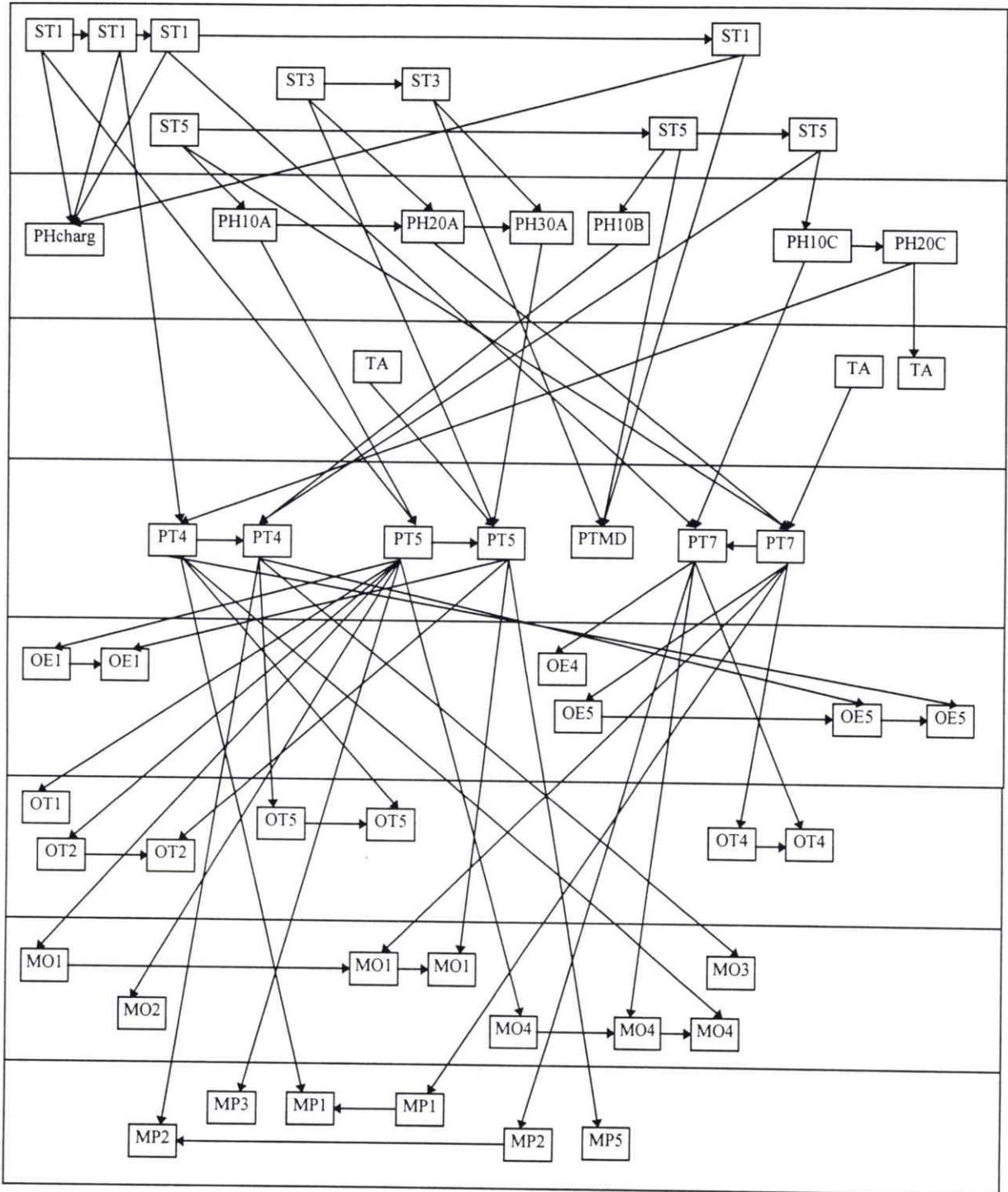


Figure V-44 : Les différents liens inter-niveaux

V-6.1.3 Blackboard "Demandes"

Le blackboard "Demandes" regroupe les informations des commandes à intégrer que l'on mémorise dans les niveaux "Demandes.Commandes", "Demandes.Produits" et "Demandes.Composants". Celles-ci doivent permettre de générer les nomenclatures, les gammes d'assemblage et les gammes de fabrication. Dans ces trois niveaux, tous les attributs ne sont pas obligatoirement évalués. Par exemple, les dates de livraison peuvent être au plus tôt, en contrepartie le client ne spécifie pas le type du produit mais uniquement la famille de produit. Les résultats qui concernent le choix des types de nomenclatures et de gammes sont mémorisés dans les niveaux "Demandes.Nomenclatures", "Demandes.Assemblages" et "Demandes.Fabrications", et le contenu des gammes, c'est-à-dire les opérations, dans le niveau "Demandes.Opérations".

(Définition du Blackboard "Demandes")	
(Définition niveau "Demandes.Commandes")	
client	: identification du client CL_f
commande	: identification de la commande CM_k
liste des produits	: liste de liens avec le niveau "Demandes.Produits"
liste des composants	: liste de liens avec le niveau "Demandes.Composants"
)	
(Définition niveau "Demandes.Produits")	
nom	: identification du produit PR_x
famille	: identification de la famille PR_x du produit Pr_x
nomenclature $NM_m(PR_x)$: lien avec le niveau "Demandes.Nomenclatures"
nombre	: nombre de produits commandés
commande	: lien avec le niveau "Demandes.Commandes"
date au plus tôt	: date de livraison au plus tôt
date au plus tard	: date de livraison au plus tard
)	
(Définition niveau "Demandes.Nomenclature")	
commande	: identification de la commande CM_k
famille $NM_m(PR_x)$: lien avec le niveau "Demandes.Produits"
produit	: identification du produit PR_x
listes des composants	: liste de liens avec le niveau "Demandes.Composants"
gamme d'assemblage	: lien avec le niveau "Demandes.Assemblages"
)	
(Définition niveau "Demandes.Assemblage")	
identification de la gamme d'assemblage	: identification de la gamme $GA_q(NM_m(PR_x))$
nomenclature $NM_m(PR_x)$: lien avec le niveau "Demandes.Nomenclatures"
liste des opérations d'assemblage	: liste de liens avec le niveau "Demandes.Opérations"
)	
(Définition niveau "Demandes.Composants")	
nom	: identification du composant CA_y
nomenclature $NM_m(PR_x)$: lien avec le niveau "Demandes.Nomenclatures"
famille	: identification de la famille CA_y du produit CA_y
nombre	: nombre de composants commandés
commande	: lien avec le niveau "Demandes.Commandes"
date au plus tôt	: date de livraison au plus tôt
date au plus tard	: date de livraison au plus tard
gamme-fabrication	: lien avec le niveau "Demandes.Opérations"
)	

(Définition niveau "Demandes.Opérations"	
identification de l'opération	: identification de l'opération Op_i
identification de la gamme $GF_q(CA_y)$: lien avec le niveau "Demandes.Fabrications"
identification de la gamme $GA_q(NM_m(PR_x))$: lien avec le niveau "Demandes.Assemblages"
numéro de l'opération	: numéro représentant le rang de l'opération dans la gamme
durée de l'opération	: durée de l'exécution de l'opération
numéro de la phase	: numéro représentant le rang de la phase dans la gamme
liste des transferts sélectionnables	: (....., TR_i ,)
liste des postes sélectionnables	: (....., PT_z ,)
liste des montages pièces sélectionnables	: (....., MP_i ,)
liste des montages outils sélectionnables	: (....., MO_i ,)
liste d'outils sélectionnables	: (....., OT_i ,)
liste des opérateurs sélectionnables	: (....., OE_i ,)
opération précédente	: pointeur vers l'opération précédente
opération suivante	: pointeur vers l'opération suivante

Exemple : soient les commandes CM1 et CM2 ci-dessous (Tableau V- 7):

	PR1	PR2	CA5
CM1	40	5	
CM2		20	15

Tableau V-7 : Exemple de commandes : CM1 et CM2

Pour la commande CM1 :

$$\begin{aligned}
 NM1(PR1) &= \{ 3 * CA2 , 1 * CA3 \} \\
 GA1(PR1) &= \{ OP9 \} \\
 GF1(CA2) &= \{ OP8 , OP9 , OP2 \} \\
 GF1(CA3) &= \{ OP7 , OP3 , OP6 , OP1 \} \\
 NM4(PR2) &= \{ 2 * CA3 , 2 * CA5 \} \\
 GA1(PR2) &= \{ OP3 , OP1 \} \\
 GF2(CA5) &= \{ OP7 , OP8 , OP7 \}
 \end{aligned}$$

Pour la commande CM2 :

$$\begin{aligned}
 NM6(PR2) &= \{ 1 * CA4 , 2 * CA5 \} \\
 GA3(PR2) &= \{ OP3 , OP1 \} \\
 GF2(CA5) &= \{ OP7 , OP9 \} \\
 GF1(CA4) &= \{ OP2 \} \\
 GF1(CA5) &= \{ OP4 , OP8 \}
 \end{aligned}$$

Blackboard associé (Figure V-45):

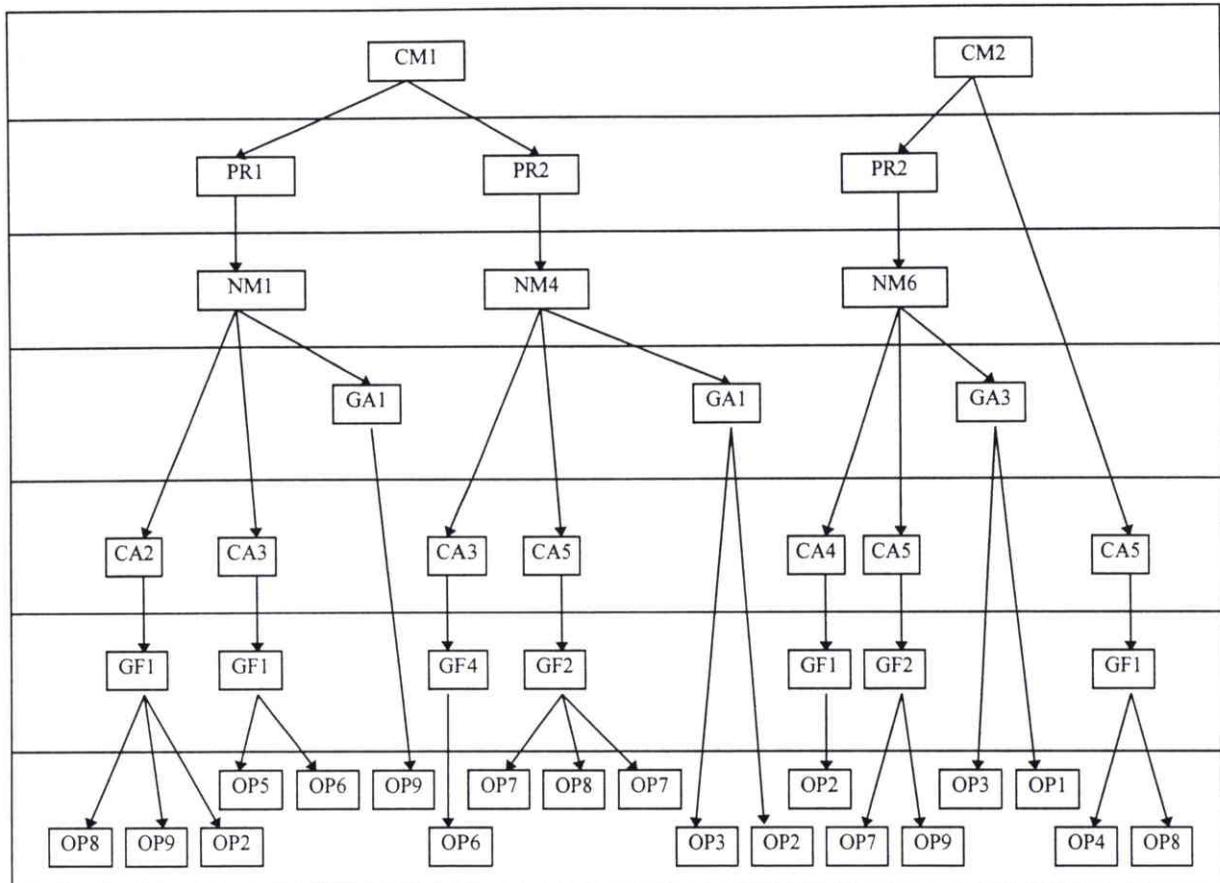


Figure V-45 : Blackboard associé à l'exemple du Tableau V-7

V-6.1.4 Blackboard "PostesCharge"

Le blackboard "PostesCharge" regroupe les informations qui concernent les opérations exécutées et programmées sur chacun des postes de travail. Dans le cadre de nos travaux, nous considérons tous les types d'opérations pouvant être exécutés sur un poste de travail (fabrication, intervention et maintenance). En effet, si l'on désire minimiser les temps d'attente, il est important de pouvoir coordonner l'ensemble de ces opérations. Pour plus de flexibilité, il est intéressant d'avoir la possibilité de choisir entre plusieurs séquences possibles et ceci sans devoir recourir systématiquement à un spécialiste. On note que dès qu'un opérateur fait évoluer une base de savoir-faire lors du traitement d'un aléa physique par exemple, il fait partager son "expérience" à l'ensemble des opérateurs intervenant sur les postes de travail de même type. Pour représenter cela, nous avons introduit les niveaux "PostesCharge.Maintenances", "PostesCharge.Interventions" et "PostesCharge.Opérations".

Dès lors que l'on s'intéresse directement aux opérations exécutées ou programmées sur les postes de travail, l'aspect convivial est important. En conséquence, nous avons introduit des grilles d'affichages et de saisies permettant d'informer les opérateurs et de les inviter à informer le système de pilotage. Les besoins définissent le nombre de grilles et à chacune d'elles correspond un niveau dans le blackboard. On note que tous les types de grilles n'intéressent pas obligatoirement tous les postes car certaines grilles peuvent être spécifiques à certains postes de travail.

(Définition du Blackboard "PostesCharge")	
(Définition niveau "PostesCharge.Postes"	
identification du PT_z	: identification du poste
phase PH_i	: lien vers le niveau "PostesCharge.Phases"
gamme de maintenance $GM_q(PT_z)$: lien vers le niveau "PostesCharge.Maintenances"
gamme d'intervention $GI_q(PT_z)$: lien vers le niveau "PostesCharge.Interventions"
affichage résultats phase	: lien vers le niveau "PostesCharge.GrilleMaintenances"
paramètres maintenance	: lien vers le niveau "PostesCharge.GrillesFinExécutions "
données production	: lien vers le niveau "PostesCharge.GrillesExécutions "
)	
(Définition niveau "PostesCharge.Phases"	
identification du PT_z	: lien vers le niveau "PostesCharge.Postes"
phase	: identification de la phase PH_i
commande	: identification de la commande CM_k
date de début	: date
date de fin	: date
coût	: coût des opérations de maintenance
liste des opérations	: liste de liens vers le niveau "PostesCharge.Opérations"
)	
(Définition niveau "PostesCharge.Maintenances"	
identification du PT_z	: lien vers le niveau "PostesCharge.Postes"
gamme de maintenance $GM_q(PT_z)$: identification de la gamme de maintenance programmée
date de début	: date
date de fin	: date
coût	: coût des opérations de maintenance
liste des opérations	: liste de liens vers le niveau "PostesCharge.Opérations"
)	
(Définition niveau "PostesCharge.Interventions"	
identification du PT_z	: lien vers le niveau "PostesCharge.Postes"
gamme d'intervention $GM_q(PT_z)$: identification de la gamme d'intervention programmée
date de début	: date
date de fin	: date
coût	: coût des opérations de maintenance
liste des opérations	: liste de liens vers le niveau "PostesCharge.Opérations"
)	
(Définition niveau "PostesCharge.Opérations"	
phase PH_i	: lien vers le niveau "PostesCharge.Phases"
gamme de maintenance $GM_q(PT_z)$: lien vers le niveau "PostesCharge.Maintenances"
gamme d'intervention $GI_q(PT_z)$: lien vers le niveau "PostesCharge.Interventions"
nom	: identification de l'opération de maintenance
numéro de l'opération	: numéro de l'opération
durée de l'opération	: temps
opérateurs sélectionnables	: liste des opérateurs sélectionnables
outillages sélectionnables	: listes des outils sélectionnables
coût	: coût de l'opération
opération suivante	: pointeur vers l'opération suivante
opération précédente	: pointeur vers l'opération précédente
)	

(Définition niveau "PostesCharge.GrillesMaintenances")	
nom	: identification de la grille
poste PT _z concerné	: lien vers le niveau "PostesCharge.Postes"
gamme de maintenance GM _q (PT _z)	: identification de la gamme de maintenance programmée
date réelle de début	: date
date réelle de fin	: date
nom de l'opérateur	: identification de l'opérateur
etc	
)	
(Définition niveau "PostesCharge.GrillesFinExécutions")	
nom	: identification de la grille
poste PT _z concerné	: lien vers le niveau "PostesCharge.Postes"
phase PH _i	: identification de la phase exécutée
date réelle de début	: date
date réelle de fin	: date
nom de l'opérateur	: identification de l'opérateur
nombre de composants	: nombre total de composants fabriqués
nombre de composants hors normes	: nombre de composants non validés
etc	
)	
(Définition niveau "PostesCharge.GrillesExécutions")	
nom	: identification de la grille
poste PT _z concerné	: lien vers le niveau "PostesCharge.Postes"
phase PH _i	: identification de la phase de fabrication exécutée
date programmée de début	: date
date programmée de fin	: date
nom de l'opérateur	: identification de l'opérateur
nombre de composants	: nombre total de composants à fabriquer
opérations à exécuter	: liste des opérations à exécuter
etc	
)	

Exemple :

Soit l'Etat des postes PT2, PT7 et PT3 à l'instant *t* :

Poste PT2 :

Exécution d'une gamme de maintenance GM3(PT2) = (OP5, OP8) avec affichage de la grille maintenance numéro 5 (GRM5) permettant d'informer l'opérateur sur des paramètres liés aux opérations de maintenance.

Programmation de l'exécution de la phase PH5 = (OP2)

Poste PT7 :

Fin d'exécution de la phase PH2 = (OP7, OP2, OP5) avec affichage de la grille fin d'exécution numéro 4 (GRFE2) permettant de récupérer les informations liées aux résultats de l'exécution.

Exécution programmée de la phase PH6 = (OP3) avec affichage de la grille exécution numéro 2 (GRE4) permettant d'informer l'opérateur sur les paramètres de lancement de l'exécution.

Poste PT3 :

Exécution d'une gamme d'intervention GI4 = (OP1, OP3, OP1) avec affichage de la grille d'exécution numéro 8 (GRE8) permettant d'informer l'opérateur sur les paramètres d'intervention.

Programmation de la phase PH4 = (OP5, OP8, OP5, OP6).

On obtient la représentation suivante (Figure V-47) :

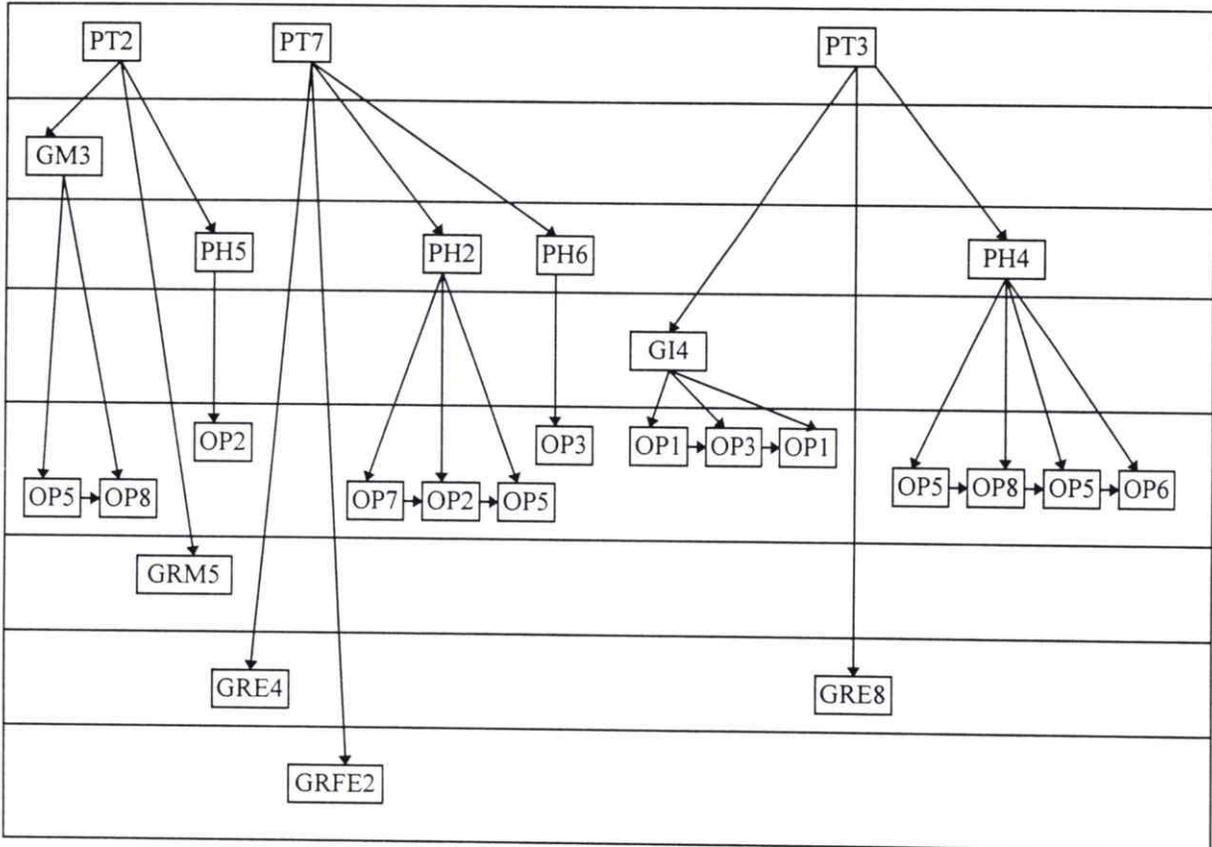


Figure V-46 : Représentation des postes et de leurs liens

V-6.1.5 Blackboards "InitSolutionsRCS"

Le blackboard "InitSolutionsRCS" est utilisé pour développer toutes les solutions possibles d'affectations des ressources. Il est construit à partir des résultats générés par le Système Interactif de Recherche de Séquences (SIRS) à partir des bases de savoir-faire.

(Définition du blackboard : " InitSolutionsRCS"

(Définition niveau " InitSolutions.Gammes"

identification de la gamme : identification de la gamme $GF_q(CA_y)$

liste des phases : liste de liens vers le niveau " InitSolutionsRCS.Phases"

)

(Définition niveau " InitSolutionsRCS.Phases"	
commande	: identification de la commande
phase	: identification de la phase
phase suivante	: pointeur vers la phase suivante
phase précédente	: pointeur vers la phase précédente
date début	: date
date fin	: date
postes activables	: liste de liens vers le niveau "InitSolutionsRCS.Postes"
opérateurs activables	: liste de liens vers le niveau "InitSolutionsRCS.Opérateurs"
coût	: prix unitaire
liste des opérations	: liste de liens vers le niveau "InitSolutionsRCS.Opérations"
)	
(Définition niveau "InitSolutionsRCS.Postes"	
nom	: identification de l'opérateur PT_z
coût	: prix unitaire
)	
(Définition niveau "InitSolutionsRCS.Opérateurs"	
nom	: identification de l'opérateur OE_i
coût	: prix unitaire
)	
(Définition niveau "InitSolutionsRCS.Opérations"	
opération	: identification l'opération
date début	: date
date fin	: date
outils activables	: liste de liens vers le niveau "InitSolutionsRCS.Outils"
montages pièces activables	: liste de liens vers le niveau "InitSolutionsRCS.MontagesPièces"
montages outils activables	: liste de liens vers le niveau "InitSolutionsRCS.MontagesOutils"
coût	: prix unitaire
opération suivante	: pointeur vers l'opération suivante
opération précédente	: pointeur vers l'opération précédente
)	
(Définition niveau "InitSolutionsRCS.Outils"	
nom	: identification de l'outil OT_i
coût	: prix unitaire
)	
(Définition niveau "InitSolutionsRCS.MontagesOutils"	
nom	: identification de l'opérateur MO_i
coût	: prix unitaire
)	
(Définition niveau "InitSolutionsRCS.MontagesPièces"	
nom	: identification du montage pièce MP_i
coût	: prix unitaire
)	

Exemple :

Soient les gammes $GF4 = [PH3]$ et $GA7 = [PH5, PH1]$,

Avec $PH3 = [OP5, OP2, OP8]$, $PH5 = [OP1, OP6]$ et $PH1 = [OP4]$

Possibilités d'affectation de ressources :

PH3 = [(PT5 ou PT4) et (OE5 ou OE7)]
 OP5 = [(OT8 ou OT6) et (MO7, MO5) et (MP4)]
 OP2 = [(OT1 ou OT8) et (MO2 ou MO5) et (MP4 ou MP1)]
 OP8 = [(OT8) et (MO5) et (MP4)]

PH5 = [(PT5 ou PT7) et (OE5 ou OE6 ou OE8)]
 OP1 = [(OT5 ou OT8) et (MO7) et (MP3)]
 OP6 = [(OT8 ou OT7) et (MO7) et (MP4,MP8)]

PH1 = [(PT8) et (OE9)]
 OP4 = [(OT2) et (MO9) et (MP6)]

- représentation de l'affectation des ressources à partir des gammes (Figure V-47)

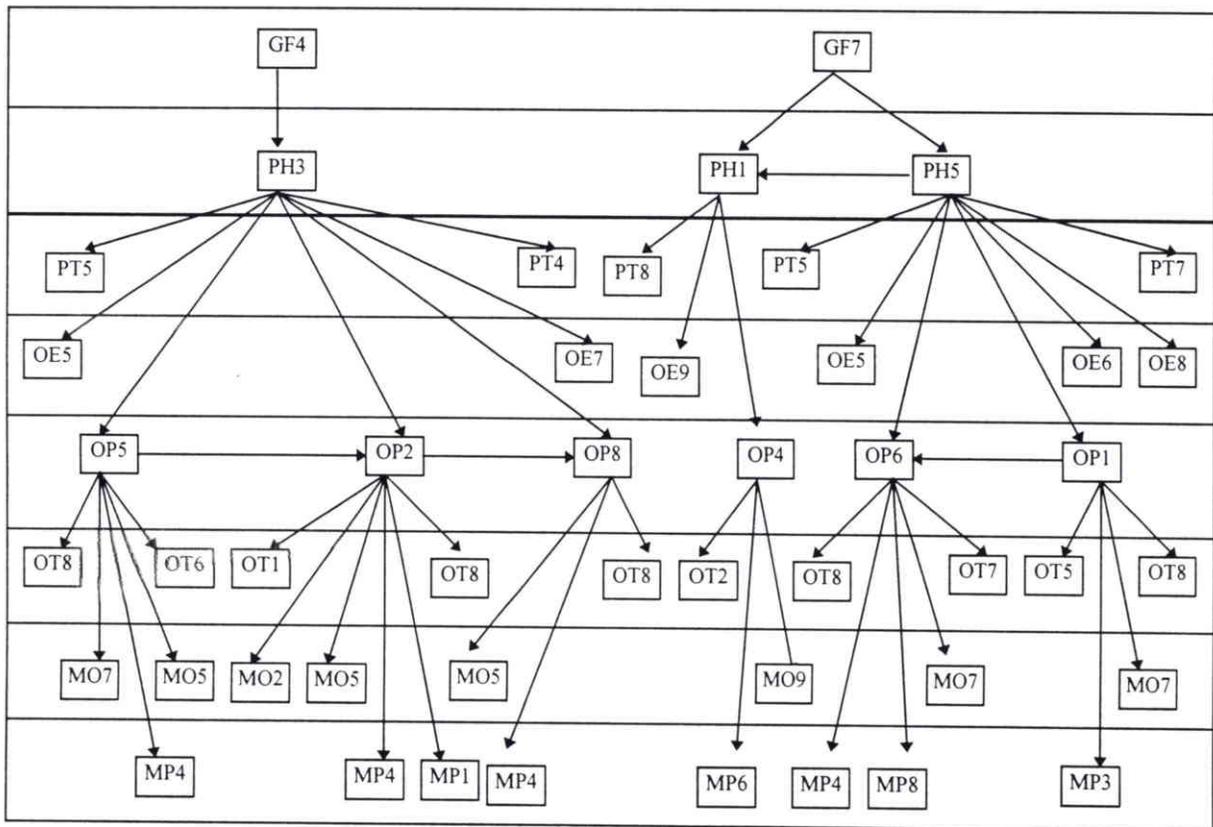


Figure V-47 : Les affectations des ressources à partir des gammes

V-6.1.6 Blackboards "InitSolutionsSTR"

Le blackboard "InitSolutions" est utilisé pour développer toutes les solutions d'affectations possibles des phases et des postes sur les systèmes de transfert

(Définition du Blackboard "InitSolutionsSTR")	
(Définition niveau "InitSolutionsSTR.Phases"	
identification de la phase	: identification de la phase PH _i
commande	: identification de la commande Cm _x
date de début d'affectation	: date de fin d'exécution de la phase PH _i sur le poste départ
durée	: temps pris par le déplacement
identification du système de transfert pièces	: liste de liens vers le niveau "InitSolutionsSTR.Transferts"
Poste arrivée PT _z	: identification du poste d'arrivée
)	
(Définition niveau "InitSolutionsSTR.Transferts"	
identification de l'affectation AFF-TR	: lien avec le niveau "InitSolutionsSTR.Phases"
identification du moyen de transfert	: identification du moyen de transfert TR _y
coût de l'affectation	: coût horaire
)	

Exemple :

- Soient les systèmes de transfert et leurs affectations possibles :

PH20C = (ST1 ou ST5)
 PH20A = (ST5 ou ST3)
 PH30A = (ST1 ou ST3 ou ST5)
 PH10A = (ST3 ou ST5)
 PH10C = (ST5)
 PH10B = (ST1 ou ST3 ou ST5)

- représentation de l'affectation des systèmes de transfert (Figure V-48)

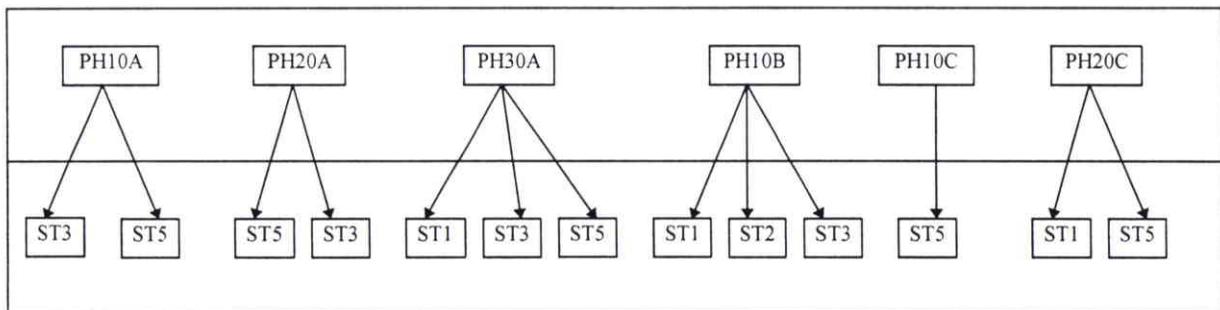


Figure V-48 : Les affectations des systèmes de transferts

V-7 Définitions de l'architecture du système proposé

Dans le cadre des travaux de cette thèse, nous ne développons pas précisément chacun des spécialistes. En effet, le développement et la mise au point de chacun d'eux nécessitent un investissement important en temps et font donc l'objet d'un développement parallèle. En

conséquence, nous introduisons les différents spécialistes nécessaires à l'exploitation des blackboards en fonction de l'objectif fixé, précisons le rôle de chacun d'eux et caractérisons le mécanisme de contrôle.

V-7.1 Définition des spécialistes

Nous rappelons que les spécialistes représentent l'expertise du domaine traité et que leur coopération permet la résolution du problème. Chacun d'eux correspond à un domaine de compétence bien précis et intervient par des actions sur les différents niveaux des blackboards.

Dans le cadre de notre approche, nous introduisons treize spécialistes :

- spécialiste **RessourCeS** noté **"SP:RCS"** qui permet l'initialisation des niveaux "Ressources.Transferts", "Ressources.MontagesOutils", "Ressources.MontagesPièces" "Ressources.Postes", "Ressources.Opérateurs" et "Ressources.Outils" à partir des informations existantes dans la base de données O2 issue du service gestion des ressources. Tous les attributs sont déjà évalués. Lors de la marche du système, le spécialiste doit informer en permanence le système O2 sur l'état des ressources représenté par le blackboard "Ressources" (Ex: outils cassés). L'initialisation du blackboard "Ressources" entraîne l'initialisation du niveau "ProgrammeFabrication.TempsAttentes" pour l'ensemble des ressources. C'est à dire que l'on doit indiquer que toutes les ressources disponibles sont en attente d'affectation. Pour chaque ressource, il y a création d'un noeud dans le niveau de la ressource considérée et d'un autre noeud dans le niveau "ProgrammeFabrication.TempsAttentes". Lors de la création d'un noeud dans le niveau de la ressource, seul l'attribut spécifiant l'identification de la ressource est évalué et dans le niveau "ProgrammeFabrication.TempsAttente", il y a évaluation des trois attributs suivants : identification de l'affectation, date de début et date de fin. Dans ce cas, le temps d'attente est délimité par la date de début de programmation et la date de fin de programmation, son exécution n'est pas assujettie à une pré-condition. Le Tableau V-8 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*	*	*	"Ressources.Transferts"
A2	*	*	*	"Ressources.MontagesOutils"
A3	*	*	*	"Ressources.MontagesPièces"
A4	*	*	*	"Ressources.Postes"
A5	*	*	*	"Ressources.Opérateurs"
A6	*	*	*	"Ressources.Outils"
A7	*			"ProgrammeFabrication.TempsAttentes"

Tableau V-8 : Les actions générées par le spécialiste RCS

- spécialiste **Demandes Non Programmées** noté **"SP:DNP"** qui permet l'initialisation des niveaux "Demandes.Commandes", "Demandes.Produits" et "Demandes.Composants" à partir des informations existantes dans la base de données O2, issues du service d'enregistrement des commandes. Tous les attributs sont déjà évalués. Lors de la marche du système, il doit informer en permanence le système O2 sur les différentes propositions. Son exécution n'est pas assujettie à une pré-condition. Le Tableau V-9 illustre un exemple d'actions générées.

Actions	Création	Modification	Suppression	Niveaux
A1	*			"Demandes.Commandes"
A2	*			"Demandes.Produits"
A3	*			"Demandes.Composants"

Tableau V-9 : Les actions générées par le spécialiste DNP

- spécialiste **SaVoir-Faire** noté "SP:SVF" permet de générer des nomenclatures, des gammes d'assemblage et des gammes de fabrication à partir du niveau "Demandes.Commandes". Pour cela il fait appel au système SIRS. Les informations concernant des choix sur les types de nomenclatures, de gammes et de leurs opérations sont stockées respectivement dans les niveaux "Demandes.Nomenclatures", "Demandes.Assemblages", "Demandes.Fabrications" et "Demandes.Opérations". A partir des informations de chaque liste d'opérations contenue dans le niveau "Demandes.Opérations", elle crée des noeuds dans le blackboard "InitSolutionsRSC". Les gammes de maintenance et d'intervention sont générées suite à une demande formulée par le service maintenance ou par l'opérateur. Parce que les gammes de maintenance et d'intervention ne requièrent pas d'affectation de ressources (pilotées), celles-ci sont directement stockées dans le blackboard "PostesCharges". Son exécution n'est pas assujettie à une pré-condition. Le Tableau V-10 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*	*		"Demandes.Nomenclatures"
A2	*	*		"Demandes.Assemblages"
A3	*	*		"Demandes.Fabrications"
A4	*	*		"Demandes.Opérations"
A5	*			"InitSolutionsRCS.Gammes"
A6	*			"InitSolutionsRCS.Phases"
A7	*			"InitSolutionsRCS.Postes"
A8	*			"InitSolutionsRCS.Opérateurs"
A9	*			"InitSolutionsRCS.Opérations"
A10	*			"InitSolutionsRCS.Outils"
A11	*			"InitSolutionsRCS.MontagesOutils"
A12	*			"InitSolutionsRCS.MontagesPièces"
A13	*			"PostesCharges.Maintenances"
A14	*			"PostesCharges.Interventions"

Tableau V-10 : Les actions générées par le spécialiste SVF

- spécialiste **Demandes Pré-Programmées** noté "SP:DPP" qui permet le chargement du blackboard "InitSolutionsRCS" à partir des informations existantes dans la base de données O2 issues de l'ordonnancement. Pour chaque gamme, le spécialiste crée l'ensemble des noeuds qui s'y rapportent dans les différents niveaux du blackboard "InitSolutionsRCS". Tous les attributs qui y figurent sont déjà évalués. Son exécution n'est pas assujettie à une pré-condition. Le Tableau V-11 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"InitSolutionsRCS.Gammes"
A2	*			"InitSolutionsRCS.Phases"
A3	*			"InitSolutionsRCS.Postes"
A4	*			"InitSolutionsRCS.Opérateurs"
A5	*			"InitSolutionsRCS.Opérations"
A6	*			"InitSolutionsRCS.Outils"
A7	*			"InitSolutionsRCS.MontagesOutils"
A8	*			"InitSolutionsRCS.MontagesPièces"

Tableau V-11 : Les actions générées par le spécialiste DPP

- spécialiste Affectations-TempsAttente noté **"SP:ATA"** qui permet d'affecter et de mettre à jour les temps d'attente des différentes phases. Pour cela il lui faut appliquer les contraintes de date de début et les contraintes de précédences. Néanmoins, la recherche de dates disponibles doit considérer impérativement tous les temps d'attente en particulier dans le niveau "ProgrammeFabrication.Phases", et les dates de début et de fin dans les niveaux "PostesCharge.Maintenances" et "PostesCharges.Interventions". Son exécution est assujettie à une pré-condition qui porte sur la disponibilité des temps d'affectation. Le Tableau V-12 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*	*	*	"ProgrammeFabrication.TempsAttente"
A2	*	*	*	"PostesCharge.Maintenances"
A3	*	*	*	"PostesCharge.Interventions"

Tableau V-12 : Les actions générées par le spécialiste ATA

- spécialiste Affectations-PosTes noté **"SP:APT"** qui permet d'affecter un poste à une phase en fonction des dates de début et de fin de la phase et des disponibilités d'attente du poste de travail. Toutefois, les dates de début et de fin considérées sont celles stockées aux niveaux "ProgrammeFabrication.TempsAttentes", "PostesCharges.Maintenances" et "PostesCharge.Interventions". Ensuite, il met à jour les temps d'attente du poste. Son exécution est assujettie à une pré-condition qui porte sur la disponibilité du poste affecté. Le Tableau V-13 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"ProgrammeFabrication.Postes"

Tableau V-13 : Les actions générées par le spécialiste APT

- spécialiste Affectations-OuTil noté **"SP:AOT"** qui permet d'affecter un outil à un poste de travail en fonction des dates de début et de fin du poste et des disponibilités d'attente de l'outil. Ensuite, il met à jour les temps d'attente de l'outil. Son exécution est assujettie à une pré-condition qui porte sur la disponibilité de l'outil affecté. Le Tableau V-14 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"ProgrammeFabrication.Outils"

Tableau V-14 : Les actions générées par le spécialiste AOT

- spécialiste **Affectations-Montage-Outils** noté "**SP:AMO**" qui permet d'affecter un montage d'outils à un poste en fonction des dates de début et de fin du poste et des disponibilités d'attente du montage d'outils. Ensuite, il met à jour les temps d'attente du montage d'outils. Son exécution est assujettie à une pré-condition qui porte sur la disponibilité du montage d'outils affecté. Le résultat se traduit par la création d'un noeud dans le niveau "ProgrammeFabrication.MontagesOutils". Le Tableau V-15 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"ProgrammeFabrication.MontagesOutils"

Tableau V-15 : Les actions générées par le spécialiste AMO

- spécialiste **Affectations-Montage-Pièces** noté "**SP:AMP**" qui permet d'affecter un montage de pièces à un poste en fonction des dates de début et de fin du poste et des disponibilités d'attente du montage pièces. Ensuite, il met à jour les temps d'attente du montage des pièces. Son exécution est assujettie à une pré-condition qui porte sur la disponibilité du montage des pièces affectées. Le résultat se traduit par la création d'un noeud dans le niveau "ProgrammeFabrication.MontagesPièces". Le Tableau V-16 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"ProgrammeFabrication.MontagesPièces"

Tableau V-16 : Les actions générées par le spécialiste AMP

- spécialiste **Affectations-Opérateurs** noté "**SP:AOE**" qui permet d'affecter un opérateur à un poste en fonction des dates de début et de fin du poste et des disponibilités d'attente de l'opérateur. Ensuite, il met à jour les temps d'attente de l'opérateur. Son exécution est assujettie à une pré-condition qui porte sur la disponibilité de l'opérateur affecté. Le Tableau V-17 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"ProgrammeFabrication.Opérateurs"

Tableau V-17 : Les actions générées par le spécialiste AOE

- spécialiste **Affectations-TRansferts** noté "**SP:ATR**" qui permet d'affecter à une phase un système de transfert en fonction de la date de fin de la phase, de la durée du transfert et des disponibilités d'attente du système de transferts. Ensuite, il met à jour les temps d'attente du système de transfert. Son exécution est assujettie à une pré-condition qui porte sur la disponibilité du moyen de transfert affecté. Le Tableau V-18 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*			"ProgrammeFabrication.Transferts"

Tableau V-18 : Les actions générées par le spécialiste ATR

- spécialiste **INT**erface opérateur noté "**SP:INT**" qui permet la manipulation des grilles d'affichages et de saisies. Il a pour rôle essentiel de divulguer les informations introduites par l'opérateur et de l'informer sur les opérations à exécuter sur le poste de travail. Les informations introduites par l'opérateur qui concernent la fin de l'exécution d'une phase se traduisent par la destruction du noeud associé dans le niveau "PostesCharge.Phases" et des noeuds associés dans le niveau "PostesCharge.Opérations". Dès lors qu'une gamme est terminée, le noeud associé dans le niveau "PostesCharge.Gammes" est détruit. Les informations qui concernent la mise hors circuit d'une ressource provoquent une destruction dans le niveau de la ressource du blackboard "Ressources" et une modification dans le niveau "PostesCharge.Phases" ou "PostesCharge.Opérations". Le Tableau V-19 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*	*	*	"PostesCharge.Grilles.Maintenances"
A2	*	*	*	"PostesCharge.Grilles.Interventions"
A3	*	*	*	"PostesCharge.Grilles.FinExécutions"
A4	*	*	*	"PostesCharge.Grilles.Exécutions"
A5		*	*	"Ressources.Transferts"
A6		*	*	"Ressources.Postes"
A7		*	*	"Ressources.Opérateurs"
A8		*	*	"Ressources.Outils"
A9		*	*	"Ressources.MontagesOutils"
A10		*	*	"Ressources.MontagesPièces"

Tableau V-19 : Les actions générées par le spécialiste INT

- spécialiste **G**estion **P**ostes de **C**harge noté "**SP:GPC**" qui permet la création des postes de charge à partir du résultat des affectations. Dès lors qu'il réceptionne des phases à réaliser, il met à jour les blackboards "InitsolutionRSC" et "PostesCharge". De la même manière, dès lors qu'une gamme est terminée, le spécialiste "SP:GPC" prend en charge la mise à jour des blackboards "ProgrammeFabrication" et "Demandes" en détruisant tous les noeuds qui se rapportent aux gammes terminées. Il est aussi chargé de la gestion d'une réaffectation de ressources lorsque l'une d'entre elles est mise hors circuit. A partir de l'identificateur de la gamme concernée, il recherche les affectations mémorisées dans le blackboard "ProgrammeFabrication" et détruit le noeud concerné par la ressource. Ensuite, il reformule une demande d'affectation par la création de noeud dans le blackboard "InitSolutionsRCS". Le Tableau V-20 illustre un exemple d'actions générées :

Actions	Création	Modification	Suppression	Niveaux
A1	*		*	"InitSolutionsRCS.Gammes"
A2	*		*	"InitSolutionsRCS.Phases"
A3	*		*	"InitSolutionsRCS.Postes"

A4	*		*	"InitSolutionsRCS.Opérateurs"
A5	*		*	"InitSolutionsRCS.Opérations"
A6	*		*	"InitSolutionsRCS.Outils"
A7	*		*	"InitSolutionsRCS.MontagesOutils"
A8	*		*	"InitSolutionsRCS.MontagesPièces"
A9	*		*	"ProgrammeFabrication.Outils"
A10	*	*		"PostesCharge.Postes"
A11	*	*		"PostesCharge.Phases"
A12	*	*		"PostesCharge.Opérations"
A13			*	"ProgrammeFabrication.Transferts"
A14			*	"ProgrammeFabrication.Phases"
A15			*	"ProgrammeFabrication.TempsAttente"
A16			*	"ProgrammeFabrication.Postes"
A17			*	"ProgrammeFabrication.Opérateurs"
A18			*	"ProgrammeFabrication.Outils"
A19			*	"ProgrammeFabrication.Montages.Outils"
A20			*	"ProgrammeFabrication.Montages.Pièces"
A21			*	"Demandes.Commandes"
A22			*	"Demandes.Produits"
A23			*	"Demandes.Nomenclatures"
A24			*	"Demandes.Assemblages"
A25			*	"Demandes.Composants"
A26			*	"Demandes.Fabrications"
A27			*	"Demandes.Opérations"

Tableau V-20 : Les actions générées par le spécialiste GPC

V-7.2 Proposition d'une architecture d'implantation

L'architecture d'implantation (Figure V-49) proposée, s'intègre au niveau de chaque atelier et communique avec les services de l'entreprise à partir d'une base de données orienté objet (BDOO). Nous remarquons que les demandes destinées à l'atelier ne sont pas obligatoirement des demandes clients, mais peuvent parvenir des services internes de l'entreprise. Le système de pilotage proprement dit est constitué de quatre types de spécialiste prenant en charge :

- l'interface avec la BDOO,
- l'affectation et le suivi de production,
- l'interface avec le système physique de production,
- l'exploitation des Savoir-Faire.

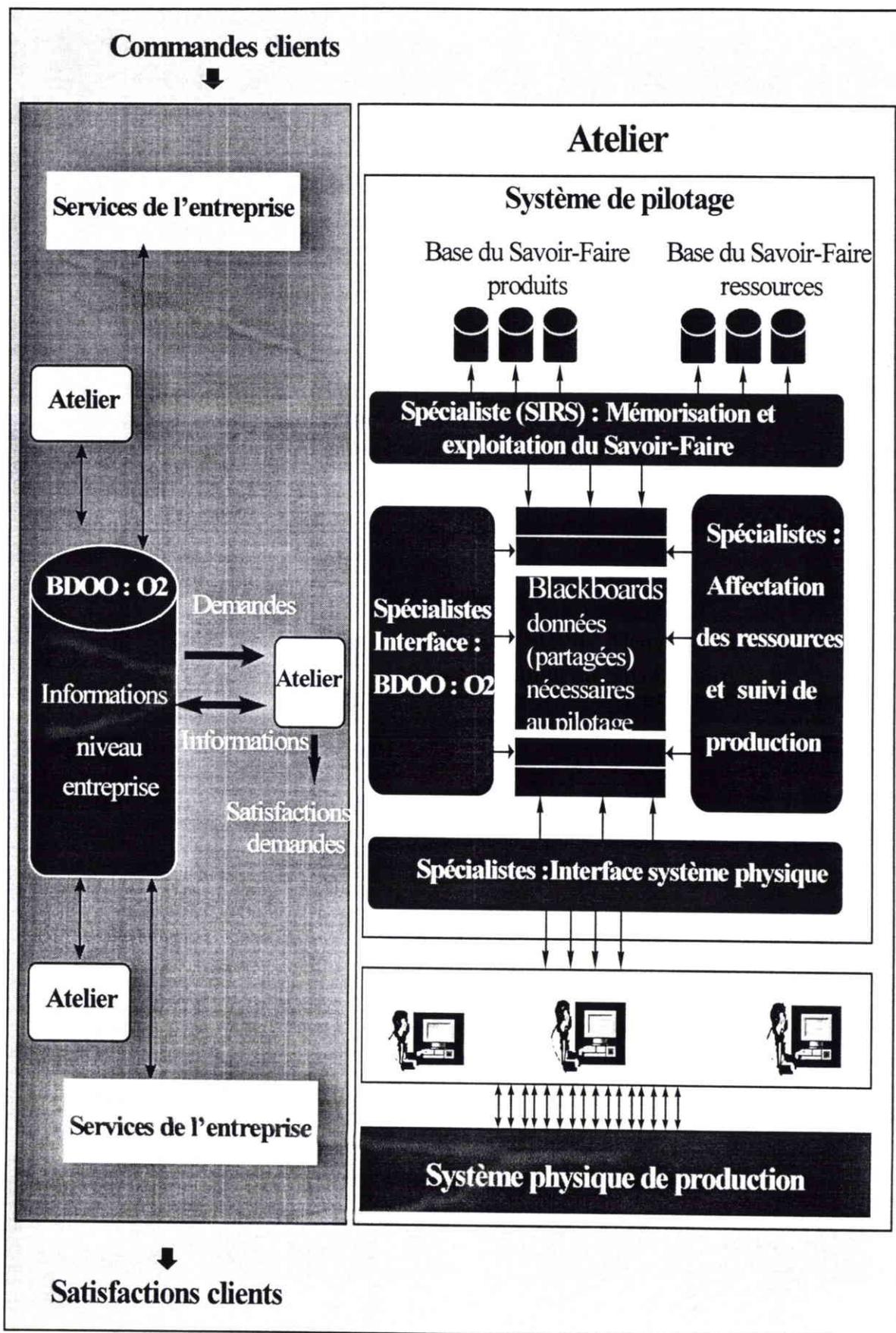


Figure V-49 : Proposition d'un architecture d'implantation

Les information manipulées sont de trois types et concernent :

- la gestion de l'entreprise, qui intéressent l'ensembles des services de l'entreprise. Ces informations sont stockés dans la BDOO. Et représentent le **potentiel de l'entreprise**.
- les ressources, l'état et la programmation du système de production. Ces informations sont stockées dans les blackboards et représentent le **potentiel physique de l'atelier**.
- les Savoir-Faire sur la fabrication des produits et l'exploitation des ressources. Celles-ci représentent le **potentiel réactif de l'atelier**, et sont stockes dans les base de connaissances modèle Grafcet-M.

V-7.3 Définition du mécanisme de contrôle

La structure de contrôle est la composante clé d'un système à blackboard. ATOME fournit un mécanisme de contrôle hiérarchique composé de sources de connaissances de contrôle appelées "Tâches" et d'une méta-source de connaissances de contrôle appelée "Stratégie".

Le niveau stratégie permet d'exprimer explicitement la stratégie de résolution du problème, en s'appuyant sur une vue globale du problème à résoudre, représentée par des blackboards "Résumés". La stratégie est introduite sous forme d'heuristiques manipulées à partir de règles de production.

Le niveau tâche permet une décomposition du problème en sous problèmes et le contrôle spécifique de chacun d'eux par des tâches. Chaque tâche est informée de l'évolution de la résolution du problème par l'intermédiaire d'événements. En conséquence, la définition du mécanisme de contrôle comprend :

- les blackboards "Résumés",
- les tâches,
- les événements,
- la stratégie et les heuristiques mises en oeuvre.

Lors de la définition des tâches et de la stratégie, il est nécessaire de définir un certain nombre de règles de contrôle, de variables de contextes et de variables locales utiles au mécanisme de contrôle. Ces définitions ne peuvent être définitives et pertinentes qu'avec l'appui de manipulations et d'expérimentations. En conséquence, nous nous limitons à présenter des règles de contrôle relativement simples dont l'objectif est d'indiquer le fonctionnement de base du mécanisme de contrôle.

V-7.3.1 Blackboard "Résumés"

Les blackboards "Résumés" représentent le problème à résoudre et l'évolution de sa résolution. Dans le cas de nos travaux, nous considérons deux aspects :

- les objectifs à réaliser représentés par les commandes avec les gammes à affecter et les gammes à réaliser,
- la performance du système représentée par l'optimisation des temps d'attente.

En conséquence nous introduisons deux blackboards "Résumés" :

- **Blackboard : "Résumé 1 : Objectifs"**: Ce blackboard est constitué à partir de la demande.

(Définition du blackboard "Résumé 1: Objectifs")	
(Définition niveau "Objectifs.Commandes"	
client	: identification du client CL_r
commande	: identification de la commande CM_k
liste des produits	: liste de liens avec le niveau "Demandes.Produits"
liste des composants	: liste de liens avec le niveau "Demandes.Composants"
)	
(Définition niveau "Objectifs.GammesAffecter"	
commande	: identification de la commande
phase	: identification de la phase
date début	: date
date fin	: date
coût	: prix
)	
(Définition niveau "Objectifs.GammesAréaliser"	
opération	: identification l'opération
date début	: date
date fin	: date
coût	: prix
)	

- **Blackboard : "Résumé 2 : Performances"**: Ce blackboard est constitué à partir des différents niveaux liés au temps d'attente.

(Définition du blackboard : "Résumé 2 : Performances")	
(Définition niveau "Performances.Phases"	
commande	: identification de la commande
phase	: identification de la phase
coût	:
liste des temps d'attente	: liste de liens vers le niveau "Performances.TempsAttentePhases"
)	
(Définition niveau "Performances.TempsAttentesPhases"	
identification de la phase PH_i	: lien vers le niveau "Performances.Phases"
coût	:
temps attente suivant	: pointeur vers le noeud suivant
temps attente précédent	: pointeur vers le noeud précédent
)	
(Définition niveau "Performances.Postes"	
poste	: identification du poste de travail
coût	:
liste des temps d'attente	: liste de liens vers le niveau "Performances.TempsAttentePoste"
)	
(Définition niveau "Performances.TempsAttentesPostes"	
identification du poste PT_z	: lien vers le niveau "Performances.Postes"
coût	:
temps attente suivant	: pointeur vers le noeud suivant
temps attente précédent	: pointeur vers le noeud précédent
)	

(Définition niveau "Performances.Opérateurs" poste : identification de l'opérateur coût : liste des temps d'attente : liste de liens vers le niveau "Performances.TempsAttenteOpérateurs")
(Définition niveau "Performances.TempsAttentesOpérateurs" identification de l'opérateur OE _i : lien vers le niveau "Performances.Opérateurs" coût : temps attente suivant : pointeur vers le noeud suivant temps attente précédent : pointeur vers le noeud précédent)
(Définition niveau "Performances.Outils" poste : identification du poste de travail coût : liste des temps d'attente : liste de liens vers le niveau "Performances.TempsOutils")
(Définition niveau "Performances.TempsAttentesOutils" identification de l'outil OT _i : lien vers le niveau "Performances.Outils" coût : temps attente suivant : pointeur vers le noeud suivant temps attente précédent : pointeur vers le noeud précédent)
(Définition niveau "Performances.MontagesOutils" poste : identification du poste de travail coût : liste des temps d'attente : liste de liens vers le niveau "Performances.TempsAttenteMontagesOutils")
(Définition niveau "Performances.TempsAttentesMontagesOutils" identification du montage MO _i : lien vers le niveau "Performances.MontagesOutils" coût : temps attente suivant : pointeur vers le noeud suivant temps attente précédent : pointeur vers le noeud précédent)
(Définition niveau "Performances.MontagesPièces" poste : identification du poste de travail coût : liste des temps d'attente : liste de liens vers le niveau "Performances.TempsAttenteMontagesPièces")
(Définition niveau "Performances.TempsAttentesMontagesPièces" identification du montage MP _i : lien vers le niveau "Performances.MontagesPièces" coût : temps d'attente suivant : pointeur vers le noeud suivant temps d'attente précédent : pointeur vers le noeud précédent)
(Définition niveau "Performances.Transferts" poste : identification du poste de travail coût : liste des temps d'attente : liste de liens vers le niveau "Performances.TempsTransferts")

(Définition niveau "Performances.TempsAttentesTransferts"
 identification du système de transfert TR_i : lien vers le niveau "Performances.Transferts"
 coût :
 temps attente suivant : pointeur vers le noeud suivant
 temps attente précédent : pointeur vers le noeud précédent
)

V-7.3.2 Définitions des tâches

Chaque tâche a pour rôle de contrôler un domaine particulier du problème à traiter. Dans le cadre de nos travaux, nous considérons essentiellement quatre domaines à contrôler :

- **tâche "Initialisation"**, dont le rôle est de contrôler les actions à mener lors de l'initialisation du système. Parmi les spécialistes manipulés, on en retient cinq qui peuvent apporter une aide à l'initialisation du système : "SP:RCS", "SP:DNP", "SP:DPP", "SP:SVF" et "SP:ATA". Le contrôle nécessaire à la manipulation de ces spécialistes ne nécessite pas un raisonnement particulier. En effet, l'importance des événements ne prime pas sur l'importance des règles à appliquer. La tâche "Initialisation" est principalement intéressée par les événements faisant référence aux diverses actions dans les blackboards "Ressources" et "Demandes".

Exemple de règles de contrôle :

Si début Alors "SP:RCS" et "SP:DNP" et "SP:DPP".

Si Modification ou Destruction dans le blackboard "Ressources" Alors "SP:RCS".

Si Création "Demandes.Commandes" Alors "SP:SVF" et "SP:ATA".

- **tâche "Suivi"**, dont le rôle est de contrôler les actions à mener dans le cadre du suivi de l'exécution des opérations programmées et de la mise à jour de l'ensemble des informations liées au suivi du système physique de production. Parmi les spécialistes manipulés, on en retient trois qui peuvent apporter une aide au suivi du programme de fabrication : "SP:GPC", "SP:INT" et "SP:SVF". Le contrôle nécessaire à la manipulation de ces spécialistes nécessite un raisonnement dirigé par les événements. En effet, son rôle est de surveiller le bon déroulement des opérations, en conséquence la tâche est tenue de réagir au moindre événement. Elle est principalement intéressée par les événements faisant référence aux diverses actions dans les blackboards "PostesCharge" et "ProgrammeFabrication".

Exemple de règles de contrôle :

Si Création "PostesCharges.Phases" Alors "SP:INT"

Si Modification "PostesCharge.Phases" Alors "SP:GPC"

Si Modification "PostesCharges.Maintenances" Alors "SP:SVF"

- **tâche "Affectations"**, dont le rôle est de contrôler les actions à mener dans le cadre de l'affectation des ressources. Parmi les spécialistes manipulés, on en retient cinq qui peuvent

apporter une aide à l'affectation des ressources : "SP:APT", "SP:AOE", "SP:AOT", "SP:AMT", "SP:AMO". Le contrôle nécessaire à la manipulation de ces spécialistes nécessite un raisonnement de type opportuniste. En effet, son rôle est de sélectionner une solution parmi un ensemble de solutions possibles. Elle est principalement intéressée par les événements faisant référence aux diverses actions dans le blackboard "InitSolutionsRCS".

La tâche fait appel à autant de règles que de ressources à affecter, c'est à dire cinq. Ces règles permettent de construire les instances de spécialistes d'affectation pour chaque événement de la liste locale d'événement y faisant référence.

Si Création "InitSolutions.Gamme.nom" = GF; Alors "SP:APT"
 Si Création "InitSolutions.Gamme.nom" = GF; Alors "SP:AOE"
 Si Création "InitSolutions.Gamme.nom" = GF; Alors "SP:AOT"
 Si Création "InitSolutions.Gamme.nom" = GF; Alors "SP:AMP"
 Si Création "InitSolutions.Gamme.nom" = GF; Alors "SP:AMO"

Les instances ainsi construites sont stockées dans l'agenda des instances de spécialistes sélectionnés (Figure V-50). Ceux-ci représentent toutes les possibilités d'affectations par rapport aux ressources sans tenir compte des disponibilités de chacune.

Une seconde sélection est menée pour établir l'agenda des instances de spécialistes exécutables. La sélection s'opère sur la partie pré-condition du spécialiste. Cette partie peut être utilisée pour tester la disponibilité de la ressource, c'est à dire la possibilité d'affecter un temps d'attente.

La tâche doit alors sélectionner une instance de spécialiste dans l'agenda des exécutables et l'exécuter. Pour cela, elle s'appuie sur la priorité affectée aux instances du spécialiste calculée à partir d'un ensemble d'heuristiques et de règles d'intégration défini au niveau de la stratégie.

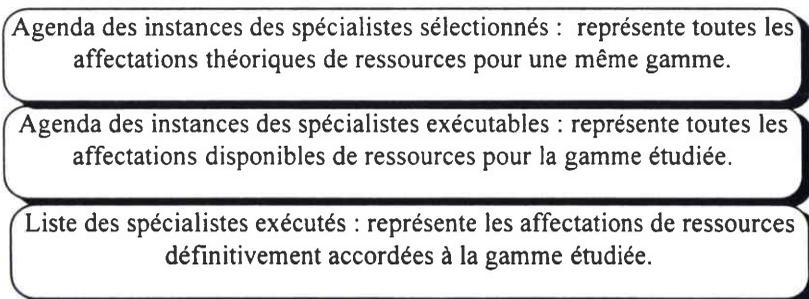


Figure V-50 : Définition du contenu des agendas de la tâche "Affectations"

- **tâche "Transferts"**, dont le rôle est de contrôler les actions à mener dans le cadre de l'affectation des moyens de transfert. Parmi les spécialistes manipulés, on en retient un qui peut apporter une aide à l'affectation des ressources : "SP:ATR". Le contrôle nécessaire à la manipulation de ce spécialiste nécessite un raisonnement de type opportuniste. En effet, son rôle est de sélectionner une solution parmi un ensemble de solutions possibles. Cette sélection est effectuée suivant une stratégie prédéfinie. Elle est principalement intéressée par les événements faisant référence aux diverses actions dans le blackboard "InitSolutionsTR".
 La tâche "Transferts" ne fait appel qu'à une seule règle :

Si Création "InitSolutionTR.Phases" Alors "SP-ATR"

Les instances ainsi construites sont stockées dans l'agenda des instances des spécialistes sélectionnés (Figure V-51). Ceux-ci représentent toutes les possibilités d'affectations par rapport aux moyens de transfert sans tenir compte des disponibilités de chacun.

Une seconde sélection est menée pour établir l'agenda des instances de spécialistes exécutables. La sélection s'opère sur la partie pré-condition du spécialiste. Cette partie peut être utilisée pour tester la disponibilité des moyens de transfert, c'est-à-dire la possibilité d'affecter un temps d'attente.

La tâche doit alors sélectionner une instance de spécialiste dans l'agenda des exécutables et l'exécuter. Pour cela, elle s'appuie sur la priorité affectée aux instances de spécialiste calculée à partir d'un ensemble d'heuristiques et de règles d'intégrations défini au niveau de la stratégie.

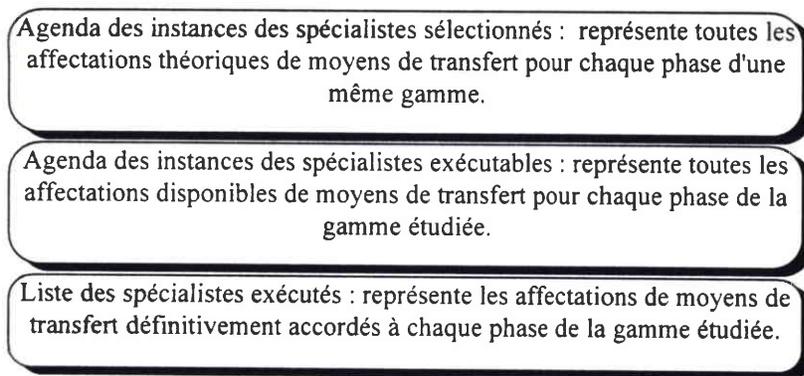


Figure V-51 : Définition du contenu des agendas de la tâche "Transferts"

Remarque : Dans le cas de tâches opportunistes, le système réitère le cycle local jusqu'à épuisement des instances de spécialistes. En conséquence, il est important d'établir une condition de rejet pour indiquer que certains spécialistes ne peuvent plus fournir de travail. En effet, dès lors qu'un spécialiste d'affectation de poste est déclenché, il est inutile de conserver les autres instances de spécialiste d'affectation de poste. Chaque cycle permet l'affectation des ressources pour une gamme. Toutefois, si lors d'une recherche d'affectation les agendas sont vides et que toutes les affectations n'ont pas été pourvues, alors le système est en échec et doit proposer une autre gamme de fabrication, voire même présenter une autre nomenclature.

V-7.3.3 Définition des événements

Les événements représentent les messages envoyés au mécanisme de contrôle par les spécialistes pour l'informer des actions menées dans les blackboards. A chaque spécialiste de juger de l'importance et de la nécessité de générer un événement. En effet, une même action peut être jugée digne d'intérêt par un spécialiste, car elle a été effectuée dans un contexte particulier, et d'aucun intérêt par un autre spécialiste. Le Tableau V-21 représente une définitions de différents événements :

Nom	SP émetteurs	Tâche destinataire	Niveau concerné	Type d'action
EVT1	SP-DNP	"Initialisation"	"Demande.Commandes"	Création
EVT2	SP-DPP	"Affectations"	"InitSolutionsRSC.Gammes"	Création

	SP-SVF SP-GPC			
EVT3	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.Phases"	Création
EVT4	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.Postes"	Création
EVT5	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.Opérateurs"	Création
EVT6	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.Opérations"	Création
EVT7	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.Outils"	Création
EVT8	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.MontagesOutils"	Création
EVT9	SP-DPP SP-SVF SP-GPC	"Affectations"	"InitSolutionsRSC.MontagesPièces"	Création
EVT10	SP-SVF	"Suivi"	"PostesCharges.Maintenances"	Création
EVT11	SP-SVF	"Suivi"	"PostesCharges.Interventions"	Création
EVT12	SP-SVF	"Transferts"	"InitSolutionsTR.Phases"	Création
EVT3	SP-SVF	"Transferts"	"InitSolutionsTR.Transferts"	Création
EVT14	SP-ATA	"Affectations"	"InitSolutionsRSC.Phases"	Création
EVT15	SP-APT	"Suivi"	"InitSolutionsRSC.TempsAttente"	Création
EVT16	SP-APT	"Suivi"	"ProgrammeFabrication.TempsAttente"	Création Modification
EVT17	SP-AOE	"Suivi"	"InitSolutionsRSC. Phases "	Création
EVT18	SP-AOE	"Suivi"	"ProgrammeFabrication.TempsAttente"	Création Modification
EVT19	SP-AOT	"Suivi"	"InitSolutionsRSC. Phases "	Création
EVT20	SP-AOT	"Suivi"	"ProgrammeFabrication.TempsAttente"	Création Modification
EVT21	SP-AMP	"Suivi"	"InitSolutionsRSC. Phases "	Création
EVT22	SP-AMP	"Suivi"	"ProgrammeFabrication.TempsAttente"	Création Modification
EVT23	SP-AMO	"Suivi"	"InitSolutionsRSC. Phases "	Création
EVT24	SP-AMO	"Suivi"	"ProgrammeFabrication.TempsAttente"	Création Modification
EVT25	SP-INT	"Initialisation"	"Ressources.Transferts"	Modification Suppression
EVT26	SP-INT	"Initialisation"	"Ressources.Postes"	Modification Suppression
EVT27	SP-INT	"Initialisation"	"Ressources.Opérateurs"	Modification Suppression
EVT28	SP-INT	"Initialisation"	"Ressources.Outils"	Modification Suppression
EVT29	SP-INT	"Initialisation"	"Ressources.MontagesOutils"	Modification Suppression
EVT30	SP-INT	"Initialisation"	"Ressources.MontagesPièces"	Modification Suppression
EVT31	SP-INT	"Suivi"	"Postes.Chages.Postes"	Modification

				Suppression
EVT32	SP-INT	"Suivi"	"Postes.ChagesPhases."	Modification Suppression
EVT33	SP-INT	"Suivi"	"Postes.Chages.Maintenances"	Modification Suppression
EVT34	SP-INT	"Suivi"	"Postes.Chages.Interventions"	Modification Suppression
EVT35	SP-INT	"Suivi"	"Postes.Chages.Opérations"	Modification Suppression

Tableau V-21 : Représentation des différents événements

V-7.3.4 Définition de la stratégie

On considère que la stratégie doit contrôler le pilotage au niveau des commandes à réaliser avec prise en compte des performances. En conséquence, elle est sensible aux actions menées dans les niveaux "Demandes.Commandes", "Demandes.Gammes", "ProgrammeFabrication.Gammes", "PostesCharge.Commandes" et "ProgrammeFabrication.TempsAttente".

Exemple de règles de contrôle :

Si "GammeAréaliser" = vide Alors Arrêt

Si début Alors Tâche "Initialisations"

Si Création "GammesAaffecter" Alors Tâche "Affectations" * RI3

Si Création "GammesAréaliser" Alors Tâche "Transferts" * RI2 et Tâche "Suivi"

Exemple de règles d'intégration :

$$RI2 = (H5 * 2)$$

$$RI3 = ((H1 * 3) (H2 * 2))$$

Exemple d'heuristiques :

H1 = (coût 7) (priorité au poste travail à coût élevé)

H5 = (coût 2)

H2 = (temps d'attente 5) (goulot d'étranglement)

H3 = (priorité 1) (poste de travail à rentabiliser)

V-7.3.5 Proposition d'une architecture

En résumé l'architecture proposée (Figure V-52) se compose de :

Six blackboards :

- le blackboard "Ressources" qui représente le potentiel du système physique de production,
- le blackboard "Demandes" qui représente les informations sur les demandes non programmées ainsi que les propositions de réalisation,

- le blackboard "ProgrammeFabrication" qui représente l'état à l'instant t de l'utilisation du système physique,
- le blackboard "IntSolutionsRCS" qui représente l'ensemble des solutions d'affectation de ressources possibles à l'instant t.
- le blackboard "IntSolutionsTR" qui représente l'ensemble des solutions d'affectation de moyens de transferts possibles à l'instant t.
- le blackboard "PostesCharge" qui représente la charge (fabrication, maintenance, intervention) de chaque poste de travail.

Quatre tâches qui sont :

- la tâche "**Initialisation**" qui permet d'initialiser les informations nécessaires au pilotage à partir des ressources disponibles et des demandes formulées. Celles-ci peuvent être programmées (ordonnancement) ou non programmées (urgente).

Elle coordonne les actions des spécialistes qui sont :

- "SP:RCS" : spécialiste de gestion des ressources,
- "SP:DNP" : spécialiste de gestion des demandes non programmées,
- "SP:DPP" : spécialiste de gestion des demande programmées,
- "SP:SVF" : spécialiste de gestion du savoir-faire ,
- "SP:ATA" : spécialiste de gestion des temps d'attente de phase.

- la tâche "**Affectations**" des ressources en fonction des capacités et des disponibilités à l'instant t de chacune d'elles.

Elle coordonne les actions des spécialistes :

- "SP:APT" : spécialiste affectation de poste de travail,
- "SP:AOE" : spécialiste affectation d'opérateur,
- "SP:AOT" : spécialiste affectation d'outil,
- "SP:AMP" : spécialiste affectation de montage de pièces,
- "SP:AMO" : spécialiste affectation de montage d'outils.

- la tâche "**Transferts**" qui prend en charge l'affectation des moyens de transfert.

Elle coordonne les actions de la spécialiste :

- "SP:ATR" : spécialiste d'affectation de moyen de transfert,

- la tâche "**Suivi**" qui prend en charge l'interface de communication avec les opérateurs.

Elle coordonne les actions des spécialistes :

- "SP:GPC" : spécialiste de gestion des postes charges,
- "SP:INT" : spécialiste de communication avec les opérateurs,
- "SP:SVF" : spécialiste de gestion du savoir-faire en maintenance et intervention.

Une **stratégie** qui permet de gérer le problème de pilotage au niveau des demandes et des performances, représentées à travers les blackboards résumés "Gammes" et "Performances".

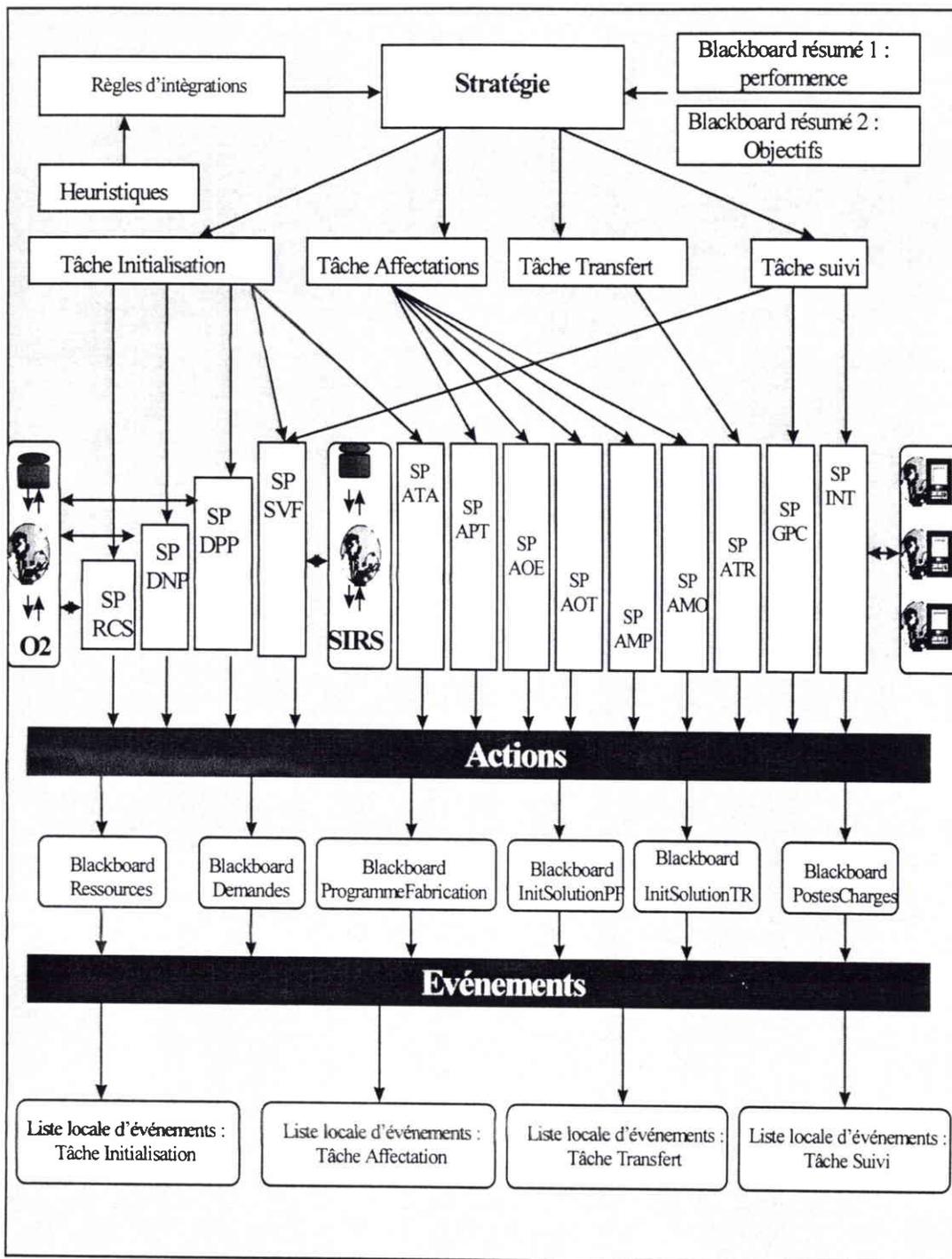


Figure V-52 : Architecture de contrôle

V-8 Conclusion

L'évolution des technologies et leur intégration dans les entreprises, en particulier les ateliers, obligent le personnel à manipuler des connaissances à caractère pluridisciplinaire et interdisciplinaire. En conséquence, le développement d'une architecture de pilotage doit prendre en compte ces deux aspects de pluridisciplinarité et d'interdisciplinarité. Ceux-ci peuvent être pris en compte dans une approche distribuée qui permet la décomposition d'un problème en sous-problèmes. Cette approche favorise la résolution globale du problème en développant un ensemble de spécialistes spécifiques à chacun des sous-problèmes. Il existe deux types d'approche distribuée :

- l'approche multi-agents qui utilise un ensemble d'agents autonomes organisés autour d'un système de communication. Pour cela, elle s'appuie sur une structure en réseaux. Ce type d'approche nous paraît bien adaptée au pilotage temps réel dont le réseau est construit en interconnectant les postes de travail.

- l'approche multi-experts qui utilise un ensemble d'agents autonomes communiquant par l'intermédiaire d'une structure de données partagée. Cette approche nous paraît bien adaptée à un pilotage basé sur l'exploitation des connaissances.

Comme nos objectifs reposent sur une réactivité basée sur l'exploitation du savoir-faire de l'entreprise, nous avons donc opté pour une architecture distribuée de type multi-experts.

Globalement, le pilotage d'atelier doit répondre à trois préoccupations majeures : l'affectation des ressources à une phase de fabrication, le transfert des produits et le suivi de fabrication. Les trois tâches ainsi introduites se distinguent de le type de raisonnement développé en fonction des objectifs. On note que les tâches d'affectation de ressources et de transfert font appel à un raisonnement opportuniste. En effet, le problème se résume à sélectionner des ressources parmi un ensemble de ressources disponibles en fonction de critères prédéfinis. Ce comportement est de type opportuniste. Par contre, la tâche de suivi de fabrication doit réagir aux événements et fait donc appel à un raisonnement guidé par les événements.

Un point essentiel d'un système multi-experts est le choix du mécanisme de contrôle. On relève que dans le cas du pilotage d'atelier le contrôle des opérations reste très hiérarchisé à cause des responsabilités de chacun.

En conséquence, les caractéristiques citées, à savoir raisonnement opportuniste, raisonnement dirigé par les événements et contrôle hiérarchisé, nous ont conduit à opter pour l'outil ATOME. En effet, les caractéristiques précitées correspondent à celles proposées par ATOME, puisqu'il utilise trois niveaux hiérarchiques de contrôle avec des possibilités de raisonnement guidé par les règles, par les événements ou opportuniste.

Conclusion

Conclusion

Les contraintes économiques actuelles engendrées par une demande en constante évolution amènent les entreprises à devoir s'adapter à une clientèle de plus en plus exigeante, n'hésitant pas à faire jouer la concurrence. Une constante parmi ces exigences est le temps. En effet, que ce soit lors de la demande d'un devis, d'une modification de commande ou encore d'une commande urgente, le client souhaite une réponse dans les plus brefs délais.

A ces contraintes économiques sont venues s'ajouter les nécessaires restructurations industrielles qui ont obligé les entreprises à se séparer de collaborateurs proches de la retraite et à sous-traiter des services. Dans le cadre de ces restructurations, de nombreuses entreprises n'ont pas traité le problème de mémorisation de l'expérience. En conséquence, se pose à l'entreprise le problème de la capitalisation et l'exploitation de son savoir-faire.

Afin d'apporter notre contribution à la problématique ainsi introduite, nous proposons une architecture distribuée multi-experts avec contrôle hiérarchique pour un pilotage réactif de système de production. Ces travaux s'inscrivent dans le cadre du projet SYCORO (SYstème de COnduite Réactive par Objectif) sur une conduite de système de production capable de prendre en compte des aspects tels que : intégration, flexibilité, réactivité et capitalisation des savoir-faire.

Pour aborder le problème de pilotage, il est nécessaire de caractériser le type de production que l'on souhaite aborder. La problématique citée précédemment concerne avant tout les entreprises ayant des productions par lots. En effet, les produits qu'on y fabrique se prêtent aisément à des modifications ou à une demande urgente et, de ce fait, engendrent les comportements cités précédemment. Ce type d'entreprise est très présent dans les industries manufacturières de transformation mécanique qui travaillent essentiellement par lots répétitifs. Ceci étant dit, nous ne nous restreignons pas aux industries manufacturières. En effet, un atelier mécanique peut exister quel que soit le type d'industrie.

Le type de production ayant été ainsi défini, nous montrons que notre approche n'est pas assujettie à un type d'atelier particulier. En effet, nous proposons d'adapter la réaction aux aléas par rapport aux disponibilités de l'atelier et non d'optimiser l'engagement des postes de travail, en faisant appel à un pilotage temps réel. L'objectif est de réagir à moindre coût en minimisant les perturbations de la production qui coûtent de l'argent et déstabilisent les services d'organisation et de planification des besoins et de la production. De manière générale nous considérons que les problèmes de flexibilité et d'adaptabilité sont très liés à la structure physique de l'atelier, par contre l'aspect réactif, tel que nous le traitons, paraît lié davantage au mode de pilotage.

La problématique ainsi définie "Double réactivité/Capitalisation du Savoir-Faire", nous analysons les différentes approches existantes et montrons qu'elles se révèlent insuffisantes pour y apporter une réponse satisfaisante. La fonction de pilotage a pour objectif d'assurer la cohérence de décision entre les ordres issus de la gestion prévisionnelle et les actions exécutées au niveau du système physique de production. Dans le cas où l'ordonnancement est intégré à la fonction de pilotage, on parle de pilotage sans ordonnancement prévisionnel. Dans le cas où l'ordonnancement est déjà traité au niveau de la gestion prévisionnelle, on parle de

pilotage à ordonnancement prévisionnel. Les différentes approches proposées, que se soit dans le cadre d'un ordonnancement prévisionnel partiel ou total, gèrent la réactivité à partir des dates de début, de fin de fabrication et de la permutabilité des postes de travail. Certaines approches sont implantées à partir de réseaux de communications permettant aux différents centres de décisions de coopérer et ainsi en améliorer les performances. Globalement, toutes les approches proposent des solutions basées sur le réordonnement de la production en temps réel. Très peu d'approches intègrent la gestion des opérateurs, des ressources et les problèmes de transfert. La quasi totalité des approches se préoccupent de gérer l'exécution des opérations et les disponibilités des postes de travail et reste donc très liée au type d'atelier auquel elle s'applique.

De part la position centrale de la réactivité dans l'approche proposée et les différentes interprétations possibles que l'on peut y associer, nous développons une analyse et proposons une définition par rapport aux aléas. La réactivité telle qu'on peut l'interpréter dans les différentes approches existantes est définie par rapport au temps réel. Dans notre approche, nous définissons la réactivité par rapport au savoir-faire de l'entreprise. La qualité d'une réactivité ne doit pas être fondée sur les dixièmes de secondes économisés lors d'une réaffectation mais sur le savoir-faire de l'entreprise. En effet, une réactivité basée sur le temps réel peut faire croire que le souci des entreprises est de modifier leur production dans la seconde et à tout moment. Dans le cas où le système de pilotage est interfacé sur chaque poste de travail et que quelques anomalies apparaissent dans un espace de temps restreint sur plusieurs postes, la production n'est plus contrôlable. Dans notre approche, nous traitons deux types de réactivité. Une réactivité dite "Amont" qui concerne les aléas de commande, ceux-ci peuvent représenter une commande urgente ou une modification de commande déjà programmée. Une réactivité dite "Aval" qui concerne les aléas physiques de production, ceux-ci peuvent représenter une ressource hors-service. C'est ainsi que nous proposons la prise en compte de quatre réactivités : réactivité à une commande urgente, réactivité à un poste HS, réactivité à une ressource d'équipement HS et réactivité à un transfert HS. Chaque réactivité est gérée par niveaux.

Pour la prise en compte du savoir-faire utile au pilotage de système de production, nous définissons les connaissances prises en compte et proposons une modélisation permettant sa capitalisation et son exploitation. Notre préoccupation première est de proposer une solution acceptable dans le milieu industriel et dans un langage qui lui soit familier. Pour cela, on considère le savoir-faire en terme d'actions dans l'exercice d'une activité par rapport au savoir qui peut être transmis indépendamment de la réactivité. Dans le cas d'un pilotage de système de production, le savoir-faire d'une entreprise regroupe les connaissances sur le produit (nomenclatures, gammes d'assemblage et gammes de fabrication) et sur les ressources utilisées (gammes de maintenance et gammes d'intervention). Toutes ces connaissances ont la particularité de pouvoir être représentées comme une liste d'actions à exécuter. Parmi les approches utilisées pour aborder les savoir-faire, nous avons opté pour un système à base de connaissances : le système SIRS (Système Interactif de Recherche de Séquences).

Parmi les possibilités offertes nous pouvons citer :

- la représentation graphique des connaissances sous forme de Grafcet-M,
- l'intégration de la notion de famille de gammes,
- la base de connaissances écrite à partir d'un langage quasi-naturel de type "Si .. Alors .. ",
- la capitalisation du savoir-faire réalisée de manière naturelle lors de la recherche d'une séquence d'actions.

Le système SIRS convient parfaitement pour aider à la capture, à la formalisation et à l'exploitation du savoir-faire. Nous rappelons que le choix de ne pas utiliser une représentation par réseau déjà existante telles que les RdP ou le Grafcet repose sur les difficultés à représenter le synchronisme et le parallélisme à partir d'un langage quasi naturel. De plus, le synchronisme et le parallélisme ne nous sont d'aucune utilité dans la représentation du savoir-faire manipulé.

L'architecture classique de fonctionnement d'un atelier est hiérarchisée par niveaux de responsabilité. Une telle structure n'est pas adaptée à l'implantation d'une réactivité par rapport au savoir-faire, car celui-ci n'est pas partagé et se trouve dilué dans les niveaux hiérarchiques. Par contre, elle paraît bien adaptée pour contrôler les actions à mener lors de la résolution d'un problème. C'est pourquoi, nous avons opté pour une architecture distribuée multi-experts avec contrôle hiérarchique. Parmi les différents outils existants, capables d'aider à l'implantation de ce type d'architecture, nous avons retenu ATOME (Another Tool for Multi-Expertise) développé au sein du CRIN de l'Université de Nancy.

Le système ATOME est un environnement d'aide au développement de système à multi-sources de connaissances et repose sur le modèle du blackboard. Il permet la séparation entre l'expertise du contrôle et l'expertise du domaine, offrant ainsi la possibilité de faire appel à différentes stratégies de résolution. Il permet le développement d'architecture composée de plusieurs types de sources de connaissances disposées suivant trois niveaux. Au niveau le plus élevé, se situe une méta-source de connaissances de contrôle dont l'objectif est de contrôler la résolution du problème en fonction d'une stratégie prédéfinie. Le niveau immédiatement inférieur est composé de plusieurs sources de contrôle appelées tâches. Chacune d'elles a la charge de contrôler les actions à mener dans un domaine spécifique du problème à résoudre et reste sous le contrôle de la stratégie. Le dernier niveau est composé de sources de connaissances spécialistes dans la réalisation de travaux spécifiques. Les informations manipulées nécessaires à la résolution du problème sont partagées au sein de plusieurs blackboards structurés par niveaux. Les spécialistes sont les seules sources de connaissances autorisées à accéder aux blackboards. Ils agissent à partir d'actions de modifications, de créations et de suppressions à tous les niveaux des blackboards. Les tâches sont informées de la pertinence d'une action à partir d'événements générés par les spécialistes. A partir de la liste d'événements réceptionnée, une tâche développe un raisonnement adapté au domaine à traiter. Un raisonnement peut être guidé par les événements, dirigé par les règles ou opportuniste. La stratégie est informée à partir d'un résumé de l'état des blackboards. Le résumé des blackboards est représentatif de l'état d'avancement de la solution du problème et permet ainsi à la stratégie de piloter sa résolution.

Dans notre approche, nous proposons une décomposition du problème de pilotage en quatre sous-problèmes principaux : l'initialisation, l'affectation des ressources d'équipement, l'affectation des transferts et le suivi d'exécution des opérations. Pour la tâche d'initialisation, un raisonnement guidé par les événements ou les règles peut convenir. Pour les tâches ayant en charge les problèmes d'affectation, nous avons opté pour un raisonnement opportuniste. En effet, leur rôle est de choisir parmi un ensemble de solutions possibles en fonction d'une stratégie prédéfinie. Pour la tâche de suivi, nous avons opté pour un raisonnement guidé par les événements. Pour aider les tâches dans la résolution de leurs problèmes respectifs, nous proposons un ensemble de spécialistes que l'on peut regrouper en quatre groupes. Un premier

groupe constitué de trois spécialistes (SP-RCS, SP-DNP, SP-DPP) qui prend en charge les problèmes d'interface avec le système de gestion de base de données O2. Un deuxième groupe constitué d'un seul spécialiste (SP-SVF) qui prend en charge la capitalisation et l'exploitation du savoir-faire. Un troisième groupe constitué de sept spécialistes (SP-ATA, SP-APT, SP-AOE, SP-AOT, SP-AMP, SP-AMO, SP-ATR) qui prend en charge les problèmes d'affectation. Un quatrième groupe constitué de deux spécialistes (SP-GPC, SP-INT) qui prend en charge le suivi d'exécution des opérations et l'interface avec chaque opérateur. En effet, leur rôle est de réagir à l'apparition d'événements. L'ensemble de ces spécialistes génère des actions sur les différents blackboards. Nous avons défini six blackboards. Un blackboard "Ressources" qui rassemble les informations concernant les opérateurs et le potentiel physique de l'atelier. Un blackboard "Demandes" qui regroupe les informations concernant les demandes urgentes. Un blackboard "ProgrammeFabrication" qui rassemble les informations concernant les affectations d'opérateurs, de postes et de ressources d'équipement déjà effectuées, un blackboard "InitSolutionPF" qui représente l'ensemble des solutions d'affectation de ressources possibles. Un blackboard "InitSolutionTR" qui représente l'ensemble des solutions d'affectation de transfert possible et enfin un blackboard "PostesCharge" qui regroupe l'ensemble des informations concernant les opérations effectuées sur chacun des postes de travail.

Après avoir défini les tâches, les spécialistes et les blackboards il reste à définir la stratégie et les blackboards résumés. Nous considérons que la stratégie doit contrôler le pilotage au niveau des commandes à réaliser et des performances réclamées par l'entreprise en fonction de la politique mis en place. C'est pourquoi, nous avons défini deux blackboards résumés. Un premier blackboard qui regroupe les informations sur les objectifs à réaliser et un deuxième blackboard qui regroupe les informations sur les performances du système d'exploitation, en particulier la représentation des temps d'attente. Pour cela, la stratégie fait appel à un certain nombre d'heuristiques représentant la stratégie à mettre en oeuvre lors du contrôle des tâches.

Il n'est pas nécessairement utile et efficace de vouloir piloter toutes les ressources de l'atelier (Ex : une machine qui ne travaille que quelques heures dans la semaine). De même, il n'est pas nécessaire de vouloir réagir à tous les aléas.

Nous souhaitons maintenant conclure au regard de l'architecture proposée. L'approche et le choix de son implantation ont permis d'apporter une contribution non négligeable à la problématique introduite. En effet, elle prend en compte les principales préoccupations telles que : la gestion d'une double réactivité, la capitalisation et l'exploitation de manière naturelle du savoir-faire, la prise en compte de l'ensemble des ressources, l'indépendance par rapport au type d'atelier, la prise en compte de plusieurs stratégies possibles et la modularité lui permettant ainsi de s'adapter aux spécificités de l'entreprise.

On note que la représentation des savoir-faire repose sur la notion de famille de gammes, ce qui permet de limiter les informations manipulées et de faire bénéficier le savoir-faire acquis sur une gamme à l'ensemble des gammes de la famille.

L'inconvénient majeur de cette approche est qu'elle requière un développement informatique conséquent.

Au terme de ce mémoire, nous pouvons envisager des perspectives de recherche à plus ou moins long terme. Un des développements à prévoir dans le futur immédiat est évidemment de développer chacune des sources de connaissances pour ainsi valider l'approche.

Parmi les développements futurs proches nous pouvons citer : l'association du système de pilotage à un système d'analyse de données pour permettre de dresser une liste des opérateurs, ressources ou autres éléments intéressants à prendre en compte lors du pilotage d'un système de production et la prise en compte dans le cadre de l'ingénierie simultanée.

A plus long terme, il est nécessaire d'envisager des applications de type intranet.

Bibliographie

A

[AMICE 89] AMICE. *Open System Architecture for CIM*, Vol. 1, Springer-Verlag, Berlin, 1989.

[AMICE 93] AMICE. CIMOSA: *Open System Architecture for CIM*, 2nd revised and extended edition, Springer Verlag, Berlin, 1993.

[Anthony 65] Anthony, R., N. *Planning and Control Systems : A Framework for Analysis*, Harvard University, Graduate School of Business Administration, Cambridge, Massachusetts, 1965.

[Archimede 91] Archimede, B. Conception d'une architecture réactive, distribuée et hiérarchisée pour le pilotage des systèmes de production, Thèse de doctorat, Université de Bordeaux I, Bordeaux, France, 1991.

[Austin 62] Austin, L., J. *How to Do Things with Works*. Oxford University Press, 1962. (Edition du Seuil, 1970, en français).

B

[Bakker 88] Bakker, H. *DFMS: A new control structure for FMS*, *Computers in Industry*, Vol.10, 1988, pp.1-9.

[Bauer et coll 94] Bauer, A., Bowden, R., Browne, J., Duggan, J. *Shop Floor Control System*, Chapman & Hall, Londres, 1994.

[Bauer et coll 91] Bauer, A., Bowden, R., Browne, J., Duggan, J., Lyons, G. *Shop Floor Control Systems. From Design to Implementation*, Chapman & Hall, Londres, 1991.

[Baumgartner et coll. 91] Baumgartner, H., Knischenski, K., Wieding, H. *CIM: Propositions pour une mise en œuvre de la Production*, Teknea, 1991.

[Bel 91] Bel, G. Pilotage et gestion d'atelier, Rapport introductif, Colloque Action de Recherche en Robotique et Productique du MRT, Paris, IIRIAM, 1991.

[Berard 83] Berard, C. Contribution à la conception de structures logicielles pour le pilotage d'atelier, Thèse de doctorat, Université de Bordeaux I, Bordeaux, France, 1983.

[Billaut 93] Billaut, J.C., Roubellat, F. A decision support system for real-time production scheduling, International Conference on Industrial Engineering and Production Management (IEPM 93), Mons, Belgique, Juin 1993, pp. 449-462.

[Bitran 89] Bitran, G., R., Tirupati, D. Hierarchical Production Planning, *MIT Sloan School Working Paper, n° 3017-89-MS*, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1989.

[Becker 60] Becker, H., S. Notes of the Concept of Commitment. *American Journal of Sociology* 66, pp. 32-40, July 1960.

[Bongaerts et coll. 96] Bongaerts, L., Wyns, J., Detand, J., Van Brussel, H., Valckenaers, P. Identification of manufacturing holons. In Pre-Proceedings of European Workshop on Agent-Oriented Systems in Manufacturing, Berlin, Allemagne, 26-27 Septembre 1996. pp. 13-29.

[**Bond 90**] Bond, H., A. Commitment : A Computational Model for Organizations of Cooperating Intelligent Agents. In Proceeding of the 1990 Conference on Office Information Systems, pp. 21-30, Cambridge, 1990.

[**Bond 90**] Bond, H., A. Distributed Decision Making in Organisation. *IEEE Transactions on Systems, Man and Cybernetics Conference*, Novembre, 1990.

[**Breuil 84**] Breuil, D. Outil de conception et de décision dans les organisations de gestion de production, Thèse d'état en Automatique, Université de Bordeaux I, Bordeaux, France, 1984.

[**Browne 88**] Browne, J. Production Activity Control-A key aspect of production control, *International Journal of Production Research*, 1988.

[**Brooks 89**] Brooks, R. A Robot that walks : Emergent Behaviors From a Carefully Evolved Network. A. I. Memo 1091, Massachusetts Institute of Technology, 1989.

C

[**Cammarata et coll. 83**] Cammarata, D., McArthur, D., Steeb, R. Strategies of Cooperation in Distributed Problem Solving. In Karlsruhe, editor, *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, V. 2, pp. 767-770, Germany, August, 1983.

[**Campagne 85**] Campagne, J.P. *Gestion de production*, Lyon, I.N.S.A. France, 1985, 183p.

[**Cantamessa 97**] Cantamessa, M. Agent-based modeling and management of manufacturing systems. *Computers in Industry*, Vol. 34, 1997, pp. 173-186.

D

[**David et coll. 89**] David, R., Alla, A. *Du Grafset aux Réseaux de Petri*, HERMES, Paris, 1989.

[**Dindeleux 92**] Dindeleux, E. Proposition d'un modèle et d'un système interactif d'aide à la décision pour la conduite d'atelier, Thèse de doctorat, Université de Valenciennes, France, 1992.

[**Dieng 90**] Dieng, R. Relation Linking Cooperating Agents. In *Proceedings of the 2nd European Workshop MAAMAW'90*, pp. 185-202, Saint-Quentin en Yvelines-France, August, 1990.

[**Dicesare et coll. 93**] Dicesare, F., Harhalakis, G., Proth, M, J., Silva, M., Vernadat, B, F. *Pratice of Petri Nets in Manufacturing*, Chapman & Hall, First edition, London, 1993.

[**Dilts et coll. 91**] Dilts, D.M., Boyd, N.P., Whorms, H.H. The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems*, Vol. 10, No. 1, 1991, pp. 79-92.

[**Doumeingts 84**] Doumeingts, G. Méthode GRAI: méthode de conception de systèmes de production, Thèse d'état en Automatique, Université de Bordeaux I, Bordeaux, France, 1984.

[**Duffie 88**] Duffie, N.A., Piper, R.S. Fault tolerant heterarchical control of heterogeneous manufacturing system entities, *Journal Of Manufacturing Systems*, Vol. 7, n°4, 1988, pp. 315-327.

[Durfee *et coll.* 89] Durfee, H., E., Lesser, R., V., Corkill, D., D. *Cooperative Distributed Problems Solving*, V. 4, pp. 83-147. Addison Wesley, 1989.

E

[Erman *et coll.* 80] Erman, L., Hayes-Roth, F., Lesser, V., Reddy, R. The Hearsay-II Speech Understanding System : Integrating Knowledge to Resolve Uncertainty, *Computing Surveys*, Vol. 12, n° 2, pp. 213-253, 1980.

F

[Ferber 90] Ferber, J. The Framework of Eco-Problem Solving. In *Proceedings of the 2nd European Workshop MAAMAW'90*, pp. 103-114, Saint-Quentin en Yvelines-France, August, 1990.

[Ferber 88] Ferber, J., Briot, J., P. Design of a Concurrent Language for Distributed Artificial Intelligence. Technical report, Université de Paris 6, 88/21, 1988.

[Ferber 88] Ferber, J., Ghallab, M. Problématiques des univers multi-agents intelligents. In *Actes des journées nationales PRC-GRECO Intelligence Artificielle*, Toulouse, pp. 295-320, mars, 1988.

G

[Gallagher *et coll.* 88] Gallagher, K., Corkill, D, Johenson,P. GBB référence Manual, Technical Report 88-66, Computer and Information Science Departement, Universtiy of Massachusetts, 1988.

[Giard 88] Giard, V. *Gestion de production*, ECONOMICA-Gestion, 1988.

[Gotha 93] Gotha. Les problèmes d'ordonnancement, *Recherche Opérationnelle*, Vol 27, n° 1, 1993, pp. 77-150.

[Gousty 88] Gousty, Y., Kieffer, J. P. Une nouvelle typologie pour les systèmes industriels de production, *Revue Française de Gestion*, 1988, pp.104-112.

[Gounthier 90] Gounthier, A. De l'optimisation des coûts de changement de fabrication vers l'atelier décentralisé d'ordonnancement, Thèse de doctorat, Institut National Polytechnique de Grenoble, France, 1990.

[Grabot 88] Grabot, B. Modélisation situationnelle et théorie des possibilités : application à l'aide à la décision d'exploitation des processus complexes, Thèse de doctorat, Université de Bordeaux I, Bordeaux, France, 1988.

H

[Haton *et coll.* 91] Haton, J.,P. Lâasri., B. Lâasri.,H. Marquis., P. *Le Raisonnement en Intelligence Artificielle*, Inter Editions, Paris, 1991

[Hayes-Roth *et coll.* 79] Hayes-Roth, B. Hayes-Roth., F., Rosenschein, S., Cammarata, S. "Modeling Planning as an Incremental, Opportunistic Process," *Proccedinings of IJCAI-79*, 375-383, 1979 (also in *Blackboard Systems*, Engelmores, R. and Morgan., T. Editors, Addison-Wesley, 1988).

[Hayes-Roth 84] Hayes-Roth, B. A Blackboard Model of Control. Technical Report HPP83-

38, Stanford University, 1984.

[**Huguet 97**] Huguet, P., Grabot, B. Modèle de référence et méthode orientée-objet pour la conception de système de pilotage d'atelier, *RAIRO - APII-JESA*, Vol. 31, n° 1, 1997, pp. 95-124.

[**Huguet 95**] Huguet, P., Grabot, B. A conceptual framework for shop-floor production activity control, *Int. J. Integrated Manufacturing Systems*. Vol.8, n° 5, 1995, pp. 357-369.

[**Huhns 87**] Huhns, M.(ed.) *Distributed Artificial Intelligence*, Morgan Kaufman, Los Atlos, CA, 1987

I

[**IMS-HMS 94**] System Components of Autonomous Modules and their Distributed Control. IMS-HMS Consortium, 1994.

J

[**Javel 93**] Javel, G. *L'organisation et la gestion de production*, Masson, Paris, 1993

[**Jones 85**] Jones, A., Mclean, C. A Production Control Model for the AMRF, Proceedings of the international ASEM, Conference on Computers in Engineering, 1985.

K

[**Kallel 85**] Kallel, G. Proposition d'une Conduite Décentralisée Coordonnée (CODECO) pour un atelier de fabrication, Thèse de doctorat, Institut National Polytechnique de Grenoble, France, 1985.

[**Koestler 89**] Koestler, A. *The Ghost in the Machine*, Arcana Books, London, UK, 1989.

[**Kouiss et coll. 95**] Kouiss, K., Pierreval, H., Mrbaki, N. Towards the use of a multi-agent approach to the dynamic scheduling of flexible manufacturing systems, International Conference on Industrial Engineering and Production Management (IEPM 95), Marrakech, Maroc, Avril 1995, pp. 118-125.

[**Kowk 94**] Kowk, A., Norrie, D. A development system for intelligent agent manufacturing software, *Integrated Manufacturing Systems*, Vol. 5, n° 4-5, pp. 64-76, 1994.

L

[**Laasri 89**] Laasri, H., Maitre, B. Flexibility and Efficiency in Blackboard Systems : Studies and Achievements in ATOME in Blackboard Architectures and Applications, Jagannathan, V., Dodhiawala, R., Baum, M. editors, Academic Press, pp. 57-73, 1989.

[**Labidi 93**] Labidi, S., Lejouad, W. De l'intelligence Artificielle Distribuée aux Systèmes Multi-Agents, Programme 2 - Calcul symbolique, programmation et génie logiciel, Projet SECOIA, Rapport de recherche n° 2004, pp. 39, Août, 1993.

[**Lissandre 1987**] Lissandre, M. *Maîtriser SADT*, Colin, Paris, 1987.

[**Le Moigne 77**] Le Moigne, J.L. *La Théorie du Système Général, Théorie de la Modélisation*, Presses Universitaires, France, 1977.

M

[Maret 97] Maret, P. Pinon., J., M. : *Ingénierie des savoir-faire*, Edition HERMES, 1997.

[Mathews 96] Mathews, J. Holonic organisational architectures. *Human Systems Management*, Vol. 15, 1996, pp. 1-29.

[Maître 89] Maître, B., Laasri, H. Flexibility and Efficiency in Blackboard Systems : Studies and Achievements in ATOME. In *Blackboard Architectures and Application*, Jagannathan, V. et Baum, S.L. Academic Press, pp. 309-322, New York, 1989.

[Mestoudian 88] Mestoudian, J. et De Crescenzo, J. *La gestion de production assistée par ordinateur*, tome 1 : planification des ressources de production et gestion de stocks, Paris : Ed.de l'usine nouvelle, 1988. 224p.

[Merec 87] Merec, C., Cohérence des décisions en planification hiérarchisée. Thèse de doctorat d'état, Université Paul Sabatier, Toulouse, France, 1987.

[Melese 90] Melese, J. *Approches Systémiques des Organisations*, Les Editions d'Organisation, Paris, 1990.

[Melinyk 86] Melinyk, S.A., Carter, P.L. Identifying the principals of effective production activity control, 29th International Conference of American Production and Inventory Control Society, St. Louis, MO, 1986.

[Merbaki 95] Merbaki, N. Une approche d'ordonnancement temps réel basée sur la sélection dynamique de règles de priorité, Thèse de doctorat, Université Claude Bernard, Lyon I, Lyon, France, 1995.

[Mercé 87] Mercé, C. Cohérence des décisions en planification hiérarchisé, Thèse de doctorat d'état-sciences, Université Paul Sabatier, Toulouse, France, 1987.

[Messaoud 86] Messaoud, D. Problèmes des décisions dans une cellule de production flexible utilisant la coopération entre robots, Thèse de doctorat, Université de Lille Flandres Artois, Lille, France, 1986.

[Mintzberg 86] Mintzberg, H. *Structure et dynamique des organisations*. Les éditions d'organisation (Paris) et les éditions Agence d'ARC inc. (Montreal), 1986.

[Müller 96] Müller, J.,P. Quinqueton., J. *IA Distribuée et systèmes Multi-Agents*, JFIADSMA 96, Editions HERMES, Paris, 1996.

N

[Newell 82] Newell, A. The Knowledge Level. *Artificial Intelligence* 18, pp. 87-127, 1982.

O

[Orlicky 75] Orlicky, J. *Material Requirements Planning*, the new way of life in Production and Inventory Mangement. MacGraw Hill, 1975.

P

[Parunak 91] Parunak, V.D.H. Characterizing the Manufacturing Scheduling Problem, *International Journal of Manufacturing Systems*, Vol.10, n° 3, 1991, pp. 241-255.

[Pluym 90] Pluym, V., D. Knowledge-based decision-making for job-shop scheduling, *International Journal of Computer Integrated Manufacturing*, Vol.3, n° 6, 1990, pp. 354-363.

[Pôl 92] Pôle et lieu d'échange en IAD. Vers une taxinomie du vocabulaire pour les systèmes multi-agents. In *Journée Systèmes Multi-agents PRC-GDR, Intelligence Artificielle*, Nancy, décembre, 1992.

[Pourcel 86] Pourcel, C. *Systèmes automatisés de production*. Cépaduès, Editions, Toulouse, 1986

R

[Roboam 93] Roboam, M. *La méthode GRAI, Principes, outils, démarche et pratique*. Teknéa, 1993.

[Roder 94] Roder, P. Visibility is the Key to scheduling success, *APICS, Planning and scheduling*, pp. 53, August 1994.

[Roubellat 88] Roubellat, F., Thomas, V. Une méthode et un logiciel pour l'ordonnancement en temps réel d'ateliers, *R.A.I.R.O., APII*, Vol.22, n° 5, 1988, pp. 419-438.

[Roy 98] Roy, D. Une architecture hiérarchisée multi-agents pour le pilotage réactif d'ateliers de production, Thèse de doctorat, Université de Metz, Metz, France, 1998.

S

[Searle 83] Searle, J., R. *Intentionality : An Essay in the philosophy of mind*. Cambridge University Press, New York, 1983.

[Searle 90] Searle, J., R. *Intentions in Communication*, chapter 19 : Collective Intentions and Actions, pp. 401-415. MIT Press, London, 1990

[Spadoni 87] Spadoni, M. Etude d'un système de génération automatique de gammes de fabrication, Thèse de doctorat, Université de Metz, Metz, France, 1987.

[Sun 94] Sun, D., Lin, L.: A dynamic job-shop scheduling framework : a backward approach, *International Journal of Production Research*, Vol. 32, n° 4, 1994, pp. 967-985.

T

[Tacquard *et coll.* 95] Tacquard, C., Baptiste, P., Manier, H. A behavioral model to schedule and control a flexible manufacturing ring : application to the push and the pull flows, *International Conference on Industrial Engineering and Production Management*, Marrakech, Maroc, Avril 1995, pp. 127-136.

[Tchaco 94] Tchaco, J., F., N. Contribution à la conception d'un système de pilotage distribué pour les systèmes automatisés de production, Thèse de doctorat, Université de Valenciennes, Valenciennes, France, 1994.

[Thomas 80] Thomas, V. Aide à la décision pour l'ordonnancement en temps réel d'atelier Thèse de doctorat, Université Paul Sabatier, Toulouse, 1880.

P

[Parunak 91] Parunak, V.D.H. Characterizing the Manufacturing Scheduling Problem, *International Journal of Manufacturing Systems*, Vol.10, n° 3, 1991, pp. 241-255.

[Pluym 90] Pluym, V., D. Knowledge-based decision-making for job-shop scheduling, *International Journal of Computer Integrated Manufacturing*, Vol.3, n° 6, 1990, pp. 354-363.

[Pôl 92] Pôle et lieu d'échange en IAD. Vers une taxinomie du vocabulaire pour les systèmes multi-agents. In *Journée Systèmes Multi-agents PRC-GDR, Intelligence Artificielle*, Nancy, décembre, 1992.

[Pourcel 86] Pourcel, C. *Systèmes automatisés de production*. Cépaduès, Editions, Toulouse, 1986

R

[Roboam 93] Roboam, M. *La méthode GRAI, Principes, outils, démarche et pratique*. Teknéa, 1993.

[Roder 94] Roder, P. Visibility is the Key to scheduling success, *APICS, Planning and scheduling*, pp. 53, August 1994.

[Roubellat 88] Roubellat, F., Thomas, V. Une méthode et un logiciel pour l'ordonnancement en temps réel d'ateliers, R.A.I.R.O., APII, Vol.22, n° 5, 1988, pp. 419-438.

[Roy 98] Roy, D. Une architecture hiérarchisée multi-agents pour le pilotage réactif d'ateliers de production, Thèse de doctorat, Université de Metz, Metz, France, 1998.

S

[Searle 83] Searle, J., R. *Intentionality : An Essay in the philosophy of mind*. Cambridge University Press, New York, 1983.

[Searle 90] Searle, J., R. *Intentions in Communication*, chapter 19 : Collective Intentions and Actions, pp. 401-415. MIT Press, London, 1990

[Spadoni 87] Spadoni, M. Etude d'un système de génération automatique de gammes de fabrication, Thèse de doctorat, Université de Metz, Metz, France, 1987.

[Sun 94] Sun, D., Lin, L.: A dynamic job-shop scheduling framework : a backward approach, *International Journal of Production Research*, Vol. 32, n° 4, 1994, pp. 967-985.

T

[Tacquard et coll. 95] Tacquard, C., Baptiste, P., Manier, H. A behavioral model to schedule and control a flexible manufacturing ring : application to the push and the pull flows, *International Conference on Industrial Engineering and Production Management*, Marrakech, Maroc, Avril 1995, pp. 127-136.

[Tchaco 94] Tchaco, J., F., N. Contribution à la conception d'un système de pilotage distribué pour les systèmes automatisés de production, Thèse de doctorat, Université de Valenciennes, Valenciennes, France, 1994.

[Thomas 80] Thomas, V. Aide à la décision pour l'ordonnancement en temps réel d'atelier Thèse de doctorat, Université Paul Sabatier, Toulouse, 1880.